Information Management Series

HP ALLBASE/4GL Developer
Reference Manual Vol. 1

For MPE/iX Systems

HEWLETT PACKARD

FINAL TRIM SIZE : 7.0 in x 8.5 in

NOTICE

FINAL TRIM SIZE : 7.0 in x 8.5 in

Printing History

New editions of this manual incorporate all material updated since the previous edition. Listed below are the editions of this manual and their dates of publication.

First Edition	September 1988
Second Edition	September 1989
Third Edition	February 1990
Fourth Edition	May 1992

Preface

MPE/iX

MPE/iX, Multiprogramming Executive with Integrated POSIX, is the latest in a series of forward-compatible operating systems for the HP 3000 line of computers.

In HP documentation and in talking with other HP 3000 users you will encounter references to MPE XL. All programs written for MPE XL will run without change under MPE/iX. You can continue to use MPE XL system documentation, although it may not refer to features added to the operating system to support POSIX (for example, hierarchical directories).

Finally, you may encounter references to MPE V, which is the operating system for the HP 3000s not based on the PA-RISC (precision architecture-reduced instruction set computing) architecture. MPE V software can be run on the PA-RISC (Series 900) HP 3000s in what is known as compatibility mode.

About This Manual

This manual is a comprehensive reference manual for application developers who are using HP ALLBASE/4GL on HP 3000 computer systems running under the MPE-XL operating system. It provides a detailed description of the features and the use of the HP ALLBASE/4GL developer application. It does

not describe the procedures and techniques for designing HP ALLBASE/4GL applications, nor does it attempt to describe the basic data processing concepts required for designing and implementing HP ALLBASE/4GL applications. The manual assumes you are familiar with the basic concepts and terminology of HP ALLBASE/4GL.

Before you attempt to use this manual, read the *HP ALLBASE/4GL Developer Self-Paced Training Guide*.

Refer to the *HP ALLBASE/4GL Developer Administration Manual* for details of the administrator application.

Manual Structure

This manual is divided into two volumes containing 14 chapters and 6 appendixes, a glossary, and an index. Volume 1 contains chapters 1 to 10, and volume 2 contains chapters 11 to 14, the appendixes, and the glossary. Both volumes contain a comprehensive index.

Chapters 1 and 2 provide an overview of the structure of HP ALLBASE/4GL and the use of HP ALLBASE/4GL menus and screens. Chapter 3 describes the rules for naming and referencing items in HP ALLBASE/4GL. Chapters 4 through 13 are reference chapters that provide a comprehensive description of the facilities of the HP ALLBASE/4GL developer. Chapter 14 outlines the procedures for generating and testing applications.

The appendixes contain a system collating sequence table, a table of ASCII values, a list of file manager errors, details about using HP ALLBASE/4GL versions, details of the MPE-XL environment for HP ALLBASE/4GL, a list of limits in HP ALLBASE/4GL, and information about application portability.

The glossary defines the meaning of some terms that are specific to HP ALLBASE/4GL.

A quick reference guide is included as a separately bound supplement to this manual.

Related Publications

In addition to this manual, the following manuals are part of the HP
ALLBASE/4GL developer documentation set:

- The *HP ALLBASE/4GL Developer Self-Paced Training Guide.*
- The *HP ALLBASE/4GL Developer Administration Manual.*

Conventions

The syntax conventions used in this manual are summarized below.

Notation	Description
COMMAND	In syntax statements, command names are shown in bold UPPERCASE letters.
*ARGUMENT	Command arguments are prefixed with the symbol * and are shown in UPPERCASE letters.
Italics	Words in Italics in syntax statements denote a parameter that you must replace with a suitable name. For example: **FILE** *CLOSE *file_name*
[ ]	In syntax statements, square brackets enclose elements that are optional. Several elements stacked inside square brackets means that you may select any one of these elements. For example: $\begin{bmatrix} A \\ B \end{bmatrix}$ You can select A, B, or neither.
{ }	In syntax statements, braces enclose elements that you must include in the command. Several elements stacked inside braces means that you must choose one of the elements. For example:

$$\left\{ \begin{matrix} A \\ B \\ C \end{matrix} \right\} \texttt{You must select A or B or C.}$$

⊔	When necessary for clarity, the symbol ⊔ is used in syntax expressions to indicate a required blank or an exact number of blanks.
underlining	When necessary for clarity, underlining indicates user input. For example:
	`MPE/iX:` hello name.account
`shading`	Shaded text represents inverse video text on the terminal screen.
⬭	The symbol ⬭ indicates a key on the terminal keyboard. For example, (Return) indicates the carriage return key.
...	In syntax expressions, an ellipsis indicates that the preceding element may be repeated.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents

FINAL TRIM SIZE : 7.0 in x 8.5 in

2. Getting Started

Contents-2

FINAL TRIM SIZE : 7.0 in x 8.5 in

3. Referencing Rules

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-6

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-7

Contents-8

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-10

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-11

FINAL TRIM SIZE : 7.0 in x 8.5 in

7. Reports

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-13

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Contents-15

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Figures

Tables

FINAL TRIM SIZE : 7.0 in x 8.5 in

1

Overview

Introduction

HP ALLBASE/4GL is an advanced fourth-generation language. It enables you to design and implement application software by defining the required results, rather than the procedures necessary to achieve those results.

HP ALLBASE/4GL provides facilities for:

■ Developing new applications.

■ Modifying existing applications.

■ Producing multiple versions of applications for different users.

■ Producing developer documentation.

This manual is a reference manual for developers who are familiar with the basic terminology and concepts of HP ALLBASE/4GL. If you haven't already done so, we suggest that you work through the *HP ALLBASE/4GL Developer Self-Paced Training Guide* before you use HP ALLBASE/4GL (and this manual) to develop your own applications.

This chapter of the *HP ALLBASE/4GL Developer Reference Manual* provides an overview of the structure of HP ALLBASE/4GL and the procedures for developing applications.

FINAL TRIM SIZE : 7.0 in x 8.5 in

HP ALLBASE/4GL System Architecture

HP ALLBASE/4GL consists of the following major functional components:

- The operating system interface.
- The data manager.
- The administrator.
- The developer.
- End user applications.

This manual provides a detailed description of the facilities of the HP ALLBASE/4GL developer. To place this description in context, this chapter provides an outline description of the other HP ALLBASE/4GL components.

The Operating System Interface

This version of HP ALLBASE/4GL operates on Hewlett-Packard HP 3000 series 900 computer systems using the MPE/iX operating system.

The HP ALLBASE/4GL operating system interface handles all the communication between HP ALLBASE/4GL and the host operating system. For the most part, you will never need to use MPE/iX commands while developing HP ALLBASE/4GL applications. However, if the need arises, you can access the MPE/iX system to execute external programs in a language other than HP ALLBASE/4GL. You can also call such external programs from within HP ALLBASE/4GL.

Data Manager

The data manager handles all access to the data files used by an application.

The data manager provides access to HP ALLBASE/SQL and HP TurboIMAGE/iX databases from HP ALLBASE/4GL.

The HP ALLBASE/4GL developer also provides all the facilities you need to create and delete KSAM (keyed sequential access method) data files and serial data files.

This lets you choose the type of data storage that is most appropriate for your site and your application.

The Administrator

The administrator is an HP ALLBASE/4GL application. The HP ALLBASE/4GL system administrator uses the administrator application to define system user names, applications and versions, databases, and various system wide defaults.

Refer to the *HP ALLBASE/4GL Developer Administration Manual* for details of the administrator application.

The Developer

The HP ALLBASE/4GL developer is also an HP ALLBASE/4GL application in itself. This is the application that you use to create end user applications.

The developer application provides you with application development facilities that can be grouped under the following headings:

■ Dictionary.

■ Screen development.

■ Logic.

■ Reports.

The developer application also includes a module builder, which creates logic and screens automatically, and number of utilities. These utilities can copy or delete components within applications, and print documentation. With the developer application, several utilities run from MPE/iX are also provided.

The following chapters of this manual describe the facilities of the developer application in detail.

End User Applications

HP ALLBASE/4GL supports two types of end user applications: base applications and versions.

A version is an additional set of definitions that operate in conjunction with a base application to customize the application for a particular end user or a group of end users. You cannot create versions of an application unless the base application exists. Appendix B describes HP ALLBASE/4GL versions.

Phases of Application Development

HP ALLBASE/4GL does not impose many restrictions on the procedure you must use to develop applications. The principal restriction is that you must define most dictionary items before you can use them in other areas of HP ALLBASE/4GL.

Typically you will create applications in five phases as follows:

- Phase 1 - Planning.
- Phase 2 - Data definition.
- Phase 3 - Screen development.
- Phase 4 - Logic definition.
- Phase 5 - Report definition.

Planning Your Application

The first step is to define the requirements of your application. As with any conventional third generation language system, the best thing to do is to get well away from your terminal and sit down with pencil and paper. As part of the task, you need to identify the field specifications, record layouts, and data files the application needs.

Data Definition

The next step is to use the HP ALLBASE/4GL dictionary facilities to define the application components you have identified. Once you have defined the necessary field specifications and record layouts, you can build the application data files from within HP ALLBASE/4GL.

The HP ALLBASE/4GL dictionary contains the following types of entries:

- Field specifications.
- Validation items.
- Storage items.
- Record layouts.
- Database items.
- Messages.
- Help screens.

Field Specifications

A dictionary field specification defines the attributes of a field in a file, a screen, or a report. Since field specifications are stored in the dictionary, the definitions of fields on screens and reports can refer to these central specifications.

The HP ALLBASE/4GL dictionary field specifications do not contain any application data in themselves. They are simply templates that define the characteristics of a field.

The dictionary field specification defines the following characteristics for a field:

- The field specification name.
- The length of the field and the number of times it is repeated.
- The edit code and justification codes for the field. These codes determine the type of data (for example, numeric, alphanumeric etc.) that the field can hold, and how it is presented on the screen or a report.
- The names of any validation tables or ranges used to validate data for this field.

- The pad character used to fill unused spaces at the beginning or end of the field.

- The external storage type for the field.

- Any data manager specific attributes required for the field.

Validation Items

The dictionary validation items are either validation tables or validation ranges. A validation table is a table of values used to test input data for a field. A validation range specifies the lower and upper limits for a range of acceptable values for a field.

Storage Items

The HP ALLBASE/4GL dictionary storage items are somewhat different from the other dictionary entries. In contrast to the dictionary field specifications, the storage items do hold application data. The HP ALLBASE/4GL dictionary storage items are:

- Variables.

- Calculated items.

- Numeric constants.

- Alphanumeric constants.

- Scratch-pad field names.

- Work areas.

- Application titles.

HP ALLBASE/4GL calculated items are variables that are evaluated each time they are called by an application. The value of a calculated item is the result of an arithmetic expression or function that is evaluated when the item is referenced.

The HP ALLBASE/4GL scratch-pad contains 100 scratch-pad fields. Scratch-pad fields are temporary storage areas that you can use for storing data within an application. HP ALLBASE/4GL allocates memory for the scratch-pad dynamically. Each scratch-pad field takes on the attributes of the

1-6 Overview

data written into it, and the combined scratch-pad uses only the amount of memory required to hold the data contained within it at any given time.

HP ALLBASE/4GL work areas are memory buffers that are similar to file record buffers. However, work areas are not associated with a data file.

Record Layouts

The record layout definitions in the HP ALLBASE/4GL dictionary define the names and sequence of fields within file records. The record layout definitions also define the fields in the records that are used as keys for reading records from indexed data files. The HP ALLBASE/4GL data manager uses the key fields to build the file indexes when you create indexed data files.

Database Items

The dictionary database items include the definitions of HP ALLBASE/SQL tables and select lists, HP TurboIMAGE/iX data sets, KSAM data files, and serial data files. Once these databases items are defined, you can physically create HP ALLBASE/SQL tables and select lists, KSAM data files, and serial data files from within HP ALLBASE/4GL.

Messages

HP ALLBASE/4GL allows you to use a number of different types of messages to communicate with application end users. Messages can be simple literal strings, or combinations of items including literals, variables, constants, and screen field data.

All messages have a code to indicate the action to be taken by the user or HP ALLBASE/4GL. You can define message codes ranging from prompts or information, to queries, warnings, errors, and aborts.

Help Screens

HP ALLBASE/4GL allows you to associate a help screen with each application screen, and with each field on a data screen. The dictionary help screen definition contains the help screen name, and the text of the screen. The *Screen Header* screen and the *Screen Field Details* screen in the screen development menu allow you to associate help screens with an application screen or a particular field on an application screen.

You can link one or more further help screens to a help screen if you need to present more information than one screen can contain.

Screen Development

HP ALLBASE/4GL supports three types of application screens: menus, data screens, and windows. The procedure for creating screens is essentially the same for all types of screens. All the HP ALLBASE/4GL screen development facilities are accessed through the screen development menu.

The first step is to complete the screen header for an application screen. The *Screen Header* screen, accessed via the screen development menu, allows you to assign a name to the application screen and define its characteristics.

You can now use the HP ALLBASE/4GL screen painter to create the screen image.

The next stage is to complete the screen field details screen for the input and output fields on data screens and windows. This screen allows you to define the attributes of the input and output fields, and the various fields for data movement to and from the screen. This step is not needed for a menu.

You can associate one or more sets of function keys with each application screen. The function keys definition screen allows you to define the text and actions for the function key sets associated with application screens.

You may also automatically create a screen when you use the module builder. The module builder is a tool within HP ALLBASE/4GL that automatically creates a screen and the logic required to maintain a file or retrieve file record details.

Logic Definition

HP ALLBASE/4GL supports four types of logic structures: processes, functions, SQL logic blocks, and decision tables.

You may automatically create a process, functions, and SQL logic blocks when you use the module builder.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Processes and Functions

A process is similar to a program in a conventional language. A function is similar to a subroutine in a conventional language.

Processes

When a process starts, HP ALLBASE/4GL initializes most of the environment.

When a process starts, the following occurs:

- All HP ALLBASE/SQL cursors, HP TurboIMAGE/iX data sets, and application data files are closed.

- All file record buffers, work areas, and screen buffers are cleared.

- Any current process, function, screen, decision table or report is terminated.

- Any incomplete HP ALLBASE/SQL transaction is reversed.

At the start of a process, the content and state of the following remains unchanged:

- Communication area fields, except for *PROCESS.

- Scratch-pad fields.

- Switches.

- Variables.

When a process terminates, control will return only to a menu. This means, for instance, that if a decision table action executes a process, control will not return to the decision table when the process has completed. If there is no menu to return to, the application terminates.

Functions

When a function is completed, HP ALLBASE/4GL resumes running the application at the point where the function was initiated.

Functions can be nested. That is, one function may initiate another function.

HP ALLBASE/4GL processes and functions consist of a number of instructions that make up the logic block of the function or process. Each logic block can contain up to 99 lines of instructions where each instruction contains one

or more logic commands. HP ALLBASE/4GL has a set of about 40 logic commands.

SQL Logic Blocks

SQL logic blocks allow HP ALLBASE/4GL and HP ALLBASE/SQL to communicate. An SQL logic block can contain up to eight HP ALLBASE/SQL commands. At run-time, the commands in an SQL logic block are invoked by an SQL command in an HP ALLBASE/4GL logic block.

Decision Tables

Decision tables can be used to implement more complex conditional testing than is available using the IF and IFLOOP logic commands.

A decision table can test up to eight questions or conditions, and initiate up to eight actions. Each decision table can test up to 31 different combinations of outcomes for the eight decision table questions, and for each combination of outcomes, execute some or all of the eight decision table actions in a specified order.

The decision table actions can include processes, functions, screens, and reports.

Report Definition

The next stage in the application development process is to define the reports. All the HP ALLBASE/4GL report generation facilities are accessible from the reports menu.

The HP ALLBASE/4GL report generator reports on the contents of one primary file. Typically, this file is the result of a query (a SELECT command) on an HP ALLBASE/SQL database. If necessary, you can specify further selection criteria to limit the records selected from the file. You can also specify additional sorting before the report is printed. The report generator also allows you to define linkages to other types of files to read information for reporting. The report generator also allows you to define the report layout, totalling, and any special logic requirements for the report.

HP ALLBASE/4GL reports consist of a number of different line groups. Line groups are available for page headings, bottom of page lines, column headings,

subheadings, subtotal lines, and the normal detail lines that are printed for each file record selected for reporting.

The procedure for defining the layout of a report is similar to the procedure for laying out a screen. The HP ALLBASE/4GL report generator includes a report painter. The painter allows you to paint report lines exactly as they will appear on the final printed report.

HP ALLBASE/SQL Interface

The interface between HP ALLBASE/4GL and HP ALLBASE/SQL consists of the following components in HP ALLBASE/4GL:

- Dictionary table definitions.
- Dictionary select list definitions.
- SQL logic blocks.

This manual contains a description of the facilities in HP ALLBASE/4GL that allow you to use HP ALLBASE/SQL databases in conjunction with HP ALLBASE/4GL applications. It assumes that you are familiar with the basic features and terminology of HP ALLBASE/SQL. Consequently, this manual does not contain detailed descriptions of any HP ALLBASE/SQL features or the syntax and usage of HP ALLBASE/SQL commands. Refer to the *HP ALLBASE/SQL Reference Manual* for more information about using HP ALLBASE/SQL.

Dictionary Items

The dictionary contains facilities so that you may define and create HP ALLBASE/SQL base tables from HP ALLBASE/4GL. The table definitions use field specifications and record layouts to define the columns of a table.

HP ALLBASE/4GL select lists are "virtual" HP ALLBASE/SQL tables. They are built from the columns of one or more base tables, and can use aggregate expressions. A select list definition uses HP ALLBASE/4GL field specifications to define the columns of the select list.

Chapter 5 of this manual contains a detailed explanation of the HP ALLBASE/4GL facilities for defining HP ALLBASE/SQL tables and select lists.

SQL Logic Blocks

HP ALLBASE/4GL SQL logic blocks provide the means of passing SQL commands between HP ALLBASE/4GL and HP ALLBASE/SQL. An SQL logic block can contain up to eight SQL commands. An SQL logic block is invoked from a process or function by the SQL command.

Chapter 11 of this manual describes SQL logic blocks.

HP TurboIMAGE/iX Interface

The interface between HP ALLBASE/4GL and HP TurboIMAGE/iX consists of the following components in HP ALLBASE/4GL:

- Dictionary data set definitions.

- Dictionary field specification (data item) definitions.

- HP ALLBASE/4GL logic commands.

- Utilities used to load HP TurboIMAGE/iX schema file details into an HP ALLBASE/4GL application.

- Database access parameters and definitions (within the administrator application).

This manual describes how to develop HP ALLBASE/4GL applications that access HP TurboIMAGE/iX databases. You will need to be familiar with the functionality and commands of HP TurboIMAGE/iX, and this manual assumes that you have a firm knowledge of the HP TurboIMAGE/iX terminology and basic features. Consequently, it does not contain detailed descriptions of any HP TurboIMAGE/iX functionality. Refer to the *HP TurboIMAGE/iX Database Management System Reference Manual* for more information about HP TurboIMAGE/iX.

The following table shows the HP ALLBASE/4GL equivalent of HP TurboIMAGE/iX terms. In most cases, HP ALLBASE/4GL terminology

will be used in this manual. However, the HP TurboIMAGE/iX data set terminology will be used to distinguish operational differences between data sets and KSAM and serial data files.

HP TurboIMAGE/iX Term	HP ALLBASE/4GL Term
Data item	Field specification
Data entry	Record
Data set	File
HP TurboIMAGE/iX root file name	Database external name

Within HP ALLBASE/4GL, field specifications and record layouts are used to define the data items and the structure of data entries in HP TurboIMAGE/iX data sets.

Chapter 5 of this manual contains a detailed explanation of the HP ALLBASE/4GL facilities for defining field specifications, record layouts and file definitions for HP TurboIMAGE/iX data sets.

HP TurboIMAGE/iX database components can be defined within HP ALLBASE/4GL, but cannot be physically created within HP ALLBASE/4GL. However, you can create HP TurboIMAGE/iX databases using the normal creation methods. HP ALLBASE/4GL provides a utility to upload HP TurboIMAGE/iX schema details to HP ALLBASE/4GL. Chapter 10 of this manual describes this utility.

Application Generation

Some parts of HP ALLBASE/4GL applications must be generated before they can be executed. The generation process converts your input into a run-time form that HP ALLBASE/4GL can execute.

The following application items require generation:

■ Calculated items.

■ Decision tables.

■ Functions.

- Messages.
- Processes.
- Record layouts.
- Reports.
- Screens.
- SQL logic blocks.

The *Generate* option on the developer main menu allows you to generate all, or selected parts of an application.

Application Documentation

HP ALLBASE/4GL allows you to enter a description of most items when you define them. The first line of the description field is a short summary of the item being defined. The next three lines are available for you to enter a more detailed description of the item.

The developer utilities menu contains an option that allows you to print some or all of the developer documentation for an application. You can print the full descriptions, short descriptions, or only the names of the required items.

Application Security

The system administrator can specify a development security code for any application. The development security code prevents anyone other than a developer who has signed on to HP ALLBASE/4GL under the correct development security code from modifying any item that has been secured by the original developer. You can secure any of the following application components:

- Field specifications.
- Decision tables.

- Record layouts.
- Reports.
- Scratch-pad name declarations.
- Screens.
- Validation ranges.
- Validation tables.

HP ALLBASE/4GL Files

HP ALLBASE/4GL uses the following types of files:

- HP ALLBASE/4GL system files.
- Application data files.
- Program files.

The following paragraphs contain a brief summary of the various HP ALLBASE/4GL files and their purpose. Appendix D contains an explanation of the operating system environment required for these files.

System Files

HP ALLBASE/4GL uses a set of system files referred to as S-files. These files hold the complete definition of each HP ALLBASE/4GL application. These files reside in the group and account identified by the MPE/iX variable HP4SPATH.

Application Data Files

Application data files are the data files (other than HP ALLBASE/SQL tables and HP TurboIMAGE/iX data sets) referenced by the HP ALLBASE/4GL application.

Application data files can be KSAM data files or serial data files. Serial data files can have fixed length records, or variable length records.

These files may reside in any group and account to which the user has access rights. When you define a data file you must assign the file a name for use within HP ALLBASE/4GL and specify the name by which the operating system knows the file. This external name may use a fully qualified file name to specify the group and account for the file.

HP ALLBASE/4GL application data files do not need to be restricted to one group or account. However, as a developer you must have the correct access permissions to be able to create the data files as required.

Program Files

HP ALLBASE/4GL uses a number of program files enabling you to develop an application. By default, these program files reside in the PUB.SYS group and account.

2

Getting Started

This section tells you how to sign on to HP ALLBASE/4GL, how to select items from menus, and how to insert and edit data in the various screen fields.

A typical HP ALLBASE/4GL session involves:

■ Logging in to the MPE/iX system.

■ Signing on to HP ALLBASE/4GL as a developer and selecting an application or version.

■ Choosing application development facilities from the HP ALLBASE/4GL developer menus to create, modify, delete, or run an application or its components.

■ Signing off the system.

MPE/iX Login

Before you can sign on to HP ALLBASE/4GL, you must log in to the host MPE/iX computer system. Your system manager must allocate a user name and login for you. The details of your login will vary depending on the needs of your particular site.

If you need more information about the MPE/iX environment for HP ALLBASE/4GL, refer to appendix D.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Running HP ALLBASE/4GL

After you have logged in to the system, you can start HP ALLBASE/4GL. The standard HP ALLBASE/4GL installation uses an MPE/iX command file called *HP4GL*. You run HP ALLBASE/4GL by typing `HP4GL` at the MPE/iX prompt. This command file starts HP ALLBASE/4GL and automatically presents the sign on screen.

Note   You cannot run HP ALLBASE/4GL on the MPE/iX system console.

The following description covers the procedure for signing on to HP ALLBASE/4GL.

Signing On to HP ALLBASE/4GL

You can sign on to HP ALLBASE/4GL as a developer and then either develop a new application, or modify an existing application. You can also create versions of existing applications. A version is a set of additional components that work with an existing application. You can use versions to adapt an application to suit the needs of a particular end user, or group of end users.

In all cases, your system administrator must complete the developer validation screen to register you as a developer. The system administrator can specify a development security code for an application or version. If you want to make changes to any secured components in the application or version, you must enter this development security code when you sign on to the application.

You can sign on to HP ALLBASE/4GL using the sign on screen or by providing the sign on information on the MPE/iX sign on command line.

The Sign On Screen

```
HP ALLBASE/4GL B.05.00 HP30601A Copyright(c) 1986-92 Hewlett-Packard Co.

                        HP ALLBASE/

                        4GL


         _____


         User Name    ██████        User Password   ░░░░░░░



                        Start    17* 18        Commit   Help    Exit
                        Again                   Data
```

To sign on to HP ALLBASE/4GL, complete this screen as follows:

1. Enter your HP ALLBASE/4GL developer user name and then press (Return).

2. If a password exists, enter it and then press (Return).

3. Press the **Commit Data** function key.

   HP ALLBASE/4GL then displays the developer sign on window.

```
HP ALLBASE/4GL B.05.00 HP30601A Copyright(c) 1986-92 Hewlett-Packard Co.

                        HP ALLBASE/

                        4GL


        _____


        User Name     _____        User Password  _____

        Application or Version to be developed      ██████████
        Development Security Code


                        Start      19* 50        Commit    Help    Exit
                        Again                     Data
```

4. Enter the name of the application or version you want to work on, and then
   press (Return). If the application or version has a development security code
   and you want to work on secured components in the application or version,
   enter the code and press (Return). Press the `Commit Data` function key
   to complete the sign on screen. (You can clear all the fields on the sign on
   screen by pressing the `Start Again` function key.)

When you finish the sign on screen, HP ALLBASE/4GL displays the developer
main menu.

Bypassing the Sign On Screen

HP ALLBASE/4GL allows you to specify your developer user name and
password, and the application name and security code at the MPE/iX prompt.
When you do this, HP ALLBASE/4GL takes you directly to the developer
main menu.

To do this you must supply the developer and application details when you
invoke HP ALLBASE/4GL. For example:

```
:HP4GL "-utricia:go4it -atraining:codeword"
```

2-4   Getting Started

HP ALLBASE/4GL first verifies that *tricia* is a valid developer user name and that *go4it* is her correct password. HP ALLBASE/4GL then verifies that *training* is an application that *tricia* has access to. If any of these verifications fail, HP ALLBASE/4GL displays an error message and then exits. If *codeword* is the correct security code for the application, tricia will be able to modify secured items.

Refer to "Command Line Options" and "Sign On Screen Bypass" later in this chapter for more information about command line options and login bypass.

HP ALLBASE/4GL Terminals

HP ALLBASE/4GL runs on most HP terminals supported by the MPE/iX operating system. It supports all the standard features of these terminals such as function keys and special editing keys. Depending on your terminal type, HP ALLBASE/4GL also makes use of the terminal's line drawing, color and touchscreen capabilities.

HP ALLBASE/4GL Terminal Initialization

At the start of each session, HP ALLBASE/4GL uses the current value of the MPE/iX variable HP4TERM to establish the terminal type and capabilities. HP ALLBASE/4GL uses a support file for your terminal to determine how to drive it. If the value of the HP4TERM variable is invalid, HP ALLBASE/4GL displays an error message and terminates.

The value of the HP4TERM variable should be set to the type of terminal that you are using. For example, if you are using an HP 2392a terminal, you should set the HP4TERM variable to the value "HP2392A" using the command:

```
SETVAR HP4TERM "HP2392A"
```

Refer to appendix D for a summary of the acceptable values for the HP4TERM variable.

If the HP4TERM variable is not set, the HP ALLBASE/4GL start-up command file assumes a default value of "HP" for the HP4TERM variable.

FINAL TRIM SIZE : 7.0 in x 8.5 in

System Character Sets

HP ALLBASE/4GL supports the full HP Roman8 extended character set. To use eight-bit characters, your terminal must support the HP roman8 character set and be in eight-bit mode.

If your terminal doesn't have line drawing capabilities, HP ALLBASE/4GL will replace any line drawing characters with equivalent ASCII characters.

Using a Touchscreen Terminal

If you want to use touchscreen operation, make sure that the terminal is in touch mode before you start HP ALLBASE/4GL. You can turn touch *off* at any time while HP ALLBASE/4GL is running, but you cannot turn touch *on* again. If you do, touch operation may not work reliably.

Accessing HP ALLBASE/QUERY

If you have HP ALLBASE/QUERY installed on your system, you can access it directly from the HP ALLBASE/4GL sign on screen. You may then make ad-hoc queries or browse an HP ALLBASE/SQL database. To do this, enter the user name `hpqm` in the *User Name* field, and accept the default application name of *hpqm*. Press the `Commit Data` function key to start HP ALLBASE/QUERY.

When you exit from HP ALLBASE/QUERY, you are returned to the MPE/iX prompt.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Developer Main Menu

```
                        HP ALLBASE/4GL Developer
    Dictionary  Screens  Logic  Reports  Module Builder  Utilities  Generate  Test














    Sign On                           2*  2    System  Activate   Help
    Screen                                      Keys     Item
```
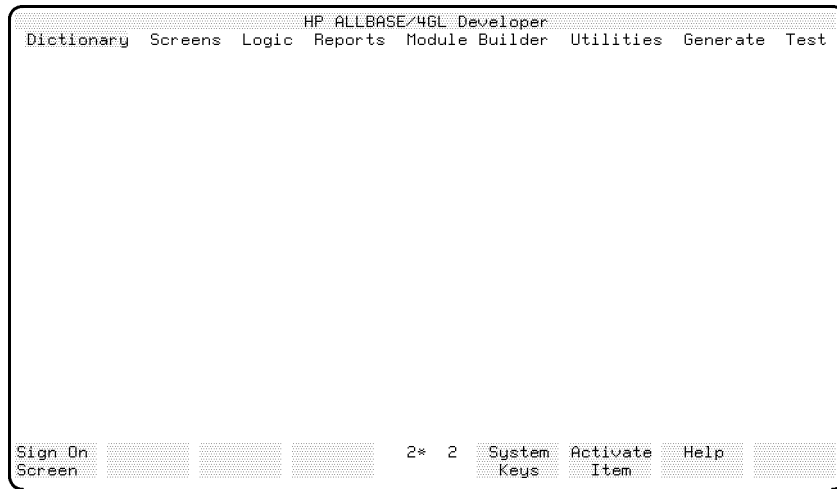
The developer main menu allows you to access other menus and facilities in the
HP ALLBASE/4GL developer application as described below.

Dictionary

Selecting this option takes you to the dictionary menu. The dictionary menu
allows you to define field specifications, validation items, data storage items,
record layouts, database items, messages, and help screens.

Screens

Selecting this option takes you to the screen development menu. This menu
allows you to describe and create menus, data screens and window screens, and
to define screen fields and function keys.

Logic

Selecting this option takes you to the logic menu. The logic menu allows you to
define functions, processes, SQL logic blocks, and decision tables.

Reports

Selecting this option takes you to the reports menu. The reports menu allows you to describe and create reports; to define record selection, sorting criteria, and linkages; and to define report output formats.

Module Builder

Selecting this option displays the module builder screen. The module builder allows you to automatically build a file maintenance or inquiry module. A module consists of a user interface screen and the HP ALLBASE/4GL logic to perform the necessary file access.

Utilities

Selecting this option takes you to the utilities menu. Options in the utilities menu allow you to print, copy, or delete application components.

Generate

Selecting this option takes you to the application generation menu. Options on this menu allow you to generate some or all components of the application. You must generate all components of an application before you can run it.

Test

Selecting this option lets you run your application from within the HP ALLBASE/4GL developer.

Selecting Menu Items

HP ALLBASE/4GL accepts input from the terminal using the MPE/iX character mode. The read terminator characters used by HP ALLBASE/4GL are [Return] and [Tab]. This means that HP ALLBASE/4GL does not process any of your input until it receives a carriage return or tab character. Keep this general rule in mind as you use HP ALLBASE/4GL menus and screens. The next section describes the various ways you can interact with an HP ALLBASE/4GL menu.

When HP ALLBASE/4GL initially displays a menu, the first item is highlighted. To activate the highlighted item, press [Return] or press the

`Activate Item` function key. HP ALLBASE/4GL then executes the selected item.

Use any of the following methods to select another item:

- Use the (Tab) key to select the item you want:

    - The (Tab) key selects the next item in the tabbing sequence, which is from left to right, top to bottom, with screen wrap-around from bottom right to top left.

    - Pressing (Shift) and (Tab) together followed by (Return) selects the next item in the reverse tabbing sequence. Reverse tabbing sequence is from right to left, bottom to top, with screen wrap-around from top left to bottom right.

- Use any combination of the cursor control keys (cursor left) (cursor right) (cursor up) and (cursor down) followed by the (Return) key to select the item you want. As you press the cursor keys, the cursor itself moves about the screen. When you press the (Return) key, HP ALLBASE/4GL selects the menu item closest to the cursor. Note that only menu items in the active menu will be selected.

    - The (cursor home) key, followed by the (Return) key, selects the top left item on the menu.

    - Pressing (Shift) and (cursor home) together, followed by the (Return) key, selects the bottom right item on the menu.

- HP touchscreens: Touch the item you want. HP ALLBASE/4GL responds to this request immediately and highlights the item.

- Type a selection abbreviation:

    - To select a menu item by abbreviation, enter the first letter of the menu item title. A single character display field, at the bottom left corner of the screen, echoes the character you type. When you press the (Return) key, HP ALLBASE/4GL highlights the item corresponding to the abbreviation. If there is more than one item with the same abbreviation, enter the same letter again and press the (Return) key again. HP ALLBASE/4GL highlights the next item corresponding to the abbreviation.

Getting Started 2-9

The following example shows the use of selection by abbreviation.

```
                              HP ALLBASE/4GL Developer
        Dictionary  Screens  Logic  Reports  Module Builder  Utilities  Generate  Test

                    ┌─────────────────────┐
                    │ Header              │
                    │ Painter             │
                    │ Details             │
                    │ Function Keys       │
                    │                     │
                    │ Catalog Display     │
                    └─────────────────────┘




         Dict.    Utility   Logic    Reports    5* 15   System   Activate   Help   Previous
         Menu     Menu      Menu     Menu               Keys     Item              Menu
```
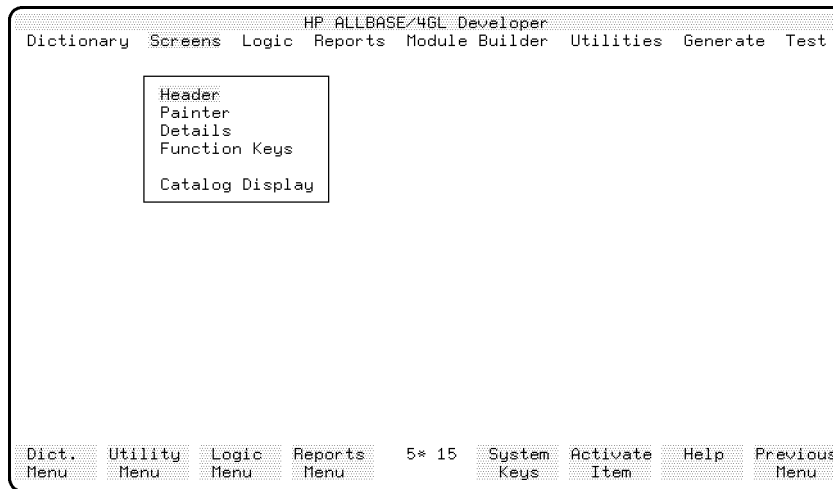
When HP ALLBASE/4GL first displays this menu, *Header* is highlighted. You can select the other items on this menu by typing the following abbreviations:

■ Press P (Return) to select *Painter*.

■ Press D (Return) to select *Details*.

■ Press F (Return) to select *Function Keys*.

■ Press C (Return) to select *Catalog Display*.

You cannot select any items from the main menu because it is not the active menu. Only one menu can be active at a time.

Executing Menu Items

You can only execute the currently highlighted menu item. To execute the highlighted menu item, press the (Return) key, or the   **Activate Item**   function key.

2-10 Getting Started

Menu Bypass

HP ALLBASE/4GL has a menu bypass facility that allows you to display screens and run processes directly without stepping through the menus.

Use this facility carefully; some screens are displayed by an HP ALLBASE/4GL process rather than being displayed directly from menus. To access one of these screens using the menu bypass, you must execute the process that calls the screen. If you try to display such a screen directly, it will not operate correctly.

The menu chart at the rear of this manual shows the HP ALLBASE/4GL developer menu structure in a diagrammatic form. This chart also shows the names of the screens, or the names of the HP ALLBASE/4GL processes that display the screens.

To display a specific menu or screen, press ⌐/⌐ and then press the ⌐Return⌐ key at any menu. The system responds by displaying a message requesting you to enter the name of the item you want to execute. Type D- followed by the name of the menu or screen you want to display, and then press ⌐Return⌐. The system responds by displaying the menu as though you had selected it via a menu. If the screen you want is displayed by a process, type ⌐/⌐ and press the ⌐Return⌐ key at any menu and then respond to the system message by typing P- followed by the name of the process.

Function Keys

In addition to the selections displayed on menus, HP ALLBASE/4GL displays application and system facilities on eight function key labels located along the bottom of the screen.

You can select the displayed facility by pressing the associated function key. The function keys ⌐f1⌐ through ⌐f8⌐ are located along the top row of the keyboard.

With HP touchscreen terminals you can select a function key facility directly by touching the function key label on the screen.

