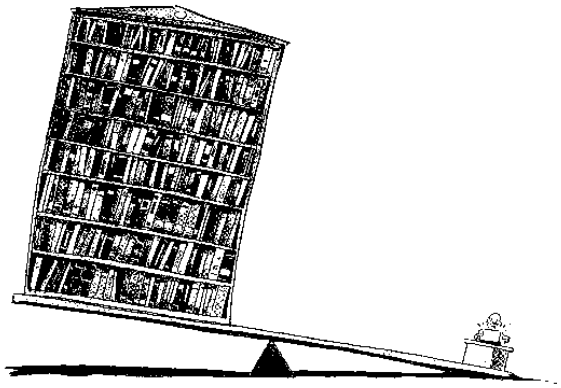
KE Texpress



User Guide

KE Software Pty Ltd

Copyright © 1993 - 2004 KE Software Pty Ltd
This work is copyright and may not be reproduced except
in accordance with the provisions of the Copyright Act.

Contents

Chapter 1 Introduction ----------1-1

Using This Guide 1-4
KE Texpress Terminology 1-5
Notation.....1-8
Terminal Capabilities.....1-9
Invocation.....1-10

Chapter 2 Using KE Texpress ----------2-1

Front-End-Menu.....2-3
Screen Layout.....2-4
Pull-Down Menus 2-5
Keyboard Accelerators 2-7
On-Line Help.....2-8
Look-up Table 2-9
Printing or Copying Information 2-10
Screen Copy 2-12
Screen Redraw.....2-14
Suspend or Invoke Shell.....2-15
Interrupt.....2-16
Text Perusal.....2-17
Database Privileges 2-18
Record Privileges 2-18
Item Privileges 2-19
Operation Privileges.....2-20

Chapter 3 Entering Information ----------3-1

Data Types.....3-3
Movement around a Form 3-4
Entering Information into Fields 3-5
Look-up Table 3-6
Hierarchies 3-8
Miscellaneous Commands 3-9
Field Editing Commands.....3-11

Chapter 4 Record Insertion----------4-1

Key Values 4-3
Key Value Assignment Database Options 4-3
Ditto Commands 4-5
Default Field Values.....4-6
Comments 4-7
Assignment Expressions 4-8
Linked Databases 4-9

Data Validation 4-10
Key Item 4-10
Hierarchy 4-10
Look-up Table 4-11
Field 4-12
Ending an Insertion 4-15
Save 4-15
Temp Save 4-17
Exit 4-18

Chapter 5 Database Querying ---------- 5-1

The Query Form 5-4
Text Queries 5-5
Word Break Characters 5-5
Noise Words 5-6
Query Commands 5-7
Query Comments 5-7
Exit 5-7
Clear Form 5-7
Single Term Query 5-7
Or Query 5-7
And Query 5-8
Phonetic Search 5-8
Stemming Search 5-8
Phrase Query 5-9
Case Search 5-9
Range Queries 5-9
Key Value Queries 5-9
Key Value Range Queries 5-10
Pattern Matching Facilities 5-11
Perform Query 5-14
The Display Form 5-15
Display Mode Commands 5-15
Exit 5-15
Movement 5-15
Among Retrieved Records 5-15
Within a Record 5-16
View comments 5-16
Select Display Form 5-17
Record Histories 5-18
Show History 5-18
View Later Version 5-18
View Earlier Version 5-18
Change Display Form 5-18
Delete History 5-18
Exit 5-18
Report Generation 5-19
Peruse records 5-20
Jump to Record 5-21

Edit Records....5-22
Delete Records....5-26
Record Privilege Level Commands....5-28
Show Privilege Level....5-28
Change Privilege Level....5-28
Re-order Records....5-29
Current Matches....5-29
For Subsequent Queries....5-30
Record Copy or Print....5-31
Current Record....5-31
All Retrieved Records....5-31
Miscellaneous Commands....5-32
Verify Matches....5-32
Set Ditto Record....5-32
Save a Dave File....5-33
Discard Record From Retrieval....5-33
Show Video....5-33
Interpret Form Expression....5-33
Additional Query Commands....5-34
Perform Separate Query....5-34
Combine Queries....5-34
Disappearing Matches....5-35

Chapter 6 Database Reports----------6-1

Create/Edit a Report Form....6-4
Select Objects....6-5
Moving Around the Report Form....6-6
Edit Commands....6-8
Retrieve Item....6-8
New Header....6-8
Insert Box....6-9
Move Object....6-9
Copy....6-10
Change Objects....6-11
Append Prompt to Item....6-12
Delete Object....6-12
Undo Last Command....6-12
Attributes....6-13
View/EditAttributes....6-13
Form Bounds....6-15
Form Order....6-17
Heading Section....6-18
Footer Section....6-18
Miscellaneous Commands....6-19
Form Status....6-19
Validation Form....6-19
Form Coordinates....6-19
Privilege Level View....6-20
Copy Form....6-20

Exit the Report Form 6-21
Save Form 6-21
Exit 6-21
Interrupt a Command 6-22
Rename Report Form 6-23
Delete Report Form 6-23
View Report Form 6-23
Copy Report Form 6-24
Generate Reports 6-25
Report all Matches 6-25
Report this Match 6-25
Moving around the Report Generation Form 6-28
Generate Report 6-28
Exit 6-29
Special Report Operators 6-30
Space Removal: $S 6-30
Space Replacement: $X 6-31
Line Removal: $L 6-31
Prompt Removal if there is no data: $P ... $P ... $E6-32
Line Replacement: $F 6-32
Line Joining: $J 6-33
Line Joining With Word Wrap: $W 6-33
Query Replacement: $Q 6-34
Record Number: $N 6-34
Multiple Column Output Support: $Cn 6-34
Embedded Control Characters: $Onnn :m 6-34
Report Pagination - texpaginate 6-35
Line Drawing - texcat 6-39

Chapter 7 Miscellaneous Operations ---------- 7-1

Database Usage Statistics 7-3
Perform a Unix Command 7-4
Version Number 7-5
Listing Databases 7-5
Database Options 7-7
Database Pathname 7-8
Database Component Pathname 7-8
Linked Database Component 7-9
General Pathnames 7-10
Screen Copy 7-10
File Copy 7-11
Insert Mode 7-11
Query Mode 7-12
Display Mode 7-12
History 7-15
Global Edit 7-15
Reports 7-15
Rebuild, Load and Batch Insertion 7-17
General 7-18

Incorrect Option Settings 7-23

Chapter 8 Videodisc Interface ----------8-1

Videodisc Commands 8-3
Examples 8-7
Connecting to the Videodisc Player 8-8
Interactive Use.....8-9

Chapter 9 Macros ----------9-1

Macro Definition and Execution 9-3
Letters Component 9-4
Sequence Component.....9-5
Modes Component 9-7
Menuopt Component 9-8
Help Component 9-9
Startup Macros 9-10
Mode Startup Macros 9-11

Appendix A Quick Reference Guides ---------- A-1

Global Edit Mode A-1
Field Editing.....A-1
Insert/Query/Edit Mode.....A-2
Display Mode A-4

Appendix B Key Types ---------- B-1

Appendix C KE Texpress Library Types ---------- C-1

Chapter 1

Introduction

Using This Guide 1-4
KE Texpress Terminology 1-5
Notation.....1-8
Terminal Capabilities.....1-9
Invocation.....1-10

Overview

The KE Texpress Information Management System is a powerful and flexible object-oriented database package encompassing database design, data capture, information retrieval and maintenance. This manual describes the retrieval, insertion and reporting components of KE Texpress which are collectively referred to as **texforms**. **Texforms** provides facilities for capturing, querying and processing data in a KE Texpress database.

This document describes all of the facilities provided by **texforms,** some of which may not be appropriate for all databases. Those not utilized by a particular application should be ignored. The major operations performed by **texforms** are those of data capture, database querying and reporting. Other operations tailor the user's working enviroment.

A database consists of a computerized collection of object descriptions called **records**. A record may take the form of a film description, a library catalogue card, a transaction receipt or even an address book entry. In these cases, the actual objects referred to are the film, the library book, the transaction and the friend or acquaintance, respectively. Each record should contain all of the information known about the corresponding object.

Texforms accepts and stores these records for later use using the data capture facility. A record is stored only after passing a series of tests set up by the DBA to ensure the accuracy and validity of the data entered. The information in this record can then be accessed by all database users.

The operation of database querying involves asking the computer questions relating to the information stored in the database. These questions, for example, could be requests for:

- All films starring Fred Astaire and Ginger Rogers.
- All of the books written by Arthur C. Clarke.
- A list of all of the transactions with BHP which occurred on Thursday.
- The names of all friends living at the beach who own a boat.

These questions are asked by entering information on Query form similar to that used during data capture. When a query is performed, all of the records stored in the database are very quickly checked to see if they match the query. The matching records can be viewed individually, summarized, modified, copied and so on.

The final operation involves reporting. This can be used for such diverse tasks as producing ordered one-line summaries of an entire database, or just printing a label for a single record. Forms can be designed which display all or part of the information contained in each record. The records to be used in the report are retrieved (using one or more queries), and can be re-ordered in any desired way and finally printed using the Report Form.

The remainder of this manual describes each step involved in the basic use of these three functions as well as the additional commands available. These additional commands, although not essential to the use of **texforms**, provide more power, greater flexibility and ease of use.

Using This Guide

This manual, the KE Texpress User Guide, is one of a suite of manuals describing the KE Texpress Information Management System.

Chapter 2 of this manual describes the basic KE Texpress interface including how to use the pull-down menu facility and the keyboard accelerators. In Chapter 3, there is a description of facilities available when entering information into a form. Chapter 4 discusses the query mechanism of KE Texpress. Chapter 5 describes how data can be entered into a KE Texpress database. The reporting mechanism of KE Texpress is covered in Chapter 6.

Chapter 7 covers miscellaneous other commands available within the Data Manager while Chapter 8 concentrates on the commands available to drive a videodisc player.

The final chapter, Chapter 9, describes the KE Texpress macro facility and is applicable for more advanced users wanting to streamline certain repetitive processes.

Appendix A provides a quick reference guide to the keyboard accelerators available in the most commonly used modes of KE Texpress.

Appendices B describes the Key types and Appendix C describes the Library types

KE Texpress can be used for a very broad range of applications. The screen layout for any application is determined by the Database Administrator DBA and can differ greatly from the user's applications. However, the basic concepts of Texpress and the interaction with the system are generally the same.

KE Texpress Terminology

The KE Texpress system is a collection of programs each of which performs a specific task. For convenience, some of these programs have been given names which are referenced throughout this manual. The program names are as follows:

KE Texpress

This name refers to the entire database system and incorporates all of the following programs.

The Administration Menu (texadmin)

This refers to the database administrator's menu which contains functions used for the design and maintenance of databases.

texforms

This refers to the retrievel, insertion and reporting components of KE Texpress.

texdesign

This program is used for the design of forms describing the type of data to be collected and queried.

KE Texpress uses a specific terminology to describe the various objects that are displayed and manipulated on the terminal screen. These terms and their definitions are as follows:

Database administrator, DBA

The Database Administrator (hereafter, DBA) is the creator of a database. This user has total control over the database, deciding the format the database shall take and the types of queries that are permitted. The DBA also has the power to delegate database privileges to other users.

Form

All user interactions with KE Texpress are handled via forms. A form is a screen image describing the type of data to be maintained and the actions to be performed on that data. It consists of a collection of headers and fields arranged in a format specified by the DBA. A form may be larger than the computer display on which it is being viewed. The various forms are Insertion form, Query and Report/Display form.

Insertion form

This is the form used by the user to enter new records, and is identical to the edit form. Only the DBA is authorised to create or tailor the insertion form. Users must have insertion privilege to be able to call up this form.

Query form

The Query form is used to enter the query terms which are matched with the data in records during a query. It contains part or all of the indexed fields in the Insertion form. Texdesign automatically generates the Query form from the Insertion form, after which it may be edited by the DBA. A database may have multiple query forms, tailored for particular users or groups of users.

Report/Display form

A Report or Display form displays information following a successfully matched query. A database may have multiple Diplays forms, tailored to display only information relevant to particular users or groups of users. The form can also contain additional reporting features, as well as altered prompts and headers.

Record

A record is a collection of data which uniquely identifies one physical object.

Header

A header is a string of characters which help describe the nature of the database. It is not directly associated with any data in a record form.

Item

Items are the basic form elements. They consist of an optional prompt followed by one or more fields.

Prompt

A prompt is a string of characters (similar to a header) which is intuitively associated with a field or collection of fields.

Field

A field is an area of the form where the data is actually entered. In **texforms**, while information is being entered into a field, it is displayed as a sequence of underscores. At all other times, it holds the data associated with the field at that time.

There are modes of operation within **texforms** which are referred to throughout the text of this document. The major modes are:

Query

This is the mode of operation in which database queries are performed. A successful query involves collecting information from a user, using that information to query the database and then displaying all of the matching records. KE Texpress automatically enters this mode upon invocation of the database. Exiting from Query mode exits the user from the database. See also Query Form.

Display

This is the mode of operation in which records matching a query are displayed on the screen. It directly follows Query mode and is entered when a query retreves at least matching record. Exiting from Display mode results in a return to Query mode. See also Report Display/form.

Insert

This is the mode of operation in which records are created. Entry into this mode is via Query or Display mode. Exiting results in a return to the previous mode. See also Insert form.

Edit

This is the mode of operation in which information in an existing record is modified. This mode is entered from Display mode. Exiting results in a return to the previous mode of operation.

Refer to Appendix A for Quick Reference Guides which describe these modes in detail and outline the entry and exit commands for each.

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. For uniformity and ease of use, only the pull-down menu option is described in the guide.

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.

Terminal Capabilities

KE Texpress is a terminal intensive program and uses as many of the terminal's capabilities as possible. These characteristics are passed to KE Texpress and this is done by setting the Unix environment variable **TERM** and defining termtype in the KE Texpress terminal directory.

The csh command to do this is:

```
setenv TERM termtype
```

The sh equivalent is:

```
TERM=termtype
export TERM
```

Some of the terminal capabilities, used by KE Texpress include highlighting, which is used extensively to visually distinguish user input from system prompts; line drawing; function keys; and European or Asian character display.

KE Texpress can be run on an enormous range of terminal types with all commands being accessible using KE Texpress's pull down menus. In order to bypass the pull down menus, keyboard accelerators are available for many KE Texpress commands. By default, KE Texpress is set up to use function keys as the keyboard accelerators for the most common commands and control characters for a range of other common commands. However, the system can be configured by the DBA or the system administrator, who can:

- Define function keys for any type of terminal.
- Change the characters used to invoke any or all of the commands supported by KE Texpress.
- Change the text strings used by KE Texpress (e.g. error messages, warnings, menu options, on-line help, etc.). Refer to KE Texpress Administration Guide for more details.

This manual describes the default settings for the commands and text strings. If these settings have been altered for a particular installation or database, the on-line help facility is automatically updated to reflect these changes.

Invocation

Texforms, the KE Texpress forms interface can be invoked by the Unix commands:

```
texpress [-ivy] dbname
```

or

```
texforms [-ivy] dbname
```

where *dbname* is the name of the database. The letters enclosed in square brackets [-ivy] are options which may be specified to alter the default functionality of **texforms**.

The command line options are:

-i Commence **texforms** in Insert mode instead of the default Query mode. Exiting from Insert mode terminates **texforms**.

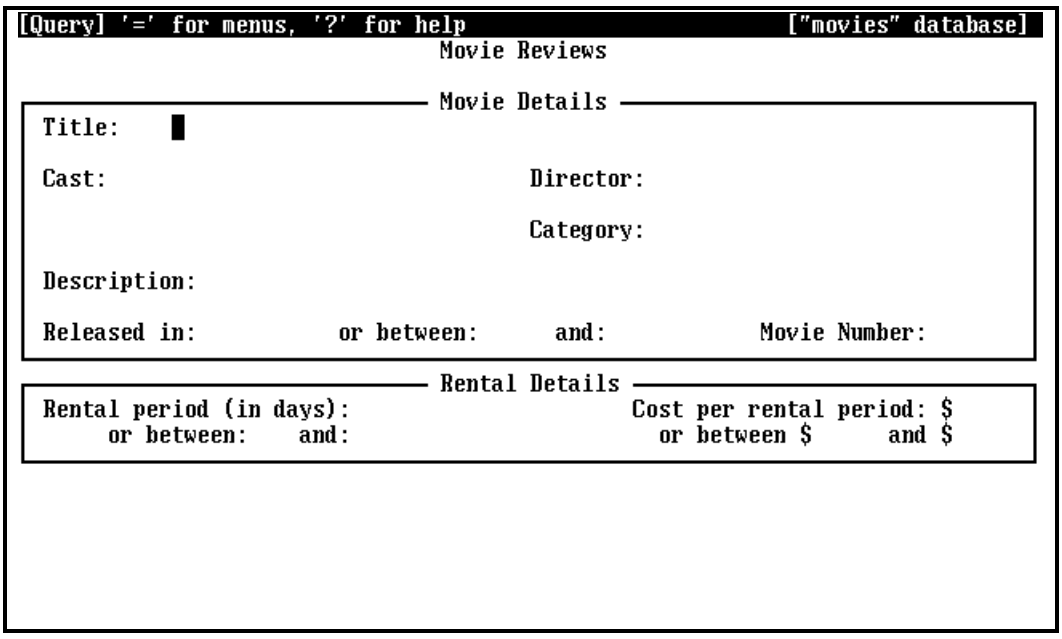
-v This option is only required if the database has been designed to contain information to access a videodisc and a videodisc player is available. It attempts to establish a connection with the videodisc player and, if successful, prevents any other users from accessing the videodisc player during the sessions. Any videodisc reference stored in a record is passed to the videodisc player to display a particular frame or sequence on the monitor.

-y Bypass the message of the day and commence Query mode immediately.

Before the database is called up, a copyright message is displayed, followed by a message of the day, if any. This is a message entered by the DBA to be shown to all users running the database and is generally used by DBAs as a bulletin board Pressing the **space bar** clears this message and calls the database.

This displays a form on the screen, generally the Query form selected for this user. (If invocation should fail then the shell environment variable, PATH, may need to be altered to include the directory in which the KE Texpress programs reside. See the system administrator for help or consult the Unix reference manuals.)

An example of a query form follows:



Query Form Example

The commands displayed throughout KE Texpress are those which the user is permitted to perform and are dependent on both the database design and the privileges of the user.

If the database is locked or closed down by the DBA for maintenance operations, it cannot be invoked, and a message displays

```
"dbname"database is closed. Try again later
```

Chapter 2

Using KE Texpress

Front-End-Menu....2-3
Screen Layout....2-4
Pull-Down Menus....2-5
Keyboard Accelerators....2-7
On-Line Help....2-8
Look-up Table....2-9
Printing or Copying Information....2-10
Screen Copy....2-12
Screen Redraw....2-14
Suspend or Invoke Shell....2-15
Interrupt....2-16
Text Perusal....2-17
Database Privileges....2-18
Record Privileges....2-18
Item Privileges....2-19
Operation Privileges....2-20

Overview

This chapter describes the general features and utlities of KE Texpress. Some of these may be used in a variety of KE Texpress modes. For example, a Look-up table can be called up in Query mode as well as Insert mode. The following sections describes each of these utilities.

Front-End Menus

Front-end menus are usually displayed when the user first calls up the system. Some users, such as OPAC users, may be taken into a Query screen as soon as they login. Various front-end menus can be set up by the DBA, tailored to the privileges and functions of the users.

Menu options can be selected in one of two ways.

First, the user can move to the option by using the arrow keys ↓ and ↑, which will cause the option to be highlighted. Pressing ↵ will then select the highlighted option.

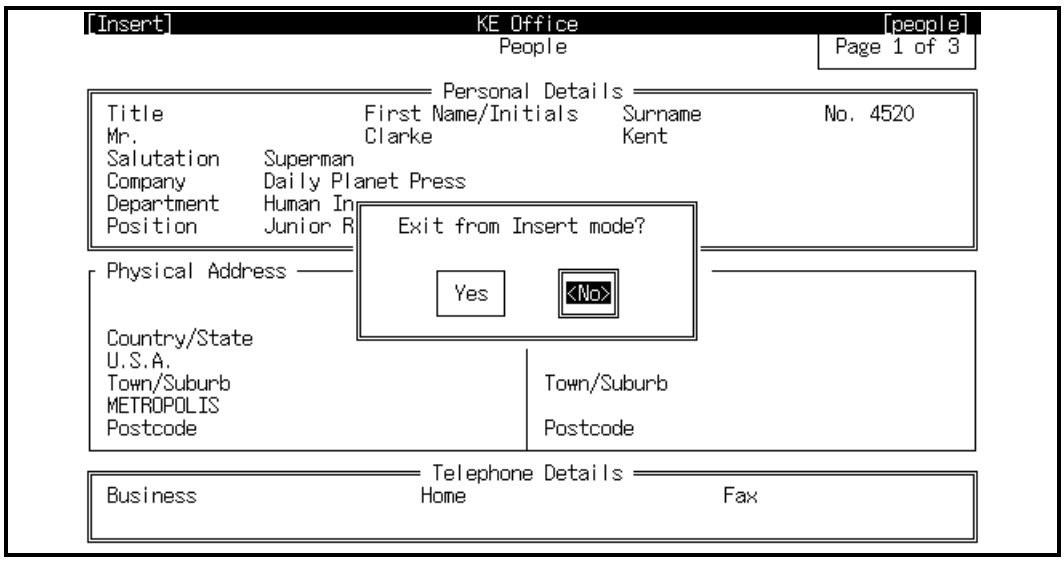
Second, the user can enter the letter or number associated with the menu. This immediately selects the option without having to press ↵.

To exit from the menu (always the last option on the menu), the user selects its associated letter or number, or alternatively, type **`x`**, **`X`**, **`Ctrl+X`** or **`F1`** (if the terminal supports function keys.)

If a menu option cannot be accessed or performed, the option will appear dimmed and it will not be possible to select it or move to it.

Screen Layout

The screen is divided into two regions: the **menu bar**, and below it the **form window** as shown below:



Screen layout

The menu bar occupies the top line of the screen. Usually it displays a mode indicator on the left, the command for pulling down menus or help, and the name of the application in the centre. When pull down menus are selected, the menu bar displays the titles under which the available commands are grouped. During data entry, the menu bar is also used to display helpful prompts.

The mode indicator displayed in square brackets at the left edge of the menu bar indicates the current mode of operation, i.e. Insert, Query, Display and Edit. The database name is displayed in square brackets at the right edge of the menu bar.

The form window displays the various Insertion, Query and Report forms associated with the database. A database can have multiple query, and display or report forms.

A form may take up more space than is available on the computer screen, i.e. it may spill over onto several screens. At the top right hand corner of the screen is often a message, e.g. `"Page 1 of 3"`. This informs the user that the screen currently displayed is the first of 3 screens. Prompts, status messages and error messages are displayed in a pop up message box. This message box whose size varies according to the particular message, pops up in the centre of the screen, overlaying the form window. When the user has responded, the message box disappears.

Pull-Down Menus

All KE Texpress commands can be invoked by selection from a pull-down menu. These menus are normally not displayed and emerge from the menu bar only when requested. The menu bar displays the titles under which the available commands are grouped. When pull-down menus are selected, the group of commands under the first title is displayed and the first command on that menu is highlighted. Selection of a command involves first moving to its menu (using ← or →) and then moving to the option (using ↑ or ↓). Once the option is highlighted, the user calls up the command by pressing ↵ . Note that keyboard accelerators cannot be used to involve commands while pull-down menus are exposed.

Some menus may display commands that cannot be selected, which will appear dimmed. This can occur when commands that are normally available are invalid under a particular set of circumstances. For example, the command to move to the next record when viewing the results of a query (**[Display] Move ⇒ Forward a record**) is invalid if only 1 record has been retrieved by the query.

The pull-down menu operations are described in the following paragraphs. Keyboard accelerators are displayed in parentheses, separated by a comma if there is more than 1.

The pull-down menu commands are:

Display pull-down menus (=)

Typing this key displays the first pull-down menu. The first available command on that menu is highlighted. If pull-down menus had previously been accessed in the current mode of operation, then the menu containing the last selected command is displayed and that command is highlighted.

Select command (↵)

Typing the ↵ key removes the menu from the screen and performs the command currently highlighted.

Move down (↓, j)

Typing either of these keys moves the menu cursor down the currently displayed menu to the next command which may be selected.

Move up (↑, k)

Typing either of these keys moves the menu cursor up the current menu to the previous command which may be selected.

Move left (←, h)

Typing either of these keys moves the menu cursor to the left adjacent pull-down menu, displaying its commands.

Move right (→, l)

Typing either of these keys moves the menu cursor to the right adjacent pull-down menu, displaying its commands.

Move to menu via title

It is possible to move directly to a pull-down menu by typing the first letter in the menu's title. Either the lower or upper case letter may be entered. If a particular letter refers to multiple titles then the pull-down menu associated with the first of these titles to the right of the current pull-down menu is displayed.

Help (?, F2)

Typing either of these keys displays a pop-up message box oulining the operation of the currently highlighted command. The last line of the message box displays a list of keyboard accelerators which can be used to invoke the command directly. Also displayed on the last line of the message box is the name of the KE Texpress help file which contains the currently displayed text.

Exit (Ctrl+X, esc)

Typing either of these keys clears the pull-down menus from the screen without performing any command on the menus.

Keyboard Accelerators

Most KE Texpress commands have keyboard equivalents. This means that the command can be invoked directly from the keyboard (rather than selection from a pull-down menu), providing fast access to heavily used commands. Typically the keyboard accelerators are control characters, however the major commands may also be invoked using function keys (provided your terminal type supports functions keys and KE Texpress has an appropriate description for your terminal).

When a pull down menu is displayed, the keyboard accelerator(s) (if any) is displayed to the right of the command name. Note that whilst any pull-down menu is displayed, keyboard accelerators cannot be used. A command may have many keyboard accelerators, however a pull down menu will display only the first one. A list of all the keyboard accelerators for a particular command can be obtained by selecting help when the command is highlighted on the pull-down menu.

On-Line Help

On-Line, context-sensitive help is always available in KE Texpress. It can be selected by typing **?** or **F2.**

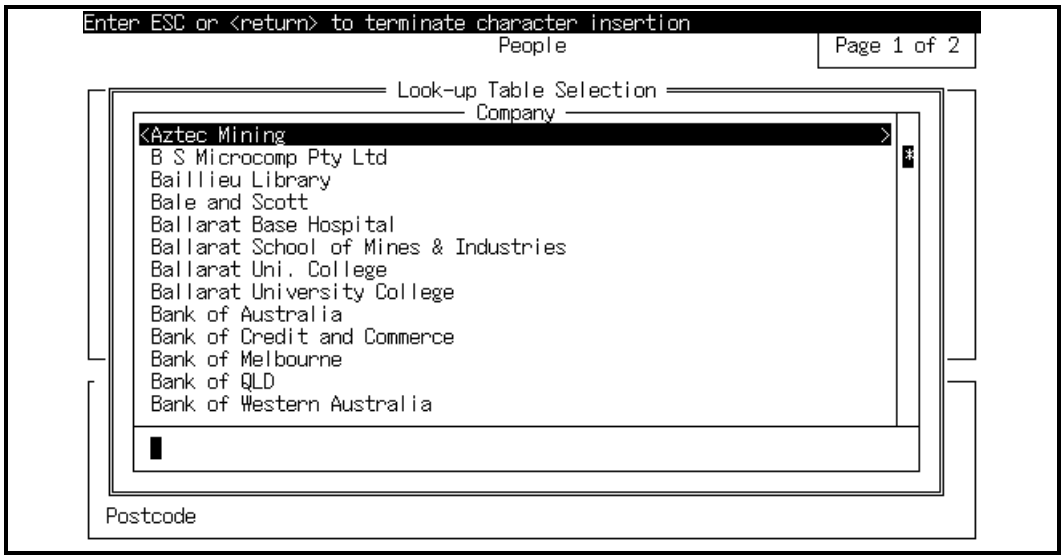
When help is selected, one or more screens of text describing the current mode of operation or the highlighted function on the pull-down menu, are displayed.