The HP ALLBASE/4GL developer application has four standard function keys associated with every screen-the keys ⌊f5⌉ through ⌊f8⌉. The actions of these keys are standard throughout HP ALLBASE/4GL and are not described elsewhere in this manual unless they are significantly different from those described below.

Standard Function Keys

The standard function keys and their actions are:

⌊f5⌉ `System Keys` Loads the standard function key set that enables you to access a number of system and terminal utilities. Refer to System Keys Function Key Set below.

⌊f6⌉ `Activate Item` (menus) Confirms your menu selection and executes the selected item.

⌊f6⌉ `Commit Data` (data screens) Confirms your data entry and terminates processing of the current screen.

⌊f7⌉ `Help` Displays the help screen associated with the current screen or field.

⌊f8⌉ `Previous Menu` Returns you to the last menu displayed by the system. This menu may not necessarily be the previous menu in terms of the HP ALLBASE/4GL menu hierarchy. The previous menu function key returns you to the menu from which you accessed the current screen. If you selected the current screen by using function keys or the menu bypass, you may not be returned to the next higher menu in the hierarchy.

System Keys Function Key Set

Pressing the `System Keys` function key on a developer screen loads the following function key set.

⌊f1⌉ `Main Menu` Returns you to the developer main menu.

⌊f2⌉ `Name Recall` loads a set of function keys that let you scroll through the item names that have been defined for an active field.

⌊f3⌉ `Clear Screen` Clears all fields on the current screen.

[f4]   `Refresh Screen` Repaints the current screen image, including the contents of all fields.

[f5]   `Previous Keys` Reloads the standard set of function keys for the screen.

[f6]   `More Keys` Loads the second set of system keys. Refer to More Keys Function Key Set below.

[f7]   `Help` Displays the help screen associated with the current screen or field.

[f8]   `Previous Menu` Returns you to the last menu displayed by the system.

Name Recall Function Key Set

These keys enable a context-sensitive name recall function. If the active field on a screen has a defined field type and item names associated with the field, the developer can use this set of function keys to recall the item names one at a time in the active field, to complete partially entered item names, or to access a catalog display screen that lists all of the names for a type or all the available types.

[f1]   `First Item` lets you return to the first item in the group.

[f2]   `Previous Item` lets you return to the previous item in the group.

[f3]   `Next Item` lets you see the next item in the group.

[f5]   `Previous Type` changes the display to the previous item type.

[f6]   `Next Type` changes the display to the next item type.

[f7]   `Help` displays the help screen for this screen.

[f8]   `Restore Field` restores the original field display.

For more information about the name recall function, refer to the "Using Name Recall" section later in this chapter.

More Keys Function Key Set

If you have loaded the system keys function key set by pressing the
`System Keys` function key on a developer screen, pressing the `More Keys`
function key loads the following function key set.

[f1] `Screen Printing` Loads the screen printing function key set. These keys allow you to print a screen image on the system printer, or on a slave printer attached to the terminal.

[f2] `Op. System` Temporarily suspends execution of HP ALLBASE/4GL and runs the MPE/iX command interpreter. When you exit from the temporary session (by typing `EXIT`), HP ALLBASE/4GL processing resumes at the point where you suspended it.

[f3] `ISQL` If HP ISQL is installed on the system, pressing this key temporarily suspends execution of HP ALLBASE/4GL and invokes HP ISQL. When you exit from HP ISQL, you are returned to HP ALLBASE/4GL.

[f4] `ALLBASE QUERY` If HP ALLBASE/QUERY is installed on the system, pressing this key suspends execution of HP ALLBASE/4GL and starts HP ALLBASE/QUERY. HP ALLBASE/QUERY automatically connects to the current application's database. When you exit from HP ALLBASE/QUERY, you are returned to HP ALLBASE/4GL.

If the current application has an open ALLBASE/SQL transaction when you press this function key, HP ALLBASE/4GL displays a warning message, allowing you to complete or reverse the transaction before starting HP ALLBASE/QUERY.

[f5] `Previous Keys` Reloads the previous function key set.

[f6] `Catalog Display` Displays the catalog display screen for the application you are running. A general catalog display screen can be accessed through any of the menus.

[f7] `Help` Displays the help screen associated with the current screen or field.

[f8]   `Exit`   Terminates the current HP ALLBASE/4GL session.

Entering and Editing Data

When HP ALLBASE/4GL first displays a data screen, it highlights the first data input field on the screen. HP ALLBASE/4GL positions the cursor at the first data entry column in the field.

You can enter data from the keyboard to the highlighted field.

HP ALLBASE/4GL accepts input from the terminal via MPE/iX's character mode, line mode. HP ALLBASE/4GL uses [Return] and [Tab] as read terminator characters. Therefore, HP ALLBASE/4GL does not process any of your input until it receives a carriage return or tab character. Keep this general rule in mind as you use HP ALLBASE/4GL screens. The next section describes the various ways you can enter, edit, and move between fields on a data screen.

Editing a Field

You can use any of the following terminal keys to edit data or to move the cursor within a data entry field:

■ The [Insert char] key toggles insert mode on or off. When you are in insert mode, typing a character inserts the character in front of the cursor, between existing characters. This causes all characters to the right of the cursor to move across the field by one character (up to the end of the field) each time you insert a character. Characters moved past the end of the field are lost.

■ The [Delete char] key deletes the character at the cursor.

■ The [Backspace] key moves the cursor one column to the left within the current field. Moving the cursor back past the left edge of the field has no effect on the field.

■ The [left arrow] key moves the cursor one column to the left within the current field. Moving the cursor back past the left edge of the field has no effect on the field.

FINAL TRIM SIZE : 7.0 in x 8.5 in

- The (right arrow) key moves the cursor one column to the right within the current field. Moving the cursor past the right edge of the field has no effect on the field.

Note

If you enter data after moving the cursor outside the current field, the cursor is immediately returned to the start of the field and the data entry begins from there.

- The (Clear line) key clears the current field from the cursor position to the right end of the field.

- The (Clear display) key clears the current field from the cursor position to the right end of the field. If you press the (Return) key immediately following the (Clear display) key, HP ALLBASE/4GL additionally clears all input fields from the current field to the bottom of the screen.

- The (Insert Line) (Return) key sequence moves the contents of the current field to the next field to the right or below the current field provided that the next field has the same edit code and length as the current field. The current contents of the next field are shuffled to the following field if the same conditions apply. The contents of the last field in the sequence is lost.

- The (Delete line) (Return) key sequence deletes the contents of the current field. If the next field to the right or below has the same edit code and length as the current field, this key shuffles the contents of the next field into the current field.

Committing a Field Entry

Complete each field by entering the required data and pressing one of the keys listed below to initiate a field commit action. When you initiate a field commit action, the system processes the current field entry and activates the next appropriate field.

Entering and Editing Data

If the field commit action you initiate commits the last field on the screen, the system processes the data in the field and the cursor remains in the field. You can then press the   Commit Data   function key to terminate the screen.

The [Return] **Key**

The [Return] key initiates a field commit action for the current field and moves the cursor to the start of the next field in the tabbing sequence. If there is no input field past the current field, the current field is processed and the cursor remains on the current field.

The [Tab] **Key**

Pressing the [Tab] key initiates a field commit action for the current field and moves the cursor to the start of the next field in the tabbing sequence. If there is no input field past the current field, the current field is processed and the cursor remains on the current field.

Correcting Mistakes

If you attempt to commit a field that contains invalid data, HP ALLBASE/4GL displays an error message and highlights the field containing the invalid data so you can correct the mistake. If you wish, you can clear the field and move the cursor to a previous field. You cannot move the cursor past the field containing the invalid data.

Moving Between Data Fields

By using the cursor keys, the [Shift] [Tab] key combination, or a touchscreen, you can move to a data field other than the next one in the normal forward sequence.

Using the Cursor Control Keys

- You can use the [up arrow], [down arrow], [left arrow], and [right arrow] keys to move the cursor to a position outside the current field. When you press the [Return] key, the current field is committed and the field closest to the cursor is made the current field. If you enter any data, between moving the cursor and pressing the [Return] key, the cursor movement is ignored and the new data is entered from the first column of the current field.

- The [home arrow] [Return] sequence initiates a field commit action for the current field and moves the cursor to the first input field in the tabbing sequence.

- Pressing [Shift] and [home arrow] together followed by [Return] initiates a field commit action for the current field, and moves the cursor to the last input

field in the tabbing sequence. If the cursor is already at the last field in the tabbing sequence, the field is processed and the cursor remains on the current field.

The (Shift) (Tab) Combination

Pressing (Shift) and (Tab) together followed by (Return) initiates a field commit action for the current field, and moves the cursor to the start of the previous field in the tabbing sequence if the cursor is at the first column of the current field.

If the cursor is not at the first column in the current field, the system moves the cursor to the first column in the field. In this case there is no need to press the (Return) key to initiate the movement. Since the cursor does not leave the current field, a field commit is not initiated.

HP Touchscreen Terminals

Touch the screen at or near the field where you want to enter or edit data. HP ALLBASE/4GL responds to this request immediately. The current field is committed and the field closest to the point you touch becomes the current field and is highlighted.

Committing Screens

When you have finished entering data on a data screen, you must press the `Commit Data` function key to confirm your entries and terminate processing of the screen.

When you press the `Commit Data` function key, the system initiates a field commit action for the current field (that is, the field occupied by the cursor) if it hasn't already been committed, or it contains data that is different from the data it contained at the time of the last field commit action for the field.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Using the Name Recall Function

The context-sensitive name recall facility available on developer screens and run-time administrator screens helps you maintain consistency in item and field names.

Activating the name recall facility puts the system into search mode. When in search mode, the function keys let you search through item names, change the display to another item type, and access the item name catalog. The message "Search mode [type]" appears on the screen's message line.

To search item names

1. To see the next or previous item, choose the `Previous Item` or the `Next Item` function key.

2. To return to the first item, choose the `FirstItem` function key.

3. To search the list for another item type, choose the `Next Type` or the `Previous Type` function key.

4. To search alphabetically through the current list of item names, press the `Next Item` key.

   If the selected field has no item names defined, the following message appears on the message line: "Search mode [type] WARNING: No items found."

Note	The item names are a list of items that are already defined for that item type. They may not always represent the correct choices for that particular field.

Using Name Completion

In addition to search item names, the name recall facility includes a name completion function. When you enter a partial name in a field, the name completion function attempts to complete the field with an existing name defined for that item type.

To use the name completion facility:

1. Type the first word or partial word of the item name you want to search for at the active field.

   The name must be a valid name already defined for the current ALLBASE/4GL application.

2. Press the `Name Recall` key.

   The field is completed, if possible, from the defined names for that field's item type. For instance, if you are in a field that requires screen names, the defined names are screen names.

3. If the name completed is not the desired name, you may browse through other names by pressing the `Next Item` or `Previous Item` function keys.

   The keys scroll you through the list of all possible item names in alphabetical order, even if the names do not match the text pattern originally entered.

Note that name completion is performed only for the right-most text in the field (that is, the right-most alphanumeric characters after a non alpha-numeric character other than - or *). Other text is ignored. For example, you can complete the record name in a field that accepts both a file name and a record name (for example, *file.record*). In this case, HP ALLBASE/4GL will default to the defined files for the search since the field is defined for a file.

To search through the Records, you must switch the item type. While performing name completion for the records, HP ALLBASE/4GL will display the first record layout that matches the text entered, which may or may not be a record defined for the file specified.

Checks are made for the amount of space left within the field before completion is allowed. If there is not enough space in the field to complete the name based upon the longest possible name for the current item type, HP ALLBASE/4GL will display a message saying there is not enough room in the field. This condition should never occur if you search for names having the same item type as the current field; however, it might occur if you have changed the item type for the search.

Using Catalog Display Function

Name recall also allows you to use the catalog display function to display all
defined names for an item type.

To show all item names for the active field

1. Choose the `Catalog Display` function key to show the names of all
   items for the item type indicated.

2. Press the [Return] key or choose the `Previous Screen` function key to
   return to the original screen.

Returning to Non-Search Mode

To return to non-search mode, complete the current field by pressing the
[Return] key or the `Restore Field` function key. The function keys revert to
their previous values and normal data entry can resume.

Note	Some fields cannot be left blank. If you cannot find an existing item name and press the [Return] key to continue, this message may appear:
	`This is a required field -- it must not be left blank.`
	Enter an item name in the field to continue.

Using the Tab Key for Name Recall

In addition to using the system function keys to activate name recall, you can
also activate the tab key for use with name recall. To do so, insert the **-n**
option into the command line used to start HP ALLBASE/4GL.

Once activated for name recall, the tab key puts you into search mode when
you are in a developer screen (such as the Field Specification screen) and the
cursor is on a field where name recall is applicable.

In search mode, the tab key lets you access item names in two ways:

1. If you do not type anything in the field, pressing the tab key will show the
   first item name (if any) defined for the field type.

2. If you have filled in the field with at least one or more characters that begin
   a valid entry for the field, pressing the tab key will cause the name recall
   facility to complete the item name.

Pressing (Shift)—(Tab) in search mode returns you to the front of the field.
Pressing (Shift)—(Tab) a second time returns you to normal mode.

Using On-Line Help

On-line help is available throughout the developer application. To use the
on-line help, simply press the `Help` function key.

HP ALLBASE/4GL help screens are associated with each field on data input
screens. There are also general help screens for each data screen, and for
menus. In many cases, the help screens have more than one "page". You can
step forwards or backwards through the help screens with the function keys on
the help screens.

HP ALLBASE/4GL may display the following function keys with help screens:

(f1)   `Field Desc` HP ALLBASE/4GL displays this key if the current
          help screen is associated with an input field that uses a dictionary field
          specification. If you press this key, HP ALLBASE/4GL displays the
          details of the dictionary field specification.

(f2)   `Display Range` HP ALLBASE/4GL displays this key if the current
          help screen is associated with an input field that has a validation range
          defined for it. If you press this function key, HP ALLBASE/4GL
          displays the validation range.

(f3)   `Display Table` HP ALLBASE/4GL displays this key if the current
          help screen is associated with an input field that has a validation table
          defined for it. If you press this function key, HP ALLBASE/4GL
          displays the contents of the validation table.

(f4)   `Screen Help` HP ALLBASE/4GL displays this key if the current
          help screen is associated with an input field on the screen. Pressing
          this key displays the help screen associated with the current screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in

[f5]   `Previous Help` HP ALLBASE/4GL displays this key if there is
            more than one page of help for the current screen or field, and you are
            not on the first page. Pressing this key redisplays the previous page of
            help.

[f6]   `More Help` HP ALLBASE/4GL displays this key if there
            is a further page of help linked to the current help screen. HP
            ALLBASE/4GL displays the next page of help when you press this
            key.

[f8]   `Exit Help` When you press this key, HP ALLBASE/4GL restores
            the original screen display and resumes processing at the point where
            you asked for help.

Printing Screen Images

HP ALLBASE/4GL allows you to print a screen image on an MPE/iX system
printer, or a slave printer attached to the terminal.

You can print a screen image using the `Screen Printing` function keys set,
which you can access by pressing the `System Keys` function key.

You can also print the current screen image by pressing (CTRL) and (P) together,
followed by (Return). If you use this method of screen printing, the destination
printer depends on the value of the MPE/iX variable HP4SCREEN_PRINT.
By default, HP ALLBASE/4GL writes the screen image to the formal file
designator HP4REP if this variable is undefined. Refer to appendix D for more
information about MPE/iX variables.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Signing Off

At the end of your HP ALLBASE/4GL session, return to the developer main menu. Press the `Sign On Screen` function key to return to the sign on screen.

To exit completely from HP ALLBASE/4GL, you can now press the `Exit` function key. You can also exit from HP ALLBASE/4GL by pressing the `Exit` function key in the *More Keys* set of system function keys.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Command Line Options

HP ALLBASE/4GL allows you to use command line options to run the program in a number of different ways. These options are available to developer users and end users. When used with options, the command line becomes:

:HP4GL [*"options"*] [*variable = value*] ...

In this command, *options* can be any of the options shown below separated by spaces.

Option	Description
-a [*application_name*] [*:application_passwd*]	This is the form of the **-a** option used by an end user. It invokes HP ALLBASE/4GL and passes control directly to the initial action defined for *application_name*. If *application_name* is not specified, the application defaults to that specified for the end user in the administrator end user validation screen. If the end user has no default application, HP ALLBASE/4GL exits with an error message. The **-u** option must be specified with this option.
-a *application_name* [*:security_code*]	This is the form of the **-a** option used by a developer. It invokes HP ALLBASE/4GL and passes control directly to the *developr* main menu (developing the application specified by *application_name*). The *security_code* field is the development security code for the application. The **-u** option must be specified with this option.
-i [*delay*]	Turns on logic tracing with a pause on trace messages of *delay* seconds. The *delay* value can be any value from 0 through 9. If *delay* is not specified, the pause defaults to 3 seconds. All messages are logged into a temporary file named *HP4TRACE* in the user's current group.

Option	Description
-l [*logo_screen*] [:*logo_application*]	This option allows you to display a customized window (*logo_screen*) on the sign on screen. If *logo_screen* is not specified or cannot be found, HP ALLBASE/4GL attempts to display the default window, *logo*. If this cannot be found, HP ALLBASE/4GL displays the standard logo window. HP ALLBASE/4GL searches the application specified by *logo_application* for the window *logo_screen*. If *logo_application* is not specified, HP ALLBASE/4GL searches the default application, *HPlogo*. If the application *HPlogo* is not defined, HP ALLBASE/4GL displays the standard HP ALLBASE/4GL logo window. Refer to Customized Sign-On, page 2 - 20 for further information.
-m [*delay*]	Sets the pause on each HP ALLBASE/4GL message to *delay* seconds. *Delay* can be any value from 0 to 9. If *delay* is not specified, the pause defaults to 3 seconds.
-n	Activates the [Tab] key for use with Name Recall function. This function allows developers to use the [Tab] key to scroll through item names previously defined for an active field on an ALLBASE/4GL screen.
-s	Turns on logging of HP ALLBASE/4GL messages. When logging is turned on, all messages are logged into a temporary file named *HP4TRACE* in the user's current group.
-u *user_name* [:*user_passwd*]	Bypasses the standard sign on screen and logs the user specified by *user_name* into the HP ALLBASE/4GL application specified by the -a option. If *user_name* is a developer, HP ALLBASE/4GL runs the developer application. The -a option must be specified with this option.

The *variable* option on this command line allows you to set values for the MPE/iX variables or job control words used by HP ALLBASE/4GL.

You can set values for the following variables and job control words:

- HP4SPATH
- HP4DATAPATH
- HP4SQLPATH
- HP4TIPATH
- HP4FSPATH
- HP4APPNPATH
- HP4DBMPATH
- HP4TERM
- NLUSERLANG (job control word)
- NLDATALANG (job control word)

Values specified on the command line apply only for the current HP ALLBASE/4GL session. The existing values (if any) for these variables and job control words are restored when you exit from HP ALLBASE/4GL.

An example of a command line that includes options and variables would be:

```
hp4gl "-m1" HP4SPATH=SGROUP
```

Refer to appendix D for more information about these variables and job control words.

Sign On Screen Bypass

Developer users and end users can bypass the HP ALLBASE/4GL sign on screen and go directly to the initial action of the application if HP ALLBASE/4GL is invoked with the **-a** and **-u** command line options. You must use the **-a** and **-u** options together.

For a developer user, the **-a** option takes two arguments: *application_name* and *security_code*. The name of the application to be developed is *application_name*, and *security_code* is its development security code (if any). If you specify *security_code*, you must precede it with a colon (:). You can still

access an application without specifying the security code for the application, but you can't modify any secured components in the application unless the correct security code is used.

For an end user, the **-a** option also uses two arguments: *application_name* and *application_passwd*. The name of the application that the user wants to run is *application_name*, and *application_passwd* is its password (if any). If the user specifies *application_passwd*, it must be preceded with a colon (:). If the user has a default application specified in the administrator, the *application_name* can be omitted.

Customized Sign On

You can display a customized logo window when an HP ALLBASE/4GL application is run with the **-l** command line option.

The **-l** option takes two arguments: *logo_screen* and *logo_application*. The argument *logo_screen* is the name of the window that you want to display on the sign on screen. The argument *logo_application* is the name of the HP ALLBASE/4GL application in which *logo_screen* is defined. If *logo_application* is specified you must precede it with a colon (:).

If the **-a** and **-u** options are used on the same command line, the **-l** option and its arguments are ignored when HP ALLBASE/4GL starts up. However, the **-l** option is processed when you return to the sign on screen from within the developer application or the end user application.

The screen *logo_screen* **must** be a window. If it is a menu or a data screen, HP ALLBASE/4GL ignores it and displays the standard HP ALLBASE/4GL logo screen. Use the HP ALLBASE/4GL screen painter to create the window. The screen cannot contain any data input or display fields, and the only valid system items for this window are the *DATE and *TIME communication area fields. Only lines 1 through 21 are displayed on the screen. The top line of the window is displayed as the second line of the sign on screen (the Hewlett-Packard copyright notice and product part number occupy the top line). If the custom window extends beyond line 14, HP ALLBASE/4GL may overwrite parts of the window when it displays the user name/password and application name/password fields.

HP ALLBASE/4GL searches for *logo_screen* in the application *logo_application*. If *logo_screen* does not exist in *logo_application*, HP ALLBASE/4GL searches in the application *HPlogo* for *logo_screen*.

If *logo_screen* is not found in either of these applications, HP ALLBASE/4GL searches for a window named *logo* in *logo_application*, and then in the application *HPlogo*.

If you omit *logo_screen* from the command line, HP ALLBASE/4GL searches for a window named *logo*. If you omit *application_name* from the command line, HP ALLBASE/4GL only searches in the *HPlogo* application. If you use the **-l** option with no arguments, HP ALLBASE/4GL searches for a window named *logo* in the *HPlogo* application. If you do not specify the **-l** option at all, HP ALLBASE/4GL uses the standard sign on window.

Getting Started   2-29

3

Referencing Rules

Introduction

This section describes the rules for specifying the names of HP ALLBASE/4GL elements, and describes the rules for referencing elements from logic blocks, screens, decision tables, and reports.

Naming and Referencing Rules

Within HP ALLBASE/4GL, you can name and reference a number of different types of elements. These elements are outlined in the following paragraphs.

Developer-Defined Elements

In the course of developing an application, you will define names for field specifications, data files, data storage elements, screens, logic elements, and reports.

The Communication Area

The communication area contains a number of fields that both you and the system can use. These fields have predefined names, although the system administrator can assign one synonym for each communication area field name.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Scratch-Pad Fields

HP ALLBASE/4GL allows you to use up to 100 scratch-pad fields for temporary data storage. Each scratch-pad field takes on the attributes (length, edit code and justification) of the data moved into it.

HP ALLBASE/4GL allocates memory for scratch-pad fields dynamically, minimizing the application's demands on system memory. This means that using scratch-pad fields may be more efficient than using variables.

HP ALLBASE/4GL Names

Any HP ALLBASE/4GL name you create must meet these requirements:

- Contain only alphabetic, numeric or underscore characters, that is, *A* to *z*, *0* to *9* and _ (underscore).

  This is the same as the requirements for the type K edit code.

- Start with an alphabetic character.

- Contain no space character.

- Be unique for each type of item you are naming.

  However, you can generally use the same name again for a different type of item. For example, you can use the same name for a screen and a report.

- Be from 1 to 8 characters long for:

    file names.
    SQL select lists.
    module names.

- Be from 1 to 16 characters long for:

    alphanumeric or numeric constants.
    application titles.
    calculated items.
    decision tables.
    field specifications.
    functions.

3-2   Referencing Rules

function key sets.
help screens.
HP TurboIMAGE/iX database names.
messages.
processes.
record layouts.
reports.
scratch-pad fields.
screens.
SQL logic blocks.
validation ranges.
validation tables.
variables.
work areas.

HP ALLBASE/4GL is case sensitive with respect to these names. You must always type the name of an element exactly the same each time you use it. For example, the variable V-Item_Cost is **not** the same variable as V-item_cost.

Although you can use up to 16 characters for some HP ALLBASE/4GL names, try to keep names as short as possible. In some cases, using the full number of characters for field specifications, file names, and record layout names may result in names that are too long to fit in the data entry fields of some screens.

HP ALLBASE/SQL Names

You may define names for the following HP ALLBASE/SQL items from inside HP ALLBASE/4GL:

- Base table names.

- Base table column names.

  HP ALLBASE/4GL uses the field specification names for the table record layout to name the columns of a base table.

Names for HP ALLBASE/SQL items defined in HP ALLBASE/4GL must meet these requirements:

- Be no more than eight characters long for HP ALLBASE/SQL base tables.

- Be no more than sixteen characters long for field specifications used to define base table column names.

- Use the characters A to Z, 0 to 9 or _ (underscore).

- Start with an alphabetic character.

- contain no extended characters (8-bit characters).

HP ALLBASE/SQL automatically changes lowercase characters in these names to uppercase.

Note   Since HP ALLBASE/4GL is case sensitive and HP ALLBASE/SQL is not, it is possible for several different HP ALLBASE/4GL names to refer to the same HP ALLBASE/SQL item. HP ALLBASE/4GL displays a warning message if you attempt to define a file name, and the same name already exists in case shifted form.

HP TurboIMAGE/iX Names

You may define names for the following HP TurboIMAGE/iX items from within HP ALLBASE/4GL:

- Data set names.

- Field specification (data item) names.

- Database names (within the administrator application).

HP TurboIMAGE/iX accepts all legal HP ALLBASE/4GL characters, except underscores (_). In addition to HP ALLBASE/4GL characters, the following characters may be used in HP TurboIMAGE/iX names:

+, -, *, /, ?, ', #, %, &, and @.

FINAL TRIM SIZE : 7.0 in x 8.5 in

These characters cannot be used for any HP ALLBASE/4GL names,
such as the file name and the field specification name. However, all legal
HP TurboIMAGE/iX characters may be used for defining external
HP TurboIMAGE/iX names, such as the Schema Data Set Name field on the
file/SQL table definition screen, and the Data Item Name field on the data
manager specific field attributes screen.

HP ALLBASE/4GL automatically changes lowercase characters in names to
uppercase when communicating with HP TurboIMAGE/iX.

Note   Since HP ALLBASE/4GL is case sensitive, and HP
            TurboIMAGE/iX is not, it is possible for several different
            HP ALLBASE/4GL names to refer to the same HP
            TurboIMAGE/iX item. For example, *F-order* and *F-Order* are
            different within HP ALLBASE/4GL, but both map to the HP
            TurboIMAGE/iX data set name *ORDER*.

Item and Field Referencing

You can reference the various elements in HP ALLBASE/4GL by specifying a
prefix indicating the type of item, a hyphen, and the item name.

FINAL TRIM SIZE : 7.0 in x 8.5 in

The following table lists the element referencing prefixes:

Prefix	Description
A-	Application title
B-	Work area buffer
C-	Alphanumeric constant
D-	Database name (HP TurboIMAGE/iX)
F-	File record field
M-	Master title
N-	Numeric constant
P-	Scratch-pad field
R-	File record buffer
S-	Screen field
U-	Calculated item
V-	Variable
W-	Work area field
*	The asterisk (*) prefixes system item names including:

- communication area fields.

- switches.

- command arguments.

- the current screen field, or a relative screen field reference.

Substring Referencing

A substring is a portion of the contents of a field. Using a substring reference allows you to refer to the portion of the contents of the field that you require.

The substring reference must occur immediately after the normal field reference (including subscripting, if applicable) but before any qualifying information such as a screen name or file name.

You can reference a field substring by specifying the starting character position number, a comma, and the number of characters in the substring. You must enclose the entire reference in square brackets.

Format

$data\_ref[start\_pos, number\_of\_chars]$

$data\_ref[start\_pos, number\_of\_chars].qualifier$

Parameters

data_ref

The name of the field. You must prefix the field name with the prefix character for the item type.

start_pos

The first character position to be included in the substring.

number_of_chars

The length, in characters, of the substring.

You can represent both the *start_pos* and the *number_of_chars* entries with one of the following items containing the required values:

- Number (maximum 255).
- Numeric constant.
- Numeric variable or calculated item.
- Screen field reference.

- Scratch-pad field reference.

- File record field reference.

- Work area field reference.

The entries specifying the *starting_position* and *number_of_characters* for the substring cannot contain substring or subscript references.

qualifier

You must always use a qualifier for S- (screen field), F- (file record field), and W- (work area field) data types when you use substring referencing. The qualifier specifies the name of the screen, file (and record layout) or work area to which the field name belongs. You must always use the current screen name as a qualifier for a substringed screen field reference. You must prefix the combined item reference with the prefix character for the item type.

Additional Information

You can refer to substrings in any alphanumeric field, but you cannot refer to substrings in the following fields:

- Numeric communication area fields.

- A scratch-pad field containing a number.

- A numeric variable with an N or S edit code.

You can include substring references in absolute screen field references (*S*nn* format) and relative screen field references. (* or *+*nn* or *-*nn* format.)

You can also reference scratch-pad fields (that is *P*nn*) by substring provided the data in the scratch-pad field is not numeric.

You cannot use substrings with references to file record buffers (R- references) or work area buffers (B- references).

Example 1

```
*[3,2]
```

This example references the third and fourth characters of the current screen field. That is, the two characters from the third character in the field.

Example 2

```
* DATE[1,2]
```

This statement references the first two characters of the communication area field *DATE. These two characters correspond to the current year, since *DATE holds the current date in the format YY/MM/DD.

Subscript Referencing

You can define a field on any of the following as having more than one occurrence:

- An HP TurboIMAGE/iX data set.

- A serial file.

- An indexed KSAM file record.

- A work file record.

- A work area.

Typically this situation would occur if all the fields held similar data. For example, you may require multiple lines of a description, and want to use the same name for each field. You can reference each field by the same name. HP SQL tables cannot use subscripted column definitions.

You can only use subscripts with dictionary field specifications. The dictionary definition (refer to chapter 5) for these fields must specify that they have more than one occurrence.

To use a subscript reference, you must enclose the subscript in parentheses immediately after the field name. The subscript reference must appear before any substring reference.

Format

F-data_reference(subscript).qualifier

W-data_reference(subscript).qualifier

Parameters

data-reference A dictionary field specification reference.

subscript The number of the occurrence of the field being referenced. This can be one of the following, containing the number of the subscript to be used:

- A number.
- A numeric constant.
- A numeric variable or calculated item.
- A screen field reference.
- A scratch-pad field reference.
- A file record field reference.
- A work area field reference.

The term specifying the number of the occurrence cannot contain subscript or substring references.

qualifier The name of the file (optionally followed by a record layout name) or the name of the work area containing the field.

Example 1

```
F-desc(*).stockdet
```

This example references an occurrence of the field *desc* in the file *stockdet*, where the current screen field contains the subscript of the actual occurrence to be referenced.

The value contained in the current screen field must not exceed the number of occurrences defined for the field *desc*.

Example 2

```
F-desc(2)[1,10].stock
```

This example references the first ten characters of the second occurrence of the field *desc* on the file buffer *stock*.

File Record Field Referencing

You can reference the content of a field in a file record by using the relevant dictionary field specification name prefixed by F-.

You must append one of the following to this field reference, to define the file reference:

■ *file_id*

■ *file_id.record_name*

where *file_id* is the HP ALLBASE/4GL name for the file and *record_name* is a record layout defined for the file. The *record_name* term is not required for reference to fields on HP ALLBASE/SQL tables or HP ALLBASE/SQL select lists.

If you don't specify a record layout name in a reference to a field on a serial data file, KSAM data file, or HP TurboIMAGE/iX data set, HP ALLBASE/4GL uses the default record layout for the field.

The dictionary field specification name must be defined for the record layout name you specify. If you don't specify a record layout name, the dictionary field specification name must be defined for the default record layout.

You can reference a substring within a non-numeric field.

For KSAM data files, HP TurboIMAGE/iX data sets, and serial data files, you can add the occurrence number immediately after the field specification name if the dictionary field specification allows more than one occurrence. The occurrence number must be in parentheses. If you don't specify an occurrence number and the field specification allows more than one occurrence, HP ALLBASE/4GL uses the first occurrence of the field.

Formats

    F-name.file_ref
    F-name[start_pos,number_of_chars].file_ref

F-name(occurrence).file_ref
F-name(occurrence) [start_pos,number_of_chars].file_ref

File Record Buffer Referencing

You can reference the entire contents of a file record buffer using the file name, prefixed by R-.

You must follow this reference with one of the following to define the file reference:

- *file_id*

- *file_id.record_name*

where *file_id* is the HP ALLBASE/4GL name for the file and *record_name* is a record layout defined for the file. The *record_name* term is not required for reference to buffers for HP ALLBASE/SQL tables or HP ALLBASE/SQL select lists.

If you don't specify a record layout name in a reference to a serial data file buffer, KSAM data file buffer, or HP TurboIMAGE/iX data set buffer, HP ALLBASE/4GL uses the default record layout for the file.

You cannot use substrings in file record buffer references.

File record buffer references are only valid in the logic commands MOVE, EXTERNAL, and DM IMAGE *LOCK.

Format

R-file_ref

Screen Field Referencing

You can reference a screen field by:

- Name.

- Absolute field number.

- Relative field number.

You can reference a substring of a non-numeric screen field, with any of these methods.

Reference by Name

You can reference a screen field by using the dictionary field specification name assigned to the field. If you use this form of referencing, you must specify the screen name as a qualifier except in the cases listed below. This forms a fully qualified screen field reference.

If the same dictionary field specification name is associated with more than one field on the same screen, you can only reference the first field by name. The first field is the field with the lowest field number.

If the dictionary field specification allows more than one occurrence, you can only use referencing by name for the first screen field that is assigned to the field specification name.

Formats

> *S-name.screen*
> *S-name.window*
> *S-name*[*start_pos,number_of_chars*].*screen*
> *S-name*[*start_pos,number_of_chars*].*window*

Limitations

Note that HP ALLBASE/4GL only maintains one set of screen field buffers at any one time. The existing contents of the screen buffers are lost when HP ALLBASE/4GL displays a subsequent screen. HP ALLBASE/4GL allows you to use a function called from a screen field to display a further screen. If you do this, any further screen references in the logic block displaying the original screen will become references to the current screen, and not the original

screen. Under these conditions, you should reference screen fields by number, or limit referencing by name to references in the logic commands listed under exceptions below.

Exceptions

Under the following circumstances, you don't need to specify the screen name as a qualifier to the field name:

- When you specify the screen field as one of the following on the screen field details screen:

    - the default data movement field identifier,

    - the primary data movement field identifier,

    - other data movement identifiers.

    In this case, you can omit the screen name. If you do specify a screen name it must be the name of the current screen.

- Some HP ALLBASE/4GL logic commands allow you to reference screen fields by name without using the screen name. This allows you to use a single function with a number of different screens, each having one or more fields with the same name. When used in this way, the command associated with the screen field reference always refers to the current screen.

    Logic commands that permit this option are:

    CLEAR

    FIELD

    MOVE

    SHOW

    TIE

Reference by Absolute Number

You can reference screen fields by their absolute field number. The first field on the screen is field number one.

When HP ALLBASE/4GL displays a window on a screen, the fields in the window are renumbered automatically. The numbering of the window

fields starts from the highest field number on the main screen plus one. HP ALLBASE/4GL renumbers the window fields in the order that they were originally defined.

Formats

$*Snn$
$*Snn[start\_pos, number\_of\_chars]$

where nn is the screen field number.

Reference by Relative Number

You can reference screen fields as a field number relative to the current screen field.

This technique allows you to reference the current field, or any other field up to nine fields before or after the current field.

Formats

* (= contents of the current field)
*+n
*-n
$*[start\_pos, number\_of\_chars]$
$*+n[start\_pos, number\_of\_chars]$
$*-n[start\_pos, number\_of\_chars]$

where n is a number from 1 to 9.

Work Area Field Referencing

You can reference the content of a work area field by using the relevant dictionary field specification name prefixed by W- .

You must follow this reference with the name of the work area.

The dictionary field specification name must be defined for the work area you specify.

You can reference a substring of any non-numeric field.

You can add the occurrence number immediately after the field specification name if the dictionary field specification allows more than one occurrence. The occurrence number must be in brackets. If you don't specify an occurrence number and the field specification allows more than one occurrence, HP ALLBASE/4GL uses the first occurrence of the field.

Formats

W-name.work_area
W-name[start_pos,number_of_chars].work_area
W-name(occurrence).work_area
W-name(occurrence)[start_pos,number_of_chars].work_area

Work Area Buffer Referencing

You can reference the contents of a work area buffer using the work area name, prefixed by B-.

You cannot use substrings in work area buffer references.

Work area buffer references are only valid in the MOVE and EXTERNAL commands.

Formats

B-work_area

Scratch-Pad Field Referencing

HP ALLBASE/4GL provides you with 100 scratch-pad fields.

You can reference scratch-pad fields by name or by number.

You can reference a substring of a scratch-pad field if its contents are not numeric.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Reference by Name

You can assign a name to a scratch-pad field using the dictionary scratch-pad fields screen. (Refer to chapter 5.) A scratch-pad field defined this way can be referenced by name.

Formats

> *P-name*
> *P-name[start_pos, number_of_chars]*

Reference by Number

You can reference scratch-pad fields by their absolute number.

Formats

> *P nn
> *P nn[start_pos, number_of_chars]*

where *nn* is the scratch-pad field number, from 00 to 99.

Scratch-pad field number 00 is the same as the communication area field *PASS.

Calculated Item Referencing

You can reference a calculated item by using the item name prefixed by U-.

You can reference a substring of a non-numeric calculated item.

Formats

> *U-name*
> *U-name[start_pos, number_of_chars]*

FINAL TRIM SIZE : 7.0 in x 8.5 in

Constant Referencing

You can reference an alphanumeric constant by using the constant name prefixed by C-.

You can reference a substring of an alphanumeric constant.

Formats

C-name
C-name[*start_pos,number_of_chars*]

Literals

Some HP ALLBASE/4GL screens and logic commands allow the use of literals. Literals can use any printable characters. Unless this manual specifies otherwise, literals must be enclosed in quotes ("). The quotes are not treated as part of the literal.

To include a single quote character within a literal, follow it immediately with another quote character. (For example, `"A quote "" within a literal"` would indicate a literal that means this: "A quote ' within a literal").

Numeric Constant Referencing

You can reference a numeric constant by using the numeric constant name prefixed by N-.

Formats

N-name
N-name[*start_pos,number_of_chars*]

Master Title Referencing

You can reference a master title by using the master title name prefixed by M-.

Format

M-name

When HP ALLBASE/4GL encounters a reference to a master title, the system first looks for an **application** title of the same name. If the application title does not exist, the system uses the named master title.

Variable Referencing

You can reference a variable by using the variable name prefixed by V-.

You can reference a substring of a non-numeric variable.

Formats

V-name
V-name[start_pos,number_of_chars]

Communication Area Field Referencing

The communication area contains fields that are accessible both by you and by HP ALLBASE/4GL. The communication area consists of fields that you can alter, and fields that are only available for reference purposes.

You can reference any communication area field by its HP ALLBASE/4GL name, or by its synonym if the system administrator has defined one.

You can reference a substring of a non-numeric communication area field.

Formats

*name
*name[start_pos,number_of_chars]

where *name* is the name of one of the communication area fields or its synonym.

HP ALLBASE/SQL Host Variable References

You can reference an ALLBASE/4GL item within an SQL logic block or SQL select list by using the full item name prefixed with a colon (:).

SQL logic blocks and SQL select lists can contain references to ALLBASE/4GL file record fields, work area fields, fully qualified screen field names, variables, calculated items, and constants. You cannot use references to scratch-pad fields or communication area fields.

You cannot use substrings in references to screen field names within SQL logic blocks and SQL select lists.

Format

:*data_ref*

where *data_ref* is the ALLBASE/4GL item name, including the item's type prefix.

For example,

`:V-due_date`	Refers to the variable *due_date*.
`:F-unit_cost.option`	Refers to the field *unit_cost* on the file record *option*.
`:S-acct_no.orders`	Refers to the field *acct_no* on the screen *orders*.

References to host variables cannot include null indicator variables. However, ALLBASE/4GL does support reading and writing null values. You can use the *NULL argument for the IF command to test for null status in ALLBASE/4GL. You can also use the *NULL argument for the MOVE command to set the ALLBASE/4GL indicator associated with an HP ALLBASE/SQL table or select list buffer field to the null status.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Action Prefixes

In addition to item and field referencing prefixes, HP ALLBASE/4GL also uses a set of prefixes to identify the action types that can be called from menus and function keys. The following table lists the action prefixes used in HP ALLBASE/4GL.

Code	Action Type
B-	Background process
D-	Data screen or menu
F-	Function logic block
H-	Help screen
I-	Internal action
P-	Process logic block
R-	Report
X-	External program

Some of these action prefixes are identical to normal prefixes. For instance, the item referencing prefix P- indicates scratch-pad fields. The action prefix P- indicates a process. HP ALLBASE/4GL differentiates between the two types of prefix according to their usage. Normal item referencing prefixes have no meanings in the situations where action prefixes are used.

When executing an X- action item, HP ALLBASE/4GL searches for the external program in the current group, and then in the groups specified by the MPE/iX system variable HPPATH.

Edit Codes

Some HP ALLBASE/4GL developer screens ask you to enter an *Edit Code* for a field or item. The edit code defines the type of data that is acceptable as screen input for the field or item.

The table below lists the available edit codes and their meanings.

Code	Meaning
X	Any printable character.
A	Alphabetic characters only (A to z, extended characters, or space).
U	Forced uppercase. Same as X edit code, but shifts all lowercase alphabetic characters to uppercase.
K	Alphanumeric characters only (A to z, 0 to 9, extended characters, and underscore).
N	Unsigned number (0 to 9, a period ".", or a comma ",").
S	Signed number. Same as N edit code, but may include + or -.
Q	Question. Y, y, N, or n only. Any characters beyond the first character in the field are ignored.
D	Date field. Must be eight characters long. The date may be either MM/DD/YY or DD/MM/YY (depending on the system-wide date format) where MM, DD, and YY represent the month, day, and year respectively.
T	Date field, defaulting to the current date.

Field specification edit codes, combined with other attributes, map to external HP ALLBASE/SQL or HP TurboIMAGE/iX data types. Chapter 5 contains details about the mapping.

Storage Type Codes

The field specification screen asks you to enter a storage type code. This code defines the way in which the data manager will physically store numeric data. For non-numeric data, the data will always be stored as alphanumeric text (storage type C).

FINAL TRIM SIZE : 7.0 in x 8.5 in

For numeric data, The table below lists the storage type codes and their meanings.

Storage Type	Meaning
C	Character (number stored in the same format as it appears on a screen)
F	Four byte, IEEE floating point
G	Eight byte, IEEE floating point (double precision)
I	Two byte 2's complement (integer)
L	Four byte 2's complement (integer)
P	Packed decimal (BCD, or binary coded decimal)

Justification Codes

Some HP ALLBASE/4GL screens ask you to enter a justification code for a field or item. The justification code determines the way data is displayed on a screen or printed on a report.

The table below lists the justification codes and their meanings.

Code	Meaning
L	Data is aligned to the left boundary of the field. Trailing blanks are replaced by the pad character.
R	Data is aligned to the right boundary of the field. Leading blanks are replaced by the pad character.
C	Data is centered within the left and right boundaries of the field.
N	No justification is performed.

FINAL TRIM SIZE : 7.0 in x 8.5 in

4

Communication Area and Switches

The Communication Area

The HP ALLBASE/4GL communication area contains a number of fields that both you and the system can access. The communication area fields are either alterable fields or read-only fields.

You can use the contents of all the communication area fields for display on screens, in reports, or in processes and functions.

The value in any communication area field can be passed to an external program via the EXTERNAL logic command.

You can reference a substring of any alphanumeric communication area field by using the substring referencing rules.

The following paragraphs explain the characteristics of each communication area field.

Alterable Communication Fields

You can use the contents of the following fields for interrogation and display, and you can alter the contents of these fields if required.

*COUNT(n)

Five fields used primarily by the report generator for counting the number of times a nominated report line group is printed. Each field is 20 characters long with no decimal places, has a type S edit code, and is right justified.

The five count fields are referenced as *COUNT(1) to *COUNT(5).

You can use these fields as numeric work variables when HP ALLBASE/4GL is not executing a report. When HP ALLBASE/4GL executes a report, the values in these fields may change.

*CROSS(n)

Five fields used primarily by the report generator for holding across-the-page totals of numeric fields. Each field is 20 characters long, with 2 decimal places, has a type S edit code, and is right justified.

The five cross totalling fields are referenced as *CROSS(1) to *CROSS(5).

You can use these fields as numeric work variables when HP ALLBASE/4GL is not executing a report. When HP ALLBASE/4GL is executing a report, the values in these fields may change.

*ERROR

This field may contain one of the following:

- The text of an HP ALLBASE/4GL message detected while an application is running.

- The text of an indexed KSAM error message detected while an application is running.

- The text of an HP ALLBASE/SQL error message detected while an application is running.

- The text of a message returned by the SQLEXPLAIN command in an SQL logic block.

You can include the text contained in this field in messages on application screens.

This field has a type X edit code, is left justified and varies in length depending on the messages placed in it. *ERROR will only contain the text of the most recent message. If no messages have been placed in the communication area field, *ERROR will be blank.

FINAL TRIM SIZE : 7.0 in x 8.5 in

*ERRORDM

This field contains the text of any HP TurboIMAGE/iX message returned by HP TurboIMAGE/iX. If no error occurs, this field is blank.

This field has a type X edit code, is left justified and varies in length depending on the messages placed in it.

*FIELDNO

Contains the current screen field number. The field is three characters long, has a type N edit code, and is right justified.

You can change the contents of this field. Depending on when you change the contents of the field, the screen processing logic changes as follows:

If HP ALLBASE/4GL encounters the change after the display processing for a field, (that is, when *ENTERED is *off*), HP ALLBASE/4GL displays the contents of the current field and immediately transfers control to the field whose number has been placed into *FIELDNO.

If HP ALLBASE/4GL encounters the change after data entry in an input field, (that is, when *ENTERED is *on*) HP ALLBASE/4GL checks to see if an error has been generated. If so, instead of remaining on the current field (the normal action with an error) HP ALLBASE/4GL transfers control to the field whose number has been placed into *FIELDNO. This action is in contrast to setting *NEWTIE or using the TIE command. In this case, HP ALLBASE/4GL processes the specified field after processing of the current field has been completed. Changing the value in *FIELDNO enables you to return the user to a field other than the field that generated the error.

If you change the value in *FIELDNO in a function called from a function key, HP ALLBASE/4GL commits the current screen field when the function exits. Screen processing resumes at the field specified by the new value in *FIELDNO.

FINAL TRIM SIZE : 7.0 in x 8.5 in

*FILENAME

Contains the external name of the KSAM data file, HP TurboIMAGE/iX data set, or serial data file currently accessed by HP ALLBASE/4GL. If HP ALLBASE/4GL is not currently accessing a file or data set, this field holds the external name of the last data file or data set accessed by the system.

For HP TurboIMAGE/iX data sets, the name contained in this field is the external schema data set name.

Note	The *FILENAME communication area field is not applicable for temporary serial files. While HP ALLBASE/4GL is accessing a temporary serial file, the name in *FILENAME is meaningless.

*FILENAME always contains the external name of the file as specified on the dictionary file definition screen.

*FILENAME is an alphanumeric field.

Dynamic Serial File Naming

This field also allows you to specify the external name of a serial file dynamically. To do this you must first specify *FILENAME as the external name of the file on the dictionary file definition screen. You can then use the MOVE command to move an MPE/iX file name into the *FILENAME field. This determines the external name of the file operated on by a subsequent FILE command using the specified HP ALLBASE/4GL file name.

Normally, HP ALLBASE/4GL appends the group and account name determined by the current value of the MPE/iX variable HP4DATAPATH to the name in *FILENAME. If the value in *FILENAME is a fully qualified filename, (that is, a filename in the form FILE.GROUP.ACCOUNT) HP ALLBASE/4GL does not prefix the name with the value of HP4DATAPATH.

HP ALLBASE/4GL obtains a file number for a file when you first access it. This same number is used for each access to the same file until it is closed. You can close a file with the FILE *CLOSE command. HP ALLBASE/4GL closes all files automatically at the end of a process. The value in *FILENAME is only used on the first access to the file when HP ALLBASE/4GL obtains

the file number. Changing the value in *FILENAME has no effect while the file is open. If you need to access a different external file using the same HP ALLBASE/4GL file name you must close the HP ALLBASE/4GL file. You can then move the new external file name into *FILENAME and access the file again. HP ALLBASE/4GL will automatically obtain a file number for the new external file and associate it with the HP ALLBASE/4GL file until it is closed.

*FUNCTION

Contains the current function name. It is blank if HP ALLBASE/4GL is not currently executing a function. The field is 16 characters long, has a type K edit code, and is left justified.

*IMSTAT

Contains the HP TurboIMAGE/iX status array, which represents status information about the last HP TurboIMAGE/iX library procedure performed.

The status array has a repeated count of ten, and each number in the array has a type S edit code of S and a type I storage code. Each number has a maximum of six characters, including the + or - sign.

The first value in the status array is the error number returned by the most recent HP TurboIMAGE/iX library procedure executed. This error number value is also contained in the *IOSTATDM communication area field.

The meaning of status arrays for each HP TurboIMAGE/iX library procedure can be found in the *HP TurboIMAGE/iX Database Management System Reference Manual*. Refer to chapter 10 of this manual for a list of the common library procedures initiated by HP ALLBASE/4GL.

*INDEXNO

Contains the number of the file index to be used for all subsequent accesses of the application's KSAM data files or HP TurboIMAGE/iX data sets. If you want to access the application data files or data sets via differing index numbers, you must set this field to the required value before each file access. The field is two characters long, has a type N edit code, and is right justified.

If you don't set this field, HP ALLBASE/4GL assumes access is to be via the primary index.

Initially, *INDEXNO is set to 1. Once you set this field, HP ALLBASE/4GL does not reset its value.

*IOSTATDM

Contains the error number returned by the most recent call to an external data manager.

Currently, this communication area field only contains error message numbers from the HP TurboIMAGE/iX data manager. The value in this field is zero if no error occurs. If errors do occur, this field contains a number corresponding to the condition encountered.

The error number is taken from the first part of the HP TurboIMAGE/iX status array, which represents status information about the last HP TurboIMAGE/iX library procedure performed. The meaning of the error numbers for each HP TurboIMAGE/iX library procedure can be found in the *HP TurboIMAGE/iX Database Management System Reference Manual.* Refer to chapter 10 of this manual for a list of the common library procedures initiated by HP ALLBASE/4GL.

The field is five characters long, has a type S edit code, a type I storage code, and is right justified.

*IOSTATUS

Contains the error message number returned by the HP ALLBASE/4GL data manager after each access to a file. The value in this field is zero if no error occurs. If errors do occur, this field contains a number corresponding to the condition encountered.

An error may also be returned by HP ALLBASE/SQL or HP TurboIMAGE/iX. In this case, one or more of the following communication area fields may contain error details: *ERROR for HP ALLBASE/SQL, or *ERROR, *ERRORDM, *IMSTAT, and *IOSTATDM for HP TurboIMAGE/iX.

FINAL TRIM SIZE : 7.0 in x 8.5 in

The type of error and number returned depend on the data file type. Refer to appendix C for a list of these conditions.

The field is six characters long, has a type N edit code, and is right justified.

*LOCKWAIT

Contains a numeric value that determines the time in seconds that HP ALLBASE/4GL waits while attempting to access a KSAM file or HP TurboIMAGE/iX data set that is locked by another MPE/iX process. If HP ALLBASE/4GL cannot access the file within the time determined by the current value in *LOCKWAIT, the data manager returns an error condition. For HP TurboIMAGE/iX data sets, HP ALLBASE/4GL attempts to access the locked file once every second for the number of seconds specified.

If the value in *LOCKWAIT is zero, HP ALLBASE/4GL attempts to access a locked file once every second for five seconds before returning an error condition.

A value of -1 in *LOCKWAIT means that HP ALLBASE/4GL waits indefinitely to access a locked file. A value of less than -1 is invalid.

*LTRACE

A numeric field used for turning on and off trace mode. When set to a non-zero value, causes HP ALLBASE/4GL to activate the trace mode facility. When set to zero, the trace mode facility is turned off. The same rules apply for trace mode asfor using the -*i* option when invoking HP ALLBASE/4GL; for instance, runtime systems and `developr` and `administ` applications cannot be traced. See chapter 13 for more details on trace mode.

EXAMPLE:

```
MOVE "1" *LTRACE
NOTE Trace mode is now on.
MOVE "0" *LTRACE
NOTE Trace mode is now off.
```

FINAL TRIM SIZE : 7.0 in x 8.5 in

*MSGDELAY

A numeric field used for changing the number of seconds HP ALLBASE/4GL will wait after a message is displayed before continuing processing. Any positive integer can be moved into this field to signify the number of seconds to wait. HP ALLBASE/4GL default delay time is three seconds.

Moving a value of zero into this field will cause HP ALLBASE/4GL to display the message and continue processing immediately.

*NEWTIE

Contains number of the next screen field to be processed. It is used when the next screen field to be processed is not necessarily the next field in sequence on the screen. For example, after testing the contents of a field you may want to skip over one or more fields.

Setting this field performs the same function as the logic command TIE.

The field is three characters long, has a type N edit code, and is right justified.

*PAGELINE

Contains the number of the current print line on the page of a report. The field is six characters long, has a type N edit code, and is right justified.

*PAGENO

Contains the current page number of a report. The field is four characters long, has a type N edit code, and is right justified.

*PASS

A general purpose alphanumeric field used for passing arguments and results.

The logic command CHECK places data into *PASS. The CHECK command places the matching position number in a validation table or a file record into *PASS if it finds a match. If no match is found, the CHECK command places zero into *PASS.

The *PASS field takes on the characteristics (length, edit code and justification) of the data stored in it.

The *PASS communication area field is identical to scratch-pad field *P00.

*PROCESS

Contains the current process name. The field is 16 characters long, has a type K edit code, and is left justified.

*RECNO

Contains the record number of the HP TurboIMAGE/iX data set record or fixed length serial data file record just read or written. The value in *RECNO is undefined after HP ALLBASE/4GL accesses other file types.

*REPORT

Contains the current report name. It is blank if HP ALLBASE/4GL is not executing a report. The field is 16 characters long, has a type K edit code, and is left justified.

*ROUTINE

Contains the current external routine name. It is blank if no external routine is called. The field is 16 characters long, has a type X edit code, and is left justified.

*ROWCOUNT

Contains a number indicating the number of rows that HP ALLBASE/SQL processes as the result of a command that changes an HP ALLBASE/SQL table.

The value in this field is the same as the value in the HP SQL communication area (SQLCA) field SQLERRD.

*SCREEN

Contains the current screen name. It is blank if HP ALLBASE/4GL is not displaying a screen. The field is 16 characters long, has a type K edit code, and is left justified.

When HP ALLBASE/4GL displays a window on a screen, the value in *SCREEN becomes the name of the current window.

*TOTALS(n)

A set of 16 fields used primarily in the report generator for holding the totals of the values of numeric fields and for accessing these totals for the current control break level. Each field is 20 characters long with two decimal places, has a type S edit code, and is right justified.

The 16 totalling fields are referenced as *TOTALS(1) to *TOTALS(16).

You can use these fields as numeric work variables when HP ALLBASE/4GL is not executing a report. When HP ALLBASE/4GL is executing a report, the values in these fields may change.

Read-Only Communication Fields

You can use the contents of the following read-only fields for interrogation and display only.

*DATE

Contains the current date. Internally, this is held in the format *YY/MM/DD*. When this field is displayed in a field with a type D or T edit code on a screen or a report, the date is displayed in the format MM/DD/YY or DD/MM/YY as specified by the system administrator.

The field is eight characters long and has a type D edit code.

*DBENAME

Contains the fully qualified external name of the DBECon file for the current application's HP ALLBASE/SQL database.

The field is 128 characters long and has a type K edit code.

*KEYS

Contains the name of the current function key set. The field is 16 characters long and has a type K edit code.

*PREVFLD

Contains the number of the last field successfully committed. It is updated when all validation checks have passed for the current field and the cursor is moved to another field.

It is set to 1 when you first enter a screen. This is the only case where *FIELDNO (the current field number) and *PREVFLD (the previous field number) are equal.

The field is three digits long, has a type N edit code and is right justified.

FINAL TRIM SIZE : 7.0 in x 8.5 in

*SUITE

Contains the name of the current application, or the name of the base application if the current application is a version of a base application. The field is 8 characters long, has a type K edit code and is left justified.

*TIME

Contains the current time of day. The field is eight characters long and holds the time in the format *hh:mm:ss*.

*USER

Contains the current user name. The field is eight characters long, has a type K edit code, and is left justified.

*VERSION

Contains the name of the current version. The field is eight characters long, has a type K edit code, and is left justified.

Switches

HP ALLBASE/4GL provides you with two different types of switches, referred to as user switches and fixed function switches. Both types of switches can be in an *on* or *off* condition. You can test the condition of any switch with the IF and IFLOOP logic commands or in decision tables.

User Switches

HP ALLBASE/4GL has eight developer-controlled software switches. You can set any of these switches to *on* or *off* with the ON and OFF logic commands.

User switches are referenced as 1 to 8 in the logic ON, OFF, IF and IFLOOP commands.

Fixed Function Switches

HP ALLBASE/4GL has a number of fixed function switches that you can use to check the status or condition of some HP ALLBASE/4GL components. The status or condition of each switch may be controlled by HP ALLBASE/4GL. You can use the logic ON command to change the status of the *BYPASS, *ENDFIELD, and *ENDLINE switches. You can also use the logic OFF command to reset the *BYPASS and *ENDLINE switches.

*BYPASS

The report generator uses this switch to indicate that a file record is, or is not selected for sorting. It also indicates that the current report line is not to be printed if it is *on*. You can set this switch *on* or *off* in functions associated with the report.

Refer to chapter 7 for details of the report processing logic.

*ENDFIELD

The screen processing logic uses this switch to determine whether input processing for a field should continue. You can set this switch *on* in a function called from a function key or a prior function on an input field.

Refer to chapter 6 for details of the screen processing logic.

*ENDLINE

The report generator uses this switch to indicate that printing of lines in the current report line group is, or is not to continue. When the switch is *on*, it indicates the end of a report line group. The *on* condition for this switch can also indicate that no more records are to be read from a report link file and no more link print lines are to be printed. You can set this switch *on* or *off* in functions associated with the report.

Refer to chapter 7 for details of the report processing logic.

*ENTERED

The screen processing logic uses this switch to indicate the status of a data entry field on a data screen. The screen processing logic sets this switch *off* before the contents (if any) of the screen buffer have been displayed in the field,

and before the user commits any data for the field. The switch is set *on* when the user initiates a commit action for the field and the data has been accepted without generating an error. You cannot set this switch with the logic ON or OFF commands.

Refer to chapter 6 for details of the screen processing logic.

*MOREREC

The HP ALLBASE/4GL data manager uses this switch when reading records from variable length record serial data files. If HP ALLBASE/4GL reads sufficient characters to fill the file record buffer without encountering the end of the record, it sets *MOREREC *on*. On the next FILE *READ command for the file, HP ALLBASE/4GL reads more characters from the same file record until it encounters the end of the record, or the file buffer is filled.

If HP ALLBASE/4GL encounters the end of the record before the buffer is filled, or the number of characters in the record matches the length of the buffer, *MOREREC is set *off* after the file read.

You cannot set this switch with the logic ON or OFF commands.

Refer to chapter 12 for more information about the FILE command.

*SHOWING

The screen processing logic uses this switch to indicate if the contents of screen fields are being displayed under the control of the logic SHOW command. The switch is set *on* if a SHOW command is being executed. You cannot set this switch with the logic ON or OFF commands.

Refer to chapter 6 for details of the screen processing logic.

FINAL TRIM SIZE : 7.0 in x 8.5 in

5

Dictionary

The HP ALLBASE/4GL dictionary is a central storage area for definitions of a
number of HP ALLBASE/4GL entities and information storage items. Each
definition or storage item is called a dictionary entry, and can be one of the
following types:

- A field specification.

- A validation item.

- A storage item.

- A record layout.

- A definition for an HP ALLBASE/SQL table, an HP TurboIMAGE/iX data
  set, a serial data file, or an indexed KSAM data file.

- An HP ALLBASE/SQL select list definition.

- A message.

- A help screen.

Dictionary Menu Screens

The following subsections describe the screens in the dictionary menu. Each
subsection describes the screen under the following subheadings:

- Screen Image.

- Field Descriptions.

- Function Keys.

- Additional Information.

Dictionary Menu

This menu allows you to access the screens used to define the characteristics of dictionary entries.

Screen Image

```
                          HP ALLBASE/4GL Developer
     Dictionary  Screens  Logic  Reports  Module Builder  Utilities  Generate  Test

      Field Specifications
      Validation Items
      Storage Items
      Record Layouts
      Database Items
      Messages
      Help Screens

      Catalog Display




     Utility   Screens   Logic   Reports    5*  4    System  Activate   Help   Previous
     Menu      Menu      Menu    Menu                Keys    Item              Menu
```
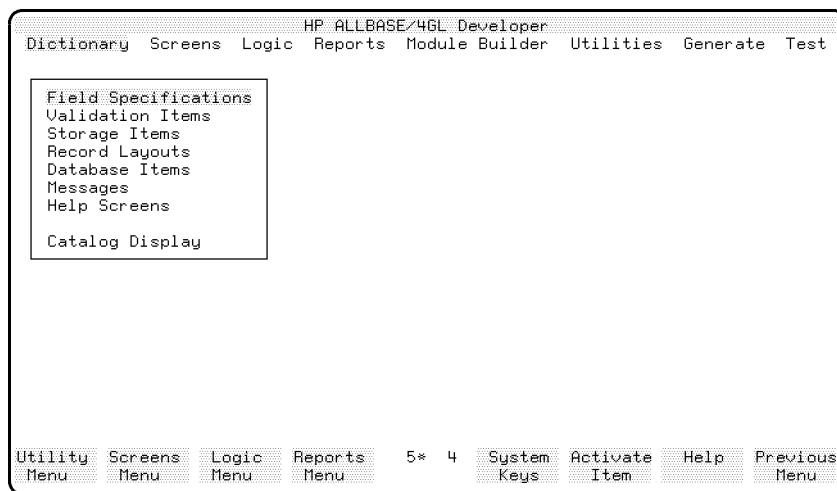
Menu Selections

The dictionary main menu allows you to access the dictionary facilities. Some screens are accessed through submenus. The dictionary facilities and submenus are listed below.

Field Specifications   This selection displays the field specifications screen, allowing you to name a field and specify its characteristics.

Validation Items   This selection displays the validation items menu. Options on this menu allow you to define validation tables and ranges.

Storage Items   This selection displays the storage items menu. Options on this menu allow you to define variables, calculated items, numeric constants, alphanumeric constants, scratch-pad field names, application titles, and work areas.

Record Layouts This selection displays the record layouts menu. Options on this menu allow you to assign a name to a record layout, and specify the details of the record layout.

Database Items This selection displays the database items menu. Options on this menu allow you to define HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, KSAM data files, and serial data files; create HP ALLBASE/SQL tables and KSAM data files; reformat KSAM data files; and delete HP ALLBASE/SQL tables and KSAM data files.

Messages This selection displays the messages screen directly, allowing you to define the message type and enter the message text.

Help Screens This selection displays the help screens screen directly, allowing you to name end-user help screens and enter the help message text.

Catalog Display This selection displays the catalog display screen, allowing you to display the names of HP ALLBASE/4GL items already created. Refer to chapter 13 for details.

Function Keys

(f1)　　　 Utility Menu　 Cancels the current screen and displays the utilities menu.

(f2)　　　 Screens Menu　 Cancels the current screen and displays the screen development menu.

(f3)　　　 Logic Menu　 Cancels the current screen and displays the logic menu.

(f4)　　　 Reports Menu　 Cancels the current screen and displays the reports menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Field Specifications Screen

This screen allows you to specify a field's attributes. A dictionary field specification is a template defining the attributes of a field on a record layout, screen, or report.

This screen also allows you to access the data manager specific field attributes screen, where you can set extra attributes for this field that are relevant to a particular data manager. Currently, you can only set HP TurboIMAGE/iX-specific attributes on this screen.

Screen Image

```
Developer                      Field Specifications              field_specs

   Field Spec. Name    ████████████████                Secured    (Y/N)

   Field Length              Repeated      Times
   Minimum Entry Length

   Edit Code            (X/A/U/K/N/S/Q/D/T)  Decimal Places
   Storage Type         (C/I/L/F/G/P)        Blank When Zero          (Y/N)
   Justification        (L/R/C/N)            Pad Character

   Validation:   Range                       Table

   Help Name

   Description



   Last Modification:       Date                  Time

Records    Ranges    Tables    Data Mgr   4* 26   System    Commit    Help   Previous
Menu                            Attribs.          Keys      Data             Menu
```

Field Descriptions

Field Spec. Name   Enter the name of the field specification (up to 16 characters).

Secured   To protect this specification from unauthorized modification, enter Y after you have completed the required fields on this screen. The default is N.

If you enter Y, this specification can only be modified by a developer who has signed on to the application under the correct development security code.

Field Length Enter the length of the field in characters.

Repeated _____ **Times** Enter a number between 1 and 255. The default value in this field is 1. Enter a number other than 1 if the field occurs more than once and all occurrences are to be referenced using the same dictionary name and a subscript. (Refer to chapter 3 for an explanation of subscript referencing.)

Minimum Entry Length Enter a number equal to the minimum number of characters acceptable for data in this field. The entry in this field defaults to 1.

Edit Code Enter a code defining the type of data in the field. The following table lists the available edit codes and their meanings.

Code	Meaning
X	Any printable character.
A	Alphabetic characters only (A to z, extended characters, or space).
U	Forced uppercase. Same as X edit code, but shifts all lowercase alphabetic characters to uppercase.
K	Alphanumeric characters only (A to z, 0 to 9, extended characters, and underscore).
N	Unsigned number (0 to 9, a period ".", or a comma ",").
S	Signed number. Same as N edit code, but may include + or -.
Q	Question. Y, y, N, or n only. Any characters beyond the first character in the field are ignored.
D	Date field. Must be eight characters long. The date may be either MM/DD/YY or DD/MM/YY (depending on the system-wide date format) where MM, DD, and YY represent the month, day, and year respectively.
T	Date field, defaulting to the current date.

Storage Type Enter a code defining the way in which the data manager will physically store data in data set fields defined by this field specification. You can only make an entry here if the field is numeric (edit code N or S). The value in this field defaults to a character storage type.

The following table lists the storage type codes and their meanings.

Storage Type	Meaning
C	Character (number stored in the same format as it appears on a screen)
F	Four byte, IEEE floating point
G	Eight byte, IEEE floating point (double precision)
I	Two byte 2's complement (integer)
L	Four byte 2's complement (integer)
P	Packed decimal (BCD, or binary coded decimal)

If you use the field specification in a data file or HP ALLBASE/SQL table, and the storage type is not supported by the data manager, HP ALLBASE/4GL will use the most appropriate storage type that is available.

For HP TurboIMAGE/iX, a default HP TurboIMAGE/iX type designator is selected by HP ALLBASE/4GL for each field. You can examine the default or change the type designator on the data manager specific attributes screen.

Justification Code Enter a code defining the way in which the data is justified within the field on entry or display. The following table lists the justification codes and their meanings.

Code	Meaning
L	Data is aligned to the left boundary of the field. Trailing blanks are replaced by the pad character.
R	Data is aligned to the right boundary of the field. Leading blanks are replaced by the pad character.
C	Data is centered within the left and right boundaries of the field.
N	No justification is performed.

Decimal Places Enter a number defining the number of decimal places for data in this field. You only need to make an entry here if the field is numeric (edit code N or S), and has a non-integer storage type (storage type codes C, F, G and P). For numeric fields, the value in this field defaults to 0.

5-6   Dictionary

Blank When Zero   This entry only applies to numeric fields (edit code N or S). If you enter Y, the field is displayed as blank on screens if its value is zero.

Pad Character   Enter a single character to replace all spaces at the beginning and/or end of the field depending on the justification.

Validation Range   If you enter the name of a validation range in this field, the system tests input data from a screen field against the values in the validation range during data entry.

Validation Table   If you enter the name of a validation table, the system checks that input data from a screen field matches one of the entries in the validation table during data entry.

Note

If you enter the names of both a validation range **and** a validation table on this screen, only data passing validation for both the validation table **and** the validation range is acceptable as input for this field.

Help Name   Enter the name of a help screen. This help screen name is automatically written to the screen fields details screen if you use the dictionary field specification to define the characteristics for a screen field. (Refer to chapter 6, Screen Development.) If the end-user requests help while entering data for the field, HP ALLBASE/4GL displays this help screen.

Description   Documentation fields. Enter a description of the field specification for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

Function Keys

[f1]   `Records Menu`   Cancels the current screen and displays the record layout menu.

[f2]   `Ranges`   Cancels the current screen and displays the validation ranges screen.

[f3]   `Tables`   Cancels the current screen and displays the validation tables screen.

f4 `Data Mgr Attribs.` Cancels the current screen and displays the
data manager attributes screen. This is the only way to access the
data manager attributes screen.

Additional Information

Mapping to SQL Data Types

The following table shows how the HP ALLBASE/4GL edit codes map onto
HP ALLBASE/SQL data types:

HP ALLBASE/4GL Edit Code	HP ALLBASE SQL Data Type	Comments
N and S	BCD(p,s)	p is length, s is scale (precision)
D and T	CHAR(8)	
Others	CHAR(length)	

Mapping to HP TurboIMAGE/iX Data Types

For the default mapping of HP ALLBASE/4GL field specification details onto
HP TurboIMAGE/iX data item types, refer to Additional Information under
the Data Mgr Specific Field Attributes Screen.

To see or alter the default settings for a specific field specification that you
have defined, press the `Data Mgr Attribs.` function key from the field
specifications screen. This displays the data manager specific field attributes
screen.

Null Value Support

Reading, detecting, and writing null values in HP ALLBASE/SQL tables are
actions supported in HP ALLBASE/4GL. HP ALLBASE/4GL associates an
indicator variable with each field in a select list or a record layout used with an
HP ALLBASE/SQL table. The status of this variable indicates whether the
field contains a null value.

5-8   Dictionary

The *NULL argument for the logic IF command allows you to test the status of the indicator variable associated with a field.

The IF test has the format:

    IF *field_reference* *NULL **THEN** *command*

Null status is transferred from one file record field to another if the value of one field is transferred to the other with the MOVE command. Moving a value from any field type other than a field on a select list or a record layout associated with an HP ALLBASE/SQL table sets the indicator variable to the non-null status. Null status is not preserved if you use the MOVE command to move a file record buffer to another buffer.

You can force the indicator variable for a field to the null status with the command:

    MOVE *NULL *field_reference*

The following table summarizes the way HP ALLBASE/4GL reads and writes null values on input and output.

Field Type	**Action**
Numeric fields	
On output	Null values are written if the indicator variable is set to null status. Otherwise the actual value is written.
On input	Null values are always converted to zero or blanks depending on the *Blank When Zero* value for the destination field.
Date Fields	
On output	Null values are written if the indicator variable is set to the null status. Otherwise the actual value of the field is written.
On input	A zero length string is returned to the date field.
Character fields	
On output	Null values are written if the indicator variable is set to null status. Otherwise the actual value in the field is written. This value includes any pad characters in the field, and is the size specified by the length of the dictionary field specification.
On input	The field is filled with the pad character.

Data Manager Specific Field Attributes Screen

This screen allows you to specify data manager specific attributes for a field specification. Currently, only HP TurboIMAGE/iX-specific attributes can be set. To display this screen, press the ░Data Mgr Attribs.░ function key from the field specification screen.

Screen Image

```
╭─────────────────────────────────────────────────────────────────────╮
│ Developer        Data Mgr. Specific Field Attributes      field_specs_dm │
│                                                                        │
│              Field Spec. Name   product_no                             │
│                                                                        │
│              Data Manager       T (T)          Action  ▊ (A/C/D)       │
│                                                                        │
│              Data Item Name     PRODUCT-NO                             │
│                                                                        │
│              Sub-item Count     1    (= Field Spec. Repeated)          │
│                                                                        │
│              Type Designator    U    (E/I/J/K/P/R/U/X/Z)               │
│                                                                        │
│              Sub-item Length       6                                   │
│                                                                        │
│                                                                        │
│                                                                        │
│                                                                        │
│                                                                        │
│ Records   Ranges   Tables   Base Fld   6* 57   System   Commit   Help   Previous │
│ Menu                        Specs.              Keys     Data            Menu     │
╰─────────────────────────────────────────────────────────────────────╯
```

Field Descriptions

Field Spec. Name   This is a display only field that contains the name of the field specification displayed on the field specification screen when you accessed this screen.

Data Manager   Enter the code to specify the data manager for which you wish to add, change or delete attributes. Currently, HP ALLBASE/4GL only supports the HP TurboIMAGE/iX data manager, so accept the default code **T**.

When you commit this field, the appropriate data manager window is displayed. Any attributes previously defined for this data manager are also

FINAL TRIM SIZE : 7.0 in x 8.5 in

displayed. If no attributes have been defined, the default attributes for the field specification are displayed.

Currently, only the HP TurboIMAGE/iX window is available.

Action   Enter the code of the action to be performed on the specified field. It can be one of the following:

Code	Action
A	Add new data manager attributes for the field.
C	Change field attributes for the data manager specified.
D	Delete field attributes for the data manager specified.

This field defaults to `A` if no field attributes have been set for the specified data manager and the default settings are displayed, or `C` if field attributes do exist for the data manager.

Data Item Name   Enter the name of the field specification as it will be known to HP TurboIMAGE/iX. This name is used as an HP TurboIMAGE/iX data item name in references to HP TurboIMAGE/iX databases. The name may contain any characters legal for HP ALLBASE/4GL names, except the underscore character (_). The following characters are also valid:

$$+, -, *, /, ?, ', \#, \%, \&, \text{ and } @.$$

This field defaults to the name in the Field Spec. Name field. Any underscore characters in the base field specification name are converted to hyphens (-).

HP TurboIMAGE/iX is not case sensitive, so HP ALLBASE/4GL automatically changes lowercase characters in this field to uppercase.

Sub-item Count   This is a display only field containing the number of times the field specification is defined. It corresponds to the number specified in the Repeated field on the field specification screen.