Typing ↵ causes the next screen to be displayed. Typing ↵ on the last help screen returns the user to KE Texpress.

To exit from help, type **F1** or **Ctrl+X.**

Look-up Tables

A Look-up Table is a list of all entries previously entered into the system at that field. Choosing an entry from this table saves the user typing it into the field, which also eliminates typographical errors. When used during record insertions or edits, Look-up Tables also keep the format of entered information consistent across users. See also Chapter 3 of this guide for details on using Look-up tables.



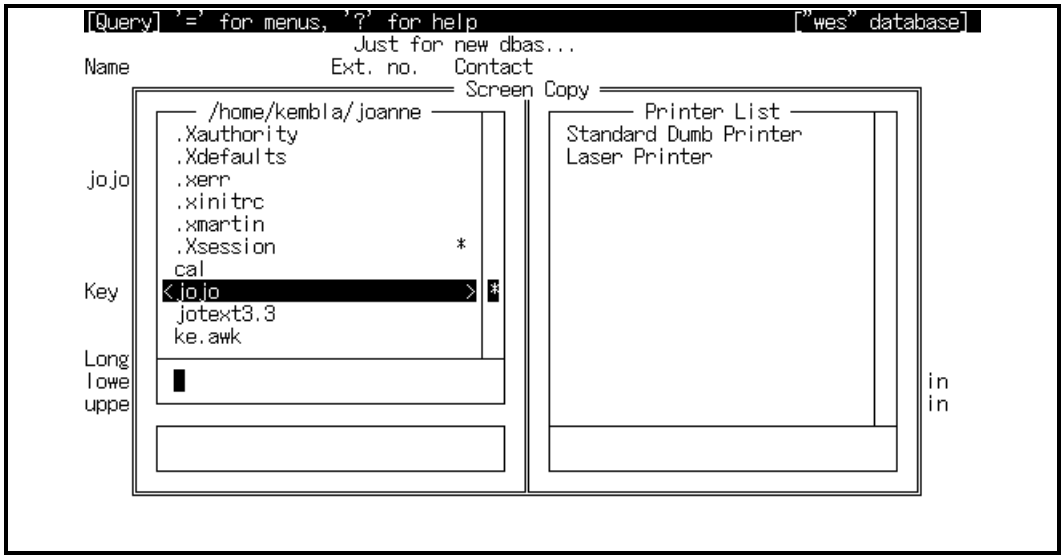
Look-up Table

Printing or Copying Information

Users can :

- Copy all information on the screen (Screen Copy), invoked by **Ctrl+G**
- Copy the current record in its entirety (**[Display] Show ⇒ Copy this match**). This is useful if the record spills over 1 screen.
- Copy all retrieved records (**[Display] Show ⇒ Copy all matches**).

Information can be copied into a file or sent directly to a printer. When any of these copy facilities are invoked, selection boxes of file names and printers will display as shown in the following figure:



Print and Copy Selection Boxes

The file selection box on the left consists of 3 sections. The top section displays all files in the current directory, including system files (starting with a . such as .cshrc). Please note that systems files should NEVER be selected for copies. Below the file section is a blank area where users can enter the leading character(s) of *existing* files, which will make the cursor jump to the first file starting with the character(s). The bottom -most section is another blank area, where users can enter the name of a *new* file.

The printer list box on the right displays all the valid printers. Users can scroll down the list by using the arrow keys ↓ or ↑, or by entering the leading character(s) of the printer in the blank section below the printer list.

Initially, the cursor will be inside the File box, in the New File Section. Here, users can enter a new file name, and when ↵ is entered, the file will be created with the information copied into it. The cursor can be moved to the Existing File Section, or the printer box by pressing the TAB key as often as necessary. Once the cursor is in the appropriate section, users can use the ↓

or ↑ key to move through the list, or enter the leading character(s) to make the cursor jump directly to the first item beginning with the character(s). Pressing **Ctrl+N** or **Ctrl+P** will scroll the list down or up.

To select a file or printer, move the cursor to it and press ↵. If an existing file is chosen, you will be prompted whether to append the information to the end of the file, overwrite the file with the information, or exit to abort the copy.

Screen Copy

A copy of the current screen image can be obtained at any time by typing **Ctrl+G** and then saved or printed using the facility described in the previous section. By default the complete screen image is copied (including the menu bar).

If the database option, **screencopy =short** is set, the information in the menu bar is not copied.

If the database option, **screencopy=block** is set, the user is requested to mark two points on the screen. All information bounded by these two points is then copied. During the marking procedure the message window displays:

```
(Mark top left corner)
```

or:

```
(Mark bottom right corner)
```

The available commands are:

Mark corner ([Screen Copy] Function ⇒ Mark)

Mark the current cursor position as a corner of the block.

Exit ([Screen Copy](Function ⇒ Exit)

Exit from the marking procedure and abort the screen copy command.

Move left ([Screen Copy] (Move ⇒ Move left)

Move the cursor left one character.

Move right ([Screen Copy] (Function ⇒ Move right)

Move the cursor right one character position.

Move up ([Screen Copy] (Function ⇒ Move up)

Move the cursor up one line.

Move down ([Screen Copy] (Function ⇒Move down)

Move the cursor down one line.

Each of the cursor movement commands may be preceded by a repeat count, indicating that the command should be performed multiple times.

E.g. **`2([Screen Copy] (Function ⇒ Move down)`** makes the cursor move down 2 lines. Also, if the upper case equivalent of the command (eg. H for left) is entered the cursor moves to the next tab stop in the appropriate direction. This provides a faster method of movement

around the screen. If an attempt is made to move the cursor outside the screen bounds, the terminal bell sounds.

Screen Redraw

Screen redrawing can be performed at any time by typing **Ctrl+R**.

This is useful if the screen happens to become jumbled, eg. if information is sent from another terminal to warn of impending database closure, or if the telephone line sends garbage over the modem.

Suspend or Invoke Shell

This is generated by the character, **Ctrl+Z**. On Unix systems with job control facilities, this signal suspends execution and returns control to the Unix command interpreter or shell. Execution recommences when the user enters **`fg.`**

For Unix systems without job control, this command causes a Unix shell to be executed. The shell to be executed is selected in the following manner:

(1) If the Unix environment variable, SHELL, is set then this program is used. For example, to execute the command interpreter, /bin/csh, the environment variable, SHELL, must be set to that value. This can be achieved from **csh** using the command:

```
setenv SHELL /bin/csh
```

or from **sh** using the commands:

```
SHELL=/bin/csh
export SHELL
```

(2) If the SHELL environment variable is not set, then the user's login shell in the password file is executed.

(3) If the password file has no shell entry for the current user, then **/bin/sh** is executed.

The termination of this shell results in a return to the exact context prior to the command being executed.

On systems with job control, the **Ctrl+Z** command takes effect immediately. On systems without job control, this command does not take effect immediately but rather is performed when the program is waiting for character input.

If usage of the suspend command is considered unsafe, it will be ignored by the system.

Interrupt

Typing the **Delete** or **Interupt** key interrupts the current operation and returns the user directly to the current menu or the shell prompt.

If usage of this interrupt facility is considered unsafe, the interrupt command will be ignored.

Text Perusal

Where large chunks of text have been split up into screen-sized records in KE Texpress, there is a text perusal facility, which allows the user to view the text consecutively, one screen at a time.

The user needs to first retrieve one or more KE Texpress records, and then while within Display mode, invoke this perusal facility **([Display] Function ⇒ Peruse records).**

The commands are as follows:

([Perusing] Move ⇒ Forward a record or Move ⇒ Move form down)

The next page of text is displayed. Moving forward from the last page of text terminates the Text perusal procedure.

([Perusing] Move ⇒ Backward a page or Move ⇒ Move form up)

The previous page of text is displayed.

([Perusing] Move ⇒ Copy text or Move ⇒ Copy a record)

Make a local copy of the text for the user's own convenience. The information may be copied into a Unix file or printed.

([Perusing] Function ⇒ Exit)

Terminate the display of text and return the user to the KE Texpress record.

Database Privileges

Privilege levels determine the operations and data available to the user. Pull-down menus reflect the users' privileges and hence users know only about operations which they can perform and data (fields or whole records) which they can view. Databases can be designed to have only one privilege level, which gives every user access to all the data.

The existence of privilege levels is generally hidden from the user.

Users with "Change record privilege" turned on in their user account, can alter a record's privilege level. To do this, the user must first retrieve the record(s), and then in display mode enter the new privilege level. **([Display]Edit ⇒ Adjust Privilege)**. This option is only available if the database has been configured to have more than one privilege level.

KE Texpress uses a network of privileges which allows a DBA to control access to data on several levels:

- Record
- Item
- Operation

To achieve this, KE Texpress uses the notion of privilege levels. The maximum number of privilege levels permitted is nine. However, the actual number of levels for a given database can be fewer and is determined at the time of database configuration (refer to Database Configuration). For a database with **n** privilege levels, the levels are numbered **0** to **n - 1**. These numbers represent ranks in a hierarchical structure where level 0 is more privileged than level 1, which, in turn, is more privileged than level 2, and so on.

The DBA assigns each new database user an insertion and display privilege level which determines access to information whenever the database is accessed.

Record Privileges

At the time of insertion, each new record is assigned a privilege level according to the insertion privilege of the user. Following this, only users with a display privilege level higher than or equivalent to the record's privilege level can retrieve and view that record when querying. In other words, for insufficiently privileged users, this record is invisible.

Item Privileges

When the database is designed (refer to Database Design in the Design Guide), each object in both the Insertion and the Query Forms is assigned a privilege level. Only items, headers and boxes with a privilege level lower than or equivalent to the user's privilege level appear on the screen whenever a form is displayed. Thus selected data within a record can be protected while the user is still allowed access to some of the information in the record.

A user's insertion privilege level determines which fields can be accessed while inserting or editing a record. A user's display privilege level determines the data which can be viewed while displaying matching records. Therefore, a user may be privileged to view certain information but not sufficiently privileged to edit that information.

Operation Privileges

There exists for each user another set of privileges based on the operations that the user is permitted to perform. These privileges are not hierarchical in structure, but are simply granted or denied. The user is permitted to perform the operation only if the privilege is granted.

The user level operations controlled by these privileges include:

- Record insertion (I)
- Record edit (E)
- Record deletion (D)
- Change record privilege level (P)
- Query (Y)
- Destructive edit (U)
- Validation override (V)
- Videodisc access (X)
- Temporary record creation (T)
- Temporary file summary (F)
- Temporary record retrieval (R)
- Report access (C)
- Report create / delete (K)
- Unix facilities (N)
- Set user options (S)

A Look-up table insertion privilege also exists and can be granted or denied individually for each Look-up table in the database.

In addition to this set of privileges, there are administrator privileges which control access to the maintenance operations of the database.

Finally, there is the DBA privilege which cannot be granted to other users but is solely that of the DBA. All privileges, other than the DBA privilege, can be either granted or denied to all registered users of the database, and are controlled by the DBA.

Chapter 3

Entering Information

Data Types....3-3
Movement around a Form....3-4
Entering Information into Fields....3-5
Look-up Tables....3-6
Hierarchies....3-8
Miscellaneous Commands....3-9
Field Editing Commands....3-11

Overview

All operations in **texforms**, eg. queries, edits, inserts, report generation etc., involve entering information into fields on various forms.

This chapter describes how to move around a form and enter information into fields.

Data Types

During data entry, the type of the field imposes restrictions on the characters accepted for data entry. If an attempt is made to enter a character not acceptable to the data type for that field, the terminal bell sounds and the character is ignored.

The data types supported include string, text, integer, real and file name.

The string data type accepts all printable characters (letters, numbers and punctuation).

The text data type accepts the same range of characters as the string type but interprets the data as individual words (separated by normal delimiter characters such as space, comma, full stop, etc.) rather than as a single string.

The integer data type accepts only data in the format:

An optional sign ('+' or '-') followed by zero or more digits.

The real data type accepts only data in the format:

An optional sign, followed by zero or more digits, optionally followed by a decimal point ('.') and zero or more digits.

The file name data type accepts text with the exception of the space character which is illegal. In addition to this, if the first character of the file name is a tilde (~) then a tilde expansion takes place. Thus, the strings, ~ and ~/, expand to the home directory of the current texpress. The string ~texpress expands to the home directory of the Unix **texpress** account.

Movement around a Form

The following commands move the cursor to fillable fields on the form. A field is defined as fillable if it can be reached and data can be entered into it. Automatically filled fields cannot be accessed by the cursor.

Note that the left and right arrow keys (→ and ←) can only be used to move to the adjacent fields on the form.

Next field (Move ⇒ Forward a field)

Move forward to the next fillable field which may be on the same item. If the cursor was at the last field on the form, move to the first field.

Previous a field (Move ⇒ Backward a field)

Move backwards to the previous fillable field which maybe on the same item. If the cursor was at the first field of the form, then move to the last field.

Next item (Move ⇒ Forward an item)

Move forward to the next fillable item. The cursor is positioned on the first fillable field of that item.

Previous item (Move ⇒ Backward an item)

Move backward to the previous fillable item. The cursor is positioned on the first fillable field of that item.

First field (Move ⇒ Start of form)

Move to the beginning of the form. The cursor is placed on the first field of the first fillable item on the form.

Last field (Move ⇒ End of form)

Move to the end of the form. The cursor is placed on the first field of the last fillable item on the form.

Move to item (Move ⇒ Find an item)

This command calls up a box, into which the user can enter the prompt or field name (or just its leading letters), which will take the cursor directly to the specified field. This is handy particularly when the field is on another screen.

Entering Information into Fields

As soon as a character is typed into a field, any previous information in the field is replaced by the new entry, and a line of underscores displays, outlining the width of the field. The menu bar will display the message:

```
"Enter ESC or <Return> to terminate character insertion"
```

Once entry has started, KE Texpress will ignore any commands, eg. to exit or perform a query, until ↵ or **ESC** is entered. The cursor will then jump to the next field on the form.

Note that *numeric* fields will only accept numeric data, whereas text fields will accept any kind of data.

Date fields often consist of 3 components, for date, month and year. Press ↵ after entering each component. The field lengths of each component will indicate the expected formats, eg. a length of 3 for month indicates "*jan*" is expected, whereas a length of 2 expects "*1*" or "*01*". Often messages will display on the menu bar as soon as data is entered into each component to specify the format, eg. Month (Jan, Feb).

Clear field (Edit ⇒ Clear text)

This command clears all characters in the current field.

Clear form (Misc ⇒ Clear form)

Data is cleared from every fillable field in the form. The cursor then moves to the first fillable field on the form.

Undo (Edit ⇒ Undo)

This command undoes the last change made anywhere on the form. If you had started entering information and wanted to undo this, you need to first finish data-entry by pressing ↵ or **ESC**, and then select this command.

Edit Field

See Field Editing Commands in this chapter.

Look-up Tables (Info ⇒ Look-up table)

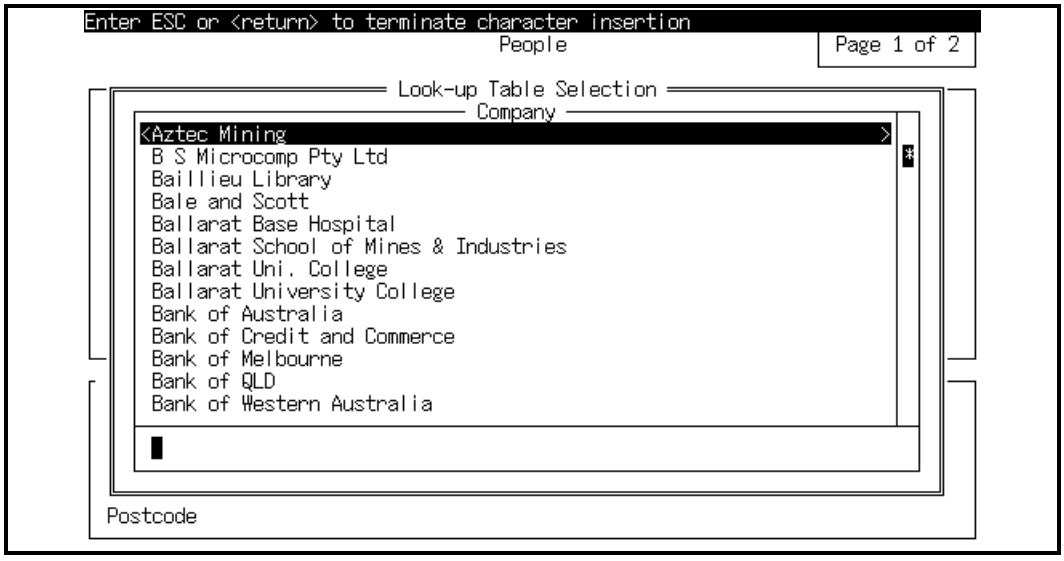
Fields which contain a restricted set of information may have Look-up Tables which are lists of all entries previously entered into the system at that field. During queries, inserts and edits, the user can choose from this list to save typing all the information into the field and this also eliminates the risk of making typos.

When used during record insertions and edits, Look-up Tables keep the format of entered information consistent across users, eg. "U.S.A" instead of "the United States of America".

When used during queries, they ensure the query term matches the format followed in the record and avoid non-retrievals caused by the query term's format being different from the data's. eg. querying on "*uk*" in the Country field will not retrieve records with "*britain*" in the same field.

Users can call up the entire Look-up Table for a field or call up a partial Look-up Table by first entering a few leading letters and then invoking the Look-up Table. If there are no Look-up Tables for this field, there will be a beep, and nothing will happen. If there is only one retrieved value, it will be automatically inserted into the field.

If there is more than one value, a list will display, as shown in the next figure.



Look-up Tables

One entry in the Look-up Table is highlighted. To the right of the list is a vertical *scroll bar*. If the entire list cannot be displayed in the selection box at one time, an asterisk (*) character is placed in the scroll bar to indicate which portion of the list is currently displayed.

Using a Look-up Table is similar to using a front-end menu. The ↑ and ↓ keys may be used to move up or down 1 entry in the list, causing the list to scroll if necessary. Entering **Ctrl+N** or **Ctrl+P** will scroll 1 screen forward or backward. Alternatively, an entry may be located by typing 1 or more of the leading letters in the entry. The letter(s) which have been typed will display below the selection list and the first entry in the list which starts with the letter(s) will become highlighted.

Once the required entry has been located and highlighted, press ↵ to select the entry and return to the screen.

To exit from the Look-up Table without selecting an entry, enter **F1**, **x** or **Ctrl+X**.

Hierarchies

A group of logically connected items which have a natural hierarchical relationship between them can be designated to form a **hierarchy**. This hierarchy becomes a special set of inter-related Look-up tables. Examples of possible hierarchies are:

- Family Genus Species Sub-species
- Country State Town

In such a hierarchy, the range of values of the second item is dependent on the value of the first item and the range of the third item is dependent on the values of both the first and second items. Thus while there are many valid *State* names throughout the world, the number of states where *Country* is **Australia** is only eight. This inter-dependency can make selection from a Look-up table much simpler.

For example, in a large database, a request for the list of known *Towns* could result in many thousands of values. Specifying towns beginning with "**m**" could reduce this list to hundreds. However, if the Town item were part of the hierarchy:

```
Country      State        Town
```

and the request is for towns beginning with "**m**" where *Country* is **Australia** and *State* is **Victoria**, then results in a far smaller list of possible Town names.

It should be noted that hierarchies work in both directions. Thus, a value in the *State* item will reduce the possible selections for countries (usually to a single value).

If a list of values is requested for an item in a hierarchy where at least one of the items above it in the hierarchy is empty, then the hierarchy becomes invalid. In this case, the action defaults to that of a simple Look-up table, and all inter-item dependencies are ignored.

Miscellaneous Commands

Link database (Info ⇒ Link database)

This command is available only if the database has been designed to extract information from linked databases during insertion.

It enables the user to view the information stored in a linked database, and is used when it is desirable to extract information from the linked database but the Key value of the linked record is not known. The cursor must first be positioned on a field of the item which represents the Key of the linked database, and the user must have an account for the linked database, i.e. be allowed to use it. The following message will display:

Executing *dbname* . . .

(where *dbname* is the name of the linked database).

On exiting from the execution of this linked database program, texforms the user returns to the current database.

Show attributes (Info ⇒ Show attributes)

Display a description of the item prompt, its coordinates, the type of retrieval method applied to it, if any, eg. whether by Key value or indexing, the ranging type (for ranged items), and whether there is a Look-up table.

For each field of the item, the co-ordinates, the length, the type of data accepted by the field, the source of the data (with the full assignment expression which is used to calculate its value, if one is used), the full validation expression for the field if one exists, its branch expression if applicable, its default value if one exists, and if the data for this field is to come from a linked database, the name of the linked database and the field within that database are displayed. All of these attributes are set-up by the database designer.

To return to the database form, press the space bar or (**Function ⇒ exit**)

Display Field Edit message (Info ⇒ Edit message)

This command provides a method of viewing the field edit message without having to enter information into the field. If the current field does not have a field edit message then the following message will display:

```
No field edit message for this field
```

Field Editing Commands

Typing new information into a line of a field replaces any information that was previously in line. To edit the existing contents of a line, these commands are used:

Move Left (Edit ⇒ Move left)

This command moves the cursor one character position to the left, unless it is already at the start of the line.

Move right (Edit ⇒ Move right)

This command moves the cursor one character position to the right, unless it is already on the last charcater in that line.

Insert character(s) (Edit ⇒ Insert)

This command allows the user to insert characters before the character currently under the cursor. Characters are inserted until the user types

esc or ↵.

Append Character(s) (Edit ⇒ Append)

This command allows the suer to append characters after the character currently under the cursor. Characters are added until the user types

esc or ↵.

Delete Character (Edit ⇒ Delete)

This command removes the character under the cursor.

Replace Character (Edit ⇒ Overstrike)

This command replaces the character currently under the cursor with the next character typed by the user.

Chapter 4

Record Insertion

Key Values 4-3
Key Value Assignment Database Options 4-3
Ditto Commands 4-5
Default Field Values....4-6
Comments 4-7
Assignment Expressions 4-8
Linked Databases 4-9
Data Validation 4-10
Key Item....4-10
Hierarchy....4-10
Look-up Table....4-11
Field 4-12
Ending an Insertion 4-15
Save....4-15
Temp Save....4-17
Exit....4-18

Overview

Records are created by entering information into fields on an **Insertion** form. If the entered data pass the pre-defined validation criteria, such as mandatory fields are filled in, the user can save the record, which is inserted into the database and it becames available to all users for queries and edits.

Only users with temporary record creation privilege or record insertion privilege (set up in their user accounts) may call up the insertion form and thereby create records.

The insertion form can be called up from any other form. Its mode indicator is set to Insert and some data may already be filled into this new record. This data can arise from either default or automatically generated values.

A default field value is a character string which is initially assigned to a field for each new insertion. This character string is intended to represent the most common value which may be assumed by that field, and can be changed at any time during data entry. See also Inserting Default Values in this Chapter.

Some items in a form can be designed to be automatic items i.e. have their field values automatically generated by KE Texpress and they cannot be directly changed by the user who is prevented from accessing these fields.

Only KE Texpress Library items may be designated as being automatic. See also Appendix C of this guide.

For details on moving around the form, using lookup tables and editing data in fields, see Chapter 2 of this guide.

Key Values

Many records have a Key item, which uniquely identifies the record. KE Texpress can maintain a list of Key item values, assign them automatically and ensure their uniqueness.

A Key item is not compulsory. If a database does not have a Key item, then ignore the following paragraphs.

Key types must be defined for each key item. The DBA can design new key types or use pre-designed ones, which are described in Appendix B of this guide.

For databases which have Key items, when each new insertion is commenced three forms of Key value assignment are possible. They are set up as database options. See Chapter 7 for more details.

Key Value Assignment Database Options

autokey=next

The (smallest) next available Key value is automatically assigned to the new record. This option is available only for Key types which contain an incrementing component. For Key types with more than one component, the same prefix (all components other than the incrementing component) as the last Key value assigned to the user is assigned. If no Key value has been assigned in this session in this session, the following message displays and the key item is left empty:

```
No appropriate next Key value available
```

autokey=personal

The Key value which is one larger than the Key assigned to the user's previous insertion, is automatically assigned to the new insertion provided that this value is available. If it is not available, no automatic Key value assignment is performed and a message displays:

```
Next Key value is already assigned
```

This option is intended to be used for entering large backlogs of data. In this case, each new record inserted by a user is the numeric successor of that user's previous insertion. This allows other users to insert records concurrently, without having to wait for the current user to save the record and thus free up the next available key value.

This option is available for incrementing Key types only.

autokey=no

No Key value is automatically assigned to the new record. The value must be entered manually. This option can be used for entering data with a random selection of Key values.

If **autokey=next** or **autokey=personal** is set as a database option, users can obtain the next available key item value when inserting records **([Insert] Info ⇒ Get next key)**. This requests the system to insert the next available key value into the record if one has not already been entered.

Ditto Commands

After the Insertion form has been displayed and all of the default, automatic and any Key items have been filled, the information describing the record can be entered.

To save the user from entering data identical to that entered previously for other records, **ditto** commands are available while inserting (and editing) records.

Several additional editing commands are available during Insert mode to make data capture simpler and more efficient.

The record which contains the information to be copied is called the **ditto record**. By default, the ditto record is the user's last record inserted or edited. As soon as the user exits texforms, the ditto record is no longer set. Users can also choose the ditto record in Insert, Edit or Display modes. See also Miscellaneous Commands in Chapter 5 of this guide.

Ditto ([Edit, Insert] Misc ⇒ Ditto)

Copy contents of current field from ditto record.

Ditto all ([Edit, Insert] Misc ⇒ Ditto all)

This command copies information into all empty fields in the insertion form from equivalent fields in the ditto record.

Change ditto ([Edit, Insert] Misc ⇒ Change ditto)

Change the ditto record. The following message appears:

```
Enter Ditto Key field n :
```

(where ***n*** is the number of the field currently being entered). The user can either exit by entering ↵ or enter the key of the new ditto record, which is then used for subsequent ditto commands. If there is no record associated with the Key value entered, the ditto record is not changed and a message displays:

```
Invalid Key value (Ditto unchanged)
```

Default Field Values

The DBA can set up default field values for each field, which represent the most commonly entered values for those fields. Users can then insert the default value for the current field or for all fields. These default field values can then be changed at any time during data-entry by the user.

Insert default ([Edit, Insert] Misc ⇒ Load default)

Insert the default value as set up by the DBA, into the current field. If the default value is a null string, the contents of the field is cleared. (To replace the previous value, use the undo command).

Insert all defaults ([Edit, Insert] Misc ⇒ Load all defaults)

Insert all default values into the appropriate fields. To reverse the result, use the **undo** command.

Comments

Every record inserted into the database can be associated with unlimited free-form comments which are independent of any data in the form. As comments are not indexed, retrieval on comments alone is slower than retrieval using indexed terms. Comments for each record are stored in the /tmp directory.

Insert Edit comments ([Insert] Misc ⇒ Edit Comments)

Add or Edit the comments for a record. This invokes a Unix text editor which is used to enter or edit the comments. The text editor is selected in the following manner:

(1) If the Unix environment variable, **EDITOR**, is set then this program is used. This variable can be set from **csh** using the command:

setenv EDITOR "*editor path-name*"

The **sh** equivalent is:

EDITOR="*editor path-name*"
export EDITOR

(2) If the EDITOR environment variable is not set then the Unix visual editor, **vi**, is used.

(3) If **vi** cannot be found, then the Unix line editor, **ed**, is used.

Exiting from comments editing returns the user to the Insertion form. When the record is written or saved, the updated comments are also saved.