Type Designator   Enter a code defining the HP TurboIMAGE/iX data item's storage type. These item types map directly to HP TurboIMAGE/iX data item type designators. The following table lists the available item types and their meanings. For further information, refer to the *HP TurboIMAGE/iX Database Management System Reference Manual*.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Code	Meaning
E	IEEE real number
I	2's complement (integer)
J	Same as I except with COBOL range restrictions
K	Absolute binary quantity (unsigned integer)
P	Packed decimal (BCD, or binary coded decimal)
R	"Classic" HP3000 real number
U	Forced uppercase; all lowercase alphabetic characters are shifted to uppercase
X	Any printable ASCII character
Z	A zoned decimal format number

The default value for this field is dependent upon the edit code and the storage type specified on the field specifications screen. The following table displays the default values.

Edit Code	Storage Type	Type Designator
X, A, or K	C	X
U	C	U
Q	C	X
D or T	C	X
N or S	C	X
N or S	I or L	I
N or S	F or G	K
N or S	P	P

Sub-item Length

Enter a number between 1 and 255. This number represents the length of each sub-item, in one of the following measurement units.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Type Designator	Unit of Measurement	Valid Sub-item Length
I, J, K	Halfwords (16 bits)	1, 2, or 4
E, R	Halfwords (16 bits)	2 or 4
U, X	Bytes (8 bits)	Any number
Z	Bytes (8 bits)	≤ 26
P	Nibbles (4 bits)	≤ 26 and a multiple of 2

The length of the data item as a whole equals the sub-item count multiplied by the sub-item length, and must be an integral number of halfwords.

For data items with type designators U, X, or Z, the data item length must be divisible by 2. That is, the product of sub-item count and sub-item length must be divisible by 2.

For data items with type designator P, the data item length must be divisible by 4. That is, the product of sub-item count and sub-item length must be divisible by 4.

In most situations, the default for this field should be satisfactory. If you enter a value in this field that does not support HP TurboIMAGE/iX field length rules, HP ALLBASE/4GL will display an error message when you press `Commit Data`.

Function Keys

(f1)   `Records Menu`   Cancels the current screen and displays the record layout menu.

(f2)   `Ranges`   Cancels the current screen and displays the validation ranges screen.

(f3)   `Tables`   Cancels the current screen and displays the validation tables screen.

(f4)   `Base Fld Specs.`   Cancels the current screen and displays the field specification screen.

Additional Information

This screen is optional. If you do not complete this screen for a field specification, and the field specification is used in a HP TurboIMAGE/iX data set, the definition for the field specification uses the default settings displayed on the screen.

These default settings are also used if you do not press the `Commit Data` function key to confirm your data entry before exiting the screen.

The following table shows the default mapping of HP ALLBASE/4GL field specification details onto HP TurboIMAGE/iX data types. It also shows the default values for the HP TurboIMAGE/iX sub-item length of each type of data type.

HP ALLBASE/4GL		HP TurboIMAGE/iX	
Edit Code	Storage Type	Type Designator	Sub-item Length
X, A, or K	C	X	Field Length + pad
U	C	U	Field Length + pad
Q	C	X	2
D or T	C	X	8
N or S	C	X	Field Length + pad
N or S	F	K	2
N or S	G	K	4
N or S	I	I	1
N or S	L	I	2
N or S	P	P	Field Length + pad

In the table above, *pad* is the smallest integer that can be used so that the data item length is an integral number of halfwords.

The data item name field defaults to the field specification name (in uppercase, with underscore characters converted to hyphens).

Validation Items Menu

This menu allows you to display the validation ranges screen, the validation
tables screen, or the Catalog Display screen.

Screen Image

```
                        HP ALLBASE/4GL Developer
     Dictionary   Screens   Logic   Reports   Module Builder   Utilities   Generate   Test

       Field Specifications
       Validation Items
       Sto
       Rec    Ranges
       Dat    Tables
       Mes
       Hel    Catalog Display

       Catalog Display




     Field                Storage   Record      8* 10    System   Activate   Help    Previous
     Specs.               Menu      Layouts              Keys     Item               Menu
```
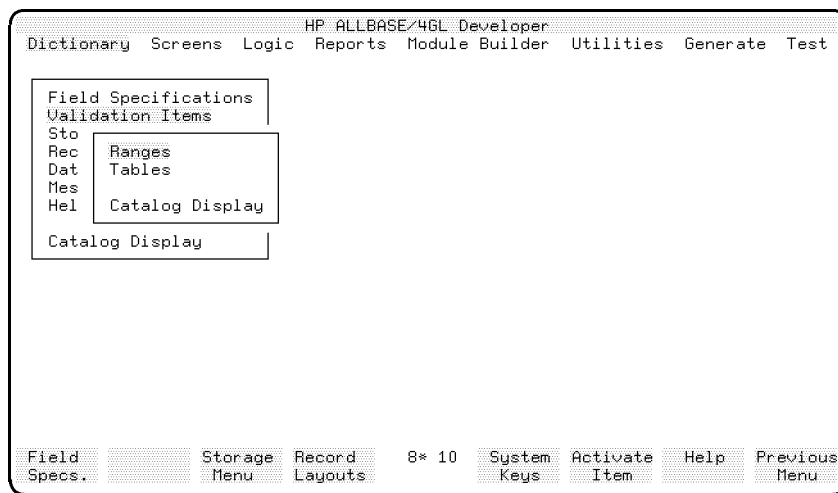
Menu Selections

Ranges   This selection displays the validation ranges screen.

Tables   This selection displays the validation tables screen.

Catalog Display   This selection displays the catalog display screen, allowing
you to display the names of the existing application components. Refer to
chapter 13 for details.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Function Keys

f1   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

f3   `Storage Menu`   Cancels the current screen and displays the storage definitions menu.

f4   `Record Layouts`   Cancels the current screen and displays the record layout menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Validation Ranges Screen

This screen allows you to define a range of values for use in validating input data or selecting records for reporting.

Screen Image

```
┌──────────────────────────────────────────────────────────────────────┐
│ Developer                     Validation Ranges                 ranges │
│                                                                        │
│                                                                        │
│             Range Name              ████████████   Secured ░  (Y/N)    │
│                                                                        │
│             Edit Code               ░  (X/A/U/K/N/S/Q/D/T)             │
│                                                                        │
│             Value  - From           ░░░░░░░░░░░░░░░░                    │
│                    - To             ░░░░░░░░░░░░░░░░                    │
│                                                                        │
│             Number of Decimal Places ░                                 │
│                                                                        │
│             Message Name            ░░░░░░░░░░░░░░░░                    │
│                                                                        │
│                                                                        │
│                                                                        │
│                                                                        │
│                                                                        │
│                                                                        │
│ Field            Tables Messages   5* 39   System  Commit  Help Previous│
│ Specs.                                     Keys    Data            Menu │
└──────────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Range Name   Enter the name of the validation range (up to 16 characters).

Secured   If you wish to protect this validation range from unauthorized modification, enter **Y** after you have completed the required fields on this screen. The entry in this field defaults to N.

If you enter **Y**, this validation range can only be modified by a developer who has signed on to the application under the correct development security code.

Edit Code   Enter a code defining the type of data in the range. The edit code defines the method by which HP ALLBASE/4GL checks the range. Refer to Additional Information.

The following table lists the available edit codes and their meanings.

Code	Meaning
X	Any printable character.
A	Alphabetic characters only (A to z, extended characters, or space).
U	Forced uppercase. Same as X edit code, but shifts all lowercase alphabetic characters to uppercase.
K	Alphanumeric characters only (A to z, 0 to 9, extended characters, and underscore).
N	Unsigned number (0 to 9, a period ".", or a comma ",").
S	Signed number. Same as N edit code, but may include + or -.
Q	Question. Y, y, N, or n only. Any characters beyond the first character in the field are ignored.
D	Date field. Must be eight characters long. The date may be either MM/DD/YY or DD/MM/YY (depending on the system-wide date format) where MM, DD, and YY represent the month, day, and year respectively.
T	Date field, defaulting to the current date.

Value - From   Enter the starting value of the range.

Value - To   Enter the last value of the range.

Number of Decimal Places   Enter the number of decimal places allowed in the number if the range is numeric (edit code N or S).

Message Name   Enter the name of a message to be displayed if data input from the screen falls outside the specified range of values. The message type (refer to Messages Screen) determines how the system reacts if the user enters incorrect data.

If you don't enter a message name in this field, the screen processing logic displays a system defined **ERROR** type message if the user enters incorrect data. Refer to chapter 6 for details of the screen processing logic.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Function Keys

(f1)   `Field Specs.` Cancels the current screen and displays the fields specification screen.

(f3)   `Tables` Cancels the current screen and displays the validation tables screen.

(f4)   `Messages` Cancels the current screen and displays the messages screen.

Additional Information

The edit code for the range determines the method of checking whether a value falls within the range.

If the validation range is numeric (that is, the edit code is either N or S), HP ALLBASE/4GL performs the range check by following strict numeric precedence.

If the validation range is not numeric, HP ALLBASE/4GL performs the range check on a character by character basis following the standard HP ALLBASE/4GL collating sequence.

The order of precedence allocated to characters within a non-numeric range is as follows:

Blanks	
Numeric digits	0 to 9
Alphabetic characters	A to z
Other characters	(punctuation characters)

The HP ALLBASE/4GL collating sequence is listed in appendix A.

Validation Tables Screen

This screen allows you to define a table of values for validating input data or selecting records for reporting.

Screen Image


```
Developer                    Validation Tables                        tables

                   Table Name    ████████████████        Secured     (Y/N)
                   Message Name  ░░░░░░░░░░░░░░░░
            Table Values
     (1)                  (18) ░░░░░░░░░░░░     (35) ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░
                               ░░░░░░░░░░░░          ░░░░░░░░░░░░


     (17)                 (34) ░░░░░░░░░░░░     (51) ░░░░░░░░░░░░

    Field      Ranges          Messages   3* 32  System   Commit    Help    Previous
    Specs.                                        Keys     Data                Menu
```

Field Descriptions

Table Name   Enter the name of the validation table (up to 16 characters).

Secured   To protect this table from unauthorized modification, enter Y after you have completed all the required fields on this screen. The entry in this field defaults to N.

If you enter Y, the table can only be modified by a developer who has signed on to the application under the correct development security code.

Message Name   Enter the name of a message to be displayed if data input from the screen does not match one of the entries in the table. The message type (refer to Messages Screen on page 5-66) determines how the system reacts if the user enters incorrect data.

If you don't enter a message name in this field, the screen processing logic displays a system defined ERROR type message if the user enters incorrect data. Refer to chapter 6 for details of the screen processing logic.

Table Values   Enter a list of up to 51 literals, each of which may consist of between 1 and 16 characters. You can use any printable characters. Trailing spaces are ignored. Don't use quotes (") unless they are part of the literal.

Function Keys

[f1]   `Field Specs.`   Cancels the current screen and displays the field specifications data screen.

[f2]   `Ranges`   Cancels the current screen and displays the validation ranges screen.

[f4]   `Messages`   Cancels the current screen and displays the messages screen.

Additional Information

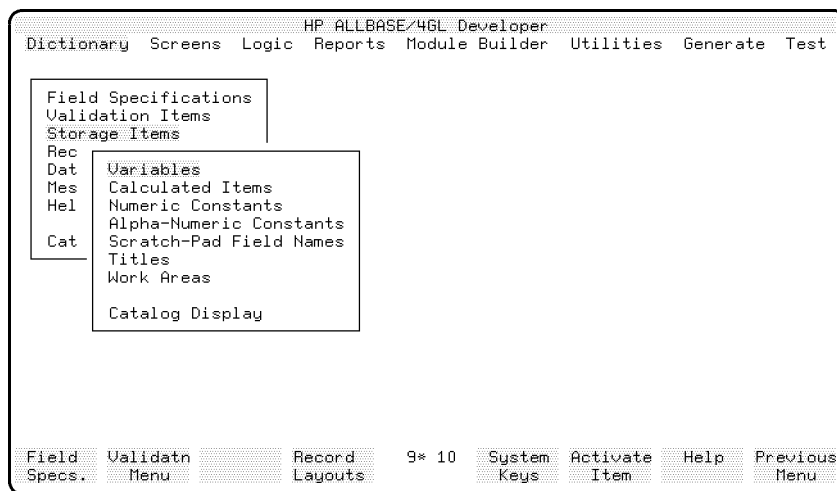When HP ALLBASE/4GL uses a validation table to check data from a screen or for reporting, it checks that the value of the data exactly matches one of the entries in the table. Trailing spaces are ignored.

HP ALLBASE/4GL stores numeric data (edit code N or S) in right justified format. Therefore, HP ALLBASE/4GL will not record a match if you compare a right justified number against a left justified validation table entry.

Storage Items Menu

The storage items menu allows you to define the attributes of the storage items in the dictionary.

Screen Image

```
                         HP ALLBASE/4GL Developer
        Dictionary   Screens   Logic   Reports   Module Builder   Utilities   Generate   Test

        ┌─────────────────────┐
        │ Field Specifications│
        │ Validation Items    │
        │ Storage Items       │
        │ Rec ┌───────────────────────────┐
        │ Dat │ Variables                 │
        │ Mes │ Calculated Items          │
        │ Hel │ Numeric Constants         │
        │     │ Alpha-Numeric Constants   │
        │ Cat │ Scratch-Pad Field Names   │
        │     │ Titles                    │
        │     │ Work Areas                │
        │     │                           │
        │     │ Catalog Display           │
        │     └───────────────────────────┘

        ┌───────┬────────┬──────────┬───────┬────────┬──────────┬──────┬──────────┐
        │ Field │ Validatn│         │ Record│  9* 10 │ System  │Activate│ Help │ Previous │
        │ Specs.│ Menu   │         │ Layouts│        │  Keys   │ Item  │      │ Menu     │
        └───────┴────────┴──────────┴───────┴────────┴──────────┴──────┴──────────┘
```

Menu Selections

Variables   This selection displays the variables screen, allowing you to define variable names and to declare their attributes.

Calculated Items   This selection displays the calculated items screen, allowing you to define calculated items. HP ALLBASE/4GL calculates the value of a calculated item each time the item is referenced.

Numeric Constants   This selection displays the numeric constants screen, allowing you to assign names and values to numeric constants.

Alpha-Numeric Constants   This selection displays the alphanumeric constants screen, allowing you to assign names and values to alphanumeric constants (alphanumeric literals).

Scratch-Pad Field Names This selection displays the scratch-pad field names screen, allowing you to assign names to scratch-pad fields.

Titles This selection displays the application titles screen, allowing you to define titles for display on application screens or printing on reports.

Work Areas This selection displays the work areas screen, allowing you to define a work area.

Catalog Display This selection displays the catalog display screen, allowing you to display the names of existing application components.

Function Keys

(f1)   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

(f2)   `Validatn Menu` Cancels the current screen and displays the validation items menu.

(f4)   `Record Layouts` Cancels the current screen and displays the record layout menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Variables Screen

This screen allows you to name a variable and declare its attributes.

Screen Image

```
 Developer                      Variables                      variables


                    Variable Name        ██████████████

                    Length                   ▒

                    Edit Code                ▒  (X/A/U/K/N/S/Q/D/T)

                    Number of Decimal Places ▒

                    Pad Character            ▒





  Work    Calc.   Numeric  AlphaNum   5* 48   System  Commit   Help   Previous
  Areas   Items   Constant Constant           Keys    Data            Menu
```

Field Descriptions

Variable Name   Enter the name of the variable (up to 16 characters).

Length   Enter the maximum number of characters that the variable may contain.

Edit Code   Enter a code defining the type of data that the variable will contain.

The following table lists the available edit codes and their meanings.

Code	Meaning
X	Any printable character.
A	Alphabetic characters only (A to z, extended characters, or space).
U	Forced uppercase. Same as X edit code, but shifts all lowercase alphabetic characters to uppercase.
K	Alphanumeric characters only (A to z, 0 to 9, extended characters, and underscore).
N	Unsigned number (0 to 9, a period ".", or a comma ",").
S	Signed number. Same as N edit code, but may include + or -.
Q	Question. Y, y, N, or n only. Any characters beyond the first character in the field are ignored.
D	Date field. Must be eight characters long. The date may be either MM/DD/YY or DD/MM/YY (depending on the system-wide date format) where MM, DD, and YY represent the month, day, and year respectively.
T	Date field, defaulting to the current date.

Number of Decimal Places   Enter the maximum number of decimal places for the contents of the variable if the variable is numeric (edit code is N or S). This field defines the accuracy of numeric data.

Pad Character   Enter a single character to be used to initialize the variable. HP ALLBASE/4GL sets every character of the variable to the value of the pad character before the variable value is assigned.

After the variable value has been assigned, the pad character replaces leading spaces in numeric variables (edit code N or S) and replaces trailing spaces in other variable types.

Function Keys

`f1` `Work Areas` Cancels the current screen and displays the work areas
screen.

`f2` `Calc. Items` Cancels the current screen and displays the
calculated items screen.

`f3` `Numeric Constant` Cancels the current screen and displays the
numeric constants screen.

`f4` `AlphaNum Constant` Cancels the current screen and then displays
the alphanumeric constants screen.

Additional Information

HP ALLBASE/4GL allocates memory space for all variables when the
application starts. As a result, using variables for one-off storage of data (for
example, counting records in one report only) results in inefficient use of the
system memory. Use scratch-pad fields for storing transient data.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Calculated Items Screen

This screen allows you to define a calculated item. A calculated item is a variable that contains the result of an algebraic expression or a function that is evaluated whenever the item is referenced. Calculated items exist in two forms, the simple form and the extended form. HP ALLBASE/4GL implements both forms of calculated item by executing a function when the item is referenced. The function assigns a value to the calculated item.

In the simple form, you specify the expression to be evaluated as a single algebraic equation using the CALC logic command. In the extended form, rather than specify an algebraic equation, you use a function with the same name as the calculated item. This function may perform any processing that is normally permitted within a function. The value of the calculated item must be set explicitly within the function. To specify a calculated item in this form, you must first define the name and attributes of the calculated item using this screen, and then use the function details screen in the logic menu to define the details of the function. Chapter 11 describes HP ALLBASE/4GL functions.

Screen Image

```
Developer                        Calculated Items           calculated_items


            Item Name                  ████████████████

            Length
            Edit Code                   (X/A/U/K/N/S/Q/D/T)
            Number of Decimal Places
            Pad Character




            Description




            Last modification:     Date                  Time

Variable Function Numeric  AlphaNum    5* 36   System   Commit    Help   Previous
         Details  Constant Constant            Keys     Data             Menu
```

Field Descriptions

Item Name   Enter the name of the calculated item (up to 16 characters). When you enter a new calculated item name, HP ALLBASE/4GL automatically creates a function with the same name.

HP ALLBASE/4GL displays an error message if you enter a name which is already in use for a function.

Length   Enter the maximum length in characters of the value that can be assigned to the calculated item by the assigning function.

Edit Code   Enter a code defining the type of data that the calculated item may contain. The following table lists the available edit codes and their meanings.

Code	Meaning
X	Any printable character.
A	Alphabetic characters only (A to z, extended characters, or space).
U	Forced uppercase. Same as X edit code, but shifts all lowercase alphabetic characters to uppercase.
K	Alphanumeric characters only (A to z, 0 to 9, extended characters, and underscore).
N	Unsigned number (0 to 9, a period ".", or a comma ",").
S	Signed number. Same as N edit code, but may include + or -.
Q	Question. Y, y, N, or n only. Any characters beyond the first character in the field are ignored.
D	Date field. Must be eight characters long. The date may be either MM/DD/YY or DD/MM/YY (depending on the system-wide date format) where MM, DD, and YY represent the month, day, and year respectively.
T	Date field, defaulting to the current date.

Number of Decimal Places   Enter the maximum number of decimal places for the contents of the calculated item if the item is numeric (edit code N or S).

Pad Character Enter a single character used to initialize the calculated item. HP ALLBASE/4GL sets every character of the item to the value of the pad character before calling the assigning function.

CALC The value of a calculated item is assigned by a function HP ALLBASE/4GL creates for this purpose. The function may be either a simple function or an extended function.

A simple function is one that contains only a single CALC command. To create a simple function, enter the details of the CALC command in this area of the screen. You don't need to use the command name CALC as part of this command expression. HP ALLBASE/4GL assumes the entry in this field is a CALC command, and displays the word CALC in a display only field the left of the command field. Refer to chapter 12 for details of the CALC command.

An extended function either has more than one step, or consists of a single step that is not a CALC command. To define a calculated item using an extended function, leave the CALC field blank. You can then use the function detail screen in the logic menu to define the extended function. Note that HP ALLBASE/4GL automatically creates a function header using the name of the calculated item as the function name, and the description fields from this screen. Refer to chapter 11 for details of function header and function details screens.

The assigning function or a function called from the assigning function may use the calculated item. Under these circumstances, the value in the calculated item is its existing value. The calculated item is not re-evaluated if it is used in the assigning function.

Description Documentation fields. Enter a description of the calculated item for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

HP ALLBASE/4GL uses the description fields and the calculated item name to create the function header for the assigning function.

Function Keys

f1   `Variable`   Cancels the current screen and displays the variables screen.

f2   `Function Details`   Cancels the current screen and displays the function details screen.

f3   `Numeric Constant`   Cancels the current screen and displays the numeric constants screen.

f4   `AlphaNum Constant`   Cancels the current screen and then displays the alphanumeric constants screen.

Additional Information

You cannot create a calculated item with the same name as a function that already exists.

Values can only be assigned to a calculated item from within a function of the same name as the calculated item.

Generating Calculated Items

When you press the `Commit Data` function key, HP ALLBASE/4GL generates the assigning function automatically. HP ALLBASE/4GL displays an error message if the function generate fails.

Using Calculated Items

In general, you can use calculated items in the same way as you would use HP ALLBASE/4GL variables, provided you don't move data into the calculated item. If you do try to move data into a calculated item, the application aborts.

Only the assigning function can move a value into a calculated item.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Numeric Constants Screen

This screen allows you to define a numeric constant.

Screen Image

```
┌─────────────────────────────────────────────────────────────────┐
│ Developer                   Numeric Constants            numerics │
│                                                                   │
│                                                                   │
│                  Numeric Constant Name    ██████████████          │
│                  Value                    ░░░░░░░░░░░░░░░░░░       │
│                  Edit Code                     (N/S)              │
│                  Number of Decimal Places                        │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│ Variable  Calc.    Work   AlphaNum   5* 45   System  Commit  Help  Previous │
│           Item    Areas   Constant           Keys    Data          Menu    │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Numeric Constant Name   Enter the name of the numeric constant (up to 16 characters).

Value   Enter the value of the constant. The constant can have up to 14 significant digits and up to 9 decimal places.

Edit Code   Display-only field. For an existing numeric constant, the display in this field is N for an unsigned constant and S for a signed constant.

Number of Decimal Places   Display-only field. For an existing numeric constant, the display in this field indicates the number of decimal places used by the constant.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Function Keys

[f1]   `Variable` Cancels the current screen and displays the variables screen.

[f2]   `Calc. Item` Cancels the current screen and displays the calculated items screen.

[f3]   `Work Areas` Cancels the current screen and displays the work areas screen.

[f4]   `AlphaNum Constant` Cancels the current screen and then displays the alphanumeric constants screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Alphanumeric Constants Screen

This screen allows you to define a constant whose value is an alphanumeric literal.

Screen Image

```
 Developer                    Alpha-Numeric Constants              constants


        Constant Name               ███████████████


        Value

        _____








 Variable   Calc.    Numeric   Titles    5* 32   System    Commit    Help   Previous
            Items    Constant                     Keys      Data             Menu
```

Field Descriptions

Constant Name   Enter the name of the alphanumeric constant (up to 16 characters).

Value   Enter a literal of up to 70 characters in length. Don't use quotes (") unless they are part of the literal.

Function Keys

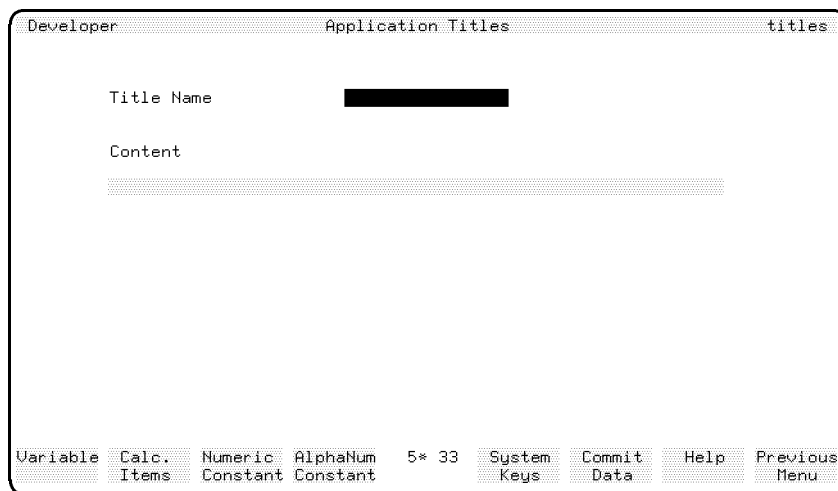(f1)   `Variable` Cancels the current screen and displays the variables screen.

(f2)   `Calc. Item` Cancels the current screen and displays the calculated items screen.

`[f3]` **`Numeric Constant`** Cancels the current screen and displays the numeric constants screen.

`[f4]` **`Titles`** Cancels the current screen and displays the application titles screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in
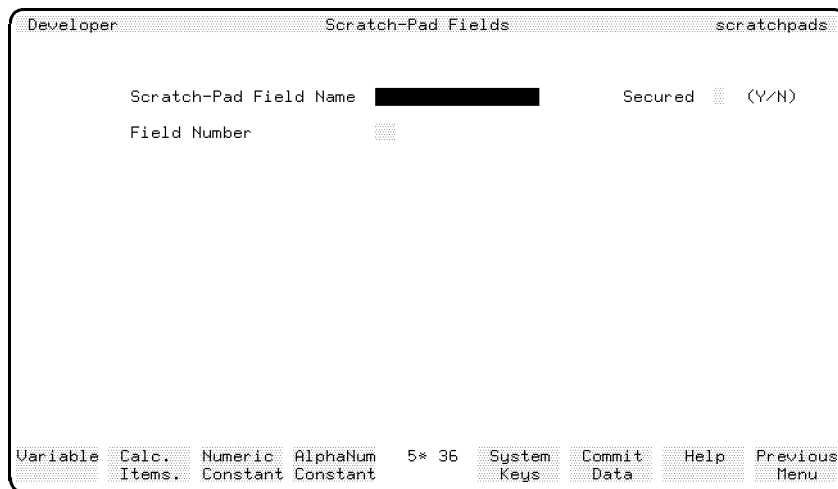
Scratch-Pad Fields Screen

This screen allows you to assign a name to a scratch-pad field.

The scratch-pad is a temporary storage area consisting of up to 100 scratch-pad fields that you can use for storing transient data. Each field takes on the attributes (length, edit code, and justification code) of the data stored within it.

HP ALLBASE/4GL allocates memory for scratch-pad fields dynamically, and consequently, using the scratch-pad for temporary storage of data can be more efficient than using variables.

Screen Image

```
┌─────────────────────────────────────────────────────────────────┐
│ Developer                  Scratch-Pad Fields          scratchpads│
│                                                                   │
│                                                                   │
│           Scratch-Pad Field Name  ███████████      Secured   (Y/N)│
│           Field Number                                            │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│                                                                   │
│ Variable  Calc.    Numeric  AlphaNum   5* 36  System  Commit  Help  Previous│
│           Items.   Constant Constant          Keys    Data          Menu   │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Scratch-Pad Field Name   Enter the name of the scratch pad field (up to 16 characters). One scratch-pad field, as identified by the field number, can have more than one name.

Secured   If you wish to protect the name for this scratch-pad name declaration from unauthorized modification, enter Y after you have completed the required fields on this screen. Otherwise enter N.

If you enter Y, the scratch-pad field name can only be modified by a developer who has signed on to the application under the correct development security code.

Field Number Enter the number of the named field (in the range 00 to 99). Note that scratch-pad field number 00 is the same as the communication area field *PASS.

Function Keys

[f1]   `Variable`   Cancels the current screen and displays the variables screen.

[f2]   `Calc. Items`   Cancels the current screen and displays the calculated items screen.

[f3]   `Numeric Constant`   Cancels the current screen and displays the numeric constants screen.

[f4]   `AlphaNum Constant`   Cancels the current screen and then displays the alphanumeric constants screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Application Titles Screen

This screen allows you to create and maintain application titles.

An application title is a literal string that can be displayed on any screen within the application or printed on reports. Any change made to an existing application title is immediately reflected through the application or related versions that use the title. You can reference application titles with the screen painter.

Screen Image

```
Developer                    Application Titles                        titles

              Title Name              ██████████████████

              Content

              ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░




Variable  Calc.    Numeric  AlphaNum   5* 33   System   Commit   Help   Previous
          Items    Constant Constant            Keys     Data            Menu
```

Field Descriptions

Title Name   Enter the name of the application title (up to 16 characters).

Content   Enter the actual information or message to be displayed.

Function Keys

[f1]   `Variable`   Cancels the current screen and displays the variables screen.

[f2]   `Calc. Items`   Cancels the current screen and displays the calculated items screen.

[f3]   `Numeric Constant`   Cancels the current screen and displays the numeric constants screen.

[f4]   `AlphaNum Constant`   Cancels the current screen and then displays the alphanumeric constants screen.

Additional Information

You can only reference an application title directly from the screen painter.

The report painter, and some HP ALLBASE/4GL logic commands allow you to reference master titles. When HP ALLBASE/4GL encounters a reference to a master title, the system first searches for an **application** title with the same name. If the application title does not exist, the system uses the appropriate master title.

Refer to the *HP ALLBASE/4GL Developer Administration Manual* for further information about defining master titles.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Work Areas Screen

This screen allows you to name a work area and specify the name of the record layout or select list used to define the work area.

Screen Image

```
Developer                        Work Areas                         work_areas


                     Work Area Name        ████████████████
                     Record Layout Name    ░░░░░░░░░░░░░░░░








   Field    Record   Record              5* 44   System  Commit   Help  Previous
   Specs.   Details  Header                       Keys    Data           Menu
```

Field Descriptions

Work Area Name   Enter the name for the work area (up to 16 characters).

Record Layout Name   Enter the name of the record layout or select list used to define the structure of the work area. The record layout or select list must exist before you can define the work area. The record layout or select list does not need to be generated before you define the work area.

Function Keys

f1   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

f2   `Record Details`   Cancels the current screen and displays the record layout details screen.

f3   `Record Header`   Cancels the current screen and displays the record layout header screen.

Additional Information

A work area is a data buffer that is similar to a file record buffer, except that a work area is not associated with a data file. A work area consists of a number of fields in a defined sequence. The structure of a work area is defined by the record layout or select list specified on this screen. The dictionary field specifications used to define the fields on the record layout determine the attributes of the fields in the work area.

To define a work area, you must first define a record layout or select list that contains the dictionary field specifications you require in the work area. You can then use this screen to assign a name to the work area, and specify the record layout or select list for the work area. You must generate the record layout or select list before the application can use the work area.

HP ALLBASE/4GL automatically clears all work areas when the application starts, and at the start of every process.

You can reference a field on a work area as *W-field_name.work_area*. The term *field_name* is the dictionary field specification name, and *work_area* is the work area name.

You can also reference the contents of a work area buffer as *B-work_area* where *work_area* is the work area name. This form of referencing is only valid in the MOVE command and the EXTERNAL command. You cannot use substrings with this form of work area referencing.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Record Layout Menu

The screens in the record layout menu allow you to name record layouts and define the structure of record layouts.

Screen Image

```
                          HP ALLBASE/4GL Developer
        Dictionary  Screens  Logic  Reports  Module Builder  Utilities  Generate  Test

          Field Specifications
          Validation Items
          Storage Items
          Record Layouts
          Dat
          Mes   Details
          Hel   Header

          Cat   Catalog Display



        Field    SQL Sel.   File     Work      10* 10   System   Activate  Help    Previous
        Specs.   Details    Defn.    Areas              Keys     Item              Menu
```
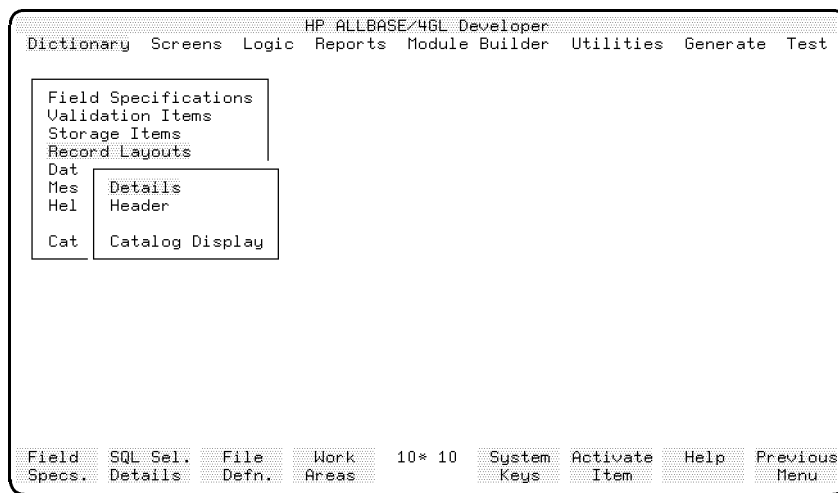
Menu Selections

Details   This selection displays the record layout detail screen, allowing you to specify the structure of a record layout.

Header   This selection displays the record layout header screen, allowing you to assign a name to a record layout and enter a description for documentation purposes.

Catalog Display   This selection displays the catalog display screen, allowing you to display the names of existing application components.

Function Keys

`f1`   `Field Specs.` Cancels the current screen and displays the field specifications screen.

`f2`   `SQL Sel. Details` Cancels the current screen and displays the SQL select list details screen.

`f3`   `File Defn.` Cancels the current screen and displays the file/SQL table definition screen.

`f4`   `Work Areas` Cancels the current screen and displays the work areas screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Record Layout Details Screen

This screen allows you to name the fields in a record layout and define their position in the record layout. The screen also allows you to define the key fields in a record layout.

Screen Image

```
 Developer                    Record Layout Details              record_details

                         Record ████████████████        Secured

  Field Number   Action (A/C/I/D)  Field Spec. Name   Key Number   Duplicate Keys
                                                                       (Y/N)
 ░░░░░          ░░░               ░░░░░░░░░░░          ░░░           ░░








  Field      Record      File      Generate   4* 34   System    Commit    Help    Previous
  Specs.     Header      Defn.     Record              Keys      Data              Menu
```

Field Descriptions

Record   Enter the name of a record that you have defined on the record layout header screen.

Secured   Display-only field indicating the status of the secured field on the record layout header screen.

Field Number   For new fields, HP ALLBASE/4GL supplies this number automatically. Otherwise, enter the number of an existing field.

Action   Enter the code of the action to be performed on the specified field number.

It can be one of the following:

Code	Action
A	Add new field.
C	Change field name or key number.
I	Insert a field before this field.
D	Delete field from layout.

Field Spec. Name   Enter the name of a dictionary field specification. This field must already be defined.

If you intend to use the record layout to define the columns of an HP ALLBASE/SQL table, the field specification cannot allow multiple occurrences of the field. HP ALLBASE/4GL uses the field specification names to define the column names for the HP ALLBASE/SQL table. HP ALLBASE/SQL converts all column names to uppercase, and does not allow extended (eight-bit) characters in column names.

Key Number   This field is optional.

If the field is to be used as a key for reading the file, you must specify the key number. Key number 1 is the primary key.

If you use the record layout to define an HP ALLBASE/SQL table, HP ALLBASE/4GL uses the key fields to define the SQL indexes for the table. The index created for key field number 1 is created as a clustering index.

The KSAM data file creation facility uses the key numbers on the default record layout to build the file index when it is building a file from the HP ALLBASE/4GL specifications. The key number defines the number of the file index that you must use to locate a record on the file using this key. If you intend to use the record layout as the KSAM data file's default record layout, you must define at least one field on the record layout as a key field.

Key numbers are also used to define the number of the file index used to locate a record on an HP TurboIMAGE/iX data set. For detail data sets, these keys are called search items. The keys defined for a record that is used to define an existing physical HP TurboIMAGE/iX data set should match the search items

already existing in the data set. Refer to Additional Information for some restrictions placed on record keys for HP TurboIMAGE/iX data sets.

Duplicate Keys Enter Y if duplicate key values are permitted. Enter N if duplicate key values are not permitted (that is, only unique key values are permitted).
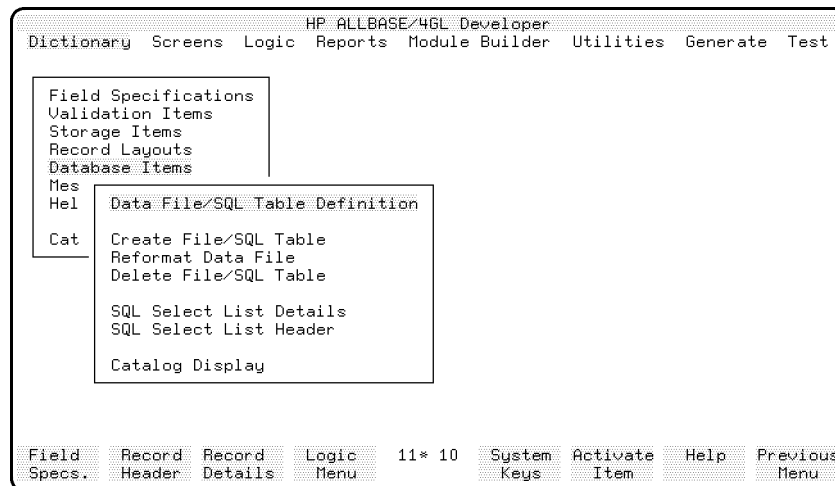
Function Keys

(f1) `Field Specs.` Cancels the current screen and displays the field specifications screen.

(f2) `Record Header` Cancels the current screen and displays the record layout header screen.

(f3) `File Defn.` Cancels the current screen and displays the file/SQL table definition screen.

(f4) `Generate Record` Generates the record layout.

Additional Information

You must generate a record layout before the application can use it.

There are certain rules about record layout keys that you need to follow within HP ALLBASE/4GL.

The KSAM file manager requires at least one key to a record so it can maintain an index of the records in a data file.

The record layout for an HP ALLBASE/SQL table does not need to have any fields defined as key fields. If you do define a field on the record layout as a key field, HP ALLBASE/4GL uses the field to create an index for the table. You cannot use a field specification with multiple occurrences as a key.

Like HP ALLBASE/SQL, HP TurboIMAGE/iX data sets cannot use record layouts with a key field specification with multiple occurrences. There are other rules for record layouts that are used in HP TurboIMAGE/iX data set definitions:

- Record layouts for master data sets cannot contain keys that allow duplicate entries.

- Record layouts for automatic master data sets must contain one key field, and no other fields.

- Record layouts for manual master data sets must contain one key field, and may contain other non-key fields.

- Record layouts for detail data sets do not need to contain any key fields, but may contain up to 16 keys.

Record Layout Header Screen

This screen allows you to assign a name to a record layout and enter a description for documentation purposes.

Screen Image

```
Developer                   Record Layout Header                   record_header


             Record Name  ███████████████           Secured     (Y/N)






             Description
             ░░░░░░░░░░░░░░░░░░
             ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░

             Last Modification:        Date             Time

  Field    Record     File     Work      5* 23  System   Commit    Help   Previous
  Specs.   Details    Defn.    Areas            Keys     Data              Menu
```

FINAL TRIM SIZE : 7.0 in x 8.5 in

Field Descriptions

Record Name   Enter the name of the record layout (up to 16 characters).

Secured   To protect this record layout from unauthorized modification, enter Y after you have completed the required fields on this screen. The entry in this field defaults to N.

If you enter Y, the record layout can only be modified by a developer who has signed on to the application under the correct development security code.

Description   Documentation fields. Enter a description of the record layout for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

Function Keys

(f1)   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

(f2)   `Record Details`   Cancels the current screen and displays the record layout details screen.

(f3)   `File Defn`   Cancels the current screen and displays the file/SQL table definition screen.

(f4)   `Work Areas`   Cancels the current screen and displays the work areas screen.

Database Items Menu

The database items menu allows you to access the screen used to define and maintain KSAM data files, serial data files, HP ALLBASE/SQL tables, HP ALLBASE/SQL select lists, and HP TurboIMAGE/iX data sets.

Screen Image

```
                              HP ALLBASE/4GL Developer
      Dictionary   Screens   Logic   Reports   Module Builder   Utilities   Generate   Test

        ┌──────────────────────┐
        │ Field Specifications │
        │ Validation Items     │
        │ Storage Items        │
        │ Record Layouts       │
        │ Database Items       │
        │ Mes ┌───────────────────────────────────────┐
        │ Hel │ Data File/SQL Table Definition        │
        │     │                                       │
        │ Cat │ Create File/SQL Table                 │
        └─────│ Reformat Data File                    │
              │ Delete File/SQL Table                 │
              │                                       │
              │ SQL Select List Details               │
              │ SQL Select List Header                │
              │                                       │
              │ Catalog Display                       │
              └───────────────────────────────────────┘


      ┌───────┬───────┬───────┬───────┬─────────┬────────┬─────────┬──────┬──────────┐
      │ Field │Record │Record │ Logic │  11* 10 │ System │ Activate│ Help │ Previous │
      │ Specs.│Header │Details│ Menu  │         │ Keys   │ Item    │      │ Menu     │
      └───────┴───────┴───────┴───────┴─────────┴────────┴─────────┴──────┴──────────┘
```

Menu Selections

Data File/SQL Table Definition   This selection displays the file/SQL table definition screen. This screen allows you to name and define HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, KSAM data files, and serial data files.

Create File/SQL Table   This selection displays the file/SQL table creation screen. This screen allows you to create new HP ALLBASE/SQL tables and KSAM data files from within HP ALLBASE/4GL.

Reformat Data File   This selection displays the data file reformat screen, allowing you to reformat an existing KSAM data file.

Delete File/SQL Table   This selection displays the data file/SQL table deletion screen, allowing you to delete existing HP ALLBASE/SQL tables or KSAM data files from within HP ALLBASE/4GL.

SQL Select List Details   This selection calls the select list details screen, allowing you to enter the details of the column definitions for the select list.

SQL Select List Header   This selection calls the select list header screen, allowing you to name an HP ALLBASE/SQL select list, and enter a description of the select list.

Catalog Display   This selection displays the catalog display screen, allowing you to display the names of existing application components. Refer to chapter 13 for details.

Function Keys

[f1]   `Field Specs.`   Cancels the current screen and displays the fields specifications screen.

[f2]   `Record Header`   Cancels the current screen and displays the record layout header screen.

[f3]   `Record Details`   Cancels the current screen and displays the record layout details screen.

[f4]   `Logic Menu`   Cancels the current screen and displays the logic menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

File/SQL Table Definition Screen

This screen allows you to define HP ALLBASE/SQL tables, KSAM data files,
HP TurboIMAGE/iX data sets, and serial data files.
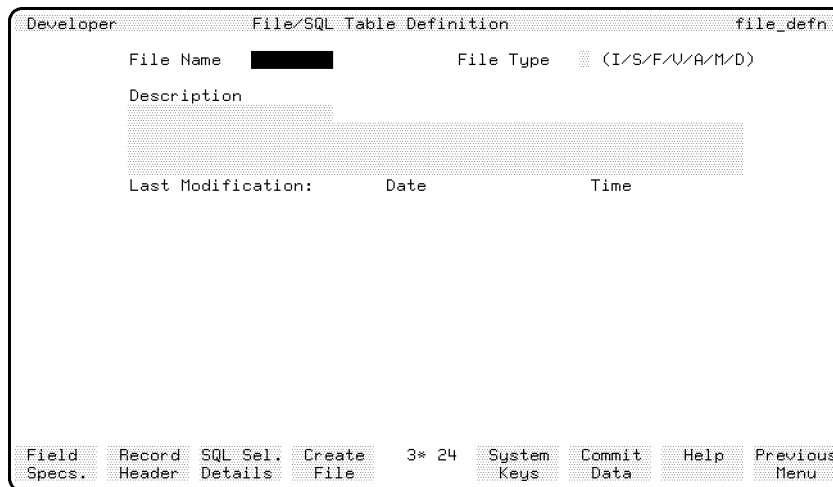
This screen uses three windows. The first window (File Type "I") allows you to
enter the details for KSAM files, and the second window (File Type "S") lets
you enter details for HP ALLBASE/SQL base tables. The third window (File
Type "M") lets you enter the details for HP TurboIMAGE/iX detail master
data sets.

Note

The term *file*, as used on this screen, refers to HP
ALLBASE/SQL base tables, HP TurboIMAGE/iX data sets,
and KSAM and serial files.

Screen Image

```
Developer              File/SQL Table Definition              file_defn

        File Name    ▮▮▮▮▮▮        File Type    (I/S/F/V/A/M/D)
        Description

        Last Modification:      Date            Time

 Field     Record  SQL Sel.  Create   3* 24   System   Commit   Help  Previous
 Specs.    Header  Details    File             Keys     Data            Menu
```

FINAL TRIM SIZE : 7.0 in x 8.5 in

Field Descriptions

File Name   Enter the HP ALLBASE/4GL internal name of the file. This is the name that is used in all file references and in the FILE logic command.

If you enter a file name, and a file with the same name in case shifted form already exists, HP ALLBASE/4GL displays a warning message.

File Type   The permissible entries in this field are:

I	KSAM (indexed) data file
S	HP ALLBASE/SQL table
F	Fixed length record serial file
V	Variable length record serial file
A	Automatic master HP TurboIMAGE/iX data set
D	Detail HP TurboIMAGE/iX data set
M	Manual master HP TurboIMAGE/iX data set

The entry in this field determines which window is displayed on this screen.

Description   Documentation fields. Enter a description of the file for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

SQL Table Definition Window

```
   External Name      PRODUCT

   SQL DBEfileset

   SQL Access Class       (1 - Public, 2 - Publicread, 3 - Private)

   Record Layout      product_rcrd




  Field    Record   SQL Sel.  Create    7* 12   System   Commit   Help   Previous
  Specs.   Header   Details   File               Keys     Data            Menu
```

FINAL TRIM SIZE : 7.0 in x 8.5 in

Field Descriptions

External Name   Enter the external name of the table as it is known in the HP ALLBASE/SQL database. Note that you cannot use a qualified name in the form *owner.table* in this field.

The entry in this field defaults to the uppercase form of the HP ALLBASE/4GL name for the table. You can type over the default if you require a different external name for the table.

The external name for an HP ALLBASE/SQL table can be up to 20 characters in length. It can be made up of any combination of uppercase letters (A to Z), digits (0 to 9), \\$, #, :, or underscores (_). The first character cannot be an underscore or a digit. You cannot enter lowercase letters in this field.

SQL DBEFileset   Enter the name of the HP ALLBASE/SQL DBEFileset that will contain the HP ALLBASE/SQL table. The entry in this field defaults to SYSTEM.

HP ALLBASE/4GL creates the table as *owner.table_name* in the specified fileset, where *owner* is the name of the HP ALLBASE/SQL owner group specified on the application header screen in the HP ALLBASE/4GL administrator.

HP ALLBASE/4GL creates the table in the database environment defined for the application on the application definition screen by the system administrator.

The DBEfileset must contain at least one DBEfile. Both the DBEfileset and the DBEfile must exist before you can create an HP ALLBASE/SQL table.

SQL Access Class   Enter 1, 2, or 3 to define the SQL access class for the table as Public, Publicread, or Private respectively. The entry in this field defaults to 1. The SQL access class determines the locking mode and the initial access mode for the table. Refer to the *HP ALLBASE/SQL Reference Manual* for more information.

Record Layout   Enter the name of an existing record layout to define the columns of the table.

The record layout must not contain any field specifications that are defined as having multiple occurrences.

HP ALLBASE/4GL uses the key fields (if any) on this record layout to define the indexes for the table.

HP ALLBASE/SQL uses the names of the fields on the record layout to define the columns of the table. HP ALLBASE/SQL automatically converts all alphabetic characters in the field specification names to uppercase.

HP TurboIMAGE/iX Data Set Definition Window

This window allows you to define automatic master data sets, manual master data sets, and detail data sets. Refer to Additional Information for a brief definition of these three types of data sets.

```
┌─────────────────────────────────────────────────────────────────┐
│        Database Name                                              │
│        Schema Data Set Name                                       │
│                                                                   │
│        Record Layout List              Default Record             │
│                                                                   │
│        1                               7                          │
│        2                               8                          │
│        3                               9                          │
│        4                              10                          │
│        5                              11                          │
│        6                              12                          │
│                                                                   │
│  Field    Record                7* 12  System   Commit   Help  Previous │
│  Specs.   Header                       Keys     Data           Menu │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Database Name   Enter the name of the database that will contain the data set. This must be the name of an existing database name defined by the HP ALLBASE/4GL administrator in the administrator application.

Schema Data Set Name   This is the external HP TurboIMAGE/iX name of the data set, up to 16 characters. The data set should exist in the database identified in the previous field.

The name may contain all characters valid in HP ALLBASE/4GL names except the underscore character (_). The name may also contain any of the following characters, which are accepted within HP TurboIMAGE/iX:

+, -, *, /, ?, ', #, %, &, and @.

FINAL TRIM SIZE : 7.0 in x 8.5 in

This field defaults to the name in the File Name field, with any underscore characters (_) converted to hyphens (-), and any lowercase characters converted to uppercase.

Default Record   Enter the number of one of the 12 record layouts for the file that is to be treated as the default record in a file record field reference. (Refer to chapter 3.) If you don't enter a record number, HP ALLBASE/4GL uses record number 1 as the default record.

Record Layout List   Enter the names of the record layouts for this file. You can specify up to 12 different record layouts.

HP ALLBASE/4GL checks each record layout to ensure that it complies with the following rules. If a record layout does not comply with these rules, it cannot be used in a HP TurboIMAGE/iX data set, and HP ALLBASE/4GL will display an error message when you press the `Commit Data` function key.

- Record layouts must not contain a key field with repeated occurrences.

- Record layouts for master data sets cannot contain keys that allow duplicate entries.

- Record layouts for automatic master data sets must contain one key field, and no other fields.

- Record layouts for manual master data sets must contain one key field, and may contain other non-key fields.

- Record layouts for detail data sets do not need to contain any key fields.

HP ALLBASE/4GL allocates a file record buffer for each record layout that you include in this list.

Each field specification contained in a record layout used in this list should match a data item in the physical HP TurboIMAGE/iX database. If a field specification does not match an HP TurboIMAGE/iX data item, an error will occur when the application is run.

KSAM/Serial File Definition Window

```
        External Name              ░░░░░░░░░░░░░░░░░░░░

        Record Layout List               Default Record      ░░

        1   ░░░░░░░░░░░░░░           7   ░░░░░░░░░░░░░
        2   ░░░░░░░░░░░░░░           8   ░░░░░░░░░░░░░
        3   ░░░░░░░░░░░░░░           9   ░░░░░░░░░░░░░
        4   ░░░░░░░░░░░░░░          10   ░░░░░░░░░░░░░
        5   ░░░░░░░░░░░░░░          11   ░░░░░░░░░░░░░
        6   ░░░░░░░░░░░░░░          12   ░░░░░░░░░░░░░


  Field    Record   SQL Sel.  Create    6* 12   System   Commit   Help   Previous
  Specs.   Header   Details   File               Keys     Data            Menu
```

Field Descriptions

External Name   Enter the name of the file as it is known to the MPE/iX
operating system, that is, the name of the file which resides on disk. It can be
either a simple or a fully qualified file name.

For KSAM files, the external file name (excluding any group and account name
specifier) cannot be longer than seven characters, and must be an acceptable
MPE/iX file name.

When you create this file, the HP ALLBASE/4GL KSAM data manager adds
the suffixes K, and S, to the external name you specify here to determine the
names of the physical files for the KSAM data file.

Normally, HP ALLBASE/4GL appends the current value of the MPE/iX
variable HP4DATAPATH to the file name. However, if you specify a fully
qualified file name, HP ALLBASE/4GL builds the file in the group and
account specified. Refer to appendix D for details of the MPE/iX variables and
group and account structure used by HP ALLBASE/4GL. If HP4DATAPATH
is undefined, HP ALLBASE/4GL builds the file in the user's current group and
account.

The external file name need not be unique. More than one HP ALLBASE/4GL
file name may refer to the same external file.

For serial files, HP ALLBASE/4GL allows you to specify the name of a file dynamically when the application is being run. To do this, you must specify the external name of the file as *FILENAME on this screen. The application logic must move the required file name into the *FILENAME communication area field before you access the file. If the serial file does not already exist, HP ALLBASE/4GL creates it when it is first accessed.

By changing the name in *FILENAME, you can use the same HP ALLBASE/4GL file definition to refer to a number of different external files. However, you must make sure that the HP ALLBASE/4GL file is closed before you attempt to access a different external file. HP ALLBASE/4GL automatically closes all data files at the start of a process. You can also close a file explicitly with the FILE *CLOSE command.

Normally, HP ALLBASE/4GL appends the current value of the MPE/iX variable HP4DATAPATH to the file name. If the value in *FILENAME is a fully qualified file name, HP ALLBASE/4GL performs the file operation on the file specified by the value in *FILENAME. If HP4DATAPATH is undefined, HP ALLBASE/4GL assumes the file is in the user's current group and account.

HP ALLBASE/4GL also allows you to define a serial file as a temporary file. To define a file as temporary, enter *TEMP in the *External Name* field. HP ALLBASE/4GL creates a temporary file when the application first attempts to write to the file. These files are created in the temporary domain of the group and account specified by the value of the MPE/iX variable HP4DATAPATH. If HP4DATAPATH is undefined, the temporary domain of the user's current group and account is used. HP ALLBASE/4GL deletes temporary files automatically when they are closed or when a process terminates.

Record Layout List Enter the names of the record layouts for this file. You can specify up to 12 different record layouts.

HP ALLBASE/4GL creates the file with a physical record layout determined by the default record. The physical file only contains one record type, even if you include more than one record layout name in this list.

HP ALLBASE/4GL allocates a file record buffer for each record layout that you include in this list.

Default Record Enter the number of one of the 12 record layouts for the file that is to be treated as the default record in a file record field reference. (Refer

to chapter 3.) If you don't enter a record number, HP ALLBASE/4GL uses record number 1 as the default record.

For KSAM files and fixed length record serial files, HP ALLBASE/4GL uses the default record layout to determine the length and layout of the physical records for the file. If you are creating a file that uses multiple record layouts, the default record layout should be the longest record layout for the file. If any record layouts are longer than the default record layout, the additional data is truncated when the record is written to the file.

When HP ALLBASE/4GL writes to a variable length record serial file, the number of characters written is determined by the length of the current record buffer rather than the default record layout. For serial files with variable length records, HP ALLBASE/4GL writes the number of characters in the current buffer. HP ALLBASE/4GL strips any trailing spaces. Note that under the MPE/iX operating system, a variable length file cannot use a record that is longer than the default record layout for the file. The maximum record length for a variable length record serial file is determined by the length of the default record for the file.

HP ALLBASE/4GL uses the key fields on the default record layout to build the KSAM file's index.

Function Keys

[f1]　　　`Field Specs.`　Cancels the current screen and displays the field specifications screen.

[f2]　　　`Record Header`　Cancels the current screen and displays the record layout header screen.

[f3]　　　`SQL Sel Details`　Cancels the current screen and displays the select list details screen.

[f4]　　　`Create File`　Cancels the current screen and displays the file/SQL table creation screen.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Additional Information

Using Existing HP ALLBASE/SQL Tables

If you want to access an HP ALLBASE/SQL table directly using HP
ALLBASE/4GL, you can enter an HP ALLBASE/4GL definition for an
HP ALLBASE/SQL table that has the same name as the existing HP
ALLBASE/SQL table. To do this, you must define a record layout with fields
that match the column definitions for the existing table.

Alternatively, you can use a SELECT command in an SQL block to access
existing tables. Refer to chapter 11 for more information about SQL logic
blocks.

Using Existing HP TurboIMAGE/iX Data Set

If you want to access an existing HP TurboIMAGE/iX data set directly using
HP ALLBASE/4GL, you can use HP ALLBASE/4GL utilities to load the data
set definition into HP ALLBASE/4GL. The upload utility, *HP4TUPLD*, uses
information from the HP TurboIMAGE/iX database schema file to create an
ASCII text file containing the data item and data set definitions. You then use
HP4ATOS to load the definitions into HP ALLBASE/4GL. This provides the
HP TurboIMAGE/iX data item definitions and data set definitions as field
specifications, record layouts, and data sets in HP ALLBASE/4GL. This is
particularly useful if you wish to use a number of existing data sets. Refer to
chapter 10 for details about this procedure.

Alternatively, you can enter an HP ALLBASE/4GL definition for the data
set, and enter the name of the existing HP TurboIMAGE/iX data set in the
Schema Data Set Name field on the file/SQL table definition screen.

HP TurboIMAGE/iX Data Set Types

There are three types of HP TurboIMAGE/iX data sets: automatic master
data sets, manual master data sets, and detail data sets. A brief definition
of each type is included here, but for complete descriptions, refer to the *HP
TurboIMAGE/iX Database Management System Reference Manual*.

A detail data set contains records with similar types of data. For instance, it
may contain details about each order that a customer submits. Detail data
sets contain search items (key items), and each data set can contain up to 16

search items. All the records with common key values are linked together in a logical chain. In the example above, the customer name or number might be the search item, and all records for a single customer would be linked together in a logical chain.

Detail data sets are related to automatic master data sets and manual master data sets by common key items.

An automatic master data set record contains one item, which is the key. When a record is added to any detail data set related to an automatic master data set, a record is automatically placed in the automatic master data set. Using the customer order example, the order date might be placed in an automatic master data set.

An automatic master data set cannot be updated separately; a detail data set record is required. If all detail data set records related to an automatic master record are deleted, the automatic master record is deleted automatically.

A manual master data set contains a unique key item and may contain other items. A record cannot be added to any detail data set related to the manual master data set unless a record containing a matching key item exists in the manual master data set. Using the customer order example, a manual master may contain customer details. You cannot add a customer order record in the detail data set until the customer details exist in the master data set.

A manual master data set cannot be automatically updated, and it may contain records not currently related to a detail data set record.

Serial File Creation

If a serial file does not already exist, HP ALLBASE/4GL automatically creates it when the application first attempts to access the file.

The process containing the FILE command that creates the file must contain a MODE *WRITE command to specify that the process has write access to the file. If the process does not contain a MODE *WRITE command, the file creation will fail.

HP ALLBASE/4GL creates fixed length serial files with a record length equal to the length of the default record layout for the file.

Note	If you use multiple record layouts for a fixed length serial file under the MPE/iX operating system, make sure that the default record layout is the longest record layout for the file. If you attempt to write to the file via a record layout that is longer than the default record layout, the additional characters will be truncated.

For variable length record serial files, HP ALLBASE/4GL also uses the length of the longest record layout for the file to determine the **maximum** length of records for a variable length record serial file.

If required, you can use an MPE/iX file equation to equate the HP ALLBASE/4GL serial file to an MPE/iX file with different default characteristics.

HP ALLBASE/4GL creates temporary serial files in the temporary domain of the group and account specified by the value of the system variable HP4DATAPATH. If HP4DATAPATH is undefined, the temporary domain of the user's current group and account is used. Application users must have appropriate access permissions for the group and account.

Using File Equations

If required, you can use an MPE/iX file equation to identify the external name of an HP TurboIMAGE/iX data set, a KSAM file, or a serial file.

If you do use a file equation for an HP ALLBASE/4GL file, the formal designator for the file must use the fully qualified name for the file. For example, if the external name for a file is *PRODUCT* and the current value for the MPE/iX variable HP4DATAPATH is *REALDATA.HP4GL* you could use a file equation in the following format:

```
FILE PRODUCT.REALDATA.HP4GL=PARTLIST.TESTDATA.HP4DEV
```

You cannot use file equations with HP ALLBASE/SQL tables.

FINAL TRIM SIZE : 7.0 in x 8.5 in

File/SQL Table Creation Screen

This screen allows you to create new HP ALLBASE/SQL tables or KSAM data files from within HP ALLBASE/4GL.

You do not need to use this screen to create serial files. If a serial file does not already exist when the application first attempts to access the file, HP ALLBASE/4GL automatically creates it.

HP TurboIMAGE/iX data sets are not physically created within HP ALLBASE/4GL. Commonly, existing HP TurboIMAGE/iX databases will be accessed, and the existing data set definition will be uploaded using HP ALLBASE/4GL utilities. Refer to chapter 10 for details about these utilities.

Screen Image

```
Developer                    File/SQL Table Creation              create_file

               File Name          ████████
               File Type
               External Name
               SQL Access Class
               SQL DBEfileset
                            Key Field List
Key                   Dup  Key              Dup  Key                    Dup
1.                         7.                    13.
2.                         8.                    14.
3.                         9.                    15.
4.                         10.                   16.
5.                         11.
6.                         12.

          Description




          Last Modification:      Date              Time


  File              Reformat  Delete   3* 34  System  Commit  Help  Previous
  Defn.             File      File            Keys    Data          Menu
```

Field Description

File Name   Enter the HP ALLBASE/4GL name of the HP SQL table or
KSAM file that you want to create. You must complete the file/SQL table
definition screen for the table or file before you can use this screen to create it.

The remainder of the fields on this screen are display only fields. When you
enter the HP ALLBASE/4GL file name, the system displays the file type, and
its external name. For HP ALLBASE/SQL tables, this screen also displays the
SQL access class and the SQL DBEFileset for the table.

For both HP ALLBASE/SQL tables and KSAM files this screen also shows
the details of the key fields defined for the record layout for the file, and the
description of the file. For KSAM files, HP ALLBASE/4GL shows the key
fields defined for the default record layout.

Function Keys

| f1 | `File Defn` Cancels the current screen and displays the file/SQL table definition screen. |

| f3 | `Reformat File` Cancels the current screen and displays the data file reformat screen. |

| f4 | `Delete File` Cancels the current screen and displays the file/SQL table deletion screen. |

Additional Information

HP ALLBASE/4GL creates HP ALLBASE/SQL tables and KSAM data files
automatically when you press the `Commit Data` function key.

HP ALLBASE/SQL Table Creation

To use this screen to create an HP ALLBASE/SQL table, you must have
resource authority for the application DBEnvironment. HP ALLBASE/4GL
transfers the ownership of the table to the SQL owner group specified for the
application on the application definition screen in the HP ALLBASE/4GL
administrator. HP ALLBASE/4GL creates the table with all columns allowing
null values

5-62   Dictionary

FINAL TRIM SIZE : 7.0 in x 8.5 in

If the table already exists and contains data, HP ALLBASE/4GL displays a warning message.

KSAM Data File Creation

To create a KSAM data file, the MPE/iX group and account for the file must exist, and you must have MPE/iX access permission for the group and account. Note that you cannot create a file in an account other than your log-in account.

If the file already exists and contains data, HP ALLBASE/4GL displays a warning message.

When you press the `Commit Data` function key to create a file, HP ALLBASE/4GL calls an external file utility program to create the file. The utility program uses a file-structure file, created by HP ALLBASE/4GL, in the group and account specified by the variable HP4FSPATH. The file-structure file is purged automatically when the KSAM file has been created.

The file creation utility will fail if the file-structure file already exists in the group and account specified by the variable HP4FSPATH.

HP ALLBASE/4GL creates three physical files whenever you create a KSAM file. For example, the HP ALLBASE/4GL file *customer* with an external name of *CUSTOMR* would have the following disk files:

Filename	Purpose
CUSTOMR	Data file
CUSTOMRK	Key file
CUSTOMRS	Structure file

All three files are created in the same group and account.

Data File Reformat Screen

This screen allows you to modify the format of an existing KSAM data
file without reentering the data in the file. You cannot use this screen to
reformat serial data files, HP SQL tables, or HP TurboIMAGE/iX data
sets, but procedures for reformatting HP ALLBASE/SQL tables and HP
TurboIMAGE/iX data sets are provided, under Additional Information.

Note   HP ALLBASE/4GL KSAM data files may be defined as having
more than one record layout per file. The file reformat utility
reformats a file according to the definition of the **default** record
layout only. You may need to restructure all record layouts for
files having multiple record layouts.

Screen Image

```
 Developer                   Data File Re-Format              reformat_file

                    File Name        ████████
                    File Type
                    External Name

                            Key Field List
 Key              Dup  Key              Dup  Key                     Dup
 1.                     7.                    13.
 2.                     8.                    14.
 3.                     9.                    15.
 4.                     10.                   16.
 5.                     11.
 6.                     12.


        Description




        Last Modification:       Date                  Time


  File    Create            Delete   3* 33  System   Commit   Help   Previous
  Defn.   File              File             Keys     Data            Menu
```

Field Description

File Name   Enter the HP ALLBASE/4GL name of the KSAM file to be
reformatted.

The remainder of the fields on this screen are display only fields. When you enter the HP ALLBASE/4GL name for the file, the system displays the MPE/iX file name, file type, key fields in the default record, and the file description.

When you press the `Commit Data` function key, HP ALLBASE/4GL displays a message requesting you to confirm the data file reformat request. Enter Y to continue, or enter N to abort the file reformat. If you continue with the data file reformat process, HP ALLBASE/4GL calls an external program to reformat the file.

Note   HP ALLBASE/4GL needs sufficient disk space for temporary storage of both the current file and the new file during the reformatting process.

Function Keys

(f1)   `File Defn` Cancels the current screen and displays the file definition screen.

(f2)   `Create File` Cancels the current screen and displays the file creation screen.

(f4)   `Delete File` Cancels the current screen and displays the file deletion screen.

(f6)   `Commit Data` Reformats the named file.

Additional Information

HP ALLBASE/4GL reformats the file to reflect changes made to the default record layout for the file. The file reformat screen allows you to reformat a data file to reflect changes made to characteristics such as length, type, justification, and number of decimal places for the fields in the default record layout for the file.

You can also use this screen to reformat a data file if you have added new fields, repositioned fields, or deleted fields in the default record layout. New fields are initialized with the pad character defined for that field.

You must generate the redefined default record layout before you use this screen to reformat a data file.

Note

To reformat a KSAM data file under the MPE/iX operating system you must be the owner of the file. The reformat process will fail if you are not the owner of the file.

When you press the `Commit Data` function key to reformat the file, HP ALLBASE/4GL calls an external file utility program. The file reformat utility will fail if a file-structure file exists in the group and account specified by the variable HP4FSPATH. This file-structure file is part of the automatic run-time data file reformatting system. It may exist if a run-only copy of the application has been unloaded using the administrator utilities application unloading screen.

File Reformat Limitations

Be careful about renaming fields on a record layout. If you do, all data in the old fields will be lost as the new field is initialized with the field's pad character.

You must also be careful about adding new key fields to a record layout, or changing existing key fields. If you add a new key to a record layout, and define that field as a unique key, HP ALLBASE/4GL will not be able to reformat the file.

Similarly, you cannot change an existing key that permits duplicates to a key that only allows unique values.

During the data file reformatting process, HP ALLBASE/4GL issues messages if the reformatting will shorten any fields, or reduce the number of occurrences of a field defined as having more than one occurrence. In these circumstances, HP ALLBASE/4GL issues a message requesting you to confirm that you want to continue with the file reformatting process.

HP ALLBASE/SQL Table Reformatting

HP ALLBASE/4GL does not provide any direct facilities to allow you to reformat HP ALLBASE/SQL tables.

FINAL TRIM SIZE : 7.0 in x 8.5 in

If you change the specification of a field, or add a new field to the table record layout, you can use the ISQL UNLOAD/LOAD facilities to reformat the table. Refer to the *ISQL Reference Manual* for more information.

HP TurboIMAGE/iX Data Set Reformatting

HP ALLBASE/4GL does not provide any direct facilities to allow you to reformat HP TurboIMAGE/iX data sets.

Refer to the *HP TurboIMAGE/iX Database Management System Reference Manual* or the *HP TurboIMAGE/iX DBchange/XL User's Guide* for more information.

Run-Time Environment Data File Reformatting

HP ALLBASE/4GL provides a mechanism for automatically reformatting existing KSAM data files in the run-time environment.

This mechanism uses the existing structure file associated with each KSAM data file (these files are indicated by the S suffix), and a file-structure file that is created during the run-time application unloading procedure.

When a run-time application is loaded into the target system, HP ALLBASE/4GL compares the structure for each existing data file, as defined by its existing structure file, and the format for the incoming data file as shown in the file-structure file. If the two formats differ, HP ALLBASE/4GL reformats the data file.

Under the MPE/iX system, the run-time environment system administrator must be the creator of the files to be able to reformat them.

The run-time data file reformatting system is described in the *HP ALLBASE/4GL Developer Administration Manual*.

File/SQL Table Deletion Screen

This screen allows you to delete existing HP ALLBASE/SQL tables and KSAM data files from within HP ALLBASE/4GL.

You cannot delete serial files or HP TurboIMAGE/iX data sets from within HP ALLBASE/4GL.

Screen Image

```
Developer                     File/SQL Table Deletion                  delete_file
                    File Name            ███████████
                    File Type
                    External Name
                    SQL Access Class
                    SQL DBEfileset
                                    Key Field List
         Key               Dup  Key              Dup  Key              Dup
         1.                     7.                     13.
         2.                     8.                     14.
         3.                     9.                     15.
         4.                     10.                    16.
         5.                     11.
         6.                     12.

         Description




         Last Modification:       Date                   Time


       File     Create  Reformat              3* 34  System   Commit   Help  Previous
       Defn.    File    File                         Keys     Data           Menu
```

Field Description

File Name Enter the HP ALLBASE/4GL name of the HP SQL table or KSAM file to be deleted. The system displays the file type, its external name, and the file description for confirmation.

When you press the Commit Data function key, HP ALLBASE/4GL displays a message requesting you to confirm the file deletion. Enter Y to confirm the deletion, or enter N to cancel the deletion.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Function Keys

(f1)   `File Defn` Cancels the current screen and displays the file/SQL table definition screen.

(f2)   `Create File` Cancels the current screen and displays the file/SQL table creation screen.

(f3)   `Reformat File` Cancels the current screen and displays the data file reformat screen.

(f6)   `Commit Data` Terminates the current screen and deletes the named file.

Additional Information

This facility deletes the actual HP ALLBASE/SQL table or data file, including any data in the table or file. It does not delete the file definition from the HP ALLBASE/4GL dictionary. The deletions screen in the utilities menu allows you to delete a file definition from the dictionary.

Select List Details Screen

This screen allows you to enter the details for an HP ALLBASE/SQL select list.

Screen Image

```
Developer                    SQL Select List Details              SQLsel_details
Select List  ████████                          Secured      (Y/N)
Select List Details




                                                                              




Field     SQL Sel. SQL Blk  Generate   3* 15   System   Commit    Help   Previous
Specs.    Header   Menu      List               Keys     Data             Menu
```
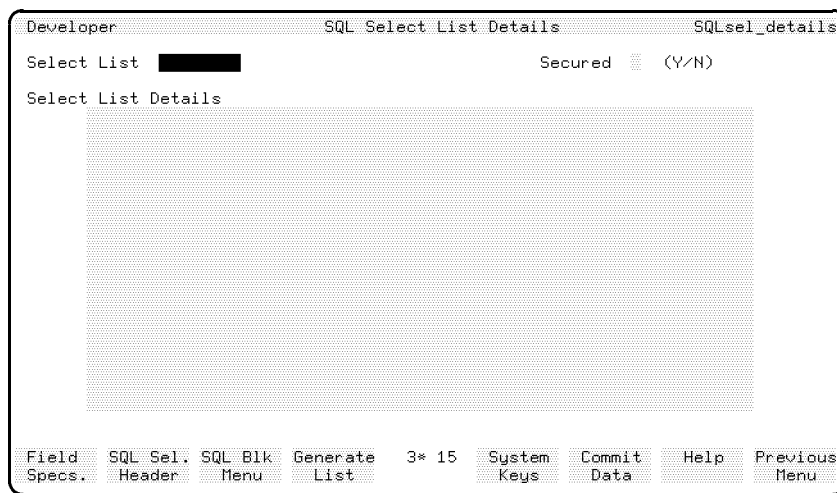
Field Descriptions

Select List   Enter the name of a select list that has been defined on the select list header screen.

Select List Details   This is a free format text entry area where you can enter the details of the select list. The entries you make in this field define the columns of the select list.

Each entry has the format:

$field\_spec\ [\ =\ column\_definition]$

In this expression, *field_spec* is the name of a dictionary field specification, and *column_definition* is a valid HP ALLBASE/SQL expression. You can use aggregate functions and expressions in the HP ALLBASE/SQL column

definitions. The field specification used to define a select list column cannot allow multiple occurrences.

The right hand side of this expression is optional. If you don't specify the right hand side of the expression, HP ALLBASE/4GL uses the field specification name as the column definition.

Use a comma (,) to separate successive entries. You can enter the definition of more than one column on a single line on this screen provided you separate the entries with a comma. Do not use a comma after the last entry.

Refer to Additional Information for an example of a select list definition.

Function Keys

[f1]   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

[f2]   `SQL Sel Header`   Cancels the current screen and displays the SQL select list header screen.

[f3]   `SQL Blk Menu`   Cancels the current screen and displays the SQL logic block menu.

[f4]   `Generate List`   Generates the select list.

Additional Information

A select list is a "virtual" HP ALLBASE/SQL table. A select list contains columns from one or more base tables or views. Each column is defined by a dictionary field specification. The values in select list columns can be:

- Taken directly from a base table column.

- Computed from a base table column value using an arithmetic expression.

- Computed from the values in various base table columns using an arithmetic expression.

- Computed from a group of values using an aggregate function (AVG, MAX, MIN, SUM, and COUNT).

- A constant, or computed from an expression involving constants.

Dictionary   5-71

HP ALLBASE/SQL Host Variable References

Column definitions for an SQL select list can contain references to HP ALLBASE/4GL file record fields, work area fields, variables and calculated items, and constants. You cannot use references to scratch-pad fields or communication area fields in SQL select lists.

SQL select list column definitions can also contain references to screen fields, provided the reference is a fully qualified reference by name. You cannot use substrings in references to screen field names.

In a select list you must precede a reference to an HP ALLBASE/4GL item with a colon (:). That is, you must express the reference in the format:

:*data_ref*

where *data_ref* is the HP ALLBASE/4GL name, including the item type prefix for the HP ALLBASE/4GL item.

Select List Generation

You must generate a select list before it can be used in the application.

When HP ALLBASE/4GL generates a select list, it uses the field specifications in the select list column definitions to create a record layout for the select list.

Select List Example

```
salesman_no = Salesman_tbl.employee_no,
month = 'Feb',
state = Salesman_tbl.sales_region,
division =Salesman_tbl.sales_division,
tot_sales = SUM(Feb_sales_tbl.amount ),
ytd_quota_pct = 100 * tot_sales / Salesman_tbl.yr_quota
```

Each entry on the left of the = sign in these expressions is a field specification that has been defined on the field specifications screen.