Assignment Expressions

It is possible to design a field of the Insertion form so that its value is generated automatically. The value for such a field may be calculated by arithmetic expressions, possibly involving some pre-defined functions. It may also be calculated based on values in other fields on the Insertion form. Thus whenever one of these other fields is altered, the expression is automatically recalculated.

The result of this is that changing one field can have an effect on other fields in the form. For example, if a Degrees Celsius field is calculated as a function of the Degrees Fahrenheit field using the formula:

$$°C = \frac{(°F - 32) * 5}{9}$$

then if the value **68** is entered in the Degrees Fahrenheit field, the value **20** is automatically calculated and immediately entered into the field for Degrees Celsius.

A numeric expression involving other fields is not calculated if all of the other fields are empty. If at least one of the other fields has a value, then all remaining empty fields assume a value of zero for the calculation of the expression.

(See Appendix F of the Design Guide for examples of assignment expressions)

Linked Databases

The Insertion form of a database can be designed to extract information from other databases, by specifying the Key value of a record in the linked database. The Insertion form for the current database must have an item which corresponds to the Key of the linked database, and when a valid Key value is entered into this item, then the information from the relevant record in the linked database is automatically copied into the linked fields of the current record, overwriting any data in those fields.

After the copy is completed, these fields cannot be accessed while the linked database Key value is present. If, however, this Key value is removed, then the data in the linked fields will become accessible to the user.

If the value entered into the linked Key item represents an invalid or unused Key value for the linked database, all of the linked fields will be cleared, and will be inaccesible until the linked Key value is removed from the current database.

By default, the information is extracted from the linked database once and is copied into the current record. Thus later altering records in the linked database will not alter the copied information in the current record.

Note that the linked Key item can be defined as a Display Only linked Key, in which case the linked information is not stored with the record but rather is extracted from the linked database every time the record is accessed. This enables the current database to reflect up-to-date information in the linked database, which acts as a Master list.

Data Validation

Each field of the Insertion form can have its own validation criteria which may be a function of its own value or dependent on the values of other fields of the form.

When the user saves the record, the system also checks additional criteria, and will only save the record if it passes these tests. (See below). Prior to saving the record, users can at any time request the system to validate the record.

Validate data ([Insert] Info ⇒ Validate data)

Users can run this command to make the system validate the current record. This indicates whether a record is valid, and if invalid, indicates the field values which do not meet the validation criteria.

The following sections describe the tests performed by the validate command.

Key Item

The Key item, if present, must have a valid Key value. If this test fails, a message will display:

```
Invalid Key value
```

The validation command check will abort, with a return to Edit mode, and the cursor will sit on at the first field of the Key item to prompt the user to edit it.

Hierarchy

If a value has been entered for an item in a hierarchy, then all items above it in that hierarchy must also have values. So for the hierarchy:

Family Genus Species Sub-species

if the Species item is filled, then both the Family and Genus items must have values. Sub-species, being below Species in the hierarchy, need not have a value. If this test should fail then a message will display:

```
Invalid hierarchy on "Fieldname". Correct entry or
Override validation?
```

Choosing Correct Entry will take the cursor is placed at the start of the prompt of the offending hierarchy item.

Choosing Override Validation is possible only for users with Validation override privilege. This causes the validation procedure to ignore this item, and allows users to go on and save the record.

Hierarchy validation also ensures that only one field component can be entered into any field of a hierarchy item. A field component is defined as a character string terminated by the end of the field or by a semicolon (**;**). If more than one field component is entered into a field of a hierarchy item, the message will display:

```
Only one field component per hierarchy item - Hit any key
to continue
```

The cursor will then sit on the first field of the invalid item.

Look-up Table

Before new entries can be inserted into Look-up tables, the system first checks that the user has Look-up table insertion privilege. This privilege can be set for each Look-up table in the database. Thus a user may be permitted to insert into one Look-up table while being denied permission to insert into another.

If a field of an item with a Look-up table contains a value which is not currently in the Look-up table and the user is not permitted to insert into that Look-up table, then a message will display:

```
Invalid entry. Choose a value from Look-up table
```

The validation command will abort, and the cursor will be taken to the first field of the offending item.

If the user can insert into the Look-up table, a message will display:

```
"string" not in Look-up table! Insert? [yes] [no]
```

(where ***string*** is the value of the field). Choosing yes causes the new value to be inserted into the Look-up table. Choosing no returns the user to the offending item, providing the user with the opportunity to re-enter or select a new value for this field.

For hierarchies, rather than testing each value separately, the system checks the existence of the combinations in the hierarchy Look-up tables. For example, for the hierarchy:

Country State Town

with data:

Australia Victoria Melbourne

the combinations:

Australia
Victoria **in** Australia
Melbourne **in** Victoria **in** Australia

are tested for existence. If a combination does not exist then the same errors as above are generated, and the user can choose to insert the combination into the look-up table hierarchy or amend it.

Field

The DBA can set up validation criteria for each field in the record. These criteria are expressions in a simple mathematical notation which is similar in appearance to the C programming language. These expressions may assign to a variable called **val**, a result of either zero which represents an invalid value, or non-zero which represents a valid value. In these expressions, the special character, #, represents the current field and **$** represents the current item. Character strings not in quotes are system identifiers of variables or other items in the form. The n^{th} field of an item is referenced by adding **_n** to the name of the item. So the second field of the current item can be referenced by **$_2**.

Some of the operations and functions supported for such expressions are listed below:

=	assignment
==	equal to
!=	not equal to
<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to
+	addition
-	subtraction
*	multiplication
/	division
%	modulus operator
.	string concatenation
&&	logical conjunctive operator (and)

|| logical disjunctive operator (or)
! logical negation operator (not)
empty string empty test
filled string not empty test
if **(**condition**)**
statement1**;**
else
statement2**;**

Some examples of validation expressions are:

val = filled(#);

The current field is compulsory.

if (filled(#))
val = (# > 3000) && (# < 4000);

If the current field is filled then its value must be between 3000 and 4000 exclusively.

if (filled($_1))
val = filled(#) && (# >= 0) && (# < 60);

If the first field of this item has been filled then the current field must also be filled and have a value not less than zero and less than sixty.

For a detailed discussion of validation expressions (and expressions in general), refer to the KE Texpress Design Guide.

If the validation test for a field should fail (it returns a value of zero), a message will display:

```
Invalid entry. View validation or correct entry?
```

The possible responses are:

Correct The user is returned to the first field of the offending item to change it.

View The screen is cleared and the offending field and its validation expression is displayed The validation expression is an expression formed using the rules as described above. Entering ↵ returns the user to the first field of the offending item to change it.

Override The validation criterion is overridden for this field and this field value is then accepted as valid. This option is available only to users with Validation override privilege. Validation cannot be over-ridden for records which have a key value that is identical to

an existing record's key in the database. In such a case, a message will display:

```
Key value is already assigned. Can't override validation
```

Ending an Insertion

To end an insertion, the user must exit from the insertion form. This can be achieved in several ways by saving the record, saving it in a temporary file, or exiting without saving.

Save ([Insert] Function ⇒ Save)

This is the only way to insert a record into the database, and the user must have record insertion privilege to do this. The information in the form is first validated (as when the command, **Validate data** is selected). If this should fail then the user is returned to Insert mode. Otherwise, the record insertion is performed with the message:

```
Performing insertion . . .
```

The user can then commence another insertion.

Part of the record insertion involves inserting any new values into the Look-up tables (or combinations into hierarchies). This is achieved silently and automatically by the system unless **User specified** is set on the field during the design of the database. This allows users to position new entries in the look-up table, and thus choose the display order of the look-up table. If **user specified** is not set, the look-up table is displayed alphabetically by the system, and the user cannot view it in any other order.

If **user specified** is set, saving the record will cause a selection box to display, listing all previously entered look-up table values. To position the new entry immediately prior to an existing entry, select the existing entry by pressing ↵. To position it after the last entry in the list, exit (without selecting) by entering **`Ctrl+X`** or **`F1`**.

A message box will pop up and ask for confirmation. Choosing yes will insert the new entry into the nominated position in the look-up table. Choosing no will take the user back to the selection box displaying the previously entered look-up table values.

If two users should happen to try to insert into the same Look-up table at the same time, then only one user (the first) will be allowed to. The second user will be suspended until the first has completed the insertion. A message will display:

```
Look-up table busy - will retry in one minute . . .
```

After 60 seconds, texforms checks to see if the first user has completed the Look-up table insertion. If so then the second user can continue. If not, the second user is suspended for another minute.

During the suspension, the second users may press the DEL (interrupt) key, to insert the record into the database and abort the attempt to insert the new

value into the Look-up table. This value can be inserted into the Look-up table the next time it is entered in the field and the record is saved. It can also be added directly to the table by the DBA, using the Look-up table edit facility (refer to the KE Texpress Maintenance Guide).

Temp Save ([Insert] Function ⇒ Temporary save)

This saves the record in a temporary file and is useful if the user has entered only part of the information for a record and does not wish to store it in the database until the information is complete. This option is available only if the user has Temporary record creation privilege, the database has a Key item and the Key item contains a valid Key value. The information is not validated and can be retrieved only once, via its Key value. Following a temporary save, the user can commence another insertion.

The information, temporarily, is retrievable only to the user who stored it and to all users with Temporary record retrieval privilege. It is retrieved by calling up an insertion form and enetering its key value in the insertion form's key item. This deletes the temporary file, so the user must either save the record, or temporarily save it again if the information is to be retained. The user must also have Record insertion or Temporary record creation privilege to retrieve from the temporary file.

Users with temporary file summary privilege can produce a summary of Key values of all records currently in the temporary file. Insertion form **([Insert]Info ⇒ list temp records)**

If the temporary file is empty, a message displays:

```
No records in temporary file
```

Otherwise the screen is cleared and a list is displayed with the heading:

```
Key values and owners of records in the temporary file
```

The order of the Key values is the order in which the associated records were placed in the temporary file. The oldest record's Key value appears first. The total list of Key values is viewed using the Text perusal facility.

Exit ([Insert] Function ⇒ Exit)

This exits the insertion without saving the record and returns the user to the display form. If the information in the Insertion form has been modified, the message:

```
Exit from Insert mode? [yes] [no]
```

is displayed, seeking confirmation of the command. Choosing no results in a resumption of Insert mode.

Chapter 5

Database Querying

The Query Form 5-4
Text Queries 5-5
Word Break Characters 5-5
Noise Words 5-6
Query Commands 5-7
Query Comments 5-7
Exit 5-7
Clear Form 5-7
Single Term Query 5-7
Or Query 5-7
And Query 5-8
Phonetic Search 5-8
Stemming Search 5-8
Phrase Query 5-9
Case Search 5-9
Range Queries 5-9
Key Value Queries 5-9
Key Value Range Queries 5-10
Pattern Matching Facilities 5-11
! (NOT) 5-11
^ (Start) 5-11
$ (End) 5-12
? (Single character) 5-12
* (wildcard) 5-12
[str] (single character range) 5-12
{str} (repeating character range) 5-12
x (where 'x' is any character) 5-12
(literal next) 5-13
Examples 5-13
Perform Query 5-14
The Display Form 5-15
Display Mode Commands 5-15
Exit 5-15
Movement 5-15
Among Retrieved Records 5-15
Within a Record 5-16
View comments 5-16
Select Display Form 5-17

Record Histories 5-18
Show History 5-18
View Later Version 5-18
View Earlier Version 5-18
Change Display Form 5-18
Delete History 5-18
Exit 5-18
Report Generation 5-19
Report this match 5-19
Report all matches 5-19
Peruse records 5-20
Jump to Record 5-21
Edit Records 5-22
Current record 5-22
All retrieved records 5-24
Delete Records 5-26
Current record 5-26
All Retrieved Records 5-27
Record Privilege Level Commands 5-28
Show Privilege Level 5-28
Change Privilege Level 5-28
Re-order Records 5-29
Current Matches 5-29
Select item/field 5-29
For Subsequent Queries 5-30
Record Copy or Print 5-31
Current Record 5-31
All Retrieved Records 5-31
Miscellaneous Commands 5-32
Verify Matches 5-32
Set Ditto Record 5-32
Save a Dave File 5-33
Discard Record From Retrieval 5-33
Show Video 5-33
Interpret Form Expression 5-33
Additional Query Commands 5-34
Perform Separate Query 5-34
Combine Queries 5-34
Disappearing Matches 5-35

Overview

The principal operation performed by KE Texpress is that of information retrieval. Information stored in the database via the Record Insertion mechanism can be retrieved, formatted, copied and displayed using the **texforms** Query facility **([Query] Function ⇒ Query)**. Query mode is then entered. However, the user must have Database Query (set up in their user account) privilege in order to perform queries. For more information on record and field privileges, see Database Privileges in Chapter 2.

The basic steps involved in performing a query are:

Entering query criteria

The information defining the query is entered into the Query form. See also Chapter 3 of this guide for instructions on how to move around the form and enter information into fields.

Performing the query

The query is performed and all records in the database which match the information defining the query and which the user is privileged to view are retrieved.

Displaying the retrieved records

The matching records are displayed and can now be stored, copied, edited, re-ordered, reported on and deleted.

The following sections describe each of these steps in detail.

The Query Form

The Query form typically contains all of the Insertion form items which are designed to be retrievable. This includes all numerically ranged items, the Key item if one exists, and all indexed items. Unindexed items may appear on the Query form and may be used in a query. However, query response will be much slower when using such items. (To improve query response, users should include at least one indexed term in the query).

Some minor distinctions between querying and insertion are worth noting:

(1) No automatic Key value assignment is performed.

(2) There are no automatically filled items.

Generally, every item on the Query form which can be seen by the user is fillable. One exception to this rule involves upper and lower bound items. If a ranged item is queried on an exact value, then querying on the associated bounds as well in the same query do not make sense. So the upper and lower bounds are unfillable if an exact value has been entered into the item. See also Chapter 3 of this guide for information on moving around the form and entering information into the fields (including the use of look-up tables).

Text Queries

Word Break Characters

Text fields contain alphabetic characters and/or numbers. Each word is a separate indexed term and is indexed unless it is a noise word. Words are defined as a string of letters and numbers terminated by the end of the field marker or by a word break character. Word break characters in query terms are ignored by KE Texpress.

The default set of word break characters includes space, full stop, comma, colon, semi-colon, slash ('/'), round brackets, ampersand ('**&**') and double quotes ('"'). This set can be changed by the DBA if required.

In text fields, punctuation is ignored, except where it serves to split words. So, for example, the text:

```
cash redistribution
```

and:

```
CASH,Re-Distribution.
```

are regarded as equivalent (given that a comma is a word separator but hyphen is not).

Noise Words

Noise words are words which are deemed unlikely to distinguish a field. They are not indexed and so are ignored by KE Texpress if they are specified as a query term. Noise words include "if", "it", "and", "at" etc. Noise words are stored in the file :

~texpress/texpress5.0/lang/english/noise

The default set of noise words is given below. It can be tailored by the DBA to meet the needs of a particular application.

a	be	of	very
about	because	or	was
all	but	so	were
also	by	than	
although	for	that	
an	here	the	
and	if	then	
another	in	there	
any	into	this	
are	is	though	
as	it	to	
at	its	too	

Query Commands

Query Comments ([Query] Misc ⇒ Query Comments)

For databases which have commands appended to records (see Chapter 4 of this guide for information on inserting comments), queries on comments can be performed. Note that as comments are not indexed, retrieval is slow.

This command calls up a message box into which the user can add up to one line of terms, which are then matched against the comments.

To exit from the message box without performing the comments query, do not enter any terms, and press ↵.

Exit ([Query] Function ⇒ Exit)

If pull-down menus are exposed, this command will close them down. Otherwise, it exits the query screen without performing the query and returns the user to the last screen or front-end menu from which s(he) came.

If information had been entered into the query form but a query was not performed, the system will first ask for confirmation:

```
Exit from Query mode ?
```

Choosing no will abort the exit and the query form will remain displayed on the screen.

Clear Form ([Query] Misc ⇒ Clear form)

This command clears all information previously entered into fields on all screens in the query form and places the cursor on the first field.

Single Term Query

Only one field on the query form is filled in. For example, if only the word *boy* is entered into a **Title** field, when the query is performed, KE Texpress will retrieve all records in the catalogue which contain the word *boy* somewhere in the Title field, such as *Boy with pail*, *African boy* etc.

Or Query

Entering information on different lines of the same field on the query form will retrieve all records which contain information specified on any of those lines. For example, if *Boyd* is entered on the first line of the **Artist Surname** field on the query form and Nolan is entered on the second line of the same field, the system will retrieve all works by either *Boyd* or *Nolan*.

And Query

If more than one word is entered on the same line of a field the search will retrieve only those records which contain all of the words in that field, regardless of the order. For example, if the words *boy* and *girl* are both entered on the same line of the **Title** field, i.e.

```
boy girl
```

Ke Texpress will find all records which contain both of these words somewhere in their title field, such as works entitled *Boy and girl* as well as *Girl and boy in Fitzroy*.

Entering information in more than one field on the query form also specifies an AND query, ie. KE Texpress retrieves records which contain all the specified information. For example, if *boy* is entered in the **Title** field and *Boyd* in the **Artist Surname** field, all works by *Boyd* and with *boy* somewhere in the title will be retrieved.

Unlike every other database system, KE Texpress retrieves records faster the more detailed the query. Users are thus encouraged to specify as much information in the query for faster retrieval and to cut down on the number of records retrieved.

Phonetic Search

Ke Texpress supports phonetic or Sounds-like retrieval on fields which are likely to contain names, such as a **Surname** field. By preceding any word in these fields by \@ (which will automatically delete the \), the search for that word is performed phonetically, thus matching words which sound similar. For example, *@smith* finds *Smith*, *Smythe* and *Schmidt***.**

Stemming Search

In text fields, retrieval on the stems or derivatives of words are available, especially the plural and singular forms and different syntaxes of the word (such as its noun, verb and adverb, etc.). When a ~ (tilde) precedes any word in the query form, the search for the stem of that word is performed. E.g. *~country* retrieves *country* and *countries*. Similarly, *~japan* finds *Japan*, *Japan's* and *Japanese*.

Note that stemming is a complex algorithm which is more intelligent than simple truncation of words. E.g. *~elect* retrieves *elects*, *electing*, *election*, etc. but not *electric* or its variation.

Phrase Query

If more than one word is entered on one line of a field, by default, the system perfoms an AND query, i.e. it searches for the specified terms in any order in the field. To specify a sequence of terms where these terms must appear consecutively in the matching field, users must enclose the terms with double quotes. For example, "*girl and boy*" will locate *Girl and boy in Fitzroy* but not *Boy and girl*. Note that phrase retrieval can be combined with the other search operators. For example, the search phrase "*@africa boy*" retrieves *African boy*.

Case Search

The default query operation ignores the case of the query terms, whether in upper or lower case, so that querying on *boy* retrieves *Boy*, *boy* and *BOY*.

To make the case significant, precede the appropriate query term with the = character. E.g. =*BOY* will not retrieve *BOY* but not retrieve *boy* or *Boy*.

Range Queries

This form of query involves specifying the lower and or upper bounds of a numeric item.

If only a lower bound is entered, then KE Texpress retrieves all records for which the item has a numeric value greater than or equal to that value. If only an upper bound is specified in the query, then KE Texpress retrieves all records for which the item has a value less than or equal to the given value.

Specifying both an upper and a lower bound in the same query retrieves records with values for the item between the two bounds given (inclusively).

Range queries may be slower than other forms of query. To improve retrieval speed, the DBA should define the range of items, so that they represent the most common query bounds, if these are known, or so that the population of records is evenly distributed across regions.

Key Value Queries

The most efficient form of query queries on the Key value of a record. As Key values are unique amongst all records, a query involving a Key value must result in either zero matches or exactly one match. This can be determined almost immediately.

If a Key value is specified in a query, no other information is required to determine the matching record (if one exists). However, if more information is specified as part of the query then all of the information is used to determine whether a matching record exists.

Key Value Range Queries

If a lower and an upper bound for the Key value are entered into the Query form, then all records with values between those bounds (inclusively) are retrieved. Both a lower and an upper bound must be specified and the two values entered must be compatible (for example, for the Key type **annn**, (alpha, number, number, number format) both values must have the same alphabetic prefix). If this is not the case, a messagewill display:

```
Invalid Key item range
```

Pattern Matching

Queries specifying unindexed terms return a far slower response than those specifying indexed fields. (Generally, only indexed fields are displayed on the query form).

Pattern-matching, which does not have indexed support, can provide powerful retrieval and so it is recommended that users combine pattern matching with queries on indexed terms in the same query to improve retrieval speed.

Pattern Matching Facilities

The examples below are referenced in the pattern matching facilities which follow, and contain sample values of the queried field.

Record 1. Congressional Federation

Record 2. Fred Constable

Record 3. Controlled reduction

Record 4. Fed by contacts

Record 5. Constable MacFarlane was fed lies

Record 6. The emaciated lions have not been fed

Record 7. Con MacIntosh

Record 8. Feed the Underprivileged

Record 9. Freed of constraints

The patttern matching facilities include:

! (NOT)

Preceding a query term with ! retrieves records which do not contain that term. There should be no spaces between the ! character and the query term.

^ (Start)

If this character is at the start of the pattern then a term which matches the remainder of the pattern must appear at the start of the field in the record. If this character appears anywhere other than at the start of the term, it is taken as a literal character which must be matched within the correspondance term. E.g. `^fed` matches only Record 4 in the pattern matching example.

$ (End)

If this character is at the end of the pattern, then a term which matches the remainder of the pattern must appear as the last term in the field. If this character appears anywhere other than at the end of the term, it is taken as a literal character which must be matched within the corresponding term. E.g. ***fed$*** matches only Record 6 in the pattern matching example.

? (Single character)

This character matches any single character at the corresponding position in the term of the record. E.g. ***f?ed*** matches Records 2(fred) and 8(feed) in the pattern matching example.

* (wildcard)

This character matches any number of characters (including zero characters) at the corresponding position in the term of the record. E.g. ***f*d*** matches Records 2(fred), 4(fred), 5(fred), 6(fred), 8(feed) and 9(feed) in the pattern matching example.

[*str*] (single character range)

This sequence implies that the next character of the term must be matched by one of the characters in the sequence, ***str***. Note that ***str*** may consist of individual characters and ranges of characters (e.g. the range, **a-m**, indicates all of the characters in the first half of the alphabet). E.g. ***f[er]ed*** and ***f[e-r]ed*** match Records 2(fred) and 8(feed) in the pattern matching example. Note that when ^ is the first character inside square brackets, it acts as a NOT, i.e. the next character of the term must NOT match one of the characters in the sequence.

E.g. ***f[!r]ed*** rejects Records 2(fred) and retrieves only Record 8(feed).

If ^ occurs anywhere but the first position inside square brackets, it is treated as a literal character, without any special function.

{*str*} (repeating character range)

This sequence can be used to match zero or more repetitions of characters which are contained within the sequence, ***str***. This sequence is interpreted as described above. E.g. ***f{er}ed*** matches Records 2(fred), 4(fed), 5(fed), 6(fed), 8(feed) and 9(freed) in the pattern-matching example.

x (where 'x' is any character)

This character must be matched by the same character at the corresponding position in the term of the record. Character case is normally ignored but as mentioned above, this can be changed if required.

\ (literal next)

This character can be used to remove the special effect, if any, of the next character in the pattern.

Examples of pattern-matching queries

```
*mac* ^con*
```

matches records 5 and 7 only in the sample records (see Pattern Matching Facilities in this section). The first pattern alone would match records 5, 6 and 7 and the second alone would match records 1, 3, 5 and 7.

The query:

```
*mac*
^con*
```

(different fields of the same item) matches all records containing either the pattern, ***mac***, or the pattern, **^con***. Thus the records 1, 3, 5, 6 and 7 are matched by this query.

The query:

!^con* fed

matches all records which do not contain the pattern, ***^con**** but do contain the indexed term ***fed***. Thus the records 4 and 6 are matched by this pattern.

Perform Query ([Query]Function ⇒ Query)

When the Query command is selected, the menu bar displays the message:

```
Performing database search . . .
```

and all information in the query form is is matched against all records in the database. In marked contrast to other systems, the more information contained in the query, the faster the query is performed. Note that the case of the query terms, whether in upper or lower case, is ignored. To make case significant, see the section Case Search in this chapter.

During the search of the database, every time the number of matches increases by a defined quantity, called the **update value**, the menu bar changes to display a message of the form:

```
Record matches found: n
```

where ***n*** represents the total number of matches found so far. The update value is initially set to **10** but can be altered (see chapter 7 of this guide). If it is set to **0**, then no such message is displayed.

When the query is completed, three responses are possible. Firstly, if no records match the information in the Query form, then the message will display:

```
No matching records
```

As a query can result in an unlimited number of matches, the DBA can impose a limit on the number of records users may retrieve from a query to ensure that user do not monopolize the machine resources. If a particular query should result in matches exceeding a user's limit, then the query will abort or fail and a message will display:

```
Too many records matching this query. Please be more
specific
```

The third possible query response, and the most common, occurs for successful queries with at least one matching record. In this case, KE Texpress automatically enters Display mode to display some or all of these records.

It is possible to interrupt a query before it has completed by typing the DEL key while KE Texpress is searching for matching records. A message displays:

```
Abort query and enter Display mode (records found so far n)?
```

Choosing no allows the query to continue. Choosing yes aborts the query and commences Display mode with the records which were found prior to the abort.

The Display Form

When a query is performed which retrives at least one matching record, the matching records are displayed on a display form. It is then possible to view the records in several alternative formats (display forms) if available, access any information linked or related to the record (s), edit the records and produce reports.

If the display form is less than half the screen in size, then KE Texpress displays as many records as will fit onto the screen. At the right edge of the menu-bar, a message will display, such as:

```
Records 1 to 16 of 23
```

or

```
Records 1 of 23
```

This indicates the total number of retrieved records (23) and which of those are currently displayed on the screen (Records 1 to 16 or Record 1)

If the database design contains information to drive a videodisc and KE Texpress was invoked with the **-v** option, then as each record is displayed, the associated videodisc commands are executed producing images on the videodisc monitor. These images could be single frames, sequences of frames displayed at pre-determined intervals or complete moving sequences with sound. The automatic display of videodisc images is applicable only if there is only one record on the screen. For more information on the use of a videodisc, refer to chapter 8.

Display Mode Commands

These commands are available in display mode for navigating through screens and records, and performing various functions.

Exit ([Display] Function ⇒ Exit)

This command exits to the query screen which will display the query terms specified in the previous query. Previously retrieved records will not be retained. To retain them while perfoming another query, see the sections **Perform Nested Query** and **Join a Query** in this chapter.

Show next record ([Display] Move ⇒ Forward a record)

This command, which is available only if there is more than one record retrieved by the query, displays the next matching record. The list is effectively circular, so that the record after the last record in the retrieved list is the first record.

Show previous ([Display] Move ⇒ Backward a record)

This command, which is available only if there is more than one record matching the query, displays the previous matching record.

Scroll Forward ([Display] Move ⇒ Forward)

When there is more than one record displayed on the screen and there is more than one screen of matching records, this command scrolls forward to the next screen of matching records. When there is only 1 record displayed at a time on the screen, this command displays the next record.

Scroll Backward ([Display] Move ⇒ Backward)

When there is more than one record displayed on the screen and there is more than one screen of matching records, this command scrolls backward to the previous screen of matching records. When there is only one record displayed at a time on the screen, this command displays the next record.

Move form up ([Display] Move ⇒ Move form up)

This command moves the form up one page in the same record unless it is already at the topmost form.

Move form down ([Display] Move ⇒ Move form down)

This command moves the form down one page in the same record unless it is already at the bottommost form..

Move form left ([Display] Move ⇒ Move form left)

This command moves the form one page to the left in the same record unless it is already at the left boundary.

Move form right ([Display] Move ⇒ Move form right)

Move the form one page to the right in the same record unless it is already at the right boundary.

View comments ([Display] Misc ⇒ View comments)

View any comments associated with this record, displayed in Text perusal mode. (See also the Text Perusal section in Chapter 2 of this guide.)