Select List Header Screen

This screen allows you to name an HP ALLBASE/SQL select list and enter a description of the select list.

Screen Image

```
┌──────────────────────────────────────────────────────────────────┐
│ Developer                  Select List Header            SQLsel_header │
│          Select List Name ▇▇▇▇▇▇▇      File Type                 │
│          Description                                            │
│          ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░  │
│          ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░  │
│                                                                  │
│          Last Modification:      Date            Time           │
│                                                                  │
│                                                                  │
│                                                                  │
│                                                                  │
│                                                                  │
│                                                                  │
│                                                                  │
│  File    SQL Sel. SQL Blk         3* 29  System  Commit  Help  Previous │
│  Defn.   Details  Menu                    Keys    Data          Menu │
└──────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Select List Name   Enter the name of the select list. The name can have up to eight characters and can use alphabetic characters, 0 to 9 and _ (underscore). The name must start with an alphabetic character.

A select list cannot have the same name as any HP ALLBASE/SQL tables, KSAM data files, serial data files or record layouts in the same application.

File Type   This is a display only field that indicates that the file type is L, for select list.

Description   Documentation fields. Enter a description of the select list for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Function Keys

[f1]   `File Defn` Cancels the current screen and displays the file/SQL table definition screen.

[f2]   `SQL Sel Details` Cancels the current screen and displays the SQL select list details screens.

[f3]   `SQL Blk Menu` Cancels the current screen and displays the SQL logic block menu.

Messages Screen

This screen allows you to name an HP ALLBASE/4GL message and to define its contents.

Screen Image

```
┌─────────────────────────────────────────────────────────────────┐
│ Developer                    Messages                    messages │
│                                                                   │
│            Name            ███████████████                        │
│                                                                   │
│            Type            ░░░░░░░  (MESS/QUERY/WARN/ERROR/ABORT)  │
│                                                                   │
│            Response Item   ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│            Help Name       ░░░░░░░░░░░░░░░░░░                      │
│                                                                   │
│                            Message Construction                   │
│            ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│            ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│            ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│            ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│            ░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░     │
│                                                                   │
│  Field    Ranges  Tables  Help     4* 24  System  Commit  Help  Previous │
│  Specs                    Screens          Keys    Data          Menu     │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Name   Enter the name of the message (up to 16 characters).

Type   The entry in this field determines the type of the message. The entry can be one of the abbreviations `MESS`, `QUERY`, `WARN`, `ERROR`, or `ABORT`. The type defines the action to be taken by the system when it displays the message. The meaning of the message type codes are as follows:

Code	Meaning and System Action
MESS	A message for information only, which requires no response from the user. The message is displayed for about three seconds.

Code	Meaning and System Action
QUERY	A question that requires a response from the user. The terminal beeps to alert the user.
WARN	A warning message. The terminal beeps to alert the user, although the user is not required to make any specific response.
ERROR	A message informing the user of an error condition that must be corrected. The terminal beeps to alert the user. If the message has been initiated by a user error in a data entry field, the cursor remains at the current field.
ABORT	A message informing the user that an irreparable system error has been detected. The user should note the error details displayed, then press ⌈Return⌉ as an acknowledgment. The HP ALLBASE/4GL session is then terminated.

Response Item For QUERY type messages only, enter the name of the field used to store the user's response. The response item is the field where the user's response is stored.

The permissible entries for this field are:

- A variable.

- A file record field reference.

- A work area field reference.

- A screen field reference.

- A scratch-pad field.

- An alterable communication area field.

The default entry in this field is *PASS.

When you define a QUERY type message, HP ALLBASE/4GL creates a response field the length of the response item. The position of the field depends on the length of the message:

- Less than 80 characters: row 24, column 1.

- Greater than 80 characters, but with sufficient space for the answer in row 24: at the end of the message.

- Greater than 80 characters, with insufficient space for the answer: the message is truncated to create sufficient space in row 24.

Help Name   For ERROR, ABORT, QUERY type messages only. Enter the name of a help screen to be displayed if the user presses the `Help` function key while HP ALLBASE/4GL is expecting a response to a message.

Message Enter the content of the message display. The message may be displayed on two lines (up to 160 characters) if required.

A message may be constructed from any combination of:

- Master titles.

- Variables or calculated items.

- Communication area field references.

- Constants.

- File record field references.

- Work area field references.

- Literals (enclosed in quotes ").

- Scratch-pad field references.

- Screen field references.

Separate items in the message definition with spaces.

If possible, restrict the message or prompt for QUERY messages to the first line. This leaves the second line clear for the user to enter a response.

Function Keys

[f1]   `Field Specs.`   Cancels the current screen and displays the field specifications screen.

[f2]   `Ranges`   Cancels the current screen and displays the validation ranges screen.

[f3]   `Tables`   Cancels the current screen and displays the validation tables screen.

[f4] `Help Screens` Cancels the current screen and displays the help
screens screen.

Additional Information

Message Display

HP ALLBASE/4GL displays messages on lines 23 and 24 on the application
screen. The message overwrites any information already displayed at that
location. When the message is removed, HP ALLBASE/4GL restores the
original display on the application screen.

Clearing the Message Display

If HP ALLBASE/4GL displays a MESS type message with a total length of
zero characters, (for example, a message that only contains the empty literal
string ""), any existing message display is cleared from the message window,
and the message time delay for the empty message is set to zero.

You can use an empty message to clear the message window, and then proceed
immediately to the next step in the current logic block.

Message Generation

When you press the `Commit Data` function key, HP ALLBASE/4GL
generates messages automatically.

Error Conditions

Messages must not exceed 160 characters in length when displayed, otherwise
HP ALLBASE/4GL truncates the message. HP ALLBASE/4GL issues an
error message if the message definitely exceeds 160 characters when it has been
generated.

HP ALLBASE/4GL also issues an error message if a message contains an
undefined data reference.

HP ALLBASE/4GL issues a warning message if a message exceeds 130
characters in length and contains scratch-pad field references.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Help Screens Screen

This screen allows you to define application help screens for the end user.

Screen Image

```
Developer                        Help Screens                         helps
                   Help Screen Name    ███████████████
                   Next Help Screen
```

Field Descriptions

Help Screen Name   Enter the name of the help screen (up to 16 characters).

Next Help Screen   Enter the name of another help screen to be displayed after this help screen. You can use more than one help screen if there is insufficient space on one screen for the necessary help information.

Message Contents   A free form area of 16 lines, each of 60 characters. Enter the text of the help message anywhere in this space.

Function Keys

[f1]   `Field Specs.`   Cancels the current screen and displays the field
specifications screen.

[f2]   `Storage Menu`   Cancels the current screen and displays the storage
items menu.

[f3]   `Messages`   Cancels the current screen and displays the messages
screen.

Additional Information

HP ALLBASE/4GL Help Display

HP ALLBASE/4GL displays help screens in response to the I-HELP internal
action, or as a result of an H- action from a menu, a function key, or a decision
table.

When HP ALLBASE/4GL receives the I-HELP internal action, it displays help
screens according to the user's current activity in the application. The I-HELP
internal action can only be initiated from a function key. The help screen
display following an I-HELP internal action follows the rules listed below:

- If the cursor is on a response field for a QUERY type message and the
  system is waiting for a response from the user, or the system has displayed
  an ERROR or ABORT message, HP ALLBASE/4GL displays the help
  screen associated with the message.

- If the cursor is on an input field, HP ALLBASE/4GL displays the help
  screen associated with the field.

- If a help screen is not associated with the field or with a message, HP
  ALLBASE/4GL displays the help screen associated with the current
  application screen.

- If there is no help screen associated with the current application screen, HP
  ALLBASE/4GL attempts to build a default help display using the text of
  the description fields on the screen header screen and the dictionary details
  of the current field. If the current screen is a menu, the default help only
  contains the description fields from the screen header. If no screen header is

available, HP ALLBASE/4GL displays a message telling the user that no help is available.

In the HP ALLBASE/4GL run-time environment, the description fields are not available. As a result, HP ALLBASE/4GL cannot build the default help screen.

When HP ALLBASE/4GL displays help screens as a result of an H-*screen_name* action, the system initially displays the help screen *screen_name* regardless of the user's current activity in the application. The user can then proceed to obtain further help using the help screen function keys. These further help screens are the help screens associated with the current field or screen.

Help Screen Function Keys

HP ALLBASE/4GL automatically allocates a set of standard function keys to each help screen. The keys are:

(f1)   `Field Desc` HP ALLBASE/4GL displays this key if the current help screen is associated with an input field that uses a dictionary field specification. If the user presses this key, HP ALLBASE/4GL displays the details of the dictionary field specification.

(f2)   `Display Range` HP ALLBASE/4GL displays this key if the current help screen is associated with an input field that has a validation range defined for it. If the user presses this function key, HP ALLBASE/4GL displays the validation range.

(f3)   `Display Table` HP ALLBASE/4GL displays this key if the current help screen is associated with an input field that has a validation table defined for it. If the user presses this function key, HP ALLBASE/4GL displays the contents of the validation table.

(f4)   `Screen Help` HP ALLBASE/4GL displays this key if the current help screen is associated with an input field on a screen. If the user presses this key, HP ALLBASE/4GL displays the help screen associated with the current application screen. The user cannot return to field level help or message level help after pressing this key.

(f5)   `Previous Help` HP ALLBASE/4GL displays this key if the current help screen is the second or subsequent screen in a sequence of help

screens. If the user presses this key, HP ALLBASE/4GL redisplays the
previous help screen.

[f6]   `More Help` HP ALLBASE/4GL displays this key if there
is a further help screen linked to the current help screen. HP
ALLBASE/4GL displays the additional help screen if the user presses
this key.

[f8]   `Exit Help` When the user presses this key, HP ALLBASE/4GL
restores the original screen display and resumes processing at the point
where the user selected the help function key.

FINAL TRIM SIZE : 7.0 in x 8.5 in

6

Screen Development

HP ALLBASE/4GL Screens

This chapter describes the HP ALLBASE/4GL screen processing logic. It also describes the screens in the screen development menu and the screen painter.

HP ALLBASE/4GL supports three types of screens. They are menus, data screens, and windows.

Menus

HP ALLBASE/4GL menus present the user with a choice of options within the application. When the user chooses an option from a menu, HP ALLBASE/4GL executes the selected option. The following actions can be executed as a result of user selections from a menu:

- Another menu
- Data screen.
- Help screen.
- Function.
- Process.
- Background process.
- Report.
- External program.

When HP ALLBASE/4GL displays a menu, any current process, function, decision table, report, or data screen is terminated immediately.

Data Screens

HP ALLBASE/4GL data screens allow the user to enter data for the application and also allow the application to display information for the user. Data screens consist of input fields, display fields, and literals or titles. Data screens can also have a scroll area for information display.

HP ALLBASE/4GL can overlay a window on a data screen. The window, which must be defined as a screen in itself, can contain data entry and display fields.

Windows

A window is a screen image that HP ALLBASE/4GL overlays on an existing screen to allow display or entry of additional data. HP ALLBASE/4GL can display a number of successive windows on the same screen.

System Items

All HP ALLBASE/4GL screens can contain system items. These are fields that display the contents of certain system or user defined fields. System items can display:

- The current date.
- Communication area fields.
- Master titles.
- Application titles.
- Numeric or alphanumeric constants.
- Variables or calculated items.
- Scratch-pad fields.

Function Keys

All HP ALLBASE/4GL screens can have one or more sets of function keys. These function keys allow the user to select various options or system functions.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Creating Application Screens

To create an application screen:

1. Create the screen header using the screen header screen. The screen header defines the screen name, its type and characteristics. The screen header also defines the name of the first help screen and function key set assigned to the screen.

2. Use the screen painter to lay out the screen image as it will appear to the user. The screen painter is a program that lets you design and modify the layout of a screen freely to suit the needs of your application.

3. For data screens and windows, use the screen fields detail screen to define the attributes of each field on the application screen image you created with the screen painter.

4. Use the function keys screen to define the text and purpose of the function keys associated with the application screen.

You must generate data screens and windows before the application can use them.

Data Screen Processing Logic

The HP ALLBASE/4GL screen processing logic controls:

■ Information display on the screen.

■ Data acceptance from fields on the screen.

■ Execution of screen field functions.

■ Movement of data to and from the screen field buffers and other data fields within HP ALLBASE/4GL.

■ The sequence of screen field processing.

The following subsections describe the screen field processing sequence, movement of data to and from screens, and data validation performed by HP ALLBASE/4GL.

Screen Development   6-3

Screen Buffers

The screen processor maintains an internal data buffer for each field on a data screen while the screen is being displayed. The attributes you define for the field in the dictionary field specification, or the attributes you enter on the screen fields detail screen define the characteristics of each screen field data buffer.

HP ALLBASE/4GL moves data between the screen field buffers and the data movement fields defined for each screen field during the screen processing sequence.

HP ALLBASE/4GL clears the screen field buffers at the start of a process, and when a SCREEN command is executed. The contents of the screen field buffers remain available to commands in the current logic block until a subsequent SCREEN command is executed.

Data Movement Fields

The screen field detail screen allows you to define a number of data movement fields for each screen field. The data movement fields for each screen field are:

- The primary data movement field.

- The default data movement field.

- Up to three other data movement fields.

Figure 6-1 shows the relationship between a screen field, the screen field buffer, and the data movement fields.
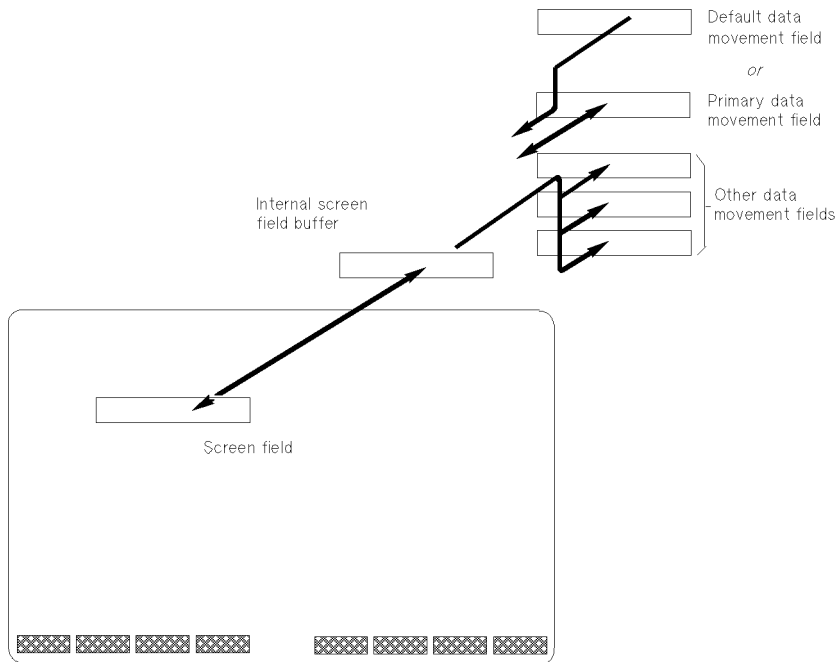
Figure 6-1. Screen Field Buffers

For any screen field, the primary data movement field and the default data movement field are mutually exclusive. You cannot define both a primary data movement field and a default data movement field for the same screen field.

Primary Data Movement File

The screen processor moves the contents of the primary data movement field to the screen buffer at the start of the processing sequence for the field. For input fields, the screen processor moves data from the screen field buffer to the primary data movement field at the end of the processing sequence for the screen field.

The primary data movement field can be any of the following:

- File record field.
- Work area field.

Screen Development   6-5

- Screen field.

- Scratch-pad field.

- Variable.

Default Data Movement Files

The default data movement field allows you to define a default entry for a field that does not have a primary data movement field.

If the screen field buffer is empty or only contains pad characters at the start of the processing sequence for a field, the screen processor moves the contents of the default data movement field into the screen buffer. If the screen field buffer already contains data, the default data movement field is not used. You can only define a default data movement field if you haven't defined a primary data movement field for the screen field. The screen processor cannot move data back into the default data movement field.

The default data movement field can be any of the following:

- Constant (numeric or alphanumeric).

- File record field reference.

- Work area field reference.

- Screen field reference.

- Scratch-pad field reference.

- Variable or calculated item.

Other Data Movement Fields

For input fields, HP ALLBASE/4GL moves data from the screen buffer to the other data movement fields when the screen field processing finishes and the input data has been validated as correct. The screen processor cannot move data from the other data movement fields to the screen buffer. The other data movement fields can be any of the field types specified for the primary data movement field.

Screen Field Functions

The screen field detail screen allows you to associate an HP ALLBASE/4GL function with each screen field. You can specify that the function is executed at the following times:

- After data has been moved from the primary or default data movement field to the screen buffer, but before the contents of the screen buffer are displayed on the screen. That is, a *Prior* function. (Input and display fields.)

- After data has been accepted from the screen and moved to the screen buffer. That is, an *After* function. (Input fields only.)

- Both before data display and after data acceptance. That is, a *Both* function. (Input fields only.)

You can also specify that the function is executed when the contents of the field are displayed by the SHOW command.

By testing the status of the switches *ENTERED and *SHOWING in the function, you can determine if HP ALLBASE/4GL is executing the function before the user enters data in the field, after data entry, or as the result of a SHOW command.

The function associated with a field can use any of the logic commands. (Some of the logic commands may terminate the current screen. For example, if the function associated with a screen field executes a process with a PROCEED command or displays a further screen with a SCREEN command, the current screen will be terminated.)

Switches and Communication Area Fields

The HP ALLBASE/4GL screen and field processing logic uses several communication area fields and switches. The function associated with a screen field can use or alter the contents of these communication area fields, or test the status of any of these switches.

Communication Area Fields

The communication area fields used during screen processing are:

- ***SCREEN** This alphanumeric communication area field contains the name of the current screen.

- ***FIELDNO** This alterable communication area field holds the number of the current screen field. You can alter the value in this field with the MOVE command in a function.

- ***NEWTIE** This alterable communication area field holds the number of the next screen field to be processed. Moving a value into *NEWTIE has the same effect as using the TIE logic command to specify the next field to be processed.

- ***PREVFLD** This read-only communication area field contains the number of the last screen field to be successfully committed. A function associated with a screen field can test the value in *PREVFLD to determine which screen field the user accessed before entering the current field. HP ALLBASE/4GL sets the value in this field to 1 at the start of a screen.

Switches

The switches used by the screen processing logic are:

- ***ENTERED** This switch indicates whether the user has entered data into a screen field. HP ALLBASE/4GL sets *ENTERED *off* before the prior function for the field is executed. *ENTERED is set *on* after the user has entered data into the field and the data has been accepted according to the edit code, minimum number of characters for the field, and validation table or range for the field.

- ***ENDFIELD** This switch indicates whether the input processing for the current field continues or is terminated. If a function called from a function key sets *ENDFIELD *on*, HP ALLBASE/4GL terminates the input field processing for the current field when the function exits.

  You can set *ENDFIELD *on* in a function called from a function key or a prior function on an input field. You can also use a TIE command in the function to specify the next field on the screen to be processed. If you do set *ENDFIELD *on*, and the user has changed the entry in the current screen field, HP ALLBASE/4GL restores the original contents of the field before it

starts processing the next field on the screen, or the field specified by the TIE command.

You can also set *ENDFIELD *on* in a function called from the function key, or in a second function called from that function.

■ ***SHOWING** This switch indicates whether the contents of the field are being displayed under the control of a logic SHOW command. The switch is set *on* when the SHOW command is active, and restored to its previous condition at the end of the SHOW command.

Screen Processing Sequence

HP ALLBASE/4GL processes screens at two different levels. These are the field level, and the screen level.

A number of factors influence the sequence of actions initiated by the screen and field processing logic. These factors are:

■ The SCREEN logic command and the D- action from a menu, decision table, or function key.

■ The function associated with each screen field.

■ Field and screen commit actions initiated by the user.

■ The TIE logic command.

When HP ALLBASE/4GL starts processing a data screen, the flow of control in the application is determined by the screen and the logic associated with the screen. Displaying a screen terminates the current screen if one is active, and clears the screen field buffers. The data screen remains active until the user or the logic associated with the screen initiates a screen commit action.

When a screen is committed, HP ALLBASE/4GL resumes running the application in one of three possible ways, depending how the screen was invoked.

■ If the screen was displayed by a SCREEN command in a logic block, control returns to the command immediately following the SCREEN command.

■ If the screen was displayed directly from a menu, control returns to the menu.

■ If you displayed the screen from a SCREEN command in a function on a field on another screen, or from a function called from a function key on another screen, HP ALLBASE/4GL firstly returns to the function and executes the remaining commands in the function logic block. Since the original screen is terminated when the second screen is displayed, control then returns to the logic block or menu that displayed the original screen.

Note   In general, you should avoid calling further data screens while a data screen is active.

☞   If you need to modify the current screen, use a window in preference to calling a further screen.

The logic SCREEN command or the D- action initiates processing at the screen level. Figure 6-2 shows the screen processing logic in diagrammatic form.

Once HP ALLBASE/4GL starts processing a screen, processing continues on a field by field basis.

FINAL TRIM SIZE : 7.0 in x 8.5 in

```
                    ┌─────────────────┐
                    │  Start of Screen │
                    └─────────────────┘
                             │
                             ▼
              ┌─────────────┐        ┌──────────────┐
              │Display Field│        │    Go to     │
              └─────────────┘        │  Next Field  │
                    │                └──────────────┘
                    ▼                        ▲
                  ╱ *FIELDNO ╲    Yes         │
                 ╱   Set?     ╲──────────────▶│
                 ╲            ╱                │
                  ╲        ╱                   │
                    │ No                       │
                    ▼                          │
       Yes        ╱ Display ╲                  │
      ◀──────────╱  Only?    ╲                 │
                 ╲           ╱                  │
                  ╲        ╱                    │
                    │ No                        │
                    ▼                           │
              ┌─────────────┐                   │
              │ Input Field │                   │
              └─────────────┘                   │
                    │                           │
                    ▼                           │
                  ╱ *FIELDNO ╲   Yes            │
                 ╱   Set?     ╲─────────────────▶
                 ╲            ╱                  │
                  ╲        ╱                     │
                    │ No                         │
                    ▼                            │
                  ╱ New TIE ╲    Yes             │
                 ╱  Field?   ╲──────────────────▶
                 ╲           ╱                   │
                  ╲        ╱                      │
                    │ No                          │
                    ▼                             │
                  ╱ End of ╲     No               │
                 ╱ Screen?  ╲────────────────────▶
                 ╲          ╱
                  ╲       ╱
                    │ Yes
                    ▼
            ┌───────────────┐
            │ End of Screen │
            └───────────────┘
```
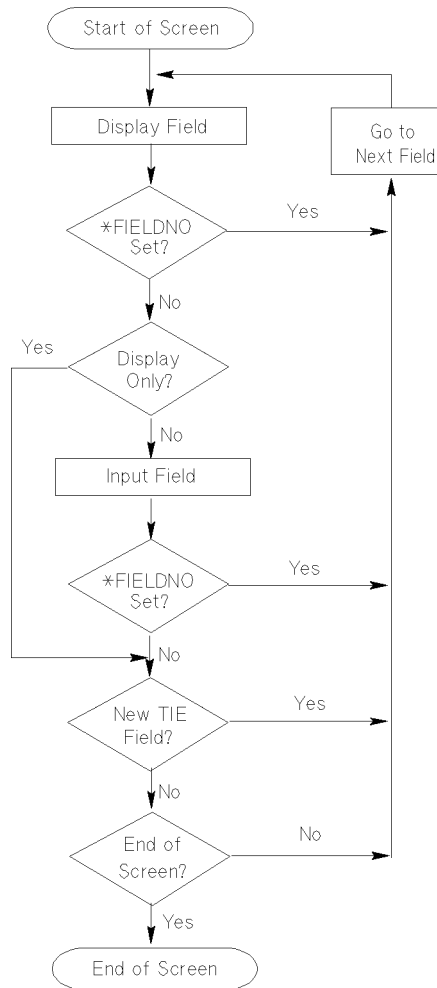
Figure 6-2. Data Screen Processing Logic

Screen and Field Commit Actions

Screen and field commit actions initiated by the user control the screen processing sequence.

A field commit action is any action (such as pressing [Return] or the [Tab] key) that moves the cursor off the current field. Field commit actions by the user control data acceptance and validation on a field by field basis.

Screen Development   6-11

A screen commit action is initiated by the internal I-COMMIT routine. This action is usually initiated from a function key on the screen. (Refer to Function Keys Screen, page 6-33.) A screen commit action can also be initiated by an EXIT *COMMIT logic command in an after function on a screen field, or in a function called from a function key.

The screen commit action terminates processing of the screen.

Field Display Logic

When HP ALLBASE/4GL processes a screen, it immediately executes the field display logic for the first numbered field on the screen. Figure 6-3 shows the field display logic.

HP ALLBASE/4GL first checks for the existence of a primary data movement field. If this field exists, the screen processor moves its contents to the screen field buffer and then performs any necessary justification and filling with the pad character. The screen processing logic only uses the contents of the default data movement field if there is no primary data movement field defined for the screen field **and** the screen field buffer is empty, or contains only pad characters.

After performing the data movement, HP ALLBASE/4GL sets *ENTERED *off*, and then executes the prior function if one is defined for the field.

After the prior function has been executed, the screen processor displays the contents of the screen buffer on the screen.

At this point, the screen processor tests the contents of *FIELDNO. (See Figure 6-2, step 2.) If the prior function has changed the value in *FIELDNO, HP ALLBASE/4GL immediately starts processing the field identified by the new value in *FIELDNO. Changing *FIELDNO in the prior function allows you to conditionally bypass the field or a group of fields.

If *FIELDNO has not been changed, HP ALLBASE/4GL checks whether the field is a display-only field or an input field (step 3 in Figure 6-2).

For display-only fields, HP ALLBASE/4GL then processes the next field in the sequence for the screen unless you have changed the value in *NEWTIE, or used the TIE logic command to specify the next field to be processed. (See Figure 6-2 step 6.) The processing of data input fields is described below.

6-12   Screen Development
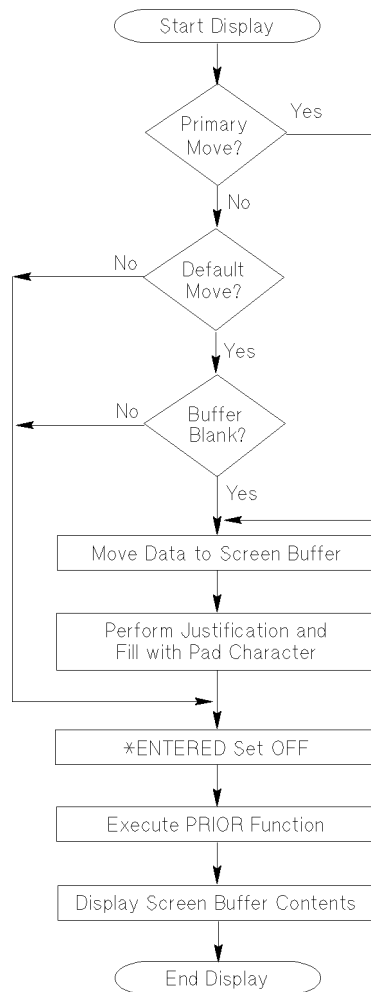
Figure 6-3. Field Display Logic

Input Field Logic

Before input processing for an input field starts, HP ALLBASE/4GL tests the status of the switch *ENDFIELD. If *ENDFIELD is *on*, the contents of the field are restored to the state that existed at the end of display processing for the field. Display processing then starts for the next screen field. You can only set the switch *ENDFIELD *on* in a function called from a function key or a

prior function on an input field. This function can contain a TIE command to specify the number of the next field to be processed.

If *ENDFIELD is *off*, HP ALLBASE/4GL starts the input processing for the field.
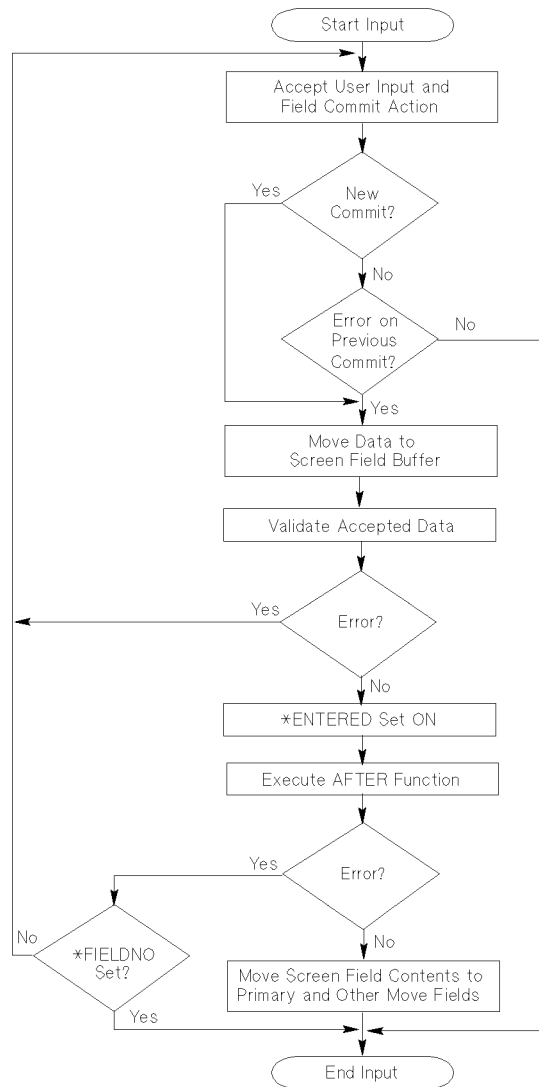
Figure 6-4. Input Field Logic

6-14   Screen Development

Figure 6-4 shows the input field processing logic. This part of the screen processing logic corresponds to steps 4 and 5 in Figure 6-2.

Input field processing starts when the user initiates a commit action for the field. A field commit action is any action that moves the cursor away from the field, such as pressing the (Return) key or the (Tab) key. HP ALLBASE/4GL also initiates a field commit action if the user presses the `Commit Data` function key and the contents of the current field have not been committed. In this case, HP ALLBASE/4GL processes the current field and then terminates the screen.

The current field is also committed if a function called from a function key changes the value of *FIELDNO. In this case, HP ALLBASE/4GL executes the function called from the function key, and then immediately commits the current screen field. After the field commit, HP ALLBASE/4GL processes the field specified by the new value of *FIELDNO.

After a commit action, HP ALLBASE/4GL checks if the action represents a new commit for the field. A commit action is considered to be a new commit if:

■ The data in the field has changed since the last commit action for the field,

   or

■ A different numbered input field has been processed since the last commit action for this field.

If the current commit for the field is not a new commit action, processing only proceeds if the previous commit action for the field generated an error.

When HP ALLBASE/4GL accepts a field commit, it tests the data against the edit code, minimum and maximum number of characters, number of decimal places, and validation range or table (if defined) for the field. If no error is detected, HP ALLBASE/4GL sets *ENTERED *on* and executes the after function for the field if one is specified.

Error Processing

If the user's input doesn't satisfy the edit code, number of decimal places, or minimum number of characters for the field, HP ALLBASE/4GL displays an internally defined error message. In this case, the cursor remains on the current field to allow the user to reenter the data.

Screen Development   6-15

If the user's input does not fall within a validation range or table associated with the field, HP ALLBASE/4GL displays the message associated with the table or range. The system actions depend on the message type. If the message is an error type message, the cursor remains on the current field to allow the user to reenter data. For other message types, HP ALLBASE/4GL displays the message, but the cursor does not remain on the current field. If the message is an abort type message, the application terminates.

If you haven't defined a message for a validation table or range associated with the field, and the user's input doesn't fall within the table or range, the screen processor displays a system defined error message. In this case, the cursor remains on the current field.

After Entry Functions

HP ALLBASE/4GL executes the after function after the automatic data validation has been completed, and data has been accepted into the screen field buffer and the switch *ENTERED has been set *on*. Note however, that HP ALLBASE/4GL has *not* yet moved the screen field data to the data movement fields.

You can use the after function to perform additional validation of the input data. You can also use the after function to move a value to *FIELDNO.

If the function displays an error type message as the result of any of the additional validation tests, HP ALLBASE/4GL tests the value in *FIELDNO. If the after entry function has not changed the value in *FIELDNO, the cursor remains on the current field to allow the user to correct the data.

If the function has changed *FIELDNO, HP ALLBASE/4GL restores the contents of the field to the state that existed at the end of the display processing for the field, and then starts processing the field identified by the contents of *FIELDNO.

Typically, you would only change the value of *FIELDNO in an after function if you wanted to return the user to an earlier screen field as the result of an error detected during data validation in the function.

If no errors are generated in the after function, the screen processor moves the contents of the screen buffer to the primary and other data movement fields (if defined). Screen processing then continues at the next field in the tabbing sequence for the screen unless you have moved a value to *FIELDNO

6-16   Screen Development

in the after function, or used the TIE logic command to change the value in
*NEWTIE. (See Figure 6-2, steps 5 and 6.) If you have changed either of these
values, processing continues at the field identified by the value in *FIELDNO
or *NEWTIE.

Note

If you want to continue processing at a field other than the
next field in the screen sequence, use the TIE command in
preference to changing the value of *FIELDNO in the after
function. The TIE command allows you to reference the next
field by name rather than by field number.

Data Validation Summary

The following paragraphs summarize the data validation that is performed by
HP ALLBASE/4GL during screen field processing.

Automatic Data Validation

HP ALLBASE/4GL performs automatic data validation on user input after
the user commits a field. This validation occurs before the data is accepted
into the screen field buffer, and before the switch *ENTERED is set *on*. The
automatic validation tests are shown in the following table.

Test	Action on Failure
Edit code	System defined error message
Number of decimal places	System defined error message
Number of characters $\geq$ minimum	System defined error message
Validation range	Developer defined message, or system defined default error message
Validation table	Developer defined message, or system defined default error message

Additional Validation

HP ALLBASE/4GL allows you to perform additional data validation in the after function on an input field. The function can perform any validation tests you require, and display a message if the data fails validation. The system action depends on the message type code.

Automatic Data Movement

The following table summarizes the automatic data movement that HP ALLBASE/4GL performs during screen field processing.

Timing	Data Movement
On display (all fields)	`Screen field buffer` $\Leftarrow$ Primary movement field or `Screen field buffer` $\Leftarrow$ Default movement field (If screen field buffer is empty)
After Commit (input fields)	`Screen field buffer` $\Rightarrow$ Primary movement field and `Screen field buffer` $\Rightarrow$ Other movement fields (If defined)

Commit Data Actions

The I-COMMIT internal routine terminates all current screen processing.

Typically you would use a `Commit Data` function key on each application data screen to invoke the I-COMMIT internal routine when the user has finished entering data. Refer to the function keys screen on page 6-33 for further details.

You can also force a screen commit by using the *COMMIT argument with the logic EXIT command in an after function on an input field, or in a function called from a function key. (Refer to chapter 12, Logic Commands.)

HP ALLBASE/4GL executes the I-COMMIT internal routine immediately if the current screen field has been processed. If the current field has not been processed, HP ALLBASE/4GL processes this field and then executes the I-COMMIT internal routine.

Following the I-COMMIT internal routine, HP ALLBASE/4GL resumes running the application in one of three possible ways, depending on how the screen was displayed.

■ If the screen was displayed by a SCREEN command in a logic block, control returns to the command immediately following the SCREEN command.

■ If the screen was displayed directly from a menu, control returns to the menu.

■ If the screen was displayed by a SCREEN command in a function on a field on another screen, or a function called from a function key on another screen, HP ALLBASE/4GL firstly returns to the function and executes the remaining commands in the function logic block. Since the original screen is terminated when the second screen is displayed, control returns to the logic block or menu that displayed the original screen.

SHOW Command Processing

The logic SHOW command displays the contents of a single screen field, or a range of screen fields on the screen. Figures 6-5 and 6-6 show the logic of the SHOW command.
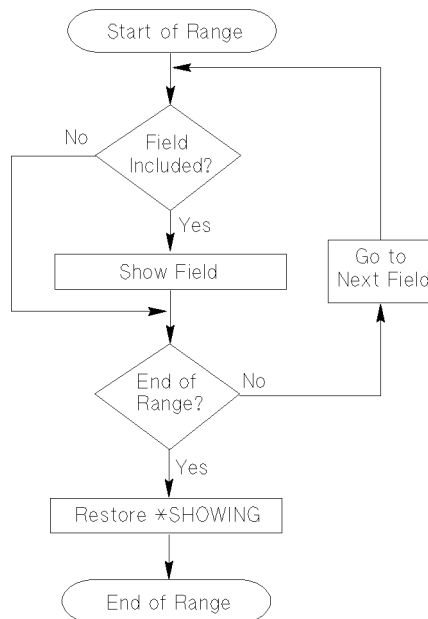
Figure 6-5. SHOW Command Logic

Figure 6-5 shows the processing that takes place when HP ALLBASE/4GL
executes the SHOW command. The SHOW command operates on a field by
field basis over the range of fields specified in the command parameters. (Refer
to chapter 12, Logic Commands.)

For each field, HP ALLBASE/4GL checks that the field is specified as included
in SHOW on the field details screen. If the field is included in SHOW, HP
ALLBASE/4GL executes the logic shown in Figure 6-6. If the field is not
specified as a show field, processing continues at the next field in the range.