Change Display form ([Display] Function ⇒Select Display Form)

Select a different Display form for viewing matching records. A selection box will appear, listing the available forms. To select the form, use the arrow keys to move to it and press ↵.

This form will remain the Display form until this command is used again to change it, or until the end of this Texpress session.

If a Display form with a built-in sort order is selected for displaying the matching records, then the records will be re-ordered according to the ordering information in the form (refer to Chapter 6) before being displayed on the new Display form.

Record Histories

History versions are created when a record is edited if the option **history=ask** or **history=yes** is set. (See also the Database Options - History section in Chapter 7 of this guide). The following commands can be used to trace the saved history of a record.

Show History ([Display]Show ⇒ Show History)

View history versions of the current record. The oldest version of the record is displayed and the mode indicator is set to **History**. The commands, **Move form left**, **Move form down**, **Move form up** and **Move form right**, as described previously, are used to position the form window. The movement commands can be preceded by a number which indicates the number of times the command should be repeated.

View Later Version ([History] Move ⇒ Function to later version)

Display the version immediately following (more recent than) the one on display. If, however, this is the current version then the command is disabled.

View to Earlier Version ([History] Move ⇒ Backward to earlier version)

Display the version immediately preceding (less recent) the one on display. If, however, this is the first version then the command is disabled.

Change DisplayvForm ([History] Function ⇒ Select display form)

Select a different Display form for viewing matching records. A selection box will appear, listing the available forms. To select the form, use the arrow keys to move to it and press ↵. This form will remain the Display form until this command is used again to change it, or until the end of this Texpress session.

If a Display form containing a built-in sort order is selected for displaying the matching records, then the records are first re-ordered according to the ordering information in the form (refer to Chapter 6) before being displayed on the new Display form.

Delete History ([History] Function ⇒ Delete version)

Delete the displayed history version of the record. The system will first ask for confirmation. Note that deleting the current version of the record (which must be performed in Display mode) also deletes all of its history versions .

Exit ([History] Function ⇒ Exit)

Exit from viewing the histories and return to the record in Display mode.

Report Generation

The following commands can be used to prepare matching records for report generation. These commands are available only to users with Report access privilege. Chapter 6, "Database Reports", has a detailed discussion of report generation facilities.

Report this match ([Display] Reports ⇒ Report this match)

Generate a report of the current record only. The Report generation form is displayed allowing the users to select the format and characteristics of the report.

On the completion of the report generation, Display mode is resumed.

Report all matches ([Display] Reports ⇒ Report all matches)

Generate a report of all matching records. The Report generation form is displayed allowing the format and characteristics of the report to be selected.

Peruse Mode

Peruse records ([Display]Function ⇒ Peruse records)

This command is available only on databases with a Key item which has an incrementing component (such as the **integer**, **annn**, **nnna** and **yynnn** Key types).

Perusing enables a user to scan forward and backward through the database in Key value order, starting from the current record. The displaying of records is always via the current Display Form. Only one record at a time displays on the screen.

The commands available during Peruse mode are similar to the equivalent Display mode commands. The commands whose functionality differs from those in Display mode are:

Next record ([Perusing] Move ⇒ Forward a record)

Move forward to the record with the lowest Key value higher than the current record's Key value, i.e. the next sequentially occurring record. This may not be the next record retrieved by the query. If no record can be found, an appropriate error message is displayed.

Previous record ([Perusing] Move ⇒ Backward a record)

Move backward to the record with the highest Key value lower than the current record's Key value, i.e. the previous sequentially occurring record. This may not be the previous record retrieved by the query. If no record can be found, an appropriate error message is displayed.

Exit ([Perusing] Function ⇒ Exit)

Terminate Perusing mode and return to Display mode.

Jump to Record ([Display] Function ⇒ Jump to Record)

If the user knows the key value of a record s/he can jump to it by specifying its key value. This command is available only on databases with a Key item.

The user will be requested to enter the key field values in turn in a pop-up box. The record with that key value is then displayed in jump mode. While in Jump mode, users can jump to another record.

The commands available during Jump mode are similar to the equivalent Display mode commands. The command whose functionality differs from that in Display mode is:

Exit ([Jump] Function ⇒ Exit)

If nested jumps have been performed, this command will daisy-chain, i.e return the user to the display of the most previous record in the stack of jump records. If the current record was the first record jumped to, Jump mode terminates, and the user is returned to Display Mode.

Edit Records

The following commands can be used to edit records already stored in the database. The user can edit either the current record or all the records retrieved from the query. Note that once these edits are saved there is no undo command to remove the effect of these operations.

When a record is edited, no check is performed to ensure that the edited data still matches the query. Therefore, in this way, an edited and now non-matching record may remain in the set of retrieved records.

The editing commands are:

Edit record ([Display] Function ⇒ Edit record)

This command enables users with Record edit privilege to edit the current record if no other user is currently editing the same record.

The user is placed in Edit mode, with access to all of the editing facilities described in Chapter 3. In addition, most of the extra commands offered for insertion are available.

One insertion feature not available is automatic Key value assignment. In fact no change at all can be made to the Key value of the record. Thus the Key item is unreachable in edit mode. Also, an edited record cannot be written to the Temporary File.

Insertion features which are supported include the ditto facilities, the default value load command, expression calculations which are performed whenever an inter-dependent field value is altered, and record validation. The Comments can also be edited. These facilities are described in Chapter 4.

Termination of record editing can be performed in one of two ways:

Save ([Edit] Function ⇒ Save)

Save the edited record. The record is first validated (as described in Chapter 4). If validation should fail, the record is not updated and the user returns to Edit mode. Otherwise the changes made become permanent and a new version of the record is created. All automatic items have their values updated.

If the user does not have Destructive edit privilege, then the unedited (old) version is saved and automatically linked to the edited (new) version, thus creating a history chain for the record. If however the user has Destructive edit privilege, KE Texpress discards the previous version unless a database option has been set to save the previous version (history). (See also the Record Histories section in Chapter 5 and the Database Options - History section in Chapter 7).

Previous versions of comments are not maintained in the history chain, but instead only the most recent version is kept, so information should not be deleted from Comments. Rather, new information should just be added to the bottom, perhaps with a note indicating the date of the addition and the name of the user who made it.

Exit ([Edit] Function ⇒ Exit)

Exit without updating the record. If the information in the Insertion form has been modified, KE Texpress asks for confirmation:

```
Exit from Edit mode? [yes] [no]
```

Choosing no resumes Edit mode.

When editing is terminated, the current record is displayed.

Edit matching records ([Global] Edit ⇒ Edit matching records)

Globally edit the matching records. This operation is available only to users with Record edit privilege and is permitted only if no other user is currently editing any of the matching records.

The command calls up an empty Insertion form, into whose fields the user can enter the new information. Information entered into each of these fields overwrites the contents of the corresponding field of each matching record.

When this option is selected, the user is placed in Edit mode with access to all of the editing facilities described in the chapter on "Entering Information". However, most of the additional commands offered for insertion are disabled. For example, although global editing commences with an empty form displayed, no change can be made to the Key value of the record as the Key item is inaccessible to the cursor. The mode indicator is set to **Global**.

Clear Field ([Global] Misc ⇒ Field clear setting)

When this is turned on, it wipes out or erases information in the current field in all matching records. Initially, this flag is off for all fields of the form. When the user selects the command, confirmation is requested:

```
Field clear setting is off. Turn on? [y/n]
```

If the setting is off, confirmation is also requested:

```
Field clear setting is on. Turn off? [y/n]
```

If during Global edit mode, data is entered into a field which has its on, then the field clear setting is automatically turned off.

Termination of form editing can be achieved in one of two ways:

Save ([Global] Function ⇒ Save)

Perform the global substitution on all matching records. Each of the matching records is updated automatically. The sequence of re-inserting each record is identical to that for individual record editing. History versions are maintained (if desired).

For users with Validation override privilege, KE Texpress first prompts with:

```
Perform record validation? [y/n]
```

Choosing no completes the global edit more quickly, but with the possibility of invalid records being inserted into the database. Choosing yes causes validation to be performed.

Various environment options can be set to alter this functionality. The setting of environment options is discussed in more detail in the "Database Options" section in Chapter 7.

If this validation phase should fail, a message will display and the user is placed in Edit mode to correct the invalid record. The editing facilities available are as described above for individual record editing. On the completion of editing, global substitutions for the remaining records continue.

If Edit mode is terminated with the Exit command then the global substitution is avoided for that record and a message will display:

```
Edit avoided for this record. Continue with remaining
records?
```

Choosing no aborts the global edit, and all subsequent records remain unchanged.

Before the re-insertion takes place, all automatic items if any have their values updated so that they reflect the date and time of the alterations and the Unix Id of the user who made them.

Exit ([Global] Function ⇒ Exit)

Exit without updating the records. If the information in the fields(s) has been modified, KE Texpress asks for confirmation:

```
Exit from Edit mode? [y/n]
```

Choosing yes aborts the global edit command being is aborted. Choosing no resumes Edit mode.

When the Global edit command is terminated, the current record is displayed on the Display form.

Delete Records

The following commands permanently delete records in the database. There is no undo command to remove the effect of these operations. The user can either delete the current record or delete all records retrieved from the query.

Delete record ([Display] Edit ⇒ Delete record)

Delete the current record from the database. This operation cannot be undone and is available only to users with Record deletion privilege. If the database has a Key item, the record's Key value is unassigned so that it can be reassigned to another new record.

When the delete command is selected, the system will first seek confirmation:

```
Delete record n? [yes] [no]
```

(where ***n*** is the relative number of this record amongst all matches.)

If the user chooses yes but a user on another terminal deleted this record while it was being viewed by the current user, a message will display:

```
Record has been deleted
```

If the user chooses yes and a user on another terminal is currently editing this record, it cannot be deleted, and a message will display:

```
Record is being edited
```

If the user chooses yes and another user has edited this record and saved it in the meantime, which means that this version of the record is no longer the current one, a message will display:

```
A newer version has been inserted
```

The query must first be repeated to retrieve the new version.

If the delete operation is successful, the user is returned to Display mode at the record following the one deleted.

Delete all Retrieved Records ([Display] Edit ⇒ Delete Matching Records)

Delete all the retrieved records from the database. This operation cannot be undone and is available only to users with Record deletion privilege. If the database has a Key item, each record's Key value is unassigned so that it can be reassigned if need be.

When the delete command is selected, the system first seeks confirmation:

```
Delete all matching records from the database?
        [yes] [no]
```

Choosing yes results in another request for confirmation:

```
All matching records will be permanently removed. Abort?
        [yes] [no]
```

Choosing no causes KE Texpress to attempt to delete all matching records. To achieve this it first locks all records and performs various checks to ensure that the operation can be performed for all the records. It aborts and gives a suitable error message if this is not possible. Otherwise it deletes each of the matching records.

If the deletion is successful, the user is returned to Query mode to commence the next query.

Record Privilege Level Commands

The following commands relate to the privilege level of the record currently being viewed. They are:

Show privilege level ([Display] Edit ⇒ Show privilege)

Display the record's privilege level if there is more than 1 privilege level set for the database. A message will display:

```
Record is at privilege level n
```

where ***n*** is the record's privilege level. This option is available only if the user is not at the lowest privilege level. (If the user is at the lowest privilege level then so must be the record.)

Change privilege ([Display] Edit ⇒ Adjust privilege)

Adjust the privilege level of all retrieved records. When selected, confirmation is first requested:

```
Change privilege level of all matching records? [yes] [no]
```

Choosing yes calls up a box, into which the user can enter the new privilege level:

```
Enter new privilege level (between x and y):
```

where ***x*** and ***y*** represent the lower and upper bounds for the new privilege level as configured for the database. The new privilege level cannot be a higher level than the user's insertion privilege level. Typing ↵ aborts the operation. Otherwise a message displays:

```
Updating privilege levels to n . . .
```

Checks are made to ensure that none of the matching records is currently being edited. If one is, an appropriate error message is displayed and the operation is aborted, i.e. the privilege levels of all retrieved records are not changed.

At the completion of the operation, system confirms the change with:

```
New privilege levels are n
```

is displayed, where ***n*** is the new privilege level of all of the matching records.

This option is available only if the user has Record privilege alteration privilege and is not at the lowest privilege level of the database.

Re-order Records

The following commands re-order the set of matching records or produce summaries of the retrieved information or re-order records for subsequent queries.

Re-order Matches ([Display] Misc ⇒ Re-order Matches)

The records matching a query are initially retrieved and displayed to the user in a random order. This command enables the user to re-order the matching records in any desired way and then display then. It is available only for queries which result in more than one matching record. This command will also produce a summary or frequency count of any field in the record nominated by the user.

The Insertion form will display with each field of the form represented as a sequence of underscores. The mode indicator is set to **Record re-ordering**. The cursor is positioned at the first prompt and can be moved around the form (see the Movement around a Form section in Chapter 3 of this guide), or by clicking the left mouse button if Tunix or WinTunix software is used. Other commands available are:

Select item/field ([Record re-ordering] Function ⇒ Select item/field)

Select the current item or field as the **List object**. The records are then re-ordered by the contents of this object.

If a field is chosen as the List object, the records are re-ordered according to the contents of this field. This ordering occurs numerically for numeric fields, alphabetically for text fields and as specified in the Look-up tables of fields of items with User specified Look-up tables.

If an item is chosen as the List object, the records are re-ordered according to the contents of this item. Thus the first field of the item in each record is tested using a field comparison as described above. If values in this field are equivalent, then the second field is tested, followed by the third and so forth. Special sorting techniques are used for the KE Texpress Library items which can be ranged (see Appendix C).

After the List object has been selected, KE Texpress prompts with:

```
Ascending or descending sorting for this item?
```

After the sort order has been selected, the following message is displayed:

```
Produce a summary of the matching records? [yes] [no]
```

Choosing no sorts the records and Display mode resumes with the matches displayed in the new order.

Choosing yes sorts the records are sorted and a summary is generated. The summary contains a one line entry for each different value of the List object. Associated with this value is a count of the number of times it appears in the matching records.The screen is cleared and the message:

```
Summary of n matching records
```

appears at the top of the screen (where ***n*** represents the total number of records which were found to match the query). Following this is a sequence of lines of the form:

```
NO        Count    String value
```

where **n** is the sequence number of this value in the ordered list, **Count** is the number of times that this value appears in the matching records and **String value** is the value of the List object. If the list object is a numerical field, the summary will show at the end of the list the average value, standard deviation and mean for the retrieved records. The summary is viewed using the Text perusal facility described in Chapter 2. The user can scroll forward or backward and copy the summary into a file or send it directly to a printer.

On termination of this summary, the user is returned to viewing the matching records in the newly imposed order with the first of these being the current record.

Exit ([Record re-ordering] Function ⇒ Exit)

Abort the re-ordering process. The matching records remain in their previous sequence.

Set ordering criteria ([Display] Misc ⇒ Set ordering criteria)

This command allows the user to re-order retrieved records for all subsequent queries. It has no effect on the records matching the current query.

The user can specify the item or field on which ordering will occur. The selection of an item or field occurs exactly as described previously in the previous Select item/field section, as do the requests for ascending or descending order and a summary of the matching records. However, no action is taken immediately. Instead, for each subsequent query, the matching records are collected and then re-ordered, and possibly summarized, as if the **Re-order matches** command had been issued. This ordering overrides any Report form order set up for the selected Display form.

To change the default re-order/summarize item, this command should be issued again. To disable this feature, the **Exit** command should be selected during the item/field selection procedure.

Copy or Print Record

The following commands allow records to be copied to a Unix file or printer, either individually or collectively. Only information which can be retrieved and viewed can be copied. See "Copying or Printing Information" in Chapter 2 for more information.

Copy this match ([Display] Show ⇒ Copy this match)

Copy the current record to a file or printer. The record appears as it does on the screen, and its comments are also copied.

Copy all Retrieved Records ([Display] Show ⇒ Copy all matches)

Copy all matching records to a file or printer. The records appear as they do on the screen and their comments are copied with them.

Miscellaneous Commands

The following commands are available on the Misc pull-down menu in Display mode:

Verify Matches ([Display] Misc ⇒ Verify Matches)

Verify that the records retrieved actually match the query. KE Texpress can very quickly ascertain possible matches to a query. However, a characteristic of the retrieval method is that a (very) small percentage of these may be false matches (they do not match the query information exactly). For some queries, Texforms does not check all possible matches to eradicate false matches at the time of the query because this can be time-consuming. Rather, it checks before displaying the record.

To set the system to perfom the check at the time of query, thus giving slower response to queries but ensuring 100% accuracy of the figure representing the total matches, the DBA can set the matchverify database option. (See Display mode options in Chapter 7 of this guide for more information.)

Running this command ensures that 100% accuracy of the total matches figure can still be obtained for any given query.

Alternatively, any command or sequence of commands which references every matching record (summarizing, reporting or viewing all matching records, etc.) automatically verfies the matches.

When the matches have been verified, KE Texpress responds with:

```
Verified n matches
```

where ***n*** is the corrected number of matching records. Any false matches are removed from the list.

This command is always available and can be repeated in subsequent queries if desired. However, more than one invocation per query is unnecessary.

Set Ditto Record ([Display] Misc ⇒ Set Ditto Record)

Set the currently displayed record as the ditto record. Subsequent ditto commands will copy data from this record. See also Ditto commands in Chapter 4.

Save a Dave File ([Display] Misc ⇒ Save a Dave File)

Write a copy of all matching records to a file in dave file format (refer to the KE Texpress Load Reference Manual for a discussion of dave file format). When this command is selected, a file name is requested. A copy of each matching record is appended to the given file preserving its contents. History versions and comments are also copied to the given file in dave file format.

No check is performed to ensure that an existing file is in dave file format before the copy is made.

Discard Record From Retrieval ([Display] Edit ⇒ Discard Record)

Discard this record from the retrieved records, without deleting it from the database. Confirmation is first sought:

```
Discard this record from the matches? [yes] [no]
```

Choosing yes removes the record from the list of matches (but not from the database) and display mode continues at the next record in the matches.

Show Video ([Display] Misc ⇒ Show Video)

Display the image or images associated with this record on the videodisc. This command is available only if the database design contains information to drive the videodisc, KE Texpress was invoked with the **-v** option, and a videodisc player is connected.

Interpret Form Expression ([Display] Misc ⇒ Interpret Form Expression)

Interpret the expression associated with the Insertion form. This command is available only if such an expression has been created by the DBA. It is often used to provide a simple interface to many of the Display mode commands by requesting a selection from the user, and generating the appropriate command sequence to perform the selection.

Additional Query Commands

The following commands allow the user to perform additional queries without losing the retrieved set of matching records.

Perform Separate Query ([Display] Function ⇒ Perform Nested Query)

This command allows the user to perform additional queries whilst **separately** retaining the currently retrieved records for later use. When this command is selected, the display screen is replaced with the query form with the mode indicator displayed in the top left hand corner of the screen now including the current nesting level in brackets. E.g. [query (2)] indicates that this is the 2nd nested query level, i.e. there are 2 distinct sets of retrieved records being held.

If a nested query retrieves some matching records, a nested display form will be called up. If no matching records are found or the user chooses to exit from nested display mode, the previous display form will be displayed. Exiting from nesting level 2 for example takes the user to level 1 and exiting again returns the user to the initial display form.

Combine Queries ([Display] Function ⇒ Join a Query)

This command allows the user to perform additional queries which **combine** the results with the currently retrieved records. When this command is selected, the display screen is replaced with the query form with a message in the right of the menu bar indicating that records are being held, e.g.

```
13 records held
```

When the query is performed, any retrieved records are first compared against the held records from the prior query and duplicated records are removed. The following messages will display :

```
Searching for duplicate matches ...
```

and

```
5 duplicates removed
```

Once the query is complete, display mode recommences. The set of records displayed now includes the original records and any new records found as a result of the joined query.

Disappearing Matches

During Display mode, in the top right hand corner of the menu bar is a message:

```
Record x of y
```

where ***x*** is the relative number of the matching record being viewed and ***y*** represents the total number of retrieved records from the query (called the **match count**). This latter value normally remains constant for the duration of Display mode. However some options can cause this value to decrease following a query.

The most obvious operations to have this effect are **Delete record**, **Delete matching records** and **Discard record**. Each of these commands removes records from the list of retrieved records. Thus the match count is reduced.

The command, **Verify matches**, can also reduce the match count by removing all of the possible matches which do not match the query exactly.

Other operations can induce the same effect. In particular, if a record returned as a possible match to the query is found to be a false match then it is removed from the list of matches the first time it is referenced. Commands such as **Verify matches** and **Re-order matches** reference all matching records, so are certain to remove the false match. Other commands such as **Forward a record** and **Backward a record** remove the false match only if it is the record which was about to be displayed.

As each record is removed from the list of matches, the number representing the total number of matches to the query is decremented. If all matches should disappear in this way, then an implied exit is performed and the user is returned to Query mode.

Following a query, if one of the matches is then edited by another user and a new version has been created, then the old version (the one which originally matched the query) is still the version which is viewed. It is **not** removed from the list of matches but cannot be edited or deleted. To access the new version, the query must be performed again.

Chapter 6

Database Reports

Create/Edit a Report Form 6-4
Select Objects....6-5
Deselect all objects....6-5
Moving Around the Report Form 6-6
Edit Commands....6-8
Retrieve Item....6-8
New Header 6-8
Insert Box....6-10
Move Object 6-10
Copy....6-11
Change Objects 6-12
Append Prompt to Item....6-13
Delete Object 6-13
Undo Last Command 6-13
Attributes....6-14
View/Edit Attributes 6-14
Exit....6-14
Forward 6-14
Backward 6-14
Next screen 6-14
Previous Screen....6-15
Next item 6-15
Previous Item 6-16
Form Bounds....6-17
Form Order....6-19
View Order 6-19
Heading Section 6-20
Footer Section 6-20
Miscellaneous Commands 6-21
Form Status 6-21
Validation Form....6-21
Form Coordinates 6-21
Privilege Level View 6-23
Copy Form 6-23
Exit the Report Form 6-24
Save Form 6-24
Exit....6-24
Interrupt a Command 6-25
Rename Report Form 6-26
Delete Report Form....6-27
View Report Form....6-28

Copy Report Form 6-29
Generate Reports 6-30
Report all Matches..... 6-30
Report this Match 6-30
Report Generation Form..... 6-31
Moving around the Report Generation Form 6-34
Exit 6-35
Special Report Operators..... 6-36
Space Removal: $S..... 6-36
Space Replacement: $X..... 6-37
Line Removal: $L..... 6-37
Prompt Removal if there is no data: $P ... $P ... $E6-38
Line Replacement: $F..... 6-38
Line Joining: $J 6-40
Line Joining with Word Wrap: $W 6-40
Query Replacement: $Q 6-41
Record Number: $N 6-41
Multiple Column Output Support: $Cn..... 6-41
Embedded Control Characters: $Onnn :m 6-41
Report Pagination - texpaginate..... 6-42
Headlines and Footlines 6-43
Data Sections..... 6-44
Header and Footer Sections..... 6-44
Start and End Sections..... 6-44
Report Section 6-46
Line Drawing - texcat 6-47

Overview

The KE Texpress reporting system is a powerful and versatile utility which enables users to perform such diverse tasks as printing a label for a single record, or generating a one-line summary of every record in the database. Reports containing arithmetic expressions such as sub-totals, averages, etc. can be generated by using the KE Texpress reporting system in conjunction with the Unix report writing tool, **awk**.

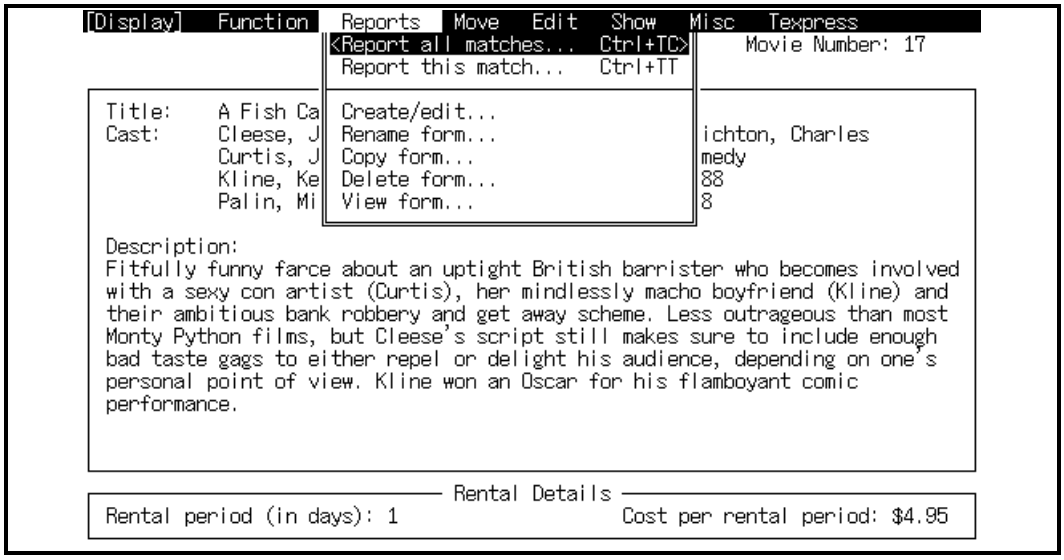
To generate a report, KE Texpress retrieves of one or more records (following one or more queries), re-orders these records according to any ordering constraints imposed on the report, and extracts from them information which can be printed, copied or manipulated as desired.

To define the information which is to be extracted from each record, a Report form is used. This special form is in fact a subset of the Insertion form. However, its actual appearance can be quite different. The ordering constraints which are to be placed on the report and the format of the report are defined in this Report form.

The output format of the report can be varied in several ways, e.g. pagination, placement of page headers, output verbosity, and compaction of information.

Finally, the report may be passed through other KE Texpress and Unix filter programs to perform further manipulation of the data.

Commands for manipulating Report forms are provided on the Report pull down menu of **texforms**. This menu is available only to users with Report access privilege and appears as in the following figure.



Report menu

Create or Edit a Report Form

Create/Edit ([Display] Reports ⇒ Create/Edit)

Create a new Report form or edit an existing one. Only users with Report create / delete privilege are permitted to perform this operation.

When the command is chosen, a selection box pops-up, displaying any existing reports, with the cursor sitting in the lower of two boxes. Press **TAB** or **Ctrl+V** or **Ctrl+W** to jump from one box to another.

To create a new report, enter a name into the lower box. This will create a Report form identical to the Insertion form, which the user can then modify.

To edit an existing report, jump to the upper box and then move to the appropriate report (using the ↓ and ↑ keys). Note that users can only edit reports which they have created. Press ↵ to select the report.

To exit without selecting a report, enter **Ctrl+X**.

Select Objects

The selection commands are used for selecting objects, and adding objects to, and removing objects from, the current group of selected objects. Any forms editing commands affect the currently selected group of objects. The selection commands do not operate on parts of items, only on whole items. For individual manipulation of prompts and fields, refer to the appropriate forms editing command.

Select objects ([Report form] Function ⇒ Select)

Add the object (item, header or box) which the cursor is currently on, to the current group of selected objects. If the object is already in the current selection then it is de-selected. An error message is displayed if the cursor is not currently on an object, i.e. header, box or item

Select block ([Report form] Function ⇒ Select block)

Select a block of objects by specifying an enclosing rectangle. The current cursor position is taken as the top left corner of the block and the user is requested by a message in the meny bar to select the position of the lower right corner. All objects completely enclosed within the designated block are added to the current selection. Objects within the block which were already selected will remain selected.

Select all objects ([Report form] Function ⇒ Select all)

Include all form objects in the current selection.

Deselect all ([Report form] Function ⇒ Deselect)

Discard the entire selection.

Moving Around the Report Form

There are various methods of cursor movement available. In all of these methods the form window automatically scrolls if required. If an attempt is made to move the cursor outside the bounds of the form window, the terminal bell sounds.