FINAL TRIM SIZE : 7.0 in x 8.5 in
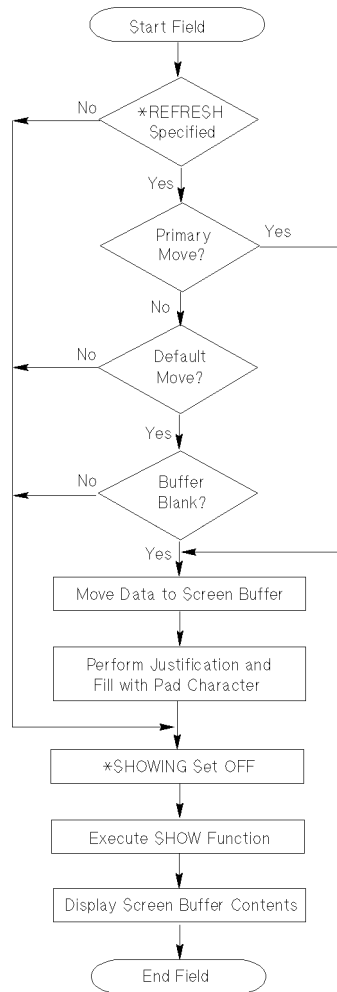
Figure 6-6. SHOW Command Field Processing Logic

If you specify the *REFRESH option with the SHOW command, the system moves data from the primary (or default) data movement field to the screen field buffer. HP ALLBASE/4GL performs any necessary justification and filling with the pad character and then sets the switch *SHOWING *on*. If a SHOW function is specified for the field, HP ALLBASE/4GL executes the function and then displays the contents of the screen buffer on the screen.

If the *REFRESH argument is not specified, HP ALLBASE/4GL does not move data from the primary or default data movement fields. HP ALLBASE/4GL sets *SHOWING *on*, executes any show function for the field, and then displays the existing contents of the screen buffer on the screen.

At the end of the field range, HP ALLBASE/4GL sets *SHOWING to the condition that existed before the SHOW command was executed.

SHOW Functions

During a SHOW command, HP ALLBASE/4GL executes the function for a screen field if the *Perform on SHOW* field on the screen field details screen is set to Y. The function can test the status of the switch *SHOWING to determine when it is being executed.

The show function cannot execute a SCREEN command. HP ALLBASE/4GL aborts the application if it encounters a SCREEN command while a SHOW command is active.

Be careful about using a SHOW command in a screen field function, and also specifying that the function is performed on show. This can cause the application to become locked in an infinite loop.

It is possible for a screen field function to change the number of fields on the screen by executing a WINDOW command. If this occurs, the range of fields for the SHOW command is truncated.

If the number of fields on the screen changes due to a WINDOW command being executed by a function on a field on the main screen, SHOW processing continues up to the last field on the main screen. If the number of fields changes while HP ALLBASE/4GL is showing fields on a window, SHOW processing terminates immediately.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Generating Screens

You must generate all data screens and windows before they can be used in an application.

You can use the `Generate Screen` function keys to generate screens individually as you define them. You can also generate all the screens in an application by selecting the *Screens* option on the application generates menu selected from the main menu.

Note	You must generate a data screen even if the screen does not
☞	contain any active data input or display fields.

Menus do not require generation.

Screen Development Screens

The following subsections describe the screens in the screen development menu.

Each subsection describes screens under the following subheadings:

■ Screen Image.

■ Field Descriptions.

■ Function Keys.

■ Additional Information.

Screen Development Menu

The screen development menu allows you to access the HP ALLBASE/4GL screen development facilities.

Screen Image

```
                          HP ALLBASE/4GL Developer
         Dictionary   Screens   Logic   Reports   Module Builder   Utilities   Generate   Test

                    ┌─────────────────────┐
                    │ Header              │
                    │ Painter             │
                    │ Details             │
                    │ Function Keys       │
                    │                     │
                    │ Catalog Display     │
                    └─────────────────────┘




  Dict.    Utility   Logic    Reports    5* 15   System   Activate   Help   Previous
  Menu     Menu      Menu     Menu               Keys     Item              Menu
```
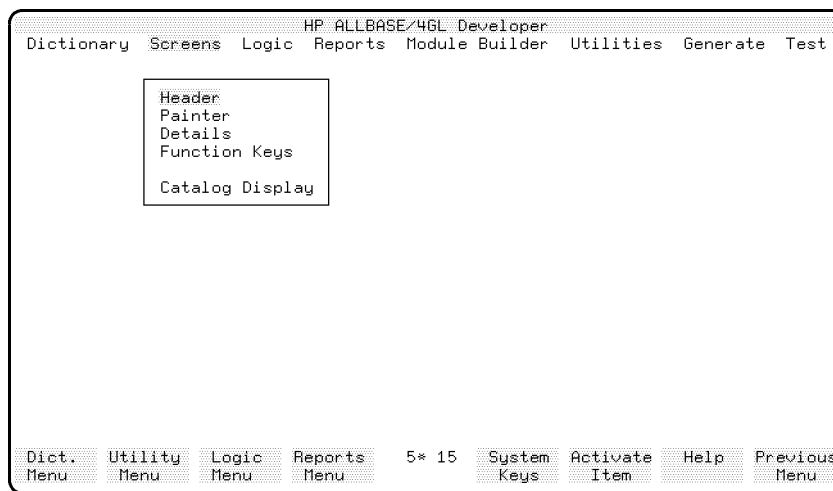
Menu Selections

Header   This selection displays the screen header screen which allows you to define the name and characteristics of an application screen.

Painter   This selection runs the screen painter. This is a program that allows you to interactively paint application screens as the end user will see them. Refer to the section on "Using the Screen Painter" for details.

Details   This selection displays the screen fields detail screen, which allows you to define the attributes of each field on an application data screen. You don't need to complete this screen for menus.

Function Keys   This selection displays the function keys screen which allows you to define the purpose and text of the function keys associated with the application screen.

Catalog Display   This selection displays the catalog display screen, allowing you to display the names of existing application components. Refer to chapter 13 for details of this screen.

Function Keys

(f1)   `Dict. Menu`   Cancels the current screen and displays the dictionary menu.

(f2)   `Utility Menu`   Cancels the current screen and displays the utilities menu.

(f3)   `Logic Menu`   Cancels the current screen and displays the logic menu.

(f4)   `Reports Menu`   Cancels the current screen and displays the reports menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Screen Header Screen

This screen allows you to name a screen and specify the attributes of the screen.

Screen Image

```
 ┌────────────────────────────────────────────────────────────────┐
 │ Developer                    Screen Header              screen_header │
 │                                                                  │
 │          Screen      ██████████████       Secured    (Y/N)       │
 │                                                                  │
 │          Screen Type               (M/D/W)                       │
 │          Automatic Numbering       (Y/N)                         │
 │          Clear Fields on Commit    (I/D/B/N)                     │
 │          Function Keys                                           │
 │          Screen Help Name                                        │
 │                                                                  │
 │          Scroll:        Top line       Bottom Line      Direction   (U/D) │
 │                                                                  │
 │                                                                  │
 │       Window Starting Line Number                               │
 │                                                                  │
 │       Description                                               │
 │                                                                  │
 │                                                                  │
 │                                                                  │
 │       Last Modification:       Date              Time           │
 │                                                                  │
 │ Function Screen    Field    Generate   3* 24   System   Commit   Help  Previous │
 │  Keys   Painter   Details    Screen             Keys     Data          Menu │
 └────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Screen   Enter the name of the screen (up to 16 characters).

Secured   If you wish to protect this screen from unauthorized modification, enter Y; otherwise enter N.

If you enter Y, this screen can only be modified by a developer who has signed on to the application under the correct development security code. The entry in this field defaults to N.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Screen Type   The entry in this field defines the type of the screen, as follows:

> M   menu
>
> D   data screen
>
> W   window screen

Don't attempt to change the entry in this field after you have painted the screen image with the screen painter. If you do, you won't be able to access the screen with the screen painter.

Automatic Numbering   This field is only active as an input field for data screens and window screens. For menus, this field defaults to Y.

If you enter Y, the screen painter assigns numbers to the screen fields automatically. The numbers start with the first field, numbering across each line, then down the screen.

If you enter N, you must supply the field numbers yourself when you're painting the screen. The field numbers must start at number 1 and be consecutive, but may be placed anywhere on the screen.

The screen painter assigns or checks the field numbers.

Clear Fields on Commit   This field is only active for data screens and window screens.

The entry in this field determines which screen fields, if any, are cleared automatically when the user presses the `Commit Data` function key.

Regardless of the entry in this field, HP ALLBASE/4GL does not clear the fields until after the next SCREEN logic command is executed. This means that data remains in the screen buffer after the user presses the `Commit Data` function key.

The entry in this field controls the automatic clearance of field types, and works in conjunction with the *Clear on COMMIT* field on the screen field details screen. If the entry on this screen specifies that the field type is cleared, or the entry on the field details screen specifies that the specific field is cleared, HP ALLBASE/4GL will clear the field when the user presses the `Commit Data` function key.

If you don't want to clear a field when the user presses the `Commit Data` function key, you must specify that the field type is not cleared on this screen, and also specify that the specific field is not cleared on the field details screen.

Enter one of the following codes in this field:

I Input fields only are cleared

D Display fields only are
cleared

B Input and display fields are
cleared

N No fields are cleared

Function Keys

Enter the name of the primary set of function keys to be displayed with this screen. The function key set need not exist when you complete this screen.

Screen Help Name   Enter the name of the first help screen to be displayed by HP ALLBASE/4GL if the user requests general help information about the current screen.

Scroll: Top line   (Data screens only.) If the screen is to contain a scroll area, enter the line number (between 1 and 22) for the top line of the scroll area. This line is included in the scroll area. Refer to Additional Information.

Scroll: Bottom line   (Data screens only.) Enter the line number (between 1 and 22) of the bottom line of the scroll area. This line is included in the scroll area. Refer to Additional Information.

Direction   (Data screens only.) If scrolling is used, this field defines the scroll direction. Enter U for upwards or D for downwards scrolling.

Content   (Data screens only.) Optional. These two fields may contain a list of data items to be scrolled. The format of this list is the same as that required for the SCROLL and DISPLAY logic commands. (Refer to chapter 12 for details of the SCROLL and DISPLAY logic commands.) You don't need to use the command name SCROLL in this area.

HP ALLBASE/4GL scrolls the items listed in these fields each time the user presses the `Commit Data` function key.

6-28   Screen Development

You can use the SCROLL or DISPLAY commands in a function logic block to display other data in this scroll area.

Window Starting Line Number   (Data screens only.) If the screen is to contain a window, this field specifies the starting line of the window display area. The starting line is included in the window area.

HP ALLBASE/4GL displays the window from the specified line number down to the bottom of the screen. Refer to chapter 12 for further details of the WINDOW command.

Description   Documentation fields. Enter a description of the screen for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

Function Keys

f1   `Function Keys` Cancels the current screen and displays the function keys screen.

f2   `Screen Painter` Cancels the current screen and starts the screen painter program.

f3   `Field Details` Cancels the current screen and displays the screen field details screen.

f4   `Generate Screen` Generates the screen.

Additional Information

You cannot specify a scroll area or window position for a window. A window may, however, reference and use the defined scroll area of the data screen on which it is displayed.

HP ALLBASE/4GL will allow you to define a scroll area that overlaps data input or output fields on the screen. However, before HP ALLBASE/4GL scrolls any data in this area, the scroll area is cleared of all text, including screen literals and special text items. In general, the scroll area of a screen should not contain any data fields or text items.

Menus cannot have a scroll area or window.

You must define a header for an application screen using this screen before you can paint the screen image with the screen painter or define the attributes of its fields.

Screen Field Details Screen

This screen allows you to define the attributes, edit code, and functions for each input and display field on an application data screen or window. You must use the screen painter to create the fields on the application screen before you can use this screen to define the field details.

Screen Image

```
 Developer                  Screen Field Details              screen_fields

         Screen  ██████████████         Sequence Number       Secured

    DATA MOVEMENT              Field Spec. Name
         Primary                                               Data<-->Screen
         Default                                               Data -->Screen
         Other 1                                               Data<-- Screen
               2                                                     "
               3                                                     "

    Input/Display   (I/D)                  Required Field      (Y/N)
        Help name                        Clear on COMMIT       (Y/N)
  Include in SHOW   (Y/N)          Automatic Flow to Next      (Y/N)

        Function                        Prior/After/Both       (P/A/B)
  Perform On SHOW   (Y/N)         After Function if Blank       (Y/N)

       Edit Code   (X/A/U/K/N/S/Q/D/T)       Justification     (L/R/C/N)
    Field Length    Min         Max          Decimal Places
           Range                             Pad Character
           Table                             Blank When Zero   (Y/N)

 Screen  Screen  Function Generate   3* 19   System  Commit   Help  Previous
 Header  Painter  Keys    Screen             Keys    Data            Menu
```

Field Descriptions

Screen   Enter the name of a screen defined on the screen header screen and painted with the screen painter.

Sequence Number   Enter a valid field number for the screen. If you enter a number that is higher than the maximum number of fields on the screen, HP

6-30   Screen Development

ALLBASE/4GL displays an error message. The highest valid sequence number for a screen is the highest field number assigned for the screen in the screen painter.

Secured   Display-only field indicating the status of the secured field on the screen header for this screen.

Field Spec. Name   Enter the name of a dictionary field specification, or leave this field blank.

If you enter a dictionary field specification name, HP ALLBASE/4GL links this screen field to the dictionary name.

If you used a dictionary field specification name to define the length of the field in the screen painter, HP ALLBASE/4GL displays the field specification name in this field.

If the *Field Spec. Name* field contains a dictionary field specification name, HP ALLBASE/4GL retrieves the specification from the dictionary. Information from the dictionary is displayed in the fields *Edit Code* to *Blank When Zero* at the bottom of this screen. While a field specification name is displayed in the *Field Spec. Name* field, you cannot change the contents of these fields.

You can enter a field specification name to retrieve a template for a field. If you delete the field specification name after HP ALLBASE/4GL has retrieved the template, you can alter the contents of any of the fields at the bottom of this screen.

Note	If you leave the *Field Spec. Name* field blank, you will not be able to reference this screen field by name. You will have to reference it by relative or absolute field number.

DATA MOVEMENT This set of fields defines the data movement fields for the screen field.

- **Primary.** Enter the name of the primary data movement field for the screen field. For all fields, the screen processor moves data from this field for display on the screen. For input fields, the screen processor moves input data from the screen field to this field. The screen processor also displays data from this field on the screen when you use the SHOW *REFRESH command. Entries for the primary data movement field can be any of the following:

  File record field reference.
  Work area field reference.
  Screen field reference.
  Scratch-pad field reference.
  Variable.

  For the second and subsequent fields on a screen, HP ALLBASE/4GL constructs a primary data movement field identifier using the dictionary field specification name for the field and the file name used in the primary data movement field identifier for the first screen field. You can type over the data movement identifier constructed by HP ALLBASE/4GL. A primary data movement identifier is not constructed for screen fields that don't use a dictionary field specification name.

- **Default.** This field is only active if you haven't defined a primary data movement field for the screen field. Enter the name of the default data movement field. The screen processor only moves data from the default movement field to the screen buffer if you haven't specified a primary data movement field **and** the screen buffer is blank. Data from the screen buffer cannot be moved into the default data movement field.

  Entries for the default data movement field can be any of the following:

  Numeric or alphanumeric constant.
  File record field reference.
  Work area field reference.
  Screen field reference.
  Scratch-pad field reference.
  Variable or calculated item.

- **Other.** Enter the names of up to three other data movement fields. For input screen fields, the screen processor moves data from the screen buffer into these fields when it finishes processing the field. The screen processor cannot move data from the other data movement fields to the screen field buffer.

6-32   Screen Development

Entries for the other data movement fields can be any of the following:

File record field reference.
Work area field reference.
Screen field reference.
Scratch-pad field reference.
Variable.

Note

If you specify a screen field reference for a data movement field, the screen name qualifier is optional. If used, the qualifier must be the name of the current screen.

Input/Display   Enter I if the field is an input field. Enter D if the field is a display-only field.

Required Field   Enter Y if the user must enter data in this field. Enter N if data entry is optional.

Help name   Optional. Enter the name of the help screen defined for this screen field. If you have entered a field specification name in the *Field Spec. Name* field, the entry in this field defaults to the help screen name defined for the field specification. You can type over this default if you wish.

Clear on COMMIT   Enter Y to specify that the field is cleared when the user commits the data for the field. Enter N to specify that the field is not cleared. The entry in this field operates in conjunction with the entry in the *Clear Fields on COMMIT* field on the screen header screen. The field will always be cleared on commit unless the entry in this field is N, **and** the entry in the screen header screen specified that the field type is not cleared.

Include in SHOW   Enter Y or N. If you enter Y, HP ALLBASE/4GL displays the contents of the screen field buffer on the screen when a SHOW command controls the field. If you enter N, HP ALLBASE/4GL does not refresh the screen display for this field when a SHOW command is executed.

Automatic Flow to Next   Automatic flow-to-next operation is not available under the MPE/iX operating system. An N will always appear in this field, requiring the user to initiate a commit action to terminate data entry for the field.

Screen Development   6-33

A commit action is any action that moves the cursor away from the field, such as pressing the ⎡Return⎤ or ⎡Tab⎤ key. When HP ALLBASE/4GL receives the commit action, it processes the data entry for the field. If no errors are detected, HP ALLBASE/4GL then moves to the next field in the sequence for the screen.

Function   Enter the name of the function to be linked to this field.

The function can execute any logic commands. However, if this function initiates a process, HP ALLBASE/4GL terminates the current screen immediately, and does not return to this screen at the end of the process.

If the function displays another screen, HP ALLBASE/4GL discards the contents of the current screen buffers. The contents of the main, default and other data movement fields remain unchanged unless the second screen or one of its functions changes them.

Prior/After/Both   The entry in this field specifies when HP ALLBASE/4GL executes the screen field function.

- Enter P to execute the function before data is displayed on the screen. (That is, a *prior* function.)

- Enter A to execute the function after data entry in the field. (This function is only executed on an input field.)

- Enter B to execute the function both before and after data entry in the field.

By testing the status of the switch *ENTERED in the function, you can test when the function is being executed. *ENTERED is *off* before the user enters data in the field, and *on* after data entry.

For display-only fields, HP ALLBASE/4GL only executes the function if you enter P or B in this field.

Perform On SHOW   Enter Y in this field if you want to execute the field function when the field contents are being displayed by a SHOW command; otherwise enter N.

By testing the status of the switch *SHOWING, you can test when the function is being executed. The switch *SHOWING is *on* when the contents of the field are being displayed by a SHOW command.

6-34   Screen Development

After Function if Blank Enter `Y` to execute the after function for the field even if the field is blank. Enter `N` to suppress execution of the after function if the field is blank.

Edit Fields The remaining fields on this screen define the type of data accepted for this field. If you have entered a dictionary field specification name in the *Field Spec. Name* field on this screen, these fields are display-only fields and cannot be altered.

If the *Field Spec. Name* field is blank, you can alter the entries in these fields. The meaning of the fields and the acceptable entries in them are the same as those for the field specifications screen in the dictionary menu. (Refer to chapter 5.)

Function Keys

`f1`	`Screen Header`	Cancels the current screen and displays the screen header screen.
`f2`	`Screen Painter`	Cancels the current screen and runs the screen painter program.
`f3`	`Function Keys`	Cancels the current screen and displays the function keys screen.
`f4`	`Generate Screen`	Generates the screen.

Additional Information

If you specify a field as not required and the minimum length is greater than one character, the user must either leave the field blank or enter at least the minimum number of characters.

You must generate a data screen or window screen before it can be used at run-time. Generation is necessary even if the screen doesn't contain any active input or display fields.

If you have entered the name of a defined dictionary field specification in the *Field Spec. Name* field, HP ALLBASE/4GL retrieves the field descriptions from the dictionary whenever you generate the screen.

Screen Development   6-35

Function Keys Screen

This screen allows you to name a function key set and define the actions performed by the keys.

Screen Image

```
Developer                        Function Keys                function_keys

        Function Keys  ████████████████

        Function Key   Action                    Label Text          Switch
                                                  Line 1   Line 2     Number

                f1
                f2
                f3
                f4
                f5
                f6
                f7
                f8




   Screen   Screen   Field           5* 21   System   Commit   Help   Previous
   Header   Painter  Details                 Keys     Data            Menu
```

Field Descriptions

Function Keys

Enter the name of the function key set (up to 16 characters). If you haven't previously defined the name, HP ALLBASE/4GL assumes you're defining a new function key set and displays the default function key set. (Refer to Additional Information.)

Action   Enter the code of the action to be carried out by HP ALLBASE/4GL when the user presses the function key. The available action codes and their meanings are:

Code	Action Type
B-	Background process
D-	Data screen or menu
F-	Function logic block
H-	Help screen
I-	Internal action
K-	Call another function key set
P-	Process logic block
R-	Report
X-	External program
blank	No action performed

To specify an action, enter the action code followed by a hyphen and the action name. For example, *D-emp_details* displays the screen *emp_details*.

Label Text   Enter the screen label text for the function key. The label can use two lines of text, each containing up to eight characters.

Switch Number   Optional. Enter a number from 1 to 8, representing one of the HP ALLBASE/4GL user switches.

If you enter a switch number in this field, HP ALLBASE/4GL displays an asterisk in the function key label on the application screen at run-time while the switch is *on*. If the key label has one line of text, the asterisk is placed in column 8 in the top line of text. If the label has two lines of text, the asterisk is placed in the bottom line of the label. The asterisk is not displayed while the switch is *off*.

Typically you would use this feature to provide the end user with a visual indication of the status of a function. For example, you could use a function associated with the function key to toggle a particular operating mode (such as an add mode, modify mode, or delete mode) on or off. By setting a switch *on*

Screen Development   6-37

or *off* in the function, you can provide the end user with a visual indication of the current operating mode.

Function Keys

(f1)   `Screen Header`   Cancels the current screen and displays the screen header screen.

(f2)   `Screen Painter`   Cancels the current screen and runs the screen painter program.

(f3)   `Field Details`   Cancels the current screen and displays the screen fields detail screen.

Additional Information

Function Key Display Logic

HP ALLBASE/4GL displays function key sets on application screens according to the following rules:

■ If the current screen has a set of function keys defined for it, these keys are displayed when the screen is executed.

■ If the current screen has a window displayed on it, and the window has its own set of function keys, the function keys for the window are displayed. If the window does not have its own set of function keys, the function keys for the main screen (if defined) are displayed.

■ If the current screen does not have a set of function keys defined for it, HP ALLBASE/4GL displays the function key set *default_keys* from the local application, or if these don't exist, the function key set *default_keys* from the administrator application.

If HP ALLBASE/4GL cannot find a set of function keys defined for a screen or window, it searches for a set of function keys with the same name in the administrator application. If no other keys can be found, HP ALLBASE/4GL displays the *default_keys* function key set from the administrator application.

If HP ALLBASE/4GL cannot read a set of function keys called by a K-action from a function key, it searches for a set of function keys with the same name in the administrator application. If no other keys can be found,

HP ALLBASE/4GL displays the *default_keys* function keys from the administrator application.

HP ALLBASE/4GL Internal Routines

You can use function keys on application screens to initiate any HP ALLBASE/4GL internal routine. To assign an internal routine to an application screen function key, enter I-followed by the internal routine name in the action field on the function keys definition screen. The internal routines are described in the following paragraphs.

I-BACKTAB   The I-BACKTAB internal routine moves the screen cursor as if the ⎡Shift⎤ + ⎡Tab⎤ keys and then the ⎡Return⎤ key sequence have been pressed. This provides faster reverse tabbing for end-users when they use one of the function keys to call I-Backtab.

I-CLEAR   This routine clears all input and display-only fields on the current screen.

I-COMMIT   The I-COMMIT internal routine terminates all current screen processing. Typically, you provide the user with a `Commit Data` function key to invoke the I-COMMIT internal routine. The user can press the function key to indicate that data entry on the screen has been completed. By default, HP ALLBASE/4GL assigns this internal routine to function key number 6 in a new set of function keys.

On a data screen or window, HP ALLBASE/4GL executes the I-COMMIT internal routine immediately if the current screen field has been processed. If the current field has not been processed, HP ALLBASE/4GL processes this field and then executes the I-COMMIT internal routine.

After the I-COMMIT internal routine, HP ALLBASE/4GL resumes running the application in one of three possible ways, depending on how the screen was displayed.

■ If the screen was displayed by a SCREEN command in a logic block, control returns to the command immediately following the SCREEN command.

■ If the screen was displayed directly from a menu, control returns to the menu.

- If the screen was displayed from a function called from a field on another screen, or a function key on another screen, HP ALLBASE/4GL firstly returns to the function and executes the remaining commands. Control then returns to the logic block or menu that displayed the original screen.

On a menu, the I-COMMIT internal routine immediately executes the selected item.

I-EXIT   The I-EXIT internal routine terminates execution of HP ALLBASE/4GL and returns to the original MPE/iX operating system level.

I-HELP   The I-HELP internal routine displays the help screen associated with the current message, field, or screen. By default, HP ALLBASE/4GL assigns this action to function key number 7 in a new function key set.

I-hpqm_call   This routine temporarily suspends execution of HP ALLBASE/4GL and starts HP ALLBASE/QUERY. The normal HP ALLBASE/QUERY welcome screen is bypassed, and the user is automatically connected to the application's HP ALLBASE/SQL database. When the user exits from HP ALLBASE/QUERY, normal HP ALLBASE/4GL processing is resumed.

If the current application has an open SQL transaction when the user presses a function key that calls this action, HP ALLBASE/4GL displays a warning message, and allows the user to complete or reverse the transaction before starting HP ALLBASE/QUERY.

I-image_dbchg_call   This routine suspends HP ALLBASE/4GL processing and invokes the HP TurboIMAGE DBchange/XL program. When the user exits from the HP TurboIMAGE DBchange/XL program, HP ALLBASE/4GL processing resumes at the point where the I-image_dbchg_call routine was called.

I-image_dbutl_call   This routine suspends HP ALLBASE/4GL processing and invokes the HP TurboIMAGE DBUTIL program. When the user exits from HP TurboIMAGE DBUTIL, HP ALLBASE/4GL processing resumes at the point where the I-image_dbutl_call routine was called.

I-image_query_call   This routine suspends HP ALLBASE/4GL processing and invokes the HP QUERY/XL program, if it is on your system. When the user exits from HP QUERY/XL, HP ALLBASE/4GL processing resumes at the point where the I-image_query_call routine was called.

6-40   Screen Development

I-isql_call   Suspends HP ALLBASE/4GL processing and invokes the HP ISQL program. When the user exits from HP ISQL, HP ALLBASE/4GL processing resumes at the point where the I-isql_call routine was called.

I-logon   The I-logon internal routine immediately terminates the current application and displays the HP ALLBASE/4GL sign-on screen. (Note that this routine name is in lowercase letters.)

Typically, you would use this internal routine to provide the end-user with a `Sign-on Screen` function key.

I-MAINMENU   The I-MAINMENU internal routine displays the application main menu. This is either the initial action specified for the application on the administrator application definition screen (Refer to the *HP ALLBASE/4GL Developer Administration Manual*), or the first menu displayed by the application if the application's initial action is a process. If no menu has been displayed, the application is terminated.

The I-MAINMENU internal routine terminates the current screen and logic block immediately.

I-PREVKEYS   Replaces the function keys on the application screen with the previous set of function keys.

I-PREVMENU   If HP ALLBASE/4GL is displaying a menu, the internal routine I-PREVMENU immediately returns control to the previous menu.

If HP ALLBASE/4GL is displaying a data screen, but the user has not entered any data on the screen, the I-PREVMENU internal routine returns control to the last menu displayed by the system. The I-PREVMENU routine immediately terminates any current process, function or decision table.

If the user has started entering data on the data screen, the I-PREVMENU internal routine initially displays a message informing the user that the screen has not been committed. A second I-PREVMENU internal routine returns control to the last menu displayed by the system without committing any data entered so far. Any current module, process, function or decision table is terminated immediately.

By default, HP ALLBASE/4GL assigns the I-PREVMENU internal routine to function key number 8 in a new function key set.

Screen Development   6-41

I-PRINT   Prints the screen and the contents of all fields. The destination printer is determined by the HP-UX variable HP4SCREEN_PRINT.

If HP4SCREEN_PRINT is undefined, HP ALLBASE/4GL writes the screen image to the formal file designator HP4REP.

If HP4SCREEN_PRINT exists, but is null (that is, its value is the empty string "") HP ALLBASE/4GL attempts to print the screen image on a slave printer attached to the user's terminal.

If HP4SCREEN_PRINT exists and contains a non-null value, screen printing is inhibited.

I-REFRESH   Repaints the screen and the contents of all fields.

I-RESET   Performs a terminal soft reset sequence.

I-SHELL   Suspends HP ALLBASE/4GL processing and invokes the MPE/iX command interpreter. When the user exits from the MPE/iX command interpreter, HP ALLBASE/4GL processing resumes at the point where the SHELL command was issued.

Users can execute a specific a command interpreter by setting the SHELL variable. MPE/iX executes the command interpreter defined by this variable. If the SHELL variable is not defined, this internal routine calls the command interpreter program CI.PUB.SYS by default. The system administrator can change this default.

I-SLAVE_PRINT   Prints an image of the current screen on a slave printer attached to the user's terminal. If no slave printer is attached to the terminal, this command times out after about five seconds and HP ALLBASE/4GL issues a message asking the user to either retry or abort the command.

I-TAB   The I-TAB internal routine moves the screen cursor as if the (Tab) key has been pressed. This internal routine is provided to accompany the I-BACKTAB internal routine.

I-TRACEMODE   Toggles the HP ALLBASE/4GL trace mode on or off. When trace is active, HP ALLBASE/4GL displays the current logic block and step for about three seconds, and then proceeds to the next step. Refer to chapter 14 for more information on trace mode.

I-TRAIN   Toggles the HP ALLBASE/4GL training mode on or off. If the system administrator has set the application user into training mode, this

6-42   Screen Development

internal action does not disable the mode. In this case, the training mode can only be disabled by the system administrator.

Default Function Key Definitions

When you're defining a new function key set, HP ALLBASE/4GL supplies a number of default function key definitions automatically.

You can set up your own default function key set in the application you're developing by a defining a function key set called *default_keys*. Each time you create a new set of function keys, HP ALLBASE/4GL will display this set of default key actions and labels. You can type over the defaults if they don't suit the screen you are working on. Note that if you define a set of default keys in your application, these keys will be displayed on an application screen unless you specify a different set of function keys on the screen header for the application screen.

If you don't define a function key set called *default_keys* in your application, HP ALLBASE/4GL displays the function key set *default_keys* from the administrator application as the default each time you create a new function key set in your application.

If you are in test mode, you may see a different set of default keys than you see when accessing other developer screens. The following tables list the default function keys for developer screens and test mode.

The first table lists the administrator application default function keys. These keys are available in both developer screens and in test mode.

Key	Action	Key Label Text
5	K-global_keys_1	System Keys
6	I-COMMIT	Commit Data
7	K-HELP	Help
8	I-PREVMENU	Previous Menu

The function key set *global_keys_1* from the administrator application supplies the definitions for the System Keys function keys to function key number 5. These keys are available in both developer screens and in test mode.

The actions and key labels for these keys are as follows:

Key	Action	Key Label Text
1	I-MAINMENU	Main Menu
2	K- print_scrn_keys	Screen Printing
3	I-CLEAR	Clear Screen
4	I-REFRESH	Refresh Screen
5	I-PREVKEYS	Previous Keys
6	K- global_keys_2	More Keys
7	I-HELP	Help
8	I-PREVMENU	Previous Menu

The *K- global_keys_2* action assigns the key set *global_keys_2* from the administrator application to the `More Keys` function key in the `System Keys` function key set. These keys are available only on the developer screen; they are not available in test mode.

The actions and key labels for this function key set are:

Key	Action	Key Label Text
2	I-SHELL	Op. System
3	I-isql_call	ISQL
4	I-hpqm_call	ALLBASE QUERY
5	I-PREVKEYS	Previous Keys
6	K-global_keys_3	More Keys
7	I-HELP	Help
8	I-EXIT	Exit

The *K- global_keys_2* action in test mode assigns the following key set *global_keys_2* to the `More Keys` function key in the `System Keys` function key set. These keys are available only in test mode.

6-44   Screen Development

The actions and key labels for this function key set are:

Key	Action	Key Label Text
2	I-SHELL	Op. System
3	I-TRAIN	Training Mode
4	I-TRACEMODE	Trace Mode
5	I-PREVKEYS	Previous Keys
7	I-HELP	Help
8	I-EXIT	Exit

The *K- global_keys_3* action assigns the key set *global_keys_3* from
the administrator application to the `More Keys` function key in the
`More Keys` function key set. These keys are available only in the developer
screens.

The actions and key labels for this function key set are:

Key	Action	Key Label Text
1	I-image_dbutl_call	T'IMAGE DBUTIL
2	I-image_query_call	T'IMAGE QUERY
3	I-image_dbchg_call	T'IMAGE DBCHANGE
5	I-PREVKEYS	Previous Keys
7	I-HELP	Help
8	I-EXIT	Exit

The *K-print_scrn_keys* action in the *global_keys_1* function key set assigns the
function key set *print_scrn_keys* from the administrator application to function
key number 2 in the `System Keys` function keys. These keys are available in
the developer screens and in test mode.

The actions and key labels for the *print_scrn_keys* function key set are:

Key	Action	Key Label Text
3	I-PRINT	To System
4	I-SLAVE_PRINT	To Local
5	I-PREVKEYS	Previous Keys

In test mode you can override the four system function key sets, *global_keys_1*, *global_keys_2*, *global_keys_3*, and *print_scrn_keys*, for a particular application by defining function key sets with the same names in your application.

Using the Screen Painter

The screen painter allows you to create or modify the screens for an application. To run the screen painter, select the *Screen Painter* option on the screens menu, or press the `Painter` function key on the other screens.

When the painter starts, it prompts you to enter the name of a screen. You must define the screen name and type on the screen header screen before you can call it up with the painter.

If you're painting a new screen, the painter presents you with a screen that is blank except for the bottom two lines.

The bottom two lines of the screen are prompt lines. The painter prompts you for input or displays messages in these lines. If you enter the name of an existing screen and the screen layout already exists, the painter displays the existing screen image.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Screen Painter Function Keys

Most screen painter functions are controlled from the terminal function keys. Although the arrangement of the screen painter function keys differs in detail for menus and data screens, the function keys consist of the following function key sets:

- *Main Keys*

- *Layout Keys*

- *Sequence Keys*

- *More Keys*

When you start work on a new screen, the painter initially loads the *Main Keys* function keys. If you use the painter to modify an existing screen, it presents the *Layout Keys* as the first set of function keys.

The *Main Keys* allow you to create input and output fields on data screens and windows, and create action items on menus. On all types of screens, the *Main Keys* also allow you to create special text items and system items.

The *Layout Keys* allow you to move, copy, or delete fields on all types of screens.

The *Number Keys* allow you to display the field numbers on automatically numbered data screens and windows, and modify the numbers on manually numbered data screens and windows.

The *More Keys* function keys provide a number of utility functions. Refer to "More Function Keys" elsewhere in this chapter for details.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Cursor Movement

You can move the cursor to any position on the painter screen using the terminal cursor control keys (cursor right), (cursor left), (cursor up), and (cursor down).

Pressing the (cursor home) key followed by (Return) moves the cursor to the top left corner of the screen. Pressing (Shift) and (cursor home) together followed by (Return) moves the cursor to the lowest and rightmost field on the screen. The (Tab) key moves the cursor from field to field on the screen. Pressing (Shift) and (Tab) together followed by (Return) moves the cursor from field to field in reverse order.

On HP touchscreen terminals, you can move the cursor to a particular location on the screen by touching the screen.

Note	If you want to use the touchscreen facility on a touchscreen terminal, make sure the terminal is in touch mode before you start the painter. Touchscreen operation may not work correctly if you turn touch mode on after the painter has started.

Screen Field and Item Types

The type of fields or items on an application screen depends on the screen type.

Menus

A menu presents the user with a choice of options in an application. Menus can have any of the following actions associated with screen action items.

Prefix	Action
B-	Background process
D-	Menu or data screen
F-	Function
H-	Help screen
P-	Process
R-	Report
X-	External program

Menus cannot contain any data input or output fields.

Data Screens and Windows

Data screens allow the user to enter data or examine information. Data screens and windows can have data input and output fields.

A data screen cannot execute another action directly. However, function logic blocks associated with data screen fields can execute other actions.

Other Screen Items

All HP ALLBASE/4GL screens can have the following item types:

- **Text literals**

- **Special text items** Special text items are text items that have specific terminal display enhancements such as inverse video, color, blinking display, and underlining associated with them. The display highlighting for special items does not depend on the system-wide values defined by the system administrator. Special text items can also use the line drawing character set on terminals that support these characters.

- **System items** A system item displays the contents of any of the following items:

  - the date.
  - communication area fields.
  - master titles.
  - application titles.

Screen Development 6-49

- scratch-pad fields.
- numeric constants.
- alphanumeric constants.
- variables or calculated items.

Creating Screen Items

The procedure for creating screen fields or items depends on the type of field or item, and the type of screen. The following subsections describe the procedure for creating new fields or items on a screen. Later subsections describe the procedures for copying, moving or deleting fields or items on a screen.

Literals

The procedure for creating text literals is the same for all types of screens.

To create a literal on a screen, move the cursor to the starting point for the literal and press (Return). You are then prompted to enter the text. Type in the text for the literal and press (Return) to terminate the literal.

To change an existing literal, move the cursor to a position on the literal and press (Return). You will be prompted to modify the text. You can do this by typing over it, or by using the terminal edit keys to insert or delete characters. Press (Return) to terminate your modifications.

System Items

The procedure for creating a system item is the same for all types of screens.

To create a system item on the screen, move the cursor to the starting point for the item and then press the `System Item` function key in the *Main Keys* function key set. The painter creates two input fields on the screen to allow you to enter the prefix for the system item, and the name of the item.

The prefixes and the corresponding items are shown below.

Prefix	Item
D	Date
*	Communication area
M	Master title
A	Application title
N	Numeric constant
C	Alphanumeric constant
V	Variable
P	Scratch-pad field
U	Calculated item

The prefix field will contain one of the system item prefixes. You can accept the displayed prefix by pressing (Return), or you can type over the prefix. By pressing the `Next Type` or `Previous Type` function keys you can obtain a display of the available types of system items.

For items other than the date, the painter prompts you to enter the item name. You can reference scratch-pad fields by name or number.

Pressing the `Restore Original` function key restores an existing system item to its original condition if you have started changing the item, but have not completed the change.

You cannot use synonyms to specify communication area fields as system items. You must always use the standard communication area field names.

Action Items

Only menus can have action items. To create an action item, move the cursor to the starting point for the item text. Press the `Action` function key in the *Main Keys* function key set. The painter prompts you to enter the prefix for the action type. (By pressing the `Next Type` and `Previous Type` function keys, you can obtain a display of the available action type prefixes.)

Enter the action type prefix and press (Return). The painter then prompts you to enter the name of the action item. Enter the name and press (Return).

The painter displays the name of the item and creates a title on the screen at the cursor position. You can type over the displayed item name to enter the text for the menu item. Press (Return) to terminate the text entry.

The name displayed by the painter is the text from the short description field on the header screen for menus, functions, processes, and reports. By choosing the text you enter in the description field on these header screens carefully, you can avoid the need to type over the menu item text on the screen.

To change the text of an action item, move the cursor to a position on the literal and press (Return). You will be prompted to modify the text. You can do this by typing over it, or by using the terminal edit keys to insert or delete characters. Press (Return) to terminate your modifications.

To change the action associated with an existing action item, move the cursor to the item and press the `Action` function key. You can now change the action type prefix, action name, and screen text for the item.

Input Fields and Output Fields

Only data screens and windows can have input and output fields. When you create an input or output field, the painter assigns some default attributes to the field. HP ALLBASE/4GL displays these attributes automatically on the screen field details screen for the appropriate field.

To create an input or output field, move the cursor to the starting point for the field and press the `Input Field` or `Output Field` function key in the *Main Keys* function key set. To define the field, you can do any of the following:

- Enter a number representing the length of the field in characters. The field details default to: required field, alphanumeric, and left justified. You can change these attributes on the field details screen.

- Enter a dictionary field specification name. The painter retrieves the field specification from the dictionary and creates a field of the correct length.

6-52 Screen Development

Using Dictionary Fields

If you use a dictionary field specification name to define the length of the field, the painter retrieves the details of the field from the dictionary and assigns these attributes to the screen field. HP ALLBASE/4GL automatically writes these details into the correct fields on the fields details screen. You cannot change these attributes unless you delete the dictionary field specification name on the screen field details screen.

If you define a screen field using a dictionary field specification name, any subsequent changes to the field specification in the dictionary or the field details screen are automatically reflected in the screen image.

Converting Fields

You can convert an input field to an output field, or convert an output field to an input field.

To convert a field from one type to the other, move the cursor to the field and press the function key for the type of field you want to convert the screen field to. The painter then asks you if you want to change the item type. Press Y and then press (Return) to confirm the change, or press N and then press (Return) to cancel the change.

Special Text Items

The procedure for creating special text items is the same for all types of screens.

To create a special text item, move the cursor to the starting point for the item and press the `Special Text` function key in the *Main Keys* function key set. The painter responds by displaying the special text function keys.