The smallest step of cursor movement is one character position or one line. For larger steps, tab movement of the cursor can be used. Tab stops are set at each eight character positions of the form window width and each six lines of the form window length. Tab movement is selected by entering the upper case equivalent of the appropriate command (eg. **H** for tab left, instead of **Ctrl+H**). A third method of cursor movement allows direct movement between words in the form. A word is defined as any sequence of non-blank characters.

Move left ([Report form] Move ⇒ Move left)

Move the cursor one character position to the left. Typing **H** moves the cursor left to the previous tab stop.

Move right ([Report form] Move ⇒ Move right)

Move the cursor one character position to the right. Typing **L** move the cursor right to the next tab stop.

Move up ([Report form] Move ⇒ Move up)

Move the cursor one line up. Typing **K** moves the cursor up to the previous tab stop.

Move down ([Report form] Move ⇒ Move down)

Move the cursor one line down. Typing **J** moves the cursor down to the next tab stop.

Next word ([Report form] Move ⇒ Next word)

Move forward to the next word. This operates on the form in a left to right, top to bottom manner with wrap-around if the form boundary.

Previous word ([Report form] Move ⇒ Previous word)

Move backward to the previous word. This operates in the form in a right to left, bottom to top manner with wrap-around if the form boundary is reached.

Each of these cursor movement commands may be preceded by a number indicating a repeat count. For example if **7→** is entered the cursor moves seven character positions to the right. Similarly, **3J** moves the cursor down three tabs. If the repetition count for a command would position the cursor outside the form bounds the cursor is moved only the maximum allowable distance in the appropriate direction.

Find prompt ([Report form] Function ⇒ Select)

A message box pops-up, requesting the user to enter the leading letters of the prompt. The cursor will then move moved to the next item prompt which begins with those letters. Searching for a prompt commences from the current item or at the item nearest the current cursor position. The form window is scrolled if required. If no matching prompt can be found, the terminal bell sounds.

Edit Commands

A report form can contain items headers or boxes, which are manipulated by the various commands.

If the form is larger than the screen, it can be scrolled horizontally or vertically, provided that the cursor is not already positioned at the edge of the form. This provides access to areas of the form not currently displayed. The form scrolls automatically if an attempt is made to move the cursor outside the current form window bounds.

Only boxes may overlap or be overlapped by another object, such as an item or a header. Items and headers cannot overlap each other, except when being moved on-route to a (non-clasing) position.

These commands are found on the Edit pull-down menu of the Report form. Users should select the position of the cursor before choosing the commands. In general it is not possible for form objects to overlap (see Form Layout). However, headers (and prompts) may be broken into several sections on the one form line.

Retrieve Item ([Report form] Edit ⇒ Extract Item)

An item previously deleted from the Report form may be re-inserted by extracting it from the Insertion form.

A selection box will display the internal ids of all the items on the Insert form. Ids of items that are already in the Report form will appear dimmed and cannot be selected. To select an item, move to it using the arrow keys ↓ and ↑ and press ↵. This will extract the item and insert it on the Report form at the current cursor position. If there is insufficient space to insert it at the current cursor position, a message will display and the item will not be extracted. Move the cursor to a blank position with enough space for the prompt and field(s) and repeat the command.

New header ([Report form] Edit ⇒ New Header)

First move the cursor to the desired position, and then choose the command. The menu bar will display a message:

```
Enter header
```

and character input is read from the keyboard and displayed on the form. Any printable character is accepted in the header text. During header insertion, character input is not permitted to overlap an existing object. Instead the cursor is automatically moved to the next available character position on the form line. The header string should be terminated by ↵ or **ESC**. If a null string results, the header is not inserted into the form.

Insert Box ([Report form] Edit ⇒ New Box)

The current cursor location is taken as the top left corner of the box. The menu bar will display a prompt requesting the user to select the position of the bottom right corner of the new box. Another prompt will then display:

```
Single or double line box?
```

Depanding on the selection, a single or double line box will be drawn on the form.

Some terminals may not be capable of displaying double lines. In this case the best approximation is displayed (typically single lines). Boxes will default to simple horizontal or vertical lines if zero height or width is selected.

Move Object ([Report form] Edit ⇒ Move)

The move command is used to relocate objects within the form. It operates on the currently selected items. (See Selecting Objects in this chapter)

A message in the menu bar will prompt the user to move the object. If nothing is currently selected, the object which the cursor is currently on will be moved. If that object is a prompt or a field then just that prompt or field will be moved. To move a complete item (i.e. prompt and field), it must first be selected before being moved. (See also Moving Around the Report form in this chapter for details on cursor movement).

Once the object(s) has/have been moved, the final position is marked by pressing F8 or **Ctrl+C** ([Report form] Function ⇒ Select position). During a move operation, it is possible to move objects across other objects on route to a new location within the form. However, objects are not allowed to clash when the position is finalized. If so, a message will display:

```
Clash at present position
```

The object(s) must be relocated in a non-clashing location. There is always room within the form for moved object(s), as they can always be replaced at their original position. If an attempt is made to move any part of the object(s) outside the form boundary, the terminal bell sounds.

Copy Object ([Report form] Edit ⇒ Copy)

The copy command is used to duplicate objects within the form. This command operates on the current selection. If nothing is currently selected the object which the cursor is currently on is copied. Note that only headers and boxes can be copied on the Report form. When the command is chosed, the original object(s) remain stationary and duplicate object(s) emerge to be relocated. As with the move command, if when the end position is marked, a clash is detected a message will display:

```
Clash at present position
```

The copied object(s) must be relocated to a non-clasing position..

It is possible to take a copy of some object(s) and then find that there is not room for it in the form. If this situation arises the copy command can be aborted by pressing the **DEL** key.

Change Object ([Report form] Edit ⇒ Change)

The change command is used to alter an existing form object's field length, or prompt, box or header. Each of the objects in the current selection may be changed in turn. If nothing is currently selected, the object which the cursor is currently on is changed. If that object is a prompt or a field, then just that prompt or field will be changed. To change a complete item (i.e. prompt and field), the item must first be selected.

To change field lengths, place the cursor over the field and select the command. Note that integer field types have a length limit of 15 characters and real field types have a length limit of 25 characters.

When the command is chosen, firstly, the object is cleared from the form and a new position may be selected. If the same form location is required the location selection command (see Selecting Objects in this chapter) may be entered immediately. The procedure then followed is similar to that for creating a new object of the same type.

It is not permissible to delete an object by changing it to a null object, e.g. selecting a field length of 0 or not entering a prompt name. If this occurs an error message is displayed, the original object is restored and the command is terminated. To remove an object from the form use the Delete command.

Append Prompt to item ([Report form] Edit ⇒ Append)

The append command is used to add a prompt to an existing item which does not have a prompt. Each of the items in the current selection or if nothing is currently selected, the item the cursor is currently on is appended to in turn

Note that fields cannot be appended to an item in the Report form.

Delete Object ([Report form] Edit ⇒ Delete)

The delete command is used to remove objects from the form. It operates on the current selection. If all objects are currently selected, confirmation is requested before the delete operation proceeds.

If nothing is currently selected, the object which the cursor is currently on is deleted. If that object is a prompt or a field, then just that prompt or field is deleted. To delete a complete item i.e. prompt and field, the complete item must first be selected.

Undo Last Command ([Report form] Edit ⇒ Undo)

This command can be used to undo the last command executed. Only commands which act on form objects may be undone. This encompasses all the commands on the Edit pull-down menu.

For example, if the previous command had moved an item to a new position within the form, then the undo command would place the item back at its original location. The undo command itself may be undone, producing a toggle effect.

Attributes

Items, fields, headers and boxes have numerous attributes which determine how they are treated by the KE Texpress. These attributes are defined by the DBA during Insertion form design. For a detailed description of attributes refer to the KE Texpress Design Guide.

View/Edit Attributes ([Report form] Define ⇒ Attributes)

This command displays the attributes of the object which the cursor is on. Editing is permitted only on the text of an item prompt or a header. All the facilities of edit mode are provided. (See Chapter 3 of this guide).

The first screen contains the attributes for the entire item, and subsequent screens contain the attributes of each individual field within the item.

The Attribute viewing commands are:

Exit ([Item attributes] Function ⇒ Exit)

Exit from attribute viewing and return to the Report Form.

Forward ([Item attributes] Move ⇒ Forward)

Move forward to the next attribute which may be edited on the current screen.

Backward ([Item attributes] Move ⇒ Backward)

Move backward to the next attribute which may be edited on the current screen.

Next screen ([Item attributes] Move ⇒ Next page)

Move to the first attribute which may be edited on the next screen. This command is disabled for header and box attribute editing.

Previous Screen ([Item attributes] Move ⇒ Previous Page)

Move to the first attribute which may be edited on the previous screen. This command is disabled for header and box attribute editing.

Next item ([Item attributes] Move ⇒ Next item)

Move to the attribute screen of the next item, header or box in the form order.

Previous Item ([Item attributes] Move ⇒ Previous Item)

Move to the attribute screen of the previous item, header or box in the form order.

Form Bounds

When a Report form is created, it is set to the same size as the Insertion form. When a report is generated, each record output occupies an area equal to the size of the form. The Report form size may need to be altered to suit the output requirements.

Change Form Size ([Report form] Define ⇒ Form Size)

Alter the form size. A sequence of dots is displayed indicating the rightmost and bottom boundaries of the form. The cursor is positioned at the bottom rightmost corner of the form. The cursor movement commands are then used to relocate the form boundaries, and so altering the form size. There is no limit on the size of a form.

The form bounds manipulation commands are:

Move left ([Form size] Function ⇒ Move left)

Move the right form boundary left, decreasing the width of the form. It is not possible to move the boundary past the edge of the right most form object.

Move right ([Form size] Function ⇒ Move right)

Move the right form boundary right, expanding the width of the form.

Move up ([Form size] Function ⇒ Move up)

Move the bottom form boundary up, decreasing the length of the form. It is not possible to move the boundary past the lowest form object.

Move down ([Form size] Function ⇒ Move down)

Move the bottom form boundary down, increasing the length of the form.

Form coordinates ([Form size] Function ⇒ Form coordinates)

This command displays the current form size, form view and cursor coordinates. The display has the format:

Size (*lines*, *cols*), View (*y*, *x* : *my*, *mx*), Cursor (*y*, *x*)

The form size states the number of lines and columns in the form. The form view indicates the virtual form window coordinates from which the form is currently being viewed. The cursor coordinates are the current form line and column position of the cursor.

Exit ([Form size] Function ⇒ Exit)

Return to Report form, which the adjusted form size becoming the current form size.

Form Order

The Report form order is the sequence in which items are sorted during report generation. By default, there is no sort order, and the report is generated in whatever order the data is retrieved unless a sort order is specified explicitly. Headers and boxes play no part in the form sort order, as they are for display purposes only.

Form order ([Report form] Define ⇒ Form order)

Specify (or change) the form sort order. The menu bar displays:

Select items in sequence

Select each item in the order desired, and each item highlighted as it is selected, with a prompt:

Ascending or descending sorting for this item?

To signify the end of the selection procedure, move the cursor to a blank area of the form and select it. Confirmation that form ordering is complete is then requested. All items need not be selected, and in such a case, sorting is performed only on the selected items.

Re-selecting selected items or trying to select headers or boxes results in an error message. Upon termination of the selection procedure, or when all items are selected, a message displays:

```
Re-ordering form . . .
```

The form sort order is then adjusted.

If no items are selected in the sort order (ie. a blank area of the form is selected) then no report sort order is designated. A message displays:

```
No report sort order
```

View Order ([Report form] Define ⇒ View Order)

This command allows the user to view the current form sort order. Each item is highlighted in turn, and a message is displayed indicating whether ascending or descending sorting applies to the item. (The next item in the form sort order is displayed when any key is typed).p41s8

Report Heading and Footer Sections

By default all lines in a Report form are printed for each record output during report generation. The Heading and Footer section commands can be used to designate certain areas of the Report form to be printed only once for each report page.

These two areas of the Report form are termed the **Heading section** and **Footer section**. The Heading section can occupy any number of adjacent lines at the top of the form. The Footer section can occupy any number of adjacent lines at the bottom of the form.

Heading Section

All lines above the line on which the cursor is positioned are designated as the header section.

To discard the heading section, move the cursor to the top line of the form and run the command

Footer Section

All lines below the line on which the cursor is positioned are designated as the footer section.

To discard the footer section, move the cursor to the last line of the form and run the command.

Miscellaneous Commands (Info Pull- down Menu)

Form Status ([Report form] Info ⇒ Form status)

Display the form name, the form type (Report) and the current status of the form, if applicable.

The form status flags are:

[Read only]: The form is read only and editing can not be performed.

[Modified]: The form has been modified since last being written to file.

If neither of these conditions apply, the status will not display.

Validate Form ([Report form] Info ⇒ Validate form)

A message will display:

```
Validating "formname" [Report form] . . .
```

A validation report is generated and displayed on the screen a page at a time (Pull down the Function menu to scroll through the pages). The validation report lists items in form sort order and describes their attributes. It is possible to have a copy of the report written to a file or sent to the printer. (See Printing or Copying Information in Chapter 2).

After the validation report has been viewed, a message is displayed indicating the validity of the form. (Report forms are always valid.)

Display Form Coordinates ([Report form] Info ⇒ Form Coordinates)

This command displays the current form size, form view and cursor coordinates. The display has the format:

Size (*lines*, *cols*), View (*y*, *x* : *my*, *mx*), Cursor (*y*, *x*)

The form size states the number of lines and columns in the form. The form view indicates the current form window coordinates. The cursor coordinates list the current form line and column position of the cursor.

Change Privilege Level View ([Report form] Info ⇒ View at Level)

For databases configured for multiple privilege levels, it is possible to view the form as it would be displayed for each distinct privilege level. This command is disabled if the database has only one privilege level. Privilege level scope is hierarchical. For example, level 0 encompasses all levels 0 to n; level 1 encompasses all levels 1 to n, and so on. Where n is the lowest privilege level for the database. Privilege level 0 is the most powerful, and privilege level 8 the most limited.

When the command is run, the system prompt for the desired privilege level viewing:

Viewing at privilege level *c*. Change to [0 - *n*] :

where ***c*** represents the current privilege level and ***n*** the maximum permissible privilege level as specified in the database configuration. When the desired privilege level for viewing the form is entered, only those objects which are included in that privilege level are displayed. If required, the form may be scrolled in order to view other areas of it. If the privilege level specified is anything other than level 0, *read only* forms editing is invoked, i.e. the form cannot be modified

Copy Form ([Report form] Info ⇒ Copy Form Image)

This calls up a selection box, and the user can specify the file or printer to which a copy of the form will be sent. The entire form is copied/printed, regardless of the number of screens it encompasses. (See Printing or Copying Information in Chapter 2 for details on using the selection box).

Exit the Report Form

Save Form ([Report form] Function ⇒ Save Form)

This command is used to save any changes made during the form editing session. While the form is being written, a message displays:

```
Writing "formname" [Report form]...
```

Exit ([Report form] Function ⇒ Exit)

Exit from forms editing. If the form had been modified since the last save, the system will first request confirmation:

```
"formname" [Report form] [Modified]. Really exit?
```

Choosing yes exits the Report form. Choosing no aborts the exit command and the Report form will continue to display.

Interrupt a Command

An interrupt can be generated by typing the **DEL** key (or the key that is set up to generate an interrupt). Interrupts are permitted during non-critical stages of form development. A message will display, the current command will be aborted and the form restored to its previous state. If interrupts are disabled, then pressing the interrupt key has no effect and the current command continues.

Rename Report Form ([Display] Reports ⇒ Rename form)

Change the name of an existing Report form. Only users with Report create/ delete privilege can perform this operation. A selection box will display all the existing reports, highlighting those reports owned by the user. Only these reports can be renamed.

To select a report, the user need only move the cursor to it using the ↓ and ↑ arrow keys, and press ↵. This will call up another selection box and the user should enter the new report name into the lower box. (Enter **Ctrl+V** or **Ctrl+W** to move between the upper and lower box). Note that the user must enter a new name into the lower box. A message will then display, confirming the name change.

Delete Report Form ([Display] Reports ⇒ Delete form)

Only users with Report create / delete privilege can perform this operation.

A selection box will display all the existing reports, highlighting only those reports owned by the user (and thus deletable). To select a report, the user need only move the cursor to it using ↓ and ↑ arrow keys and press ↵. The system will prompt for confirmation and if the user chooses yes, will display a message confirming the deletion.

View Report Form ([Display] Reports ⇒ View form)

This command allows users to view all reports set up for this database, regardless of their owner.

A selection box will display all the existing reports. To select a report, the user need only move the cursor to it using the ↓ and ↑ arrow keys and press ↵.

The report will then display, in Report form mode. The report cannot be changed here. To change the report, use the Create/Edit command on the Reports pull-down menu. See Create/Edit report in this chapter.

Copy Report Form ([Display] Reports ⇒ Copy Form)

Copy an existing Report form. Only users with Report create/delete privilege can perform this operation.

Any report form, regardless of who it is owned by, can be copied.

The command causes a selection box to pop-up, listing all existing report forms. To select a report, users need only move the cursor to it, using the ↓ and ↑ arrow keys, and press ↵. This causes another selection box to display. The user should enter a new name into the lower box (use **Ctrl+V** or **Ctrl+W**) and press ↵. A message will then display, confirming the copy.

Generate Reports

To generate a report, it is necessary to first identify the records to be reported. This is achieved by performing one or more queries. A report can then be generated on the record currently being viewed, or on all the retrieved records. While in Display mode, the user can select on of the following commands:

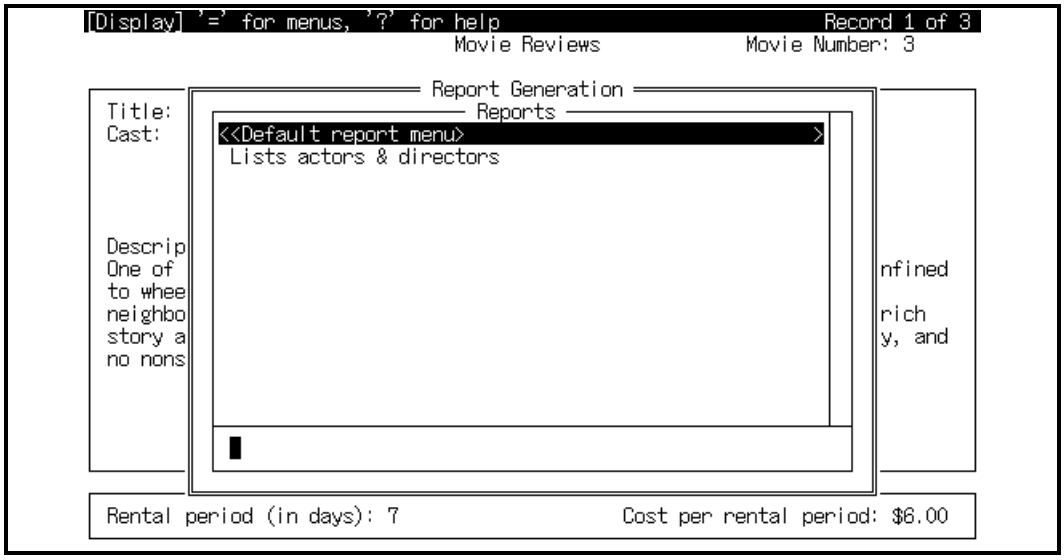
Report all Matches ([Display] Reports ⇒ Report all Matches)

Generate a report of all retrieved records.

Report this Match ([Display] Reports ⇒ Report this Match)

Generate a report of the record currently being displayed.

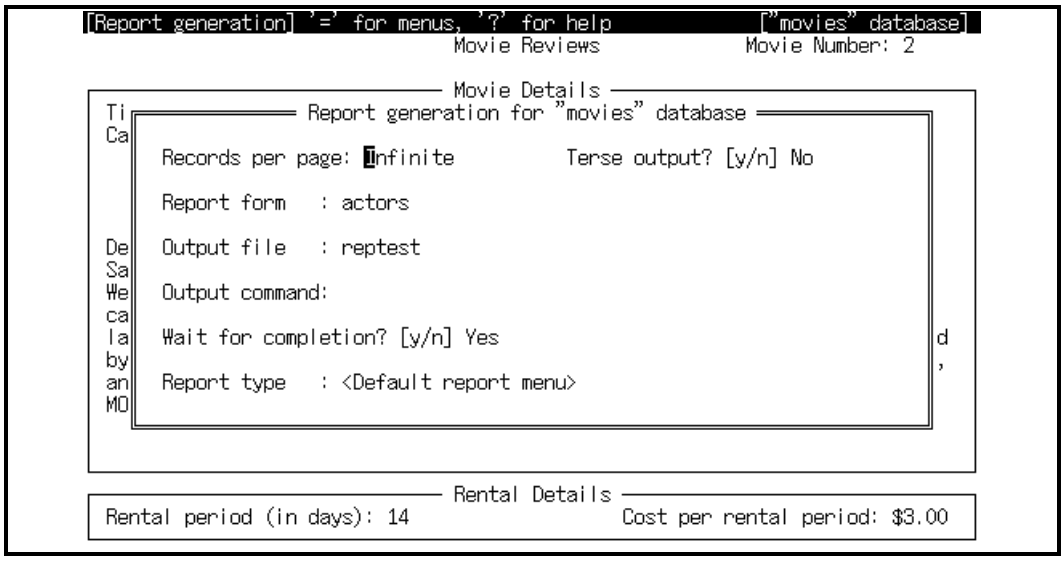
Both the "Report all matches" and "Report this match" commands cause the pre-defined report generation options (set up by the DBA) to display as in the following figure:



Report generation

To select a format, move the cursor to it using the arrow keys and press ↵.

This calls up a report generation form, where the user can specify various report options, as shown in the following figure:



Report Generation Form

The report generation options are set out below:

Records per page

This option provides a simple way of paginating the report output. It indicates the maximum number of records to appear on each page of output from the report generator (only the last page can have fewer records than this). Each Report form can have heading and/or footer sections which appear once at the top and bottom of each page respectively. In between these sections is the data for each of the records to be printed on that page. A form-feed character is generated between pages to ensure that, at least on printers, the next page is correctly aligned.

This option is, naturally, numeric and thus accepts only a positive integer value. If, however, the value entered is **0,** it is replaced by the word **Infinite.** This indicates that the heading and footer sections are to appear only at the start and end of the report output respectively. The output is not paginated in this case. The initial value of this option is **Infinite.**

Terse output

The **Terse output** option provides an abbreviated, more readable output format, which is suitable for records which are ordered on some field(s) and where those fields have much data in common. If the first ordered field value of the next record to be printed is identical to the corresponding value in the previous record, then the value is not output ,but rather is left blank. If the first ordered field is left blank, then the second ordered field is tested against the corresponding value in the previous record. If it too is identical, then it is also left blank. This continues on through the items in the form sort order. The last item in the sort order is always output.

When the report is also paginated, then the first record on every page is output in full. Terse Output then continues from this record.

For Terse mode reporting, the KE Texpress Library items are considered to be one field. The initial value of this option is **No**.

Report form

Indicates the name of the Report form which contains the specification of the report. A Report form name is compulsory.

Output file

This option allows the user to specify a file name for capturing the output of the report generator. The file is created if required. If it existed previously then it is over written. An Output file name is optional.

Output command

Specifies a standard Unix command line which takes the output of the report generator as its input. This is one method of printing a report as it is produced, or perhaps of viewing it a page at a time. A typical Unix command which could be used is **lp,** which sends the resulting output to the default printer.

Another command which could be used is **awk,** which is a Unix report writing utility. The output from KE Texpress's reporting facility can be shaped to provide convenient input for awk, thus simplifying the development of complex reports. For more information on awk, refer to the Unix manuals.

An Output command is optional.

Wait for completion

As the report generator can produce output to the screen, the texforms has the option of waiting for the report to complete, or performing the report generation in the background.

In the latter case, the user's terminal is then freed up so the user can do other tasks. A message will display when the report is completed. A report which outputs information to the screen should not be performed in background, as it may result in loss of control of the terminal and jumbled terminal settings. The initial value of this option is **Yes**.

Report type

This field displays the name of a particular report format, if one has been selected. A report format is a pre-defined set of report generation options which is referenced by name. If report formats have been created by the DBA, the Look-up command can be used to view and select a report format.

To enter a value for a particular option, it is necessary only to move to that option and enter the appropriate text. (See Field Editing Commands described in Chapter 3).

The Terse output and Wait for completion options differ from the other options in that they can assume only one of two values, **Yes** or **No**. Thus the normal editing commands are disabled. The user can enter `Y` for Yes or `N` for No.

Moving around the Report Generation Form

Forward ([Report Generation] Function ⇒ Foward)

Move forward to the next option on the form. The option after the last option is the first.

Backward ([Report Generation] Function ⇒ Backward)

Move backward to the previous option on the form. The option before the first option is the last.

Generate Report ([Report Generation] Function ⇒ Perform)

When all of the options are set as required, the command is used to generate the report. This command first ensures that a valid Report form has been specified. If an error is found then an "invalid Report form" message is displayed and the user is returned to the Report generation form. The Output File and Output Command options are optional. If there are no errors, the screen is cleared and a message displays:

```
Performing report generation . . .
```

If the user is waiting for completion, another message displays at the bottom of the screen when report generation is complete:

```
Report generation complete - [ Hit any key to  continue ]
```

If the user is not waiting for completion, the report is generated in background, and the user immediately returns to Display mode. When the report has completed, a message appears on the screen:

```
Your "formname" report has completed
```

To redraw or refresh the screen (remove the message), enter **Ctrl+R**.

A report which outputs information to the screen should not be performed in background, as it may result in loss of control of the terminal and unusual terminal settings.

Exit ([Report Generation] Function ⇒ Exit)

Exit from Report generation (without generating a report) and return to viewing the record(s) in Display mode.

Special Report Operators

KE Texpress's reporting facility provides numerous operators to aid in the creation of specialised reports, such as address labels, catalogue cards and specimen tags. These operators are invoked by embedding special character sequences into Report forms. The user must first call up the Report form (**[Display]Reports ⇒ Create/edit**) and then insert these special operators into the Report form as headers or prompts. (See also Creating or Editing a Report Form at the start of this chapter.) The operators can also be embedded into record data, although this practice is not recommended.

During report generation, these embedded character strings invoke particular functions, but are themselves always removed from the form before the record is output.

During Display mode, the embedded character sequences are not processed, and are displayed as normal headers.

The report operators are listed below, with examples of a Report form, data, and report output. The examples contain first an image of the Report form in use. This includes any prompts and fields and the header or headers necessary to invoke the operator. After this is an example of one data record which is to be loaded into the form. Finally, the actual output from the report generator is displayed.

Space Removal: $S

Remove all spaces between the last non-blank character and this string.

Example, the form:

Name:_____$S

with the data:

Bill Smith

results in the output:

Name:**Bill Smith**

Space Replacement: $X

Replace the string with any spaces removed from the line by preceding **$S** operators.

Example, the form:

```
Name: _____[$S_____$X]
```

with the data:

```
Bill          Smith
```

results in the output:

```
Name:[Bill Smith]
```

Line Removal: $L

Remove this line only if it is empty. This operator must appear at the start of the line.

Example, the form:

```
$L_____________________
$L_____________________
```

with the data:

```

Bill
```

results in the single line of output:

```
Bill
```

Prompt Removal if there is no data: $P ... $P ... $E

If there are no non-blank characters between the second **$P** and the **$E** (typically the field of the item) then everything between the first **$P** and the **$E** is removed (typically the prompt and its field).

Example, the form:

```
$PTitle$P _____$E
$L$PAge$P_____$E
```

with the data:

Sir

results in the single line of output:

Title **Sir**

Line Replacement: $F

After this line, insert a blank line for each line removed by any of the operators (such as **$L**).

Example, the form:

```
$L $P Month $P_____$E
$L $P Year $P _____$E$F
```

with the data:

1960

results in the two lines of output:

Year **1960**

Line Joining: $J

Precede the string with the next line of the Report form, only if the resulting line width would be less than or equal to the width of the Report form.

Example, the form:

```
Name___$S____$S $JYear___
Month___$PDay of the month$P ____$E
```

with the data:

```
            Bill  Smith
1980
            Mar
```

results in the single line of output:

```
Name  Bill Smith Month Mar Year 1980
```

Line Joining with Word Wrap: $W

Replace this string with the next line of the Report form. Interpret the resultant line as text and if it is wider than the Report form word wrap it on to the next line.

Example, the form:

```
Comments  _____$S $W
_____$S $W
_____
```

with the data:

```
            This is a long
sentence broken into
several lines of text.
```

results in the output:

```
Comments  This is a long sentence
broken into several lines of text.
```

Query Replacement: $Q

This operator may appear anywhere in the Report form. It is replaced by a representation of the query performed to capture the records for report generation. Each field in the query which had a value specified, is output one per line (in form order) alongside the item prompt (or item Id if the prompt is null).

Typically this operator would be used in the heading section of a report.

The standard output format can be adjusted by a DBA by editing the database text file. Refer to the KE Texpress Interface Guide for more details.

Record Number: $N

Replace this operator with the number of the record within the report, as it occurs. (At the completion of the report, this value equals the total number of records processed by the report).

Multiple Column Output Support: $C*n*

This operator may appear anywhere in the Report form. It ensures that a multiple of ***n*** records is output by the report generator. If additional records are required, a series of blank lines (the number of lines in the Report form) are output. This can be useful when producing ***n***-column output using the Unix utility, **pr** (refer to the Unix manual) where on some systems, each column must have the same number of entries.

Embedded Control Characters: $O*nnn* :*m*

Substitute the octal character, ***nnn*** (exactly three digits), into the output and assume that it consumes ***m*** character positions. This is useful for embedding printer escape sequences into Report forms for producing effects such as bolding, italics, special symbols, etc.

Report Pagination - texpaginate

KE Texpress provides a facility to perform more sophisticated pagination of reports which can be used when the "Records per page" option of report generation is not sufficient.

The facility, called **texpaginate**, is a filter which acts in a fashion similar to the Unix command **pr**. The **texpaginate** program reads the standard input and writes to the standard output. It can be invoked from the shell, or included in the "Output command" option on the report generation form.

The **texpaginate** program is invoked by a command of the form:

```
tpr [-wn] [-ln] [-ohn] [-ovn] [-h'l'c'r'] [-f'l'c'r']
```

The command line options are:

-w*n* Set the page width to *n*. The default is 79 characters.

-l*n* Set the page length to ***n***. The default is 62 lines.

-oh*n* Set the horizontal page offset to ***n*** characters. The default is 0 characters.

-ov*n* Set the vertical page offset to ***n*** lines. The default is 0 lines.

-h, l, c, r,

Define a headline to contain three text components with content l (between the first 2 commas), c (between the second and third commas) and r (after the third comma).The text *l* is left justified, *c* is centred and *r* is right justified with respect to the page width.

-f, l, c, r,

Define a footline to contain three text components with content l (between the first 2 commas), c (between the second and third commas) and r (after the third comma). The text *l* is left justified, *c* is centred and *r* is right justified with respect to the page width.

-d *dbname* Use the message files of this dbane *database*

The texpaginate pagination program provides the following facilities:

texpaginate - Headlines and Footlines

A headline appears at the top of every page, followed by a blank line. A footline appears at the bottom of every page and is preceded by a blank line. Each may contain a left justified component, a centred component and a right justified component. Any or all of the components may be empty. A headline and a footline is designated the required text in the appropriate three part command line argument to texpaginate.

Each text component in the headline or footline may contain one or more of the following conversions: (Note that text components should be enclosed in double quotes)

%h The current hour (12 hour clock).

%H The current hour (24 hour clock).

%n The current minute.

%a "am" or "pm" as appropriate.

%A "AM" or "PM" as appropriate.

%d The day of the month.

%m The current month as a number.

%M The current month as a 3 character string.

%L The long (full) name of the current month.

%y The current year (2 digits).

%Y The current year (4 digits).

%p The current page number.

%% A % character.

E.g. to display a report on the screen with the current date in the format 1 January 1900 centred in the headline, the following command can be entered in the command-line of the report generation program:

```
texpaginate -h, ,"%d %L %Y", , | more
```

texpaginate - Data Sections

A data section is intended to represent the contents of each record. It is introduced by a line containing only the letter **D** at the start of the line. All subsequent lines, up to the next section marker, are considered to be the one record.

If a record does not fit on the remainder of the current page, then the next page is commenced (after outputing the footline, a page feed and the headline).

texpaginate - Header and Footer Sections

The header section is intended to represent the information to be placed at the top of each page. It is introduced by a line containing only the letter **H** at the start of the line. All subsequent lines, up to the next section marker, are considered to be the header.

Whenever a new page is commenced, the current header section is output (immediately after the headline). The header section can be redefined by the appearance of a second line containing only an **H** character.

Similarly, the footer section is intended to represent the information to be placed at the bottom of each page. It is introduced by a line containing only the letter **F** at the start of the line. All subsequent lines, up to the next section marker, are considered to be the footer.

Whenever a page is completed, the current footer section is output (followed immediately by the footline). The footer section can also be redefined.

Both header and footer sections are optional.

texpaginate - Start and End Sections

The start section is output after the headline but before the header section, on the first page only. This can be useful when it is necessary to include the specified query in a report. It is introduced by a line containing only the letter **S** at the start of the line.

The end section is output after the all data has been read and the footer section has been output. It is generally used to output calculated totals, etc. It is introduced by a line containing only the letter **E** at the start of the line.

Both start and end sections are optional.

texpaginate - Report Section

The report section is included only if a line containing a single letter **R** at the start of a line appears in the input data stream. The report section simply appends after the end section a line indicating the number of data sections processed.

The report section is optional.

Line Drawing - texcat

Ke Texpress provides a facility to incorporate the line and box drawing capabilities of the user's printer into the report generation.

Texcat is invoked by a command of the form:

```
texcat [-d] [-otermtype] [files]
```

where

-d uses the default options, of + in corners, - as a horizontal line and | as a vertical line.

-o*termtype* specified The printer's line drawing facilities must first have been in a file called *termtype* and compiled using **texterm.** See Line Drawing Character Set and Installing a Terminal Definition in Chapter 5 of the KE Texpress Interface Guide.

files Get the data input from this file. It is not necessary to use this option if **texcat** is run from report generation, as the retrived record(s) will be the input for **texcat**.

Chapter 7

Miscellaneous Operations

Database Usage Statistics....7-3
Perform a Unix Command....7-4
Version Number....7-5
Listing Databases....7-5
Database Options....7-7
- Database Pathname....7-8
- Database Component Pathname....7-8
- Linked Database Components....7-9
- General Pathnames....7-10
- Screen Copy....7-10
- File Copy....7-11
- Insert Mode....7-11
- Query Mode....7-12
- Display Mode....7-12
- History....7-15
- Global Edit....7-15
- Reports....7-15
- Rebuild, Load and Batch Insertion....7-17
- General....7-18

Incorrect Option Settings....7-23

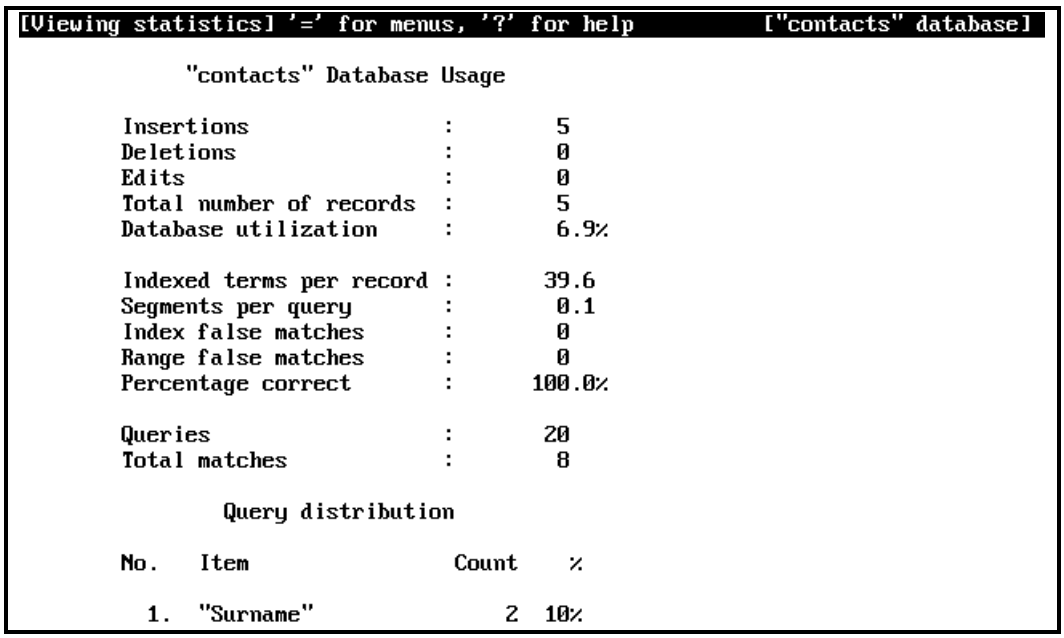
Overview

This chapter describes the use of each of the remaining facilities which can appear on the Query Mode menus. Also described is a facility for listing database names, owners and other information. The final section of this chapter provides information on the numerous database options incorporated into the KE Texpress system

Database Usage Statistics ([Query] Texpress ⇒ Database usage)

Display a summary of statistics indicating database usage. The exact format of the statistics displayed is dependent on the database usage to date.

An example is given in the following figure.



Database Usage

The display shows three blocks of information. The first block lists the total number of insertions, deletions and edits, the total number of records in the database, and the percentage of the database capacity used so far.

The second block lists, for the DBA only, the average number of indexed terms per record and the average number of segment matches per query. Segment matches are described in more detail in the KE Texpress Design Guide. For all users of the database this block lists the number of index and range false matches found, and the percentage of all matches that were correct (i.e. not an index false match).

The third block of information lists the total number of queries performed and the total number of matches.

Finally, if there has been at least one (non-trivial) query 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.

Perform a Unix Command ([Query] Texpress ⇒ Run command)

A selection box will appear, headed run a command. In the box is a prompt :

```
Enter command:
```

Once the required command is entered, the screen is cleared and the command executed. On the completion of the command, any key can be typed to return to the Query Mode.

The command is executed via the preferred shell for the current Unix account. Hence it can assume any standard path or utilize any aliases. It can also take advantage of the tilde expansion facility. The preferred shell is selected in the following manner:

(1) If the Unix environment variable, SHELL is set, then this program is used.

(2) If this environment variable is not set, then KE Texpress looks in the password file for the account's login shell and executes this.

(3) If the password file has no shell for the account, then /bin/sh is executed.

Display Version Number ([Query] Texpress ⇒ About KE Texpress)

Display the current KE Texpress version number.

Listing Databases

People with access to KE Texpress database also have access to a program which lists information on all KE Texpress databases. This program is called **texlist** and can be executed from the Unix shell.

In its simplest form, this program is invoked by typing the command line:

texlist

This lists in alphabetic order, the names of all KE Texpress databases with it's DBA's Unix user Id in brackets. The database names are output in two column format.

Several options alter the output of this program. In general, it can be invoked by a command of the form:

texlist [-bdlnprs] [-u*uid*] [*dbname* ...]

Sections enclosed by [and] are optional. The character string, **bdlnprs**, represents a set of single character options which can be selected in any combination. The full list of options is as follows:

-l Produce a long listing of the databases, 1 database per line. In addition, a string of the form displays for each database:

***nnn* records *mm.m*%**

The figure *nnn*, represents the number of records in the database and the figure *mm.m* represents the percentage of database capacity which has been used. For uninitialized databases, the record count and the percentage of capacity figures are replaced by the text:

Uninitialized

-b Sort the databases numerically by the total size, in (bytes) of all of the files which make up each database. The smallest database is listed first. Databases of the same size are sorted by name.

-d Sort the databases alphabetically by the DBA's Unix User Id. Databases with the same DBA are sorted by database name.

-n Sort the databases numerically by the number of records in each. The smallest database is listed first.Databases with the same number of records are sorted by name.

-p Sort the databases numerically by the percentage of capacity used. The smallest database is listed first.Databases with the same percentage of capacity used are sorted by name.

-r Reverse the order, i.e. alphabetic lists are produced in reverse alphabetic order and numeric lists are produced in decreasing order of value.

-s Display the total size, (in bytes) of all of the files which make up the database. The size is displayed in the form:

s.nnc

where *s.nn* is a real number indicating the size of the database to two decimal places, If *c* is **b**, the size is given in bytes. If *c* is **K**, the size is given in kilobytes (1024 bytes). If *c* is **M**, the size is given in megabytes (1048576 bytes).

-u*uid* Restrict the output to databases for which *uid* is the DBA. This option can appear several times on the command line, in which case the output is restricted to databases for which the DBA is one of the *uid*s specified. If *uid* is not a valid Unix Id then an appropriate error message is generated along with a usage message.

If two or more conflicting sorting options are specified on the command line, only the last of these options is performed.

Finally, one or more database names can be used, after any options, on the command line, in which case the output is restricted to information concerning those databases only. If an invalid database name is given, then an appropriate error message is generated and the name is ignored.

Database Options

Setting Database Options

It is possible to alter the operation of KE Texpress by changing the **database options**.

The user/database options, ie. those for a particular Unix user and a particular database, may be altered using the ***dbname*opts** environment variable. For csh, the command is:

```
setenv dbnameopts "option1 option2 ..."
```

and for sh, the commands are:

```
dbnameopts ="option1 option2 ..."
export dbnameopts
```

The user-wide options, ie. those for all databases for a particular Unix user may be altered using the **TEXPRESSOPTS** environment variable. For csh, the command is:

```
setenv TEXPRESSOPTS "option1 option2 ..."
```

and for sh, the commands are:

```
TEXPRESSOPTS ="option1 option2 ..."
export TEXPRESSOPTS
```

If environment variables are to be used, they are generally set in the **.cshrc** (for csh) or **.profile** (for sh) file for the appropriate Unix User so that they become set whenever that Unix user logs in.

Database-wide options, ie. those for anyone using a particular database, may be altered by the DBA. Refer to the KE Texpress Design Reference Guide for more information.

System-wide options, ie. those for anyone using any KE Texpress database, may be altered by the superuser or the user, **texpress**. Refer to the KE Texpress Design Guide for more information.

System-wide options are evaluated first, followed by database-wide options, then user-wide options and finally user/database options. More than one of these methods may be used to set options. It may follow that a particular option is set to different values in the several methods used. The resulting value for that option is the one last evaluated, e.g. user/database options over-ride system-wide ones.

For options that are assigned a textual value, the text should be enclosed within double quotes if it contains more than one word.

The entire list of database options is listed below. Some of these options, however, are applicable only to DBAs.

Database Pathname Options

The following options define the pathnames of directories where databases are located. It is strongly recommended that absolute pathnames be used:

dbpath=*pathname*

The value, *pathname*, defines the directory under which there is a directory in the database name, in which all files in the database reside. The dbpath option provides a sweeping method of setting all the options, datapath, filepath, keypath, lutpath and filepath, to the same directory.

linkdbpath=*pathname*

The value, ***pathname***, defines the directory under which there are directories in the names of linked databases, in which all files in those linked databases reside. The linkdbpath option provides a sweeping method of setting all the options, linkdatapath, linkkeypath and linkfilepath, to the same directory.

Database Component Pathname Options

The following options define the pathnames of directories where component files of databases reside. Again it is strongly recommended that absolute pathnames be used:

datapath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which the data file in that database resides.

filepath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which all files in that database, other than those defined by the options datapath, keypath, lutpath, recpath and segpath, reside.

keypath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which the key file in that database resides.

lutpath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which the lookup table files in that database reside.

recpath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which the record descriptor file in that database resides.

segpath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in the database name, in which the segment descriptor file in that database resides.

Linked Database Component Pathname Options

The following options define the pathnames of directories where component files of linked databases reside. Again it is strongly recommended that absolute pathnames be used:

linkdatapath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in each linked database name, in which the data file in the linked database resides.

linkfilepath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in each linked database name, in which all files in the linked database, other than those defined by the options linkdatapath and linkkeypath, reside.

linkkeypath=*pathname*

The value, ***pathname***, defines the directory under which is a directory in each linked database name, in which the key file in the linked database resides.

General Pathname Options

The following options define the pathnames of files not in databases. It is strongly recommended that absolute pathnames be used:

duppath=*pathname*

The value, ***pathname***, defines a file which is updated with information concerning every insert, edit and delete operation performed on the database. This information is stored in a format suitable for loading (using **texload**) into another KE Texpress database. This option can be used to provide a dynamic duplication facility.

pkgpath=*pathname*

This option has been provided for use within KE Texpress applications and may be used increasingly in future applications. A current use is that **pkgpath** may be set to the pathname of the application directory, under which a directory named **help** may be created. Under this directory reside files containing the text of help messages, to be accessible using the help command while an option on a front-end menu is selected.

tmppath=*pathname*

The value, ***pathname***, defines a directory in which KE Texpress temporary files are created.

Screen Copy Options

screencopy=all
screencopy=block
screencopy=short

This option is used when a copy of the screen is requested. The default value is **all**. In this case, in a screen copy, the entire screen including the menu bar, is copied. When **screencopy** is set to **block**, Texpress prompts the user to mark on the form window the top left and bottom right corners of the block to be copied. When the option is set to **short**, only the form window is copied. The menu bar is not copied.

File Copy Options

The following options are used when information is copied to a file:

file=append
file=ask
file=overwrite

The default setting is **ask**. When an attempt is made to copy information to an existing file, if the option is set to **ask**, Texpress prompts for one of three possible actions, append, overwrite, or exit. When the option is set to **append** or **overwrite**, the append or overwrite action is automatically carried out.

filename=*pathname*

This option defines the name of the file which is automatically used for record and screen copies, as well as for copying from text perusal.

Insert Mode Options

autokey=next
autokey=no
autokey=personal

The default setting is **next**. This setting ensures that the next available Key value will be assigned to a record in Insert mode. If the option is set to **no**, this automatic assignment will not be done. If the setting is **personal**, the Key value assigned will be the personal next value for a given user.

autokeytries=*nnn*

The default value for this option is **500**. When assigning the next available Key, Texpress scans forward until a valid Key value is found. However it is possible that the validation constraints preclude all Key values. To avoid infinite testing of Key values, Texpress is limited to ***nnn*** attempts, scanning forward, to assign a Key value. If all of these attempts should fail, an appropriate error message is given and no Key value assignment occurs. The value of this option is also used as a limit when scanning forward through key values in Peruse mode. Texpress is limited to ***nnn*** attempts to find the next Key value when searching for the next record to peruse.

inspriv=*n*

This option is used to set the insertion privilege level to ***n***. The default value is the level of least privilege for the specific user in the specific database. If ***n*** is a value lower than the lowest level in the database is use, then it is reset to that lowest level. If ***n*** is a value higher than the

working privilege level for the user, then it is reset to that working privilege level.

Query Mode Options

fldbased=no
fldbased=yes

The default setting for this option is **no**. When a value is entered in one of several fields of an item in Query mode, the records retrieved from the query will be those where that value occurs in any field of the item if **fldbased** is set to **no**, but only those where the value occurs in that specific field if **fldbased** is set to **yes**.

matchfreq=*nnn*

The default value for this option is **10**. Its value specifies that, during retrieval or global edits, the displayed count of the number of records processed is updated whenever that number increments by ***nnn***.

qryform=*string*

There is an un-named default query form, but other named query forms may be designed. This option is used to set the query form to ***string***.

Display Mode Options

chronolog=edit
chronolog=insert

By default, records displayed on an unsorted display form are not in any particular order. However the **chronolog** option may be used to specify that the records currently in the database are to be displayed in the order in which they were edited, or in the order in which they were inserted. In order to achieve this order an index rebuild must be performed after the option is set to either **edit** or **insert**.

direction=lefttoright
direction=righttoleft

The default value for this option is **lefttoright**. It specifies the direction in which characters in all strings displayed on the screen by KE Texpress are to be ordered. It may be set to **righttoleft** when KE Texpress is used with languages in which this is the natural direction of reading.

dispcache=*n*

The default value for this option is **1**. Whenever a switch occurs between Display forms, by default the new form is read from its file. This option can be used to specify that KE Texpress keep the last ***n*** accessed Display forms in a memory cache to allow faster switching between forms.

dispform=*string*

The default display form is the insertion form. This option is used to set the default display form to ***string***.

matchhold=*nnn*

During Display mode, KE Texpress retains some information, referred to as a record description, about each matching record. Physical limitations impose a limit on the number of record descriptions which can be held in memory. If the matches from a query exceed this limit, then the record descriptions are written to a temporary file. The effect of this is that the query may become a litttle slower. The value of matchhold is set, to **1024** on most machines, at the time of installation of KE Texpress, but may subsequently be altered. It should be noted that making this value too high can be taxing on the machine's resources and can result in a degradation of KE Texpress performance generally. The total memory of the machine should be considered before altering the value from its initial setting.

matchlistby=*itemid*

When this item is set, the retrieved matches are automatically sorted, before display, on the item with id ***itemid***.

matchsearch=forward
matchsearch=backward

The default value for this option is **forward**. By setting the option to **backward**, the order of the retrieved records may be reversed.

matchshow=auto
matchshow=jump
matchshow=no
matchshow=yes

The default value for this option is **yes**. The values **yes** and **no** specifiy whether or not terms matching the query terms are highlighted. The value **auto** specifies that, when displaying a record, the form window is to be repositioned so that the first matching term in the record is included in that window. The value **jump** applies only to very slow machines, where the time taken to identify a matching term and

reposition the window is excessive. This value specifies that the first screen of the record is to be displayed before the window is repositioned to include the matching term.

matchsort=ascending
matchsort=descending

The default value for this option is **ascending**. This option is used only in conjunction with **matchlistby** or **matchsummary**. When one of these options is set to an item id, by default the order of sort used is ascending. If the option **matchsort** is set to **descending**, the order of sort is reversed.

matchsummary=*itemid*

When this item is set, the retrieved matches are automatically summarised, before display, on the item with id ***itemid***.

matchverify=view
matchverify=yes

The default value of this option is **view**. The value determines whether or not each retrieved record is checked for exact matching before all matches are displayed, or only, as is the default, when that record is to be viewed or used in some other way. KE Texpress uses a fast retrieval mechanism which retrieves partial matches. For a well configured database, there should be fewer than 0.1% false matches. (For range queries, however, this percentage can be higher, but in this case depends on the configuration and the query.) When any operation is performed on the retrieved records subsequent to the initial retrieval (report generation, scanning through the matches on the screen and so on) the records are checked to ensure that they are true matches before they are processed by the operation, whether that is Display, Report Generation, Summary, or anything else, and any that are not exact matches are discarded. The Misc menu in Display mode provides a command to verify the matches. Each retrieved record is checked and false matches are discarded. Exact matching is time consuming, and full advantage is not taken of the fast retrieval capability of KE Texpress. It is preferable to ensure that the configuration is such that the incidence of false matches is negligibly low. However if it is essential that exact matching occurs with each retrieval, the **matchverify** option may be assigned the value **yes**.

History Options

history=ask
history=no
history=yes

The default value for this option is **no**. In this case, when a record is saved after an edit, the old (history) version of the record will not be saved. If the value is yes, the history version is automatically saved after an edit. If the value is **ask**, when a record is saved after an edit, KE Texpress questions whether the old version is to be saved, and requests a yes or no answer.

Global Edit Options

globeditval=ask
globeditval=no
globeditval=yes

The default value for this option is **ask**. In this case, whenever a global edit occurs, KE Texpress questions whether records are to be validated and requests a yes or no answer. The **globedit** option may be set to either **yes** or **no**, to avoid the question, and specify that validation either will or will not be carried out with each global edit.

Report Options

When the Default report menu is selected to report one or all matches, KE Texpress enters Report Generation mode and displays a screen on which information must then be entered before report generation is performed. The information consists of the records per page, whether terse or not, the report form, the output file, the output command, and whether or not to wait for completion of the report. Each of these may be set to a default value which will be entered automatically on the screen in Report Generation mode, using some of the options listed below. In addition it is possible to avoid entry to the Report Generation mode screen by creating a list of pre-designed report types. This list then provides alternatives to the Default report menu and automates report generation. There are options associated with the use of pre-designed report types.

repcmd=*cmd*

This option sets the default output command in report generation to ***cmd***.

repfile=*pathname*

This option sets the default output file in report generation to ***pathname***.

repform=*name*

This option sets the default report form in report generation to ***name***.

repmanual=no
repmanual=yes

The default value for this option is **no**. If pre-designed report types exist, when the command to report one or all matches is used, KE Texpress displays a selection box containing a list of the report types. By default, if a pre-designed report type is selected, KE Texpress does not proceed to the Report Genertaion mode screen on which values such as the report form and output command are entered. Instead, KE Texpress uses the pre-designed values for these entities and automatically performs the report generation. However, if the **repmanual** option is set to **yes**, when a pre-designed report type is selected, KE Texpress does display this screen, and any of the pre-designed values already automatically entered on the screen may be added to, or overwritten with other values. The report generation does not proceed until the command to perform it is given.

repmenu=no
repmenu=yes

The default value for this option is **yes**. When the command to report one or all matches is given, KE Texpress displays a selection box of report types, which contains any pre-designed report types, and, by default, an option providing access to the Default report menu. If the option repmenu is set to **no**, the option for the Default report menu is not listed.

repperpage=*name*

This option sets the records per page in report generation to ***nnn***.

repterse=no
repterse=yes

The default value for this option is **no**. In this case the report output is by default not terse. Terse output by default may be specified by setting this option to **yes**.

repwait=no
repwait=yes

The default value for this option is **yes**. In this case the wait for completion value is by default yes. Setting repwait to **no** sets the default wait for completion value to no.

Rebuild, Load and Batch Insertion Options

batch=exclusive
batch=shared

The default value for this option is **exclusive**. In this case, prior to commencement of a batch insertion, the database is closed, so that the insertion procedure can be completed as quickly as possible . If the option has the value **shared**, the database is not closed and the batch insertion progresses much more slowly.

loadfreq=*nnn*

The default value for this option is **100**. During a rebuild or a load, the monitoring program displays the number of records processed, and the percentage of the whole database processed. The value ***nnn*** specifies the number of records processed between updates of this screen information.

loadmemory=*nnn*

The default value for **loadmemory** in each KE Texpress installation is set at the time of installation, usually at **1024**, and specifies, in kilobytes, the memory allocated by the data rebuilding and data loading programs. It may subsequently be altered if necessary to improve the speed of these operations. The value should not be excessive relative to the total memory of the machine, or degradation of the performance of the machine will occur during a rebuild or load.

loadtemp=no
loadtemp=quiet
loadtemp=yes

The default value for this option is **yes**. When the program **texload** is used to load data into a KE texpress database which has a Key, by default if a record fails validation, it is saved in the temporary file and is reported in the error report. If the option **loadtemp** is set to **no**, any record failing validation is not saved in the temporary file, but is reported in the error report. If **loadtemp** is set to **quiet**, a record failing validation is not saved in the temporary file, and is not reported in the error report.