```
 _____
| Set highlighting attributes then press return                      |
|_____|
|Inverse   Half   Blinking  Under    1*  1   Line      Color   Abort |
|Video     Bright           Line             Drawing*   Keys    Edit |
 --------------------------------------------------------------------
```

The special text function keys are toggle keys that turn the selected mode *on* or *off*. When the mode corresponding to a key is *on*, the painter displays an

FINAL TRIM SIZE : 7.0 in x 8.5 in

asterisk in screen text for the key. Select the highlighting you require and then press (Return).

You can now enter the text of the special text item. Press (Return) to terminate the text entry.

Using Colors

You can define the color of a special text item for display on color terminals. To set the color of a special text item, press the `Color Keys` function key. The painter responds by displaying the color keys. Press the key corresponding to the color you want. The selected color is shown by an asterisk in the function key label for the color. (This selection defaults to white.)

You can assign color attributes to a special text item even if you're working on a monochrome terminal. If you're working on a color terminal, the special text item will appear in the selected color.

Note

Use blue carefully. Blue appears very dark on some color terminal screens, and can be hard to read. Use cyan (light blue) in preference.

Most color terminals are not capable of supporting color and half-brightness for the same item. If you do specify both color and half-brightness for the same item, the color attribute takes precedence on color terminals.

Monochrome terminals ignore the color attributes.

Editing Special Text Items

You can change the text of a special text item by typing over the existing text.

To change the display highlighting of a special text item, move the cursor to the item, and press the `Special Text` function key. The painter responds by displaying the special text function keys. You can change the highlighting of the item by making a new selection from the function keys. Press (Return) when you have selected the highlighting you want.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Converting Text Items

The screen painter allows you to convert a screen literal to a special text item.

To convert a screen literal, move the cursor to a position on the literal and press the `Special Text` function key. The painter displays the message *Convert normal text to special text?* To convert the text, enter Y. The painter responds by displaying the special text function keys. Select the highlighting you want, and then press (Return). The painter converts the screen literal to a special text item.

You cannot convert a special text item to a normal screen literal directly. If you want to convert a special text item to a different type of screen item, delete the special text item and then reenter it as the item type you want.

Line Drawing Character Set

On terminals that support the line drawing character set, you can use these characters for any special text item. To use the line drawing character set, press the `Line Drawing` function key after you have selected the special text function key. You can use the other display highlighting function (inverse video, blinking display, etc) with the line drawing characters.

In the line drawing character set, the normal keyboard characters are replaced by the characters shown in the figure below.

```
Line draw keyboard overlay here
```

If you display a line drawing special text item on a terminal that doesn't support the line drawing character set, HP ALLBASE/4GL substitutes suitable standard characters for the line drawing characters.

Screen Development   6-55

Clashing Items

The screen painter does not allow you to define fields or screen items that overlap or clash with existing items on the screen. If you attempt to define an item that clashes with an existing item, the screen painter displays a warning message.

Clashes may result from a change to the dictionary definition of a field (that is, the length of the field has been increased in the dictionary) or changes to a system item. In this case, the painter reads the screen image, and temporarily shortens the clashing field. The clashing field is indicated by error highlighting and is filled with the string "#*** ...". The painter expands the clashing field to its correct length if you move it to a location where it does not clash with another item.

Display Highlighting

The system administrator at the end user site defines the display highlighting for all application screen items except special text items. Display highlighting for special text items depends on the display attributes you choose when you define the item.

The following table lists the display highlighting used in the developer application. The released default highlighting for the run-time environment is the same as the developer environment highlighting.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Item	Highlighting
Banner line	(Row number 1 of the screen) Half-bright inverse video, cyan
Menus	
Selected item	Half-bright inverse video, green
Other items	Half-bright normal video, yellow
Active item	Full-bright inverse video, yellow
Data screens	
Non-active input field	Half-bright inverse video, yellow
Active input field	Full-bright inverse video, green
Input field in error	Full-bright inverse video, blinking, red
Display only field	Full-bright normal video, green
Text Items	Half-bright normal video, cyan
System Items	Half-bright normal video, cyan
Messages	
Message	Half-bright inverse video, green
Query	Half-bright inverse video, green, terminal bell
Warning	Half-bright inverse video, yellow
Error	Half-bright inverse video, red, terminal bell
Abort	Half-bright inverse video, red, terminal bell
Response item field	Full-bright inverse video, green

FINAL TRIM SIZE : 7.0 in x 8.5 in

Available Space Keys

The `Available Space` function key is in the *Main Keys* function key set for menus. For data screens and windows, the `Available Space` function key is in the *Sequence Numbers* function key set.

The available space function key displays the amount of space (in characters) between the current cursor position and the beginning of the next field. It also displays the details of the field or system item at the cursor position if the item has been temporarily shortened due to a clash with another item.

Layout Function Keys

The screen painter layout function keys allow you to move, copy or delete fields and items on the application screen. Press the `Layout Keys` function key on the *Main Keys* function key set to display the layout function keys.

Moving and Copying Fields

The screen painter allows you to move or copy individual items, groups of items, or defined blocks of items. You cannot move or copy fields to locations that cause screen clashes. If you attempt such a move or copy operation, the painter displays a warning message.

To move or copy an individual item or field, move the cursor to the item and then press the `Move` or `Copy` function key. Position the cursor at the new location for the item and press (Return) to complete the move or copy. You can continue moving or copying the selected item until you press the `End Function` function key.

To move or copy a group of fields, move the cursor to the first field in the group and press the `Move` or `Copy` function key. Pressing the `Move` or `Copy` again includes the next field in the tabbing sequence for the screen in the group of fields to moved or copied. The painter highlights the fields that are included in the group. When you have included all the fields you want to move (or copy), position the cursor at the new location for the first field in the

group, and press Return. You can continue to move or copy the group of fields until you press the End Function key.

Block Move and Copy Functions

To move or copy a block of fields, you must first define the block with the Begin Block and End Block function keys.

The Begin Block and End Block function keys allow you to define a rectangular area or block on the screen. All fields that are wholly or partly enclosed in the rectangular area are moved or copied as part of the block. To mark a block, move the cursor to the top left corner of the block that you want to mark, and press the Begin Block function key. Move the cursor to the bottom right corner of the block and press the End Block function key. Note that the cursor does not need to be on a field or screen item while you are marking the begin block and end block positions.

You can now press the Move or Copy function key and position the cursor at the new position for the top left corner of the block. Press Return to move or copy the block. You can continue moving or copying the block until you press the End Function function key.

Deleting Items

You can delete individual items, a group of items, or a block of items on the application screen.

To delete an item, move the cursor to the item and press the Delete function key. The painter responds with the message *Are you sure you want to delete?* Enter Y to confirm the deletion or press any other key to abort the deletion.

To delete a group of items, move the cursor to the first item in the group, and press the Delete function key. The painter responds with the message *Are you sure you want to delete?* Pressing the Delete function key again includes the next item in the tabbing sequence in the group of items selected for deletion. The items selected for deletion are highlighted as you select them. Continue pressing the Delete function key until you have selected all the

items you want to delete. You can now enter `Y` to confirm the deletion. Any other entry aborts the deletion.

You can delete a block of items. All items that are wholly or partly included in the rectangular area defined by the begin block and end block positions are deleted.

To delete a block of items, use the `Begin Block` and `End Block` function keys to mark the block of items you want to delete, and then press the `Delete` function key. The painter requests you to confirm the deletion. Enter `Y` to confirm the deletion. Any other entry aborts the deletion.

Column Entry Mode

The column entry mode simplifies the entry of tabulations or vertical lists on the screen. When column entry mode is on, pressing `Return` to terminate an entry moves the cursor to the next screen line and positions it directly below the beginning of the previous field.

To select column entry mode, press the `Column Mode` function key. This key toggles the column entry mode *on* or *off*. Pressing the key a second time turns the column mode off. The painter displays an asterisk in the key label when column mode is on.

Sequence Numbers Keys

The sequence numbers keys allow you to display the field numbers on the screen, or renumber the fields on manually numbered data screens.

To display the field numbers, press the `Sequence Numbers` function key, and then press the `Display Numbers` function key.

Press the `Auto Number` function key to resequence the fields on a manually numbered screen. You must resequence the fields on a manually numbered screen if you add an additional field or delete an existing field. The painter does not allow you to save a manually numbered screen if the fields need renumbering.

6-60   Screen Development

To renumber the fields on a manually numbered data screen or window, press the `Modify Numbers` function key. You can now move to the field, press `Return`, and type over the numbers displayed in the screen fields. Press `Return` after you enter each number. You must start numbering the fields from 1, and you must number the fields sequentially and without breaks. Press the `End Modify` function key to terminate the modify numbers mode. You cannot save a manually numbered screen if the field numbering sequence is incorrect.

More Function Keys

The *More Keys* function key set provides a number of utility functions. The function keys and their actions are summarized below.

`f1`   `Abort Painter` Aborts the current screen image, and terminates the painter program.

`f2`   `Print Screen` Writes the screen image to formal file designator HP4REP.

`f3`   `Identify Screen` Displays the current screen name, the screen type, and the application name.

`f4`   `Refresh Screen` Redisplays the current screen image.

`f5`   `Save Screen` This key allows you to save the current screen image to a screen name. The painter returns you to the existing image when it has saved the screen.

`f6`   `Overlay Screen` This key allows you to overlay an existing screen image on another screen so you can use it as a template.

`f7`   `Another Screen` This key optionally allows you to save the current screen, and then restarts the painter so you can paint another screen.

`f8`   `Main Keys` Returns you to the main function keys set.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Saving Screens

Function keys in the `More Keys` set allow you to save a screen and then continue work on the same screen, or save the screen and continue work on a different screen.

Press the `Save Screen` function key to save a screen image, and then continue work on the same screen. When you press the key, the painter prompts you for a name to save the image to.

When you press the `Another Screen` function key, the painter asks you if you want to save the current screen. Enter `Y` to save the current screen, or enter `N` to abandon the screen. The painter then prompts you for the name of the next screen.

You cannot save a screen that contains clashing fields, and you cannot save manually numbered screens if the field numbering sequence is incorrect.

Leaving the Painter

You can terminate the painter program by pressing the `Exit` function key. The system saves the current screen image and then terminates the painter program. You cannot exit if the screen contains clashing fields, or if a manually numbered screen has incorrectly numbered fields.

To abandon the current screen and terminate the painter program, press the `Abort Painter` function key. The painter requests you to confirm the abort. Enter `Y` to confirm the abort, or press any other key to cancel the abort command.

FINAL TRIM SIZE : 7.0 in x 8.5 in

7

Reports

This section describes the HP ALLBASE/4GL report generator. It describes the HP ALLBASE/4GL reporting environment and the report processing logic.

This section also describes the screens in the reports menu and the report painter. These are the facilities you use to define application reports and to specify the layout of the printed report.

HP ALLBASE/4GL Report Environment

This subsection provides an overview of the HP ALLBASE/4GL reporting environment. Refer to the descriptions of the reports menu screens for a more detailed explanation of the requirements for completing the reports screens.

Primary Report File

The HP ALLBASE/4GL report generator reports on one file referred to as the primary report file. The primary report file can be one of the following:

- Indexed KSAM data file.

- Serial data file.

- HP ALLBASE/SQL table or view.

- Select list.

- HP TurboIMAGE/iX data set.

The report generator reads records from the primary report file and prints them in the report in the format you specify.

You can specify selection and sorting criteria to select and sort records from the primary report file.

The report generator allows you to specify file linkages to read records from KSAM files or HP TurboIMAGE/iX data sets, other than the primary report file, during report generation.

Report Output

HP ALLBASE/4GL can direct the report output to the user's terminal, a slave printer attached to the terminal, or an MPE/iX formal file designator. If the report is directed to a formal file designator, the stationery type and number of copies specified in HP ALLBASE/4GL can be overridden with a file equation.

By writing the report to a file designator, it is possible for the end-user site to customize the output destination for a report, or even alter the destination for different runs of the same report.

By default, HP ALLBASE/4GL directs the report to device class LP.

Record Selection

HP ALLBASE/4GL allows you to specify selection criteria for selecting records for reporting for up to 99 fields on the primary report file. For each selection field, you can specify a range of acceptable values. You can express the starting and ending values of the range in terms of any of the following:

- Number or literal.

- Numeric or alphanumeric constant.

- Variable or calculated item.

- Screen field reference.

- Scratch-pad field reference.

- File record field reference.

- Work area field reference.

- Communication area field.

You can also use a validation table or validation range for record selection.

If you specify selection criteria for more than one field in the file record, the report generator only reports records that satisfy all the selection criteria.

You can also use an after selection function to implement more complex record selection if required.

Report Sorting and Control Breaks

The HP ALLBASE/4GL report generator operates on a line group basis. It prints detail line groups of a report in the sequence determined by the sorting fields specified for the report. The report generator prints line groups other than detail lines (for example, total line groups) at *control breaks* in the report sequence. A control break occurs when the value of a sort field differs from the value of the same field in the previous record.

Report Sorting

You can specify up to eight fields on the primary file record as sort fields. The report generator uses the contents of these fields to sort the records into the required sequence for printing.

You can specify that the sorting order for each field is either ascending or descending. The sequence of definition of the sort fields defines the *sort levels*. Level 1 is the highest sort level and level 8 is the lowest sort level. The sort levels determine the priority of the sort fields and the sequence of the sorted output. For example, consider three sort fields as follows:

Level	Field
1	Company
2	Department
3	Employee

In this case, the report generator sorts the values of the *Employee* field within the values of the *Department* field and then sorts the results within the *Company* field. The end result is a report that lists departments within each company, and lists the employees in each department.

You can also specify that no sorting is required. However you may still need to specify the sort fields to allow the report generator to print total line groups at control breaks.

Appendix A lists the HP ALLBASE/4GL collating sequence.

Control Breaks

A control break occurs when the value of a sort field changes from one record to the next. The level of the sort field causing the control break determines the level of the control break.

You can define total line groups and subheading line groups for each control break level. At a control break, the report generator prints the corresponding total line group for the old value of the sort field and a subheading line group for the new value of the sort field. The report generator only prints these line groups if you have defined them.

At each control break, the report generator prints the total line group corresponding to the level of the control break. For example, if the control break is on sort level 3 then the report generator prints a T3 line group.

A control break on one sort level automatically initiates control breaks on all lower sort levels if they exist. For example, a control break on level 3 initiates control breaks on levels 4 to 8.

Report Line Types

The HP ALLBASE/4GL report generator uses a number of different types of lines. The line type determines how and when the line is printed. HP ALLBASE/4GL treats all lines the same way when you define them. This means that a page heading can appear exactly the same as a detail line or a bottom of page line.

You can define a number of line groups within most line types, and each group can consist of one or more lines. This allows you to split a report line group over several physical lines on the printed report. For example, you could use several physical lines to print a banner heading on each page of a report. You can also use multiple physical printed lines in the body of a report if each file record contains more information than can be accommodated on one physical line.

When the report generator prints a line group, it normally prints all the lines for the line group. (You can suppress printing of some or all lines for a line group by setting the switches *BYPASS or *ENDLINE *on*. Refer to Report Processing Logic.)

Each line group is identified by a line type code and a group number. A suffix on the line group code identifies the lines within each line group. For example, D2.05 refers to line number 5 for line group D2.

The various report line types and groups are:

Line Type	Group Numbers	Number of Lines	Description
P	1	1 to 99	Top of page headings
C	1	1 to 3	Column headings
B	1	1 to 9	Bottom of page lines
D	1	1 to 99	Detail lines
D	2 to 9	1 to 99	Link detail lines
E	1 to 9	1 to 99	Extra lines
H	1 to 8	1 to 99	Subheading lines
T	1 to 8	1 to 99	Subtotal lines
TF	1	1 to 99	Final total lines

All report line types except type C column heading lines can contain the following data types:

■ Literals.

■ Application titles.

■ Master titles.

■ Numeric or alphanumeric constants.

■ Variables or calculated items.

■ Screen field references.

■ Scratch-pad field references.

- File record field references.

- Work area field references.

- Communication area fields.

Type C lines are treated as literals for their entire length.

Each line group is printed when it is required. The report generator prints a type D1 line group for every record that is selected from the primary report file, and prints subheading and subtotal line groups at every corresponding control break.

The link detail line groups D2 to D9 are available to print the details read from link files. The report generator prints these line groups when they are specified on the file linkages screen.

The PRINT Command

You can print any line group from a function within a report with the PRINT command. When you print a line group with the PRINT command, HP ALLBASE/4GL treats the line group as a detail line group regardless of its type. The PRINT command prints the line group on the next available space on the report page using the spacing defined for the line group. For example, the command PRINT P1 does not necessarily start a new page to print the line group. The report generator only starts a new page if you specified a page skip for the line group P1 on the report line header screen.

File Linkages

File linkages allow you to access files other than the primary report file while the report is being printed. You can specify up to 999 file linkages per report. The report generator executes file linkages on a line group basis.

You can link to a secondary file from any report. However, the secondary file must be either a KSAM data file or an HP TurboIMAGE/iX data set. You cannot link to a secondary HP ALLBASE/SQL table or select list, or a serial data file.

You can define a file linkage for any print line group. The link is initiated by the line group, not a line within the group. If you define a file linkage, the report generator searches the link file for a specific record immediately before

7-6   Reports

printing the line group that initiates the link. This allows you to look up and retrieve data from other files and include that data in the initiating report line group. The data read from the link file is available to all lines in the initiating line group and to any other line group in the report.

You can use a file linkage to print records from the link file. You can specify a link print line group and an end-of-link print line group for each file linkage. If you specify these line groups, the report generator prints the link print line group for each successful read of the link file. The report generator prints the end-of-link print line group when any of the following conditions occur:

- The end of the link file is reached.

- The value in the key field in the record just read does not match the value specified for accessing the link file.

- A post-read function sets the switch *ENDLINE *on*. This switch indicates that no more records are to be read from the link file.

You can specify links from a print line group to more than one link file. If you do specify more than one link, the order of specification determines the order of processing the links.

You can also specify that the link is critical. That is, you can specify that the report generator must find the specified record in the link file on the first file read. If the file read fails because the specified record does not exist or the end of the link file is reached, the report generator sets the switch *ENDLINE *on* and does not print the initiating line group. If the file read fails for any other reason, HP ALLBASE/4GL aborts the report.

If you don't specify that a link is critical, and the first file read fails, HP ALLBASE/4GL clears the file record buffer to spaces so the report can proceed.

You can also specify selection criteria to be applied to printing of records from link files. You must specify the selection criteria individually for each linkage, even if you have specified a number of linkages to the same file and you want to apply the same criteria to each linkage.

Switches and Communication Area Fields

HP ALLBASE/4GL uses a number of switches and communication area fields during report generation.

Switches

The switches used during report generation are:

- ***BYPASS** The status of this switch indicates whether the current physical line in a line group should be printed. If *BYPASS is *on*, the report generator does not print the current physical line. However, the report generator still executes the after-print function for the line if one is defined.

- ***ENDLINE** The status of this switch indicates whether processing of a line group should continue. If *ENDLINE is *on*, The report generator stops printing lines within the line group.

Communication Area Fields

The communication area fields used during report generation are:

- ***COUNT(1) to *COUNT(5)** Five numeric fields optionally used to count the number of times a given line group is printed.

- ***CROSS(1) to *CROSS(5)** Five numeric fields optionally used to accumulate across-the-page totals for a physical print line on the report page. These fields are cleared after each physical line is printed.

- ***TOTALS(1) to *TOTALS(16)** Sixteen numeric fields used for accumulating totals for numeric fields.

- ***REPORT** An alphanumeric field that contains the name of the report being executed.

Report Functions

HP ALLBASE/4GL allows you to execute functions at the following times while a report is being executed:

- At the start of the report.
- After each record from the primary report file or link file has been tested against each set of selection criteria defined for the record.

7-8   Reports

- After each record has been read from a link file.

- Before each physical line is printed.

- After each physical line is printed.

- At the end of the report.

The start-of-report and end-of-report functions can execute any logic commands except REPORT. The other functions in a report can execute any logic commands except SCREEN and REPORT. If any function contains a PROCEED command, HP ALLBASE/4GL terminates the report immediately and executes the process.

Functions within a report can test or change the status of any of the report generator switches. The functions can also read or alter the values in any of the communication area fields used by the report generator.

Totalling Facilities

The report generator uses the 16 communication area fields *TOTALS(1) to *TOTALS(16) to accumulate totals during report generation. These fields are numeric fields and can be used for accumulating totals for any numeric fields, including other *TOTALS fields, line counter fields *COUNT(n) and cross add fields *CROSS(n). You can print the values in the *TOTALS fields in any report line type except type C print lines.

Totalling Matrix

If you define sort levels for a report on the report sorting screen, the report generator constructs an internal totalling matrix when it executes the report. The totalling matrix has 16 columns, with each column corresponding to one of the totalling fields. The number of rows in the matrix is the number of sort levels (n) for the report, plus one additional row corresponding to final totals for type TF report lines. The matrix can have a maximum of nine rows.

The report totalling matrix can be represented as follows:

	Total Field				
Sort Level	1	2	3	...	16
TF					
1					
2					
. . .					
n					

During report generation, you can use a function to read the value in any of the *TOTALS fields. The values in the *TOTALS fields are:

- The values in the row corresponding to the sort level initiating the control break if the report generator *is* printing a type T line group.

- The values in row number n of the totalling matrix if the report generator is not printing a type T line group.

For example, when the report generator is printing a T3 line group, the *TOTALS fields contain the values from row number 3 of the matrix.

Accumulating Totals

While HP ALLBASE/4GL is executing a report, the report generator accumulates the values of fields requiring totalling into row number n of the matrix. When a control break for level n occurs, the report generator accumulates the level n totals into all higher levels of the matrix.

At each control break, the report generator processes the totals for the sort level initiating the control break and all lower sort levels. If you have defined any print line groups for this control break, the report generator prints the line groups. After the report generator has processed the control break, it sets

the totals for that level and all lower levels to zero even if no line groups are printed.

Report Calculations

You can use functions within reports to modify values in the totals fields or perform other calculations. A typical use for these calculations would be determining percentages or ratios for printing in the report.

If you use a function that is not associated with a total (type T) print line type to change the contents of any of the *TOTALS fields, the change is reflected through **all** levels of the totalling matrix.

At a control break, the report generator accumulates the totals in the current row of the totalling matrix into the higher rows before it processes the current type T line group. The report generator then processes the current type T line group, and finally clears the current level and all lower levels in the totalling matrix.

This means that the effect of changes to the values in the *TOTALS fields that occur while the type T line group is being processed are restricted to the current line group. They do not have any effect on other levels of the totalling matrix.

As a result, you can use a function associated with a type T line group to change the contents of the *TOTALS fields without changing other levels in the totalling matrix.

Since changes to *TOTALS fields that are initiated by processing of a type T line group don't have any effect outside the line group, you cannot use a *TOTALS field to accumulate totals for fields on a type T line group. The *TOTALS field receives the value for the current line group, and is cleared immediately after the line group is printed. If you do need to accumulate totals for fields on type T lines, you can use a before print function to accumulate the total into a variable or scratch-pad field.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Reporting From an HP ALLBASE/SQL Database

The HP ALLBASE/4GL report generator can be used to create reports from HP ALLBASE/SQL tables, existing views, or select lists. To generate a report from a select list, you must define the select list in the dictionary.

To generate a report from an HP ALLBASE/SQL database, you must declare and open a cursor using the SELECT command in an SQL logic block. You can use an SQL command in the start-of-report function to call an appropriate SQL logic block.

The start-of-report function must not contain a FILE *NEXT command after the SELECT command. The report generator issues an implicit FILE *FIRST command while it is creating the report. This command generates an error condition if you have already used a FILE *NEXT command. The FILE *FIRST command is not permitted for SQL cursors in HP ALLBASE/4GL logic blocks.

If you are generating a report from an HP ALLBASE/SQL select list, the SQL SELECT statement can perform the selection and sorting required for the report. In this case, you will use the HP ALLBASE/4GL report generator to format and paginate the report. However, the report generator's selection and sorting facilities are available to perform any selection and sorting required in addition to that performed by the SELECT command. In general, better reporting performance will be obtained by using HP ALLBASE/SQL to perform any necessary record selection and sorting.

File Linkages

The HP ALLBASE/4GL report generator allows you to define linkages from SQL objects (base tables, existing views or select lists) to KSAM data files or HP TurboIMAGE/iX data sets. However, you cannot define linkages to SQL tables or select lists, or serial data files.

Reporting From an HP TurboIMAGE/iX Database

When you report from an HP TurboIMAGE/iX database, HP ALLBASE/4GL accesses the database in the DM IMAGE *MODE currently set for the database. If no mode has been specified, it opens in the default *READSHAR mode. This mode should be satisfactory for most reports.

If you wish to set an access mode, the DM IMAGE *MODE command must be executed before the report is executed. If you wish to change a current database access mode, the database must be closed before a second DM IMAGE *MODE command will have an effect on the mode in which HP ALLBASE/4GL accesses the database.

The HP ALLBASE/4GL report generator allows you to define linkages from HP TurboIMAGE/iX data sets to KSAM data files or other HP TurboIMAGE/iX data sets. However, you cannot define linkages to SQL tables or select lists, or serial data files.

Defining a Report

The following paragraphs summarize the procedure for defining a report. Defining a report involves the following steps.

■ **Complete the report header screen.** This screen allows you to:

- assign a name to the report.

- name the output formal file designator.

- specify the

  primary report file.

  report destination.

type of stationery for the report.

physical page parameters for the report.

names of the start-of-report and end-of-report functions (optional).

- enter a description of the report for documentation purposes.

■ **Define the report sorting fields.** This step is optional. However, if you want to sort the records that are read from the primary file or define control breaks to print subtotal and subheading lines in the report, you must define sort fields for the report. The report sorting screen allows you to name the fields in the primary report file that are used for sorting records before printing. The order in which you define the sort fields defines the sort levels and control break levels for the report.

■ **Define the report selection criteria.** This step is also optional, and generally only applies to reports that use HP TurboIMAGE/iX data sets, KSAM data files, or serial files. Record selection for reports that use HP ALLBASE/SQL tables is usually performed using a SELECT statement. If you don't define any selection criteria, the report generator reports on every record read from the primary report file. The selection criteria screen allows you to specify selection criteria for fields on the primary report file. You can also use this screen to specify selection criteria for records read from link files.

■ **Define the report lines.** You must use the report line header screen to define the lines that make up each line group. The report line header screen allows you to define the characteristics of each report line. This screen also allows you to define the before-print and after-print line spacing and name the before-print and after-print functions.

■ **Define the report linkages.** This step is also optional. If you want to read information from KSAM data files or HP TurboIMAGE/iX data sets other than the primary report file, you can specify a file linkage to read the information. Linkages are defined on a line group basis. You can use the information read from the link file in all lines in the initiating line group, or you can print records from the link file in link print line groups.

■ **Paint the report lines.** The report line painter allows you to determine the layout of each report line as it will appear on the printed page. All report lines (except lines in type C line groups) can consist of fields and literals.

7-14   Reports

The report line painter allows you to define the literals and the contents of each field that make up the report line.

■ **Generate the report.** You must generate a report before it can be executed. To generate a report, press the `Generate Report` function key that is defined on the screens in the reports menu. You can also generate a report by selecting the *Reports* option on the application generation menu. This menu is accessed by selecting the *Generate* option on the developer main menu.

Report Processing Logic

The HP ALLBASE/4GL report processing logic can be thought of as operating on two levels. In this manual, the two levels are referred to as the report control logic, and the report line group logic. The report control logic is the overall controlling logic for a report, and the report line group logic is the logic that controls the printing of a single line group within a report. In a typical report, HP ALLBASE/4GL executes the line group logic many times.

Report Control Logic

The overall report control logic can be thought of as being subdivided into three components. These components are:

■ Report initialization logic.

■ Report record processing logic.

■ Report completion logic.

Report Initialization

Figure 7-1 shows the report initialization logic.
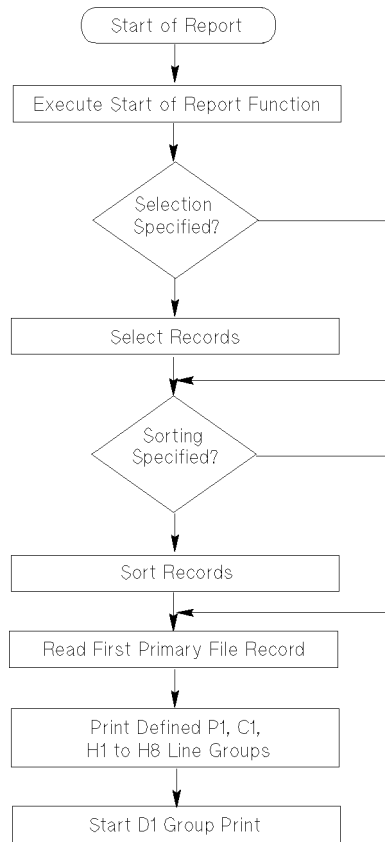
Figure 7-1. Report Initialization Logic

Report generation is initiated by a REPORT logic command, or an R- action from a menu. At the end of the report, control returns to the next command in the logic block that initiated the report, or to the menu that initiated the report.

At the start of a report, the report generator executes the start-of-report function if you have specified one. The start-of-report function can execute any logic command except REPORT. A typical use for this function is to display a screen to obtain user input for report selection criteria. If the report uses an HP ALLBASE/SQL data base, this function can execute an SQL logic block containing a SELECT command to retrieve data for reporting.

7-16   Reports

The report generator then performs any necessary record selection and sorting, and writes the selected and sorted records to a temporary file. It then prints the page heading, column heading, and subheading line groups for all defined control break levels and then enters the report record processing logic.
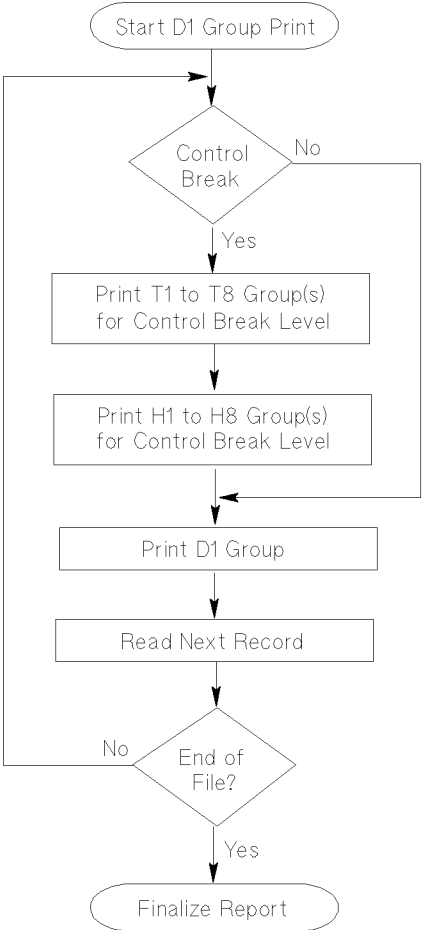
Figure 7-2. Report Record Processing Logic

Report Record Processing Logic

Figure 7-2 shows the report record processing logic.

The report record processing logic is executed for each record in the temporary file of selected and sorted records. For each record, the report generator checks if the record initiates a control break. If no control break is present, the report generator prints a type D1 line group and then reads the next record.

If the record just read initiates a control break, the report generator prints the type T and type H line groups corresponding to the initiating control break level, and for all lower level control breaks. For example, if you define a report with four control break levels, a control break at level 2 initiates printing of type T line groups and type H line groups for level 2, level 3, and level 4 control breaks.

When the report generator detects the end of the temporary report file, it executes the report completion logic.

Report Completion Logic

Figure 7-3 shows the report completion logic.
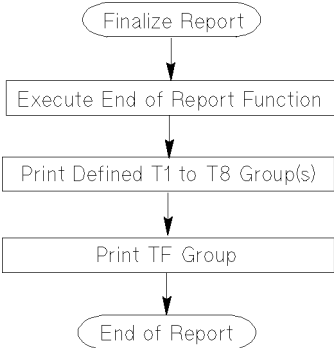
Figure 7-3. Report Completion Logic

After the last record for reporting has been processed, the report generator executes the end-of-report function. The report generator then prints all the control break type T line groups and prints the type TF grand total line group. Control then returns to the logic block or menu that initiated the report.

7-18   Reports

Report Line Group Logic

The report generator executes the report line group logic for every line group it prints, regardless of the line group type. Figure 7-4 shows the line group logic.

The first actions in the line group logic are to set *BYPASS to *off* and execute the before-print function for the current line within the group.

For the first line only in a line group, the report generator checks if there are file linkages defined for the line group. If you have defined linkages for the group, the report generator tests the status of *ENDLINE. If *ENDLINE is *off*, the report generator performs the initial read of each link file and executes the post-read functions for the file linkages. If *ENDLINE is *on*, processing of the current line ceases. This allows you to conditionally bypass the link file read and terminate the line by setting *ENDLINE *on* in the before-print function.

For all lines in the line group, the report generator then checks the status of the switch *ENDLINE. (Note that for the first line in the group, this is the second time that *ENDLINE is tested if you have defined a file linkage for the line group.) If *ENDLINE is *off*, the report generator executes the line printing logic (See Figure 7-5). If *ENDLINE is *on*, the report generator terminates processing of the current line group and processes any defined link print line groups.

After the line printing logic has been executed for any given line within a group, the report generator checks if there are any more lines defined for the group. If