General Options

bells=no
bells=yes

The default value for this option is **yes**. Certain status and error messages in KE Texpress provide for the terminal bell to sound. By default this bell sound occurs with each such message. If the **bells** option is set to **no**, the bell sound does not occur.

cmd=command

The KE Texpress pull down menu contains a Unix command option. The option **cmd** is used to set the default value for this Unix command to ***command***.

copyform=*string*

The command to copy all matches or copy the current match may be chosen from the Show menu in Display Mode. The data is copied on a form which is by default the Insertion form. When the error file from a rebuild or data load shows record data in appropriate fields of a form, the form used is by default the Insertion form. However in both cases the form used may be specified to be a particular display form by setting the **copyform** option to the name of that form.

dbname=*name*

This option is unusual in that it is not set explicitly. The dbname option automatically has as its value the name of the database being used.

finditem=ids
finditem=prompts

The default value for this option is **prompts**. KE Texpress provides a command to find an item in a form. When this command is used, by default KE Texpress displays a box for the entry of the prompt, or leading letters identifying that prompt. When the **finditem** option has the value **ids**, KE Texpress supplies a box for the entry of the item id, or leading letters identifying it.

findstr=*string*

KE Texpress maintains in a buffer the string last entered when finding an item by a prompt or an id. This buffer is by default empty when the command to find an item is first used during a session in the database. The **findstr** option may be used to set the buffer to contain the string ***string***.

flush=no
flush=yes

The default value for this option is **yes**. It can occur in KE Texpress that on entry of some illegal command, a message is displayed for a short time, then any pending input is cleared (flushed). Thus any characters typed while the message is displayed are lost. If the **flush** option is set to **no**, pending input is queued rather than flushed.

freelist=no
freelist=yes

The default value for this option is **yes**. The data file in a KE Texpress database develops, over time, locations which once contained data, but no longer do. KE Texpress maintains a list of these free locations, known as the free list. When a record is inserted into a KE Texpress database, or an edited version of a record is saved, KE Texpress consults the free list, attempting to find a sufficiently large free location in which to save the record in question. If no large enough free location is found, the data in the record is appended to the end of the data file. Use of the free list promotes reuse of otherwise wasted space in the data file. It is possible to specify that the free list is not to be used in this way, but that records are automatically to be appended to the end of the data file, by setting the **freelist** option to **no**.

funckeys=no
funckeys=yes

The default value for this option is **yes**. KE Texpress maintains, for various terminal types, a mapping between certain function keys and the character sequences generated by them. When one of these mapped character sequences is detected, it is interpreted as input from a Function key. If the **funckeys** option is set to **no**, this interpretation does not occur.

langcode=*string*

This option is used when a multibyte character set is used with KE Texpress. The option may take any one of a set of values restricted to just the names of the character sets provided in the directory **~texpress/etc/chrmaps**.

lutinsert=ask
lutinsert=no
lutinsert=yes

The default value for this option is **ask**. During validation when a record is saved, if an item contains a value not already in the item's Lookup Table, KE Texpress requests a yes or no response to the question of whether the value is to be entered into the Lookup Table. If the answer is no, the record can not be saved. If lutinsert is set to **no**, the question is avoided but the value is not entered into the Lookup Table. If the option is set to **yes**, the question is avoided and the value is automatically entered into the Lookup Table.

lutlimit=*nnn*

When the Lookup Table command is used without typing leading characters, when inserting data, edting, or querying, KE Texpress provides access to all the values in the Lookup Table. Some Lookup Tables are extremely large, and access to the entire Lookup Table in this way is very demanding of the memory resources of the machine. It is preferable if several leading characters are entered into the field before the Lookup Table command is used, because KE Texpress then provides access to just that subset of the Lookup Table with values beginning with those leading letters. To guard against inefficient use of Lookup Tables, and consequent overtaxing of the machine's resources, it is possible to set the option **lutlimit** to some number, ***nnn***. When the Lookup Table command is used, if the number of values to be accessed cexceeds ***nnn***, a message informs the user that there are too many Lookup Table entries, and states that KE Texpress is displaying the first ***nnn***.

macro=*string*

This option is used to define a macro with structure ***string***. Multiple macros may be specified by using the macro option repeatedly with different values of ***string***. For a complete description of Macros refer to the appropriate chapter in the KE Texpress User Guide.

macroshow=*n*

The default value for this option is **0**. If the value is any other integer, ***n***, then during execution of the sequence of commands in any macro, there is a pause of ***n*** seconds. In addition screen redrawing is enabled during the execution of the macros. If some error condition is encountered in the macro, KE Texpress displays a box indicating where the breakdown of the macro occurred. The option can therefore be used as a tool to debug macros.

output=both
output=file
output=printer

The default value for this option is **both**. When any command is used to copy information in KE Texpress, such as one or all retrieved records, or the screen image, or the contents of a Lookup Table, by default KE Texpress displays two selection boxes. One of which may be used to specify a filename, if copying to a file is required. The other may be used to specify a printer, if the information is to be printed. If the option **output** is set to **file**, or **printer**, the file selection box alone, or the printer selection box alone is displayed.

pagedesc=*name*

When using the **texlabel** command, the name of a page description file should by default be specified on the command line, in the form **-p*name*** Alternatively, if the **pagedesc** option is set to ***name***, the command line specification is not necessary.

profile=yes
profile=no

The default value for this option is no. The option is set to yes if used with a database which is to act as a profiling database, i.e. a database which contains data for queries used to check incoming data. When profile is set to yes, automatically the segment option is set to no.

readonly=no
readonly=yes

The default value for this option is **no**. When the option is set to yes, updating any form is disallowed.

segment=no
segment=yes

The default value for this option is **yes**. The retrieval mechanism used by KE Texpress uses two levels of indexing of data, a record level and a segment level, where a segment contains a number of records. If the segment option is set to **no**, only a single level of indexing, that at the record level, is used. When the **profile** option is set to **yes**, the **segment** option is set to **no**. In general this option should not be used, except with a profiling database.

sortexact=no
sortexact=yes

The default value for this option is **no**. When sorting in KE Texpress, the case of the values being sorted is not significant. When **sortexact** is set to **yes**, the opposite applies.

sortignore=*string*

This option is used to specify in *string*, a comma separated list of noise words which are to be ignored in the sorting process.

termtype=*string*

This option is used to specify the terminal type name to be ***string***.

videocmd=*command*

This option is specifies the command used to connect to the viseodisc recorder. See Chapter 8 for more details.

videodisp=auto
videodisp=manual

The default value for this option is **auto**. When KE Texpress has been used with the -v option specifying use of a videodisc player, and the connection has been made to the videodisc player, and there is video data in the video item in the record retrieved, KE Texpress automatically sends the appropriate command to the videodisc player to display the still shots or moving sequences specifies in the record. If the videodisp option is set to manual, KE Texpress does not send the commands to the videodisc player until the option to show the video is selected from the Show menu in Display mode.

Incorrect Option Settings

Two obvious types of error may occur when setting options. No error message appears at the time the option is set. However an error message is subsequently displayed, when a KE Texpress program is started.

The first type of error is the use, as the name of an option, of a word that is not in the set of valid option names. In this case the subsequent error message is of the form:

```
Illegal option: "word"
```

where ***word*** is the rejected name.

The second type of error is an assignment, to an option with a valid name, of a value not in the list of acceptable values for that option. The subsequent error message shows the illegal value, listing the acceptable values in brackets afterwards. The message is of the form:

```
Illegal options setting: "bells=ring",  ("yes", "no")
```

If conflicting value settings for the same option (such as **bells=no** and **bells=yes**) are included in the database options, the effective setting is the last one evaluated.

Chapter 8

Videodisc Interface

Videodisc Commands 8-3
Examples 8-7
Connecting to the Videodisc Player 8-8
Interactive Use.....8-9

Overview

KE Texpress has the capability of interfacing to a videodisc player. A KE Texpress library item on the Insertion form is used to communicate with the player. Information may be entered into this item in a high level language which is converted into the actual commands needed to drive the videodisc player. The high level language consists of simple commands for controlling the facilities of the videodisc player.

When matching records are displayed, the data associated with the video KE Texpress library item is converted to the instructions required and sent to the videodisc player. Also, whenever the user enters a KE Texpress command which affects the displayed record, the player is halted. The user may restart the current videodisc sequence by typing the Display mode command, **Ctrl+V**. This aborts any current activity on the player before re-sending the video commands.

As validation is not performed when the data is entered for the video fields, it is possible that the user may enter incorrect video commands. Errors are detected only when the record is displayed. If an error is encountered a suitable message is displayed on the video monitor.

As an aid to texformss who wish to see information or images stored on a particular disc, the videodisc interface may be run interactively directly from the Unix shell. This allows the user to experiment with the video set-up.

The videodisc player models currently supported by KE Texpress are listed in Appendix J of the Ke Texpress Design Guide.

Videodisc Commands

This section explains in detail the high level language used to communicate with a videodisc player. It examines each command individually and concludes with some sample sequences used to produce certain effects.

Where a number of commands are required for a particular record a "**;**" is used to separate them. If a command is interrupted by KE Texpress, it is aborted and any queued commands are discarded.

number

A command which consists of a single number displays a still picture of the given frame number. Note that the allowed range of frame numbers is from 0 to 54,000.

number-number

This format allows a sequence of frame numbers to be shown at normal viewing speed. The player will stop when the second frame number is reached. Note that if the second frame number is less than the first, the sequence is viewed backwards.

number*:*number or ***number-number*:*number***

The :number construct trailing the frame number(s) indicates the time in seconds to delay between frames. Thus the command 234-244:5 displays each frame between 234 and 244 for 5 seconds.

repeat [*number*] (*commands*)

This is the repetition command. It continuously executes the sequence of commands. An optional number of iterations may be given, otherwise the commands are repeated forever, or until interrupted.

slow(*number-number*)

This command displays a sequence of frames at one third the normal playing speed. If the concluding frame number is less than the starting frame number, the sequence is played backwards.

fast(*number-number*)

Similar to the slow command except that the picture is shown at three times normal playing speed.

pause(*number*)

It may be necessary to insert time delays into video sequences. This command will cause the player to delay for number seconds before executing the next command. The video screen is left unchanged.

bell(*number*, *number*)

This command can be used to sound the bell. The first number is the pitch of the bell which must be in the range of 1 to 10. The second number is the duration of the sound in 10th's of a second. Again the range is from 1 to 10.

audio([a | b] , on | off)

This command is used to control the two audio channels available on a disc player. The channels are labelled **a** and **b**. The second argument indicates whether the selected channel is to be turned **on** or **off**. Thus to turn on the second channel the command audio(b, on) is required. The simplified command audio(on) indicates that both channels are to be turned on. By default, both channels are enabled.

frame(on | off)

Frame numbers can be enabled or disabled by this command. If enabled the video player displays the current frame number on the screen. By default, frame numbers are not shown.

text(*number*, *number*, *arg1*, *arg2*, . . .)

If the video player is capable of displaying teletext this command allows the texforms to place text on the screen. The text added is superimposed over any images currently on the screen. The first number is the row in which the text is placed. The rows are numbered from 0 to 22 with 0 being the top row. The second number is the column number. The columns are numbered from 0 to 39 where 0 is the left hand column. If the string is to be centered then the column number may be set to a "**?**". The third and subsequent arguments contain the text to be placed on the screen. Note that each text argument must be a field Id (indicating that the contents of the field should be displayed) or a string within double quotes.

flash(on | off)

This command indicates that all subsequent text placed on the screen is to be flashing (on) or steady (off). By default, flashing is off.

double(on | off)

Teletext permits two character sizes. This command either selects double height lettering or reverts to normal height. By default, double height lettering is disabled.

colour(*type*)

This command allows the colour of the text placed on the screen to be modified. The available colours are: **white**, **red**, **green**, **blue**, **yellow**, **pink** and **cyan**. The selected colour remains in effect until the next colour command or a clear command. The default colour is white.

clear

This command clears the screen leaving it black. It also resets all the text (flash, double, colour) and audio attributes (audio) to their default settings.

Three other optional command sequences may be assigned by the DBA to change the default behaviour of the player. The three options are **header**, **before** and **after**. The header option allows the DBA to define commands which are executed as soon as the videodisc player is initialized. These commands may typically be words of introduction.

The before option contains commands which are executed before each set of commands is sent to the video player. Thus if all the text written to the screen was to be red then the command:

before=colour(red)

may be used. This would save the texforms entering this command for each record. The after option is similar to before except that it is executed after all commands associated with a record have completed.

Before any commands are sent from KE Texpress to the videodisc player for a particular record, the text and audio attributes are reset to their default states.

The KE Texpress command, **texvidopts** is used to set the header, before and after options. It functions in the same way as **texopts** (refer to the KE Texpress Design manual).

The contents of a particular field may be referenced within the video command language. This allows the texforms to display data on the screen which comes from a record stored in the database. For example, assuming the database has an item with the item Id "name", the video information field could comprise:

colour(yellow); text(22, ?, name)

which would display the text of the first field of name in yellow centered on the bottom line of the video screen. The second field of the name item could be referenced using name_2. Field references may be interspersed with text, for example:

text(22, 0, "This is a ", name, " (red).")

would produce the text "This is a kangaroo (red)." on the video screen, assuming that the name field contained the data "kangaroo".

Examples

(1) Display frame 20 on the screen.

20

(2) Step through frames 345 to 365, pausing at each frame for 3 seconds. Also have the word "Kangaroos" displayed in the centre of the bottom line.

text(22, ?, "Kangaroos"); 345-365:3

(3) Play frames 100 to 200 with the word "Emu" in blue and double height centered on the second bottom line. Then play frames 200 to 225 in slow motion with the word "Wombats" displayed on the second bottom line with normal height letters and green colouring.

double(on); colour(blue); text(21, ?, "Emu"); 100-200;

double(off); colour(green); text(21, ?, "Wombats");

slow(200-225)

Connecting to the Videodisc Player

Connection to the videodisc player is achieved by consulting the video option which may be defined in the KE Texpress options file for a particular database. The options file is accessible using the KE Texpress command **texopts**. The video option may also be set using the environment variables, **TEXPRESSOPTS** or ***dbname*opts**.

The video option is used to define the command required to invoke the video player interface program. An example setting is:

```
video="/u/texpress/bin/texvideo -s dbname /dev/tty5"
```

The first argument is the name of the videodisc program used to connect to the videodisc player and accept commands from KE Texpress. The second argument, "-s", sets the rate for communication with the player to 1200 baud. The default setting is 9600 baud. The third argument is the name of the database being used and the fourth is the Unix device (terminal port) to which the player is connected.

Interactive Use

It is possible to connect directly to the video player. This is normally used to test command sequences interactively or to find the frame numbers of pictures stored on a disc. The invocation sequence for interactive use is:

```
texvideo [-abs] dbname device
```

The "-a" option indicates that audio channel "a" is to be turned off before every command sequence. The "-b" option indicates that channel "b" is to be turned off. By default, both channels are turned on before each command sequence. The "-s" option indicates that the communication rate between the program and the videodisc player is 1200 baud. The default rate is 9600 baud. The database name must also be specified. The device is the name of the Unix port to which the video player is connected.

After connecting to the videodisc the prompt **ready** is displayed. Commands may then be entered. A command sequence must be entered on the one line. When Return is pressed the commands are converted and sent to the videodisc player. Thus to place the word "hello" on the screen in pink the texforms should enter:

ready (*prompt from videodisc program)*

```
colour(pink); text(0, 0, "hello")
```

Entering **Ctrl+D** on an otherwise blank line terminates the connection with the videodisc player.

Chapter 9

Macros

Macro Definition and Execution 9-3
Letters Component 9-4
Sequence Component 9-5
Modes Component 9-7
Menuopt Component 9-8
Help Component 9-9
Startup Macros 9-10
Mode Startup Macros 9-11

Overview

This chapter describes the use of the KE Texpress macro facility. Macros can be used for a variety of reasons, including:

- Reducing a complex series of KE Texpress operations to a single keyboard command.
- Performing a sequence of commands automatically whenever a particular mode of KE Texpress is commenced.
- Automatically performing a set of KE Texpress commands which implement an entire KE Texpress session.
- Altering the KE Texpress **texforms** interface by hiding many of the KE Texpress puul-down menu option and providing a simple list of alternatives from which a user can make a selection.

Macro Definition and Execution

KE Texpress provides a mechanism which allows a single keystroke to be mapped to a series of KE Texpress commands. Such maps are called *macros*. They are defined as KE Texpress database options, and as such they are read when KE Texpress is initiated. Macros, in general, take the following form:

```
macro=letters:sequence:modes:menuopt:helpfile
```

Consider the following example. The macro can be defined through the environment for **sh** texformss by:

```
dbnameopts="macro=^A:^T^Fname\n^K:Query:Name Look-up:"
export dbnameopts
```

while **csh** users can use the command:

```
setenv dbnameopts="macro=^A:^T^Fname\n^K:Query:Name
Look-up:"
```

where *dbname* is the name of the database for which the macro is to be available. The macro is invoked by the keys, **Ctrl+A**, and moves the cursor to the first field of an item with a prompt having leading letters, "name", and displays its Look-up table. The macro is valid during Query mode. Also, an option, "Name Look-up", is created on a Macros pull down menu so that the macro can be invoked in this manner.

When a valid macro is invoked, KE Texpress takes commands from the *sequence* component of the macro, until all the characters have been consumed. Commands are then accepted as normal from the user (or until another macro is invoked). When a macro is executing, the terminal screen is not updated. Only at the completion of the macro is the screen refreshed. Two exceptions to this rule are, if the macro requests any user input, or if the macro definition explicitly turns screen updating on.

KE Texpress does not support nested macros. One macro cannot invoke another macro.

The following sections describe the components of the macro definition.

Letters Component

The *letters* component of a macro definition is a sequence of zero or more characters which defines a series of keystrokes used to invoke the macro. If the user enters any of these characters while in the appropriate mode the macro will be executed. Each character may be represented in one of the following ways:

- A letter (upper or lower case), number or punctuation character.
- A ^ character followed by an upper case letter, representing a control character, eg. **Ctrl+D**
- A function key name surrounded by curly brackets e.g. **{F3}** or **{FIND}**.
- An extended character surrounded by curly brackets e.g. **{,C}**.
- A \ character followed by a symbol as defined below:

\E	escape character
\n	newline character
\b	backspace character
\t	horizontal tab character
\r	carriage return character
\f	form feed character
\v	vertical tab character
\\	backslash character
nnn	character having octal value *nnn*

For example, the letter component required for a macro to be invoked by Ctrl+A, or function key F1, or by the letter **Ç**, is:

^A{F1}{,C}

Unlike the definition of the characters used to invoke commands under KE Texpress, the first named character is not the keyboard equivalent character shown on the pull down menu option.

Sequence Component

The *sequence* component of a macro definition contains a series of characters which are placed in the input queue when the macro is invoked. These characters represent commands which are executed as though entered by the user. The character sequence contains zero or more KE Texpress command characters where each character may be one of the following:

- A letter (upper or lower case), number or punctuation character.
- A ^ character followed by an upper case letter, representing a control character.
- A function key name surrounded by curly brackets e.g. **{F3}** or **{FIND}**.
- An extended character surrounded by curly brackets e.g. **{,C}**.
- A \ character followed by a symbol as defined in the table in the previous section.
- A KE Texpress function surrounded by curly bracket, e.g. **{insrec}**. The function names can be obtained by positioning the cursor on the appropriate pull-down menu option and selecting the help facility, which will display the function name at the bottom right hand corner of the help screen.
- A **$ or %** character followed by a letter indicating that a certain operation should be performed at this point in the execution of the macro (see below).

The series of characters making up the sequence component replace any user input up until the last character has been executed, or until an illegal command is encountered. In the case of an illegal command being entered, the rest of the macro is discarded and control is returned to the user.

As a macro takes effect immediately after an invoking keystroke is typed, the first command in the macro is executed relative to the current cursor position. Hence, using a macro to traverse a menu is dangerous as the next time the menu is invoked, the current command will have changed to the previously selected command.

Several special operators may be embedded in the sequence component. They are:

$r, %r - Screen redraw toggle

The first **$r** or **%r** in the sequence causes the screen to refresh and subsequent commands will update the screen. This allows the execution of the macro to be observed by the user. A second **$r** in the sequence will turn off screen updating. In this way it is possible to control what the user observes on the screen as a macro is being executed.

$s[*message*], **%s** [message]- Screen message

At the point of encountering this operator in the execution of the macro, the ***message*** is displayed in a message box in the centre of the screen. This facility is useful in providing feedback to the user when it is necessary to perform a sequence of KE Texpress commands which may take a long time. For example, a sequence component of:

$s[Reading Look-up table Wait ...]^K

reads a Look-up table and displays a message informing the user of this action.

$i[*string*], **%i** [string]- Input

At the point of encountering this operator in the execution of the macro, the screen is refreshed and the user can enter data up until one of the characters in ***string*** is typed. At this point the matching character is returned as input and the macro is resumed. Screen redrawing remains as it was prior to the use of the input operator. The ***string*** is in the same format as the letter component of a macro. The **[*string*]** section is optional and if not specified, the newline character is assumed to terminate user input and resume execution of the macro. For example, a sequence component of:

```
^T^Fname\n$i[\n\E]$r^Y
```

accepts user input in the item *name* up to either a newline or escape character is typed. A query is then performed. Screen refreshing is turned on before the query is commenced so that the user may observe the query as it is performed.

When using either the redraw, status message or input operator it is important to remember that the operation are performed sequentially in the order in which they are set out. Hence refresh must be turned on before the query is commenced in the above example otherwise the screen will be refreshed after the query is complete.

Modes Component

The *modes* component of a macro definition indicates the modes for which the macro is valid. Where more than one mode is valid, the modes should be separated by commas. Each mode name must match exactly the Mode indicator appearing in square brackets at the left side of the menu bar. Character case is significant.

If the last character in the mode name is *****, trailing characters in the Mode indicator are ignored. Thus the mode name, **Q*** matches all Mode indicators commencing with **Q**.

Menuopt Component

The *menuopt* component of a macro definition contains the character sequence which will appear as an option on the special **Macros** pull down menu. If a macro is defined for a certain mode and the macro definition contains a menuopt component then an extra menu name will appear with the title **Macros**. Any macros defined for the current mode with menuopt components will appear on this menu. If a keyboard equivalent character is to be shown the menuopt string must contain that character.

For example, to display the text "Name Look-up Ctrl+A" as an option on the Macros pull-down menu, the following menuopt component can be used:

```
Name Look-up Ctrl+A
```

If the menuopt component ends with an underscore, the pull-down menu option will be displayed with a menu separator (a line) drawn directly below it on the Macros pull-down menu.

If the menuopt component is omitted from the macro definition, the macro does not appear on the Macros pull down menu and can only be invoked from the keyboard. If there are no macros for a particular mode which include a menuopt component, then the Macros pull down menu is not displayed in that mode.

Help Component

The name of a help file for the macro can be listed here. When the user highlights the macro entry on the pull-down menu and selects the help facility, the contents of this help file are displayed in a help box.

Startup Macros

A facility which allows a macro to be defined which is executed automatically when KE Texpress is invoked. A macro sequence defined as:

macro=:*sequence*::

(i.e. the *letters*, *modes* and *menuopt* components are empty) is a startup macro. There can be only one startup macro for a database.

A batch job can be simulated by defining a startup macro that performs the task and then exits.

Mode Startup Macros

A startup macro may also be defined for each KE Texpress mode. When the appropriate mode is commenced or recommenced, the mode startup macro is invoked automatically. A mode startup macro is defined in the following manner:

macro=:*sequence*:*mode*:

(i.e. their *letters* and *menuopt* components are empty).

The menuopt component can also be used to list a series of mode names, so that the Macros is invoked when any of these modes are commenced or re-commenced.

Appendix A
Quick Reference Guides

Global Edit Mode Quick Reference

Ctrl+G Copy screen image to a printer or Unix file.

Ctrl+R Redraw the screen.

Ctrl+Z Job control - stop process or invoke a shell.

? Display helpful information.

Ctrl+S Flow control - stop screen output.

Ctrl+Q Flow control - restart screen output.

Field Editing Quick Reference

Ctrl+H Move left one character position.

Ctrl+L Move right one character position.

Ctrl+I Insert characters before the current character.

Ctrl+A Append characters after the current character.

Ctrl+D Delete the character under the cursor.

Ctrl+O Overstrike character under cursor with the next (single) character typed.

Ctrl+C Copy data into the Copy buffer.

Ctrl+E Enter data from the Copy buffer into the field.

Ctrl+U Undo the previous change.

↵, ESC Terminate Text edit mode.

Insert/Query/Edit Mode Quick Reference

Not all commands are available at all times during editing. Some are specific to a particular operation (e.g. **Ctrl+Y** is only applicable in Query mode).

Function Pull-down Menu

Ctrl+TT	Write record to the temporary file (Insert mode only).
Ctrl+W,F4	Write this record to file (not in Query mode).
Ctrl+Y,F3	Perform the query (Query mode only).
Ctrl+X,F1	Exit Edit mode without performing the operation.

Move Pull-down Menu

Ctrl+F	Move forward to the next fillable field.
Ctrl+B	Move backward to the previous fillable field.
Ctrl+N	Move to the next fillable item.
Ctrl+P	Move to the previous fillable item.
Ctrl+TB	Move to the beginning of the form.
Ctrl+TE	Move to the end of the form.
Ctrl+TF	Find a prompt by its leading letters.

Info Pull-down Menu

Ctrl+K,F7	Display the contents of a Look-up table.
Ctrl+TV	Validate the form contents (not in Query mode).
Ctrl+TN	Assign the next available Key value (Insert mode only).
Ctrl+TA	Display the current item's attributes.
Ctrl+TP	Print the field edit message.
Ctrl+TO	Execute the linked database program.

Clear Data

ESC	Clear the current field.
Ctrl+TC	Clear the form of all data.
Ctrl+TL	Load default value into field.
Ctrl+TW	Wipe out / don't wipe out field (Global edit mode only).

Misc Pull-down Menu

Ctrl+V	Ditto (fill current field with data from previous record).
Ctrl+TI	Ditto all empty fields from the ditto record.
Ctrl+TD	Select a new ditto record.
Ctrl+TY	Edit the comments (Query mode - search comments text).

Display Mode Quick Reference

The following is a summary of commands available during Display mode. Not all commands are available at all times. The list of available commands depends on the design of the forms and on the actual query performed.

Function Pull-down Menu

Ctrl+TC	Generate a report of all matching records.
Ctrl+TT	Generate a report of the current matching record.
Ctrl+TP	Peruse records by Key value.
Ctrl+TJ	Jump to a record by Key value.
Ctrl+TV	Verify the matching records.
Ctrl+TK	Set current record as the ditto record.
Ctrl+TW	Write the matching records in dave file format.
Ctrl+Y,F3	Join another query to the current set of matching records.
Ctrl+X,F3	Exit Display mode and return to Query mode.

Move Pull-down Menu

Ctrl+F	Display the next matching record.
Ctrl+B	Display the previous matching record.
Ctrl+H	Move the form window half a screen to the left.
Ctrl+J	Move the form window half a screen down.
Ctrl+K	Move the form window half a screen up.
Ctrl+L	Move the form window half a screen to the right.

Edit Pull-down Menu

Ctrl+E	Edit the current record.
Ctrl+TE	Globally edit all matching records.
Ctrl+D	Delete the current record from the database.
Ctrl+TI	Delete all matching records from the database.
Ctrl+TP	Show the current record's privilege level.
Ctrl+TA	Adjust the privilege level of all matching records.
Ctrl+TD	Discard the current record from the match array.

Re-Order Records

Ctrl+TL	Re-order/summarize the matching records.
Ctrl+TO	Set default re-order/summarize criteria.

Show

Ctrl+TY	Display the current record's comments.
Ctrl+TH	View history (saved) versions of the current record.
Ctrl+V	Re-display the video sequence.
Ctrl+TN	Select a new Display form (Insertion form or Report form).
Ctrl+TF	Move the form window to the next matching term.
Ctrl+TB	Move the form window to the previous matching term.
Ctrl+I	Execute form expression.

Copy

Ctrl+C	Copy all matching records to a Unix file or printer.
Ctrl+O	Copy the current record to a Unix file or printer.

Appendix B
Key Types

This section describes the Key types which are distributed with KE Texpress. Each Key type contains a given number of fields of the appropriate data type. In general, these attributes may not be altered. However, the field lengths, validation and prompt may be changed if necessary. A Key type may be designed by a DBA, after which it can be used by any DBA in the design of any database.

For an automatic Key value assignment to occur using the multi-field Key types, it is necessary to use the same prefix as that of the previous Key value entered by the user during the current session. Hence, at the commencement of a session, it is generally necessary to enter a prefix before any automatic Key value assignment is possible. The exception to this rule is when an assignment expression or a default value is used to automatically enter a value for the prefix.

annn

The annn Key type has the following format:

```
Key _ ______
```

It consists of two fields. Field 1 accepts string data. Its default validation is:

```
val = filled(#);
```

Field 2 accepts an integer. Its default validation is:

```
val = filled(#) && (# > 0);
```

Automatic Key values can be generated for this Key type.

integer

The integer Key type has the following format:

```
Key ______
```

It consists of a single field accepting an integer. Its default validation is:

```
val = filled(#) && (# > 0);
```

Automatic Key values can be generated for this Key type.

nnn.nnn

The nnn.nnn Key type has the following format:

```
Key _____ _____
```

It consists of two fields. Both fields accept integer values and have default validation of:

```
val = filled(#);
```

Automatic Key values can be generated for this Key type.

nnna

The nnna Key type has the following format:

```
Key ______ _
```

It consists of two fields. Field 1 accepts an integer. Its default validation is:

```
val = filled(#) && (# > 0);
```

Field 2 accepts string data. Its default validation is:

```
val = filled(#);
```

Automatic Key values can be generated for this Key type.

string

The string Key type has the following format:

```
Key __________
```

It consists of a single field accepting a string. Its default validation is:

```
val = filled(#);
```

yynnn

The yynnn Key type has the following format:

```
Key __ ______
```

It consists of two fields. Field 1 accepts integer data. Its default validation is:

```
val = filled(#);
```

Field 2 accepts an integer. Its default validation is:

```
val = filled(#) && (# > 0);
```

Automatic Key values can be generated for this Key type. If no value is supplied for the first field, then the last two digits of the current year are assigned.

Appendix C

KE Texpress Library Types

This section describes the currently available KE Texpress Libraries and their formats. The fields of these items have been defined to accept a restricted set of values during data entry.

ddmmyyyy

This has the following format:

```
Date __/__/____
```

Field 1 accepts an integer value in the range, 1 to the number of days in the month corresponding to field 2. Field 2 accepts a integer value in the range, 1 to 12. Field 3 accepts a 4 digit integer value which is non-negative. This library may be designated automatic and may also be ranged.

For range queries involving lower bounds, if either field 1 or field 2 is empty, it assumes the value, 1. If field 3 is empty, it assumes the value, 0.

For range queries involving upper bounds, if field 3 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, 12. If field 1 is empty, it assumes the value of the number of days in the corresponding month.

ddmmmyyyy

This has the following format:

```
Date __  ___  ____
```

Field 1 accepts an integer value in the range, 1 to the number of days in the month corresponding to field 2. Field 2 accepts a 3 character value which is one of:

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

Field 3 accepts a 4 digit integer value which is non-negative. This library may be designated automatic, and may also be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 1. If field 2 is empty, it assumes the value, Jan. If field 3 is empty, it assumes the value, 0.

For range queries involving upper bounds, if field 3 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, Dec. If field 1 is empty, it assumes the value of the number of days in the corresponding month.

ddmmyy

This is identical to the ddmmyyyy library except that only two digits may be entered for the year value. It has the format:

Date __/__/__

This library may be designated automatic and may also be ranged.

ddmmmyy

This is identical to the ddmmmyyyy library except that only two digits may be entered for the year value. It has the format:

Date __ ___ __

This library may be designated automatic and may also be ranged.

ddmonyyyy

This library has the following format:

Date __ _____ ____

Field 1 accepts an integer value in the range, 1 to the number of days in the month corresponding to field 2. Field 2 accepts a character value which is one of:

January February March April May June July August September October November December

Field 3 accepts a 4 digit integer value which is non-negative. This library may be designated automatic, and may also be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 1. If field 2 is empty, it assumes the value, January. If field 3 is empty, it assumes the value, 0.

For range queries involving upper bounds, if field 3 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, December. If field 1 is empty, it assumes the value of the number of days in the corresponding month.

mmddyyyy

This has the month field before the day field but otherwise is identical to the ddmmyyyy KE Texpress library. It has the format:

Date __/__/____

This library may be designated automatic and may also be ranged.

mmmddyyyy

This is similar to the mmddyyyy library except that the month field accepts a three character abbreviation of the month name (as described above). It has the format:

Date ___ __ ____

This library may be designated automatic and may also be ranged.

mmyy

This has the following format:

Date __/__

Field 1 accepts a integer value in the range, 1 to 12. Field 2 accepts a 2 digit integer value which is non-negative. This library may be designated automatic and may also be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 1. If field 2 is empty, it assumes the value, 0.

For range queries involving upper bounds, if field 2 is empty, it assumes the value, 0. If field 1 is empty, it assumes the value, 12.

monddyyyy

This is similar to the ddmonyyyy library (as described above) except that the first field represents the month and the second field represents the day of the month. It has the format:

Date _____ __ ____

This library may be designated automatic and may also be ranged.

yyyymmdd

This is similar to the ddmmyyyy KE Texpress library except that the order of the fields is reversed. It has the following format:

Date ____/__/__

This library may be designated automatic and may also be ranged.

yyyymmmdd

This is similar to the ddmmmyyyy KE Texpress library except that the order of the fields is reversed. It has the following format:

Date ____ ___ __

This library may be designated automatic and may also be ranged.

yyyybc

This has the format:

Date ____ __

Field 1 accepts a 4 digit integer value which is non-negative. Field 2 accepts a 2 character string which can take one of the values:

BC AD

This library may be ranged. For range queries, if field 1 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, **AD**.

hhmm12

This has the format:

Time __:__ __

Field 1 accepts an integer value in the range 1 to 12. Field 2 accepts an integer value in the range 0 to 59. Field 3 accepts one of the strings:

AM PM

This library may be designated automatic and may also be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 12. If field 2 is empty, it assumes the value, 0. If field 3 is empty, it assumes the value, AM.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, 59. If field 3 is empty, it assumes the value, PM.

hhmm24

This has the format:

Time __:__

Field 1 accepts an integer value in the range 0 to 23. Field 2 accepts an integer value in the range 0 to 59. This library may be designated automatic and may also be ranged.

For range queries involving lower bounds, if either field is empty, it assumes the value, 0.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 2 is empty, it assumes the value, 59.

latdm

This has the format:

Latitude __ Deg _____ Min _

Field 1 accepts an integer value in the range 0 to 90. Field 2 accepts a real value in the range 0 to 59.99. Field 3 accepts a 1 character direction, one of '**N**', '**n**', '**S**' or '**s**'. This library may be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 0. If field 3 is empty, it assumes the value, **N**. If field 2 is empty, it assumes the value, 59.99, for queries where field 3 is **S** or **s**. It assumes the value, 0, otherwise.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 3 is empty, it assumes the value, **N**. If field 2 is empty, it assumes the value, 59.99, for queries where field 3 is **N** or **n**. It assumes the value, 0, otherwise. Note also that in any comparisons, south is considered less than north.

latdms

This has the format:

Latitude __ Deg __ Min __ Sec _

Field 1 accepts an integer value in the range 0 to 90. Field 2 accepts an integer value in the range 0 to 59. Field 3 accepts an integer value in the

range 0 to 59. Field 4 accepts a 1 character direction, one of '**N**', '**n**', '**S**' or '**s**'. This library may be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 0. If field 4 is empty, it assumes the value, **N**. If either field 2 or field 3 is empty, it assumes the value, 59, for queries where field 4 is **S** or **s**. It assumes the value, 0, otherwise.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 4 is empty, it assumes the value, **N**. If either field 2 or field 3 is empty, it assumes the value, 59, for queries where field 4 is **N** or **n**. It assumes the value, 0, otherwise. Note also that in any comparisons, south is considered less than north.

longdm

This has the format:

Longitude ___ Deg _____ Min _

Field 1 accepts an integer value in the range 0 to 180. Field 2 accepts an real value in the range 0 to 59.99. Field 3 accepts a 1 character direction, one of '**E**', '**e**', '**W**' or '**w**'. This library may be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 0. If field 3 is empty, it assumes the value, **E**. If field 2 is empty, it assumes the value, 59.99, for queries where field 3 is **W** or **w**. It assumes the value, 0, otherwise.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 3 is empty, it assumes the value, **E**. If field 2 is empty, it assumes the value, 59.99, for queries where field 3 is **E** or **e**. It assumes the value, 0, otherwise. Note also that in any comparisons, west is considered less than east.

longdms

This has the format:

Longitude ___ Deg __ Min __ Sec _

Field 1 accepts an integer value in the range 0 to 180. Field 2 accepts an integer value in the range 0 to 59. Field 3 accepts an integer value in the range 0 to 59. Field 4 accepts a 1 character direction, one of '**E**', '**e**', '**W**' or '**w**'. This library may be ranged.

For range queries involving lower bounds, if field 1 is empty, it assumes the value, 0. If field 4 is empty, it assumes the value, **E**. If either field 2 or field 3 is empty, it assumes the value, 59, for queries where field 4 is **W** or **w**. It assumes the value, 0, otherwise.

For range queries involving upper bounds, if field 1 is empty, it assumes the value, 0. If field 4 is empty, it assumes the value, **E**. If either field 2 or field 3 is empty, it assumes the value, 59, for queries where field 4 is **E** or **e**. It assumes the value, 0, otherwise. Note also that in any comparisons, west is considered less than east.

Userid

This consists of an item with one field which accepts text input of up to eight characters. It has the format:

User Id _____

This library may be designated automatic.

video

This consists of an item with one field which accepts text input initially up to 25 characters. It has the format:

Video frames _____

This library is used to enter information such as frame numbers which can be used to drive a videodisc.

Index

A

Administrator, 1-5
And query, 5-8
annn Key type, B-1
Append
    To field, 3-11
    To Prompt, 6-12
Assignment
    Key value, B-1
Attribute
    Backward, 6-13
    Exit, 6-13
    Forward, 6-13
    Mode, 6-13
    Next page, 6-13
    Previous Page, 6-13
Automatic
    Items, 4-2
    Key value, B-1
Awk, 6-3

B

Backward
    Attribute, 6-13
    Field, 3-4
    History, 5-18
    Item, 3-4
    Option, 6-28
    Record, 5-16
Bounds
    Down, 6-15
    Exit, 6-16
    Form, 6-15
    Left, 6-15
    Move menu, 6-15
    Right, 6-15
    Up, 6-15

C

Capabilities
    Terminal, 1-9
Case search, 5-9
Celsius from fahrenheit expression, 4-8
Change
    Display Form, 5-17
    History Display Form, 5-18
    Privilege, 5-28
    Report Form Name, 6-23
    Report Form Object, 6-11
Clear
    Form, 3-5, 5-7
    Text, 3-5
Combine
    Query, 5-34
Command summary
    Insert mode, A-2
Commands
    Videodisc, 8-3
Comments, 4-7, 5-16
    Editor, 4-7
Connection
    Videodisc, 8-8
Coordinates
    Form, 6-15, 6-19
Copy
    All matching records, 5-31
    Current Record, 5-31
    Current record, 2-17
    Report form, 6-24
    Report Form Object, 6-10
    Screen, 2-12
Create/Edit
    Report form, 6-4
Cursor position
    Form, 6-15, 6-19

D

Data
    Capture, 1-2
    Protection, 2-18
Data section
    texpaginate, 6-37
Data security, 2-18
Data type
    Filename, 3-3
    Integer, 3-3
    Real, 3-3
    String, 3-3
    Text, 3-3
Data Validation, 4-10
Database
    Design, 1-2
    Options, 7-7
    Parameters, 7-7
    Query, 1-2
    query privilege, 5-3
    Querying, 5-3
    Statistics, 7-3
Database Administrator, 1-5
Database component pathname options, 7-8
Database pathname options, 7-8
Database querying, 1-2
Database reporting, 1-2, 1-3
Database-wide options, 7-7
Date
    Library type, C-1, C-2, C-3, C-4
    Library type, C-3
Dave File, 5-33
DBA, 1-5
dbnameopts, 8-8
ddmmmyy
    Library type, C-2
ddmmmyyyy
    Library type, C-1
ddmmyy
    Library type, C-2
ddmmyyyy
    Library type, C-1
ddmonyyyy
    Library type, C-2
Default values, 4-2
Delete, 3-11
    History, 5-18
    Matching records, 5-27
    Record, 5-26, 5-35
    Report, 6-23
Delete object
    Report form, 6-12
Deselect Objects
    Report Form, 6-5
Disappearing matches, 5-35
Display
    Comments, 5-16
    Pull-down menus, 2-5
    Version Number, 7-5
Display form, 6-30
    Change, 5-17
Display mode, 1-7, 5-14, 6-30, 8-2
    Commands, 5-15
    Form, 5-15
    Options, 7-12
Display mode Quick reference, A-4
Ditto, 4-5
    All, 4-5
    Change ditto record, 4-5
    Commands, 4-5
    Set ditto record, 5-32

E

Edit
    Command, 6-8
    Comments, 4-7
    Mode, 1-7, 5-22
    Record, 5-22
Edit mode Quick reference, A-2
End of form, 3-4
End section
    texpaginate, 6-37
Entering Information into Fields, 3-5
Environment variable
    SHELL, 7-4
European character set, 9-4
Exit, 2-6, 2-12, 4-18, 5-7, 5-15, 5-23, 5-25, 6-21
    History, 5-18
    Jump mode, 5-21
    Perusing mode, 5-20
    Report menu, 6-29
    Text perusal, 2-17
Expressions, 4-8

Extract item
Report form, 6-8

F

Fahrenheit to celsius expression, 4-8
False match, 5-35
Field, 1-6, 4-12
Edit message, 3-10
Editing Commands, 3-11
Validation, 4-12, 4-13
Wipe out flag, 5-24
File copy
Options, 7-11
Find an item, 3-4
Find Prompt
Report Form, 6-7
Footer section
texpaginate, 6-37
Footlines
texpaginate, 6-36
Form, 1-5
Coordinates, 6-15, 6-19
Cursor position, 6-15, 6-19
Display, 5-15, 5-18, 6-30
Expression, 5-33
Key, B-1
Order, 6-17
Privilege level View, 6-20
Query, 5-4
Size, 6-15, 6-19
Status, 6-19
View, 6-15, 6-19
Window, 2-4
Form bounds, 6-15
Down, 6-15
Exit, 6-16
Left, 6-15
Right, 6-15
Up, 6-15
Form size, 6-15
Form status
Modified, 6-19
Read only, 6-19
Forward
Attribute, 6-13
Field, 3-4
History, 5-18
Item, 3-4
Option, 6-28
Record, 5-15
Front-end menu, 2-3

G

General options, 7-18
Pathname, 7-10
General utilities, 2-2
Global
Mode, 5-24
Global edit, 5-24
Options, 7-15

H

Header, 1-6
Header section
texpaginate, 6-37
Headlines
texpaginate, 6-36
Help, 2-6
hhmm12
Library type, C-4
hhmm24
Library type, C-5
Hierarchy, 3-8
Validation, 4-10
History, 5-23
Backward, 5-18
Change display form, 5-18
Delete, 5-18
Exit, 5-18
Forward, 5-18
Mode, 5-18
Options, 7-15
View, 5-18

I

Image display, 5-33
Incorrect option settings, 7-23
Indicator
Mode, 2-4
Information retrieval, 1-2
Insert, 3-11
Record, 4-15

Insert box
Report form, 6-9
Insert mode, 1-7
Options, 7-11
Quick reference, A-2
Insert record number, 6-34
Insertion form, 1-6
Integer
Data type, 3-3
integer Key type, B-1
Interpret Form Expression, 5-33
Interrupt, 2-16
Query, 5-14
Report Generation, 6-22
Invocation
texforms, 1-10
Item, 1-6
Attributes, 3-9
Privileges, 2-19

J

Job control, 2-15
Join a query, 5-34
Jump mode
Exit, 5-21
Jump to Record, 5-21

K

KE Texpress, 1-2, 1-5
KE Texpress library, 5-29
KE Texpress library types, C-1
Key, 1-4
Automatic, B-1
Key assignment
Next available, 4-3
None, 4-4
Personal next, 4-3
Key form, B-1
Key item
Validation, 4-10
Key type
annn, B-1
integer, B-1
nnn.nnn, B-2
nnna, B-2
string, B-2
yynnn, B-3
Key value, 4-3
Assignment, B-1
Key value queries, 5-9
Key value range queries, 5-10
Keyboard accelerators, 1-9
Keys
Perusing, 5-20

L

latdm
Library type, C-5
latdms
Library type, C-5
Latitude
Library type, C-5
Library, 1-4
Library type
Date, C-1, C-2, C-3, C-4
ddmmmyy, C-2
ddmmmyyyy, C-1
ddmmyy, C-2
ddmmyyyy, C-1
ddmonyyyy, C-2
hhmm12, C-4
hhmm24, C-5
latdm, C-5
latdms, C-5
Latitude, C-5
Login Id, C-7
longdm, C-6
longdms, C-6
Longitude, C-6
mmddyyyy, C-3
mmmddyyyy, C-3
mmyy, C-3
monddyyyy, C-3
Time, C-4, C-5
Userid, C-7
video, C-7
yyyybc, C-4
yyyymmdd, C-4
yyyymmmdd, C-4
Line
joining, 6-33
joining (word-wrap), 6-33
Removal, 6-31
Replacement, 6-32

Line drawing
texcat, 6-39
Link database, 3-9
Linked database, 4-9
Component pathname options, 7-9
List object, 5-30
Listing Databases, 7-5
Load all defaults, 4-6
Load default, 4-6
Login Id
Library type, C-7
longdm
Library type, C-6
longdms
Library type, C-6
Longitude
Library type, C-6
Look-up table, 2-9, 3-6, 4-11
Insertion privilege, 4-11
Validation, 4-11
Lower bound items, 5-9

M

Macros, 9-2
Definition, 9-3
Execution, 9-3
Help, 9-9
Letters component, 9-4
Menuopt component, 9-8
Modes component, 9-7
Sequence component, 9-5
Startup, 9-10
Mark corner, 2-12
Match count, 5-35
Matching record
Copy, 5-31
Delete, 5-27
Menu bar, 2-4
Menus
Pull-down, 2-5
Message of the day, 1-10
Message window, 2-4
Miscellaneous Commands, 5-32
mmddyyyy
Library type, C-3
mmmddyyyy
Library type, C-3
mmyy
Library type, C-3
Mode, 2-4, 5-29
Display, 1-7, 5-14
Edit, 1-7
Global, 5-24
History, 5-18
Indicator, 5-18
Insert, 1-7
Perusing, 5-20
Query, 1-7, 5-3, 5-4
Re-order, 5-29
Startup Macros, 9-11
Modified
Form status, 6-19
monddyyyy
Library type, C-3
Move down, 2-5, 2-12
Report form, 6-6
Move form
Down, 5-16
Left, 5-16
Right, 5-16
Up, 5-16
Move left, 2-6, 2-12, 3-11
Report form, 6-6
Move object
Report form, 6-9
Move right, 2-6, 2-12, 3-11
Report form, 6-6
Move to
Start of form, 3-4
Move to menu via title, 2-6
Move to next word
Report form, 6-6
Move to previous Word
Report Form, 6-7
Move up, 2-6, 2-12
Report form, 6-6
Movement around a form, 3-4
Moving around
Report form, 6-6

N

New header
Report form, 6-8
Next
Attribute page, 6-13
Page, 2-17

Next available Key assignment, 4-3
Next item, 6-13
nnn.nnn Key type, B-2
nnna Key type, B-2
No automatic Key assignment, 4-4
Noise Words, 5-6

O

Object-oriented, 1-2
On-line help, 2-8
Operation
    Privileges, 2-20
Options
    after, 8-5
    before, 8-5
    Database, 7-7
    Database-wide, 7-7
    header, 8-5
    screenall, 2-12
    screenblock, 2-12
    screenshort, 2-12
    System-wide, 7-7
    video, 8-8
    Videodisc, 8-5
Or query, 5-7
Order
    Form, 6-17
Output command, 6-27
Output file, 6-27
Overstrike, 3-11
Overview, 3-2, 4-2, 6-3

P

Page
    Next, 2-17
    Previous, 2-17
Paginating reports, 6-35
Parameters
    Database, 7-7
PATH
    Shell variable, 1-10
Pattern Matching Facilities, 5-11
Pattern-matching operators
    ! (Not), 5-11
    $ (End), 5-12
    (literal next), 5-13
    * (Wildcard), 5-12
    ? (Single character), 5-12
    [str] (single character range), 5-12
    ^ (Start), 5-11
    {str} (repeating character range), 5-12
    examples, 5-13
Perform
    a Unix Command, 7-4
    Nested Query, 5-34
    Query, 5-14
    Separate Query, 5-34
Personal next Key assignment, 4-3
Perusing
    Backward, 5-20
    By Key value sequence, 5-20
    Exit, 5-20
    Forward, 5-20
    mode, 5-20
Phonetic search, 5-8
Phrase query, 5-9
Precision
    Data type, 3-3
Previous
    Attribute Page, 6-13
    Item, 6-14
    Page, 2-17
Printing or copying information, 2-10
Privilege
    Database query, 5-3
    Edit record, 5-22
    Look-up table Insertion, 4-11
    Record deletion, 5-26, 5-27
    Record edit, 5-24
    Record privilege alteration, 5-28
    Report access, 5-19
    Temporary record creation, 4-17
    Validation override, 4-13, 5-24
Privilege level
    Record, 5-28
    View, 6-20
Privileges, 2-18
    in General, 2-18
    Item, 2-19
    Operation, 2-20
    Record, 2-18
Prompt, 1-6
    Removal, 6-32
Pull-down menu, 2-5
    Keyboard Accelerators, 2-7

Q

Query, 1-2, 5-3
  Case search, 5-9
  Combine, 5-34
  Comments, 5-7
  Form, 1-6, 5-4
  Interrupt, 5-14
  Join, 5-34
  Mode, 1-7, 5-3, 5-4
  Perform, 5-14
  Range, 5-9
  Replacement, 6-34
Query mode
  Options, 7-12
Query mode Quick reference, A-2
Query type
  Range, 5-9
Quick reference
  Display mode, A-4
  Edit mode, A-2
  Field Editing Commands, A-1
  Global Edit mode, A-1
  Insert mode, A-2
  Query mode, A-2

R

Range queries, 5-9
Re-order
  current matches, 5-29
  Mode, 5-29
  Records, 5-29
Read only
  Form status, 6-19
Real
  Data type, 3-3
  Precision, 3-3
Rebuild, load and batch insertion option, 7-17
Record, 1-2, 1-6
  Copy, 5-31
  Deletion, 5-26, 5-35
  Deletion privilege, 5-26
  Discard from Retrieval, 5-33
  Edit, 5-22
  Edit privilege, 5-22, 5-24
  Global edit, 5-24
  History, 5-18, 5-23
  Insert, 4-15
  Match verification, 5-32, 5-35
  Print, 5-31
  Privilege level, 5-28
  Privileges, 2-18
  Re-ordering, 5-29
Record Histories, 5-18
Record Privilege Level Commands, 5-28
Records per page, 6-26
Redraw
  Screen, 2-14
Repeating Cursor Movements
  Report Form, 6-7
Report
  All matches, 5-19, 6-25
  Embed control characters, 6-34
  Footer section, 6-18
  Formats, 6-28
  Generation, 6-25, 6-28
  generation, 5-19
  Generation form, 5-19
  Header section, 6-18
  Heading section, 6-18
  Insert record number, 6-34
  Line joining, 6-33
  Line joining (word-wrap), 6-33
  Line removal, 6-31
  Line replacement, 6-32
  Multiple column output, 6-34
  Pagination, 6-35
  Prompt removal, 6-32
  Query replacement, 6-34
  Space removal, 6-30
  Space replacement, 6-31
  This match, 6-25
  this match, 5-19
  View Order, 6-17
Report Form
  Create/Edit, 6-4
  Deselect Objects, 6-5
  Select all Objects, 6-5
  Select Block, 6-5
  Select object, 6-5
Report form, 1-6, 6-3, 6-18, 6-27
  Change name, 6-23
  Change object, 6-11
  Copy, 6-24

Copy object, 6-10
Delete object, 6-12
Extract item, 6-8
Find Prompt, 6-7
Info menu, 6-19
Insert box, 6-9
Move down, 6-6
Move left, 6-6
Move object, 6-9
Move right, 6-6
Move to next word, 6-6
Move to previous Word, 6-7
Move up, 6-6
Moving around, 6-6
New header, 6-8
Repeating Cursor Movements, 6-7
Undo Command, 6-12
Report form Layout, 6-8
Report Generation
Interrupt, 6-22
Report menu, 6-3, 6-25
Report section
texpaginate, 6-38
Report; Options, 7-15
Reporting, 1-2, 1-3
Retrieval
Discard Record, 5-33
Return to Unix, 2-15

S

Save, 4-15, 5-23
Dave File, 5-33
Global edit, 5-24
Report form, 6-21
Screen
Copy, 2-12
Redraw, 2-14
Screen copy
Options, 7-10
Screen layout, 2-4
screenall, 2-12
screenblock, 2-12
screenshort, 2-12
Scroll
Backward, 5-16
Forward, 5-16
Security, 2-18
Select all Objects
Report Form, 6-5
Select Block
Report Form, 6-5
Select command, 2-5
Select item/field, 5-29
Selecting Objects
Report Form, 6-5
SHELL
Environment variable, 7-4
Shell variable
PATH, 1-10
Single term query, 5-7
Size
Form, 6-15, 6-19
Space
Removal, 6-30
Replacement, 6-31
Special Report Operators, 6-30
Start section
texpaginate, 6-37
Startup Macros.
Mode, 9-11
Statistics, 7-3
Status
Form, 6-19
Stemming search, 5-8
Stop process, 2-15
String
Data, 3-3
string Key type, B-2
Summarize, 5-30
Suspend or Invoke Shell, 2-15
System-wide options, 7-7

T

Temporary record creation privilege, 4-17
Temporary Save, 4-17
Terminal capabilities, 1-9
Terminology, 1-5
Terse output, 6-26
texcat
Line drawing, 6-39
texdesign, 1-5
texforms, 1-2, 1-5, 1-10
texopts, 8-5, 8-8
texpaginate, 6-35

Data section, 6-37
End section, 6-37
Footer section, 6-37
Footlines, 6-36
Header section, 6-37
Headlines, 6-36
Report section, 6-38
Start section, 6-37
texpress, 1-10
TEXPRESSOPTS, 8-8
Text
Data type, 3-3
Text perusal, 2-17, 5-30
Exit, 2-17
texvideo, 8-9
texvidopts, 8-5
The Administration Manager, 1-5
Tilde expansion, 7-4
Time
Library type, C-4, C-5
Type
Key, B-1

U

Undo, 3-5
Undo Command
Report Form, 6-12
Upper bound items, 5-9
User's retrieval limit, 5-14
Userid
Library type, C-7
Using this manual, 1-4
Utilities, 2-2

V

Validate Form, 6-19
Validation
Data, 4-10
Field, 4-12, 4-13
Hierarchy, 4-10
Key item, 4-10
Look-up table, 4-11
Override privilege, 4-13
Validation override privilege, 5-24
Verification of record matches, 5-35
Verify record matches, 5-32
Version of record, 5-18
Video
Library type, C-7
Videodisc, 5-33
after, 8-5
before, 8-5
Commands, 8-3
Connection, 8-8
Errors, 8-2
Examples, 8-7
header, 8-5
Interactive use, 8-9
Interface, 8-2
invocation, 1-10
Language, 8-3
Options, 8-5
Validation, 8-2
video, 8-8
View
Form, 6-15, 6-19
Privilege level, 6-20
Report Form, 6-23
Report Order, 6-17
View/Edit Attributes, 6-13

W

Wipe out flag, 5-24
Word Break Characters, 5-5

Y

yynnn Key type, C-3
yyyybc
Library type, C-4
yyyymmdd
Library type, C-4
yyyymmmdd
Library type, C-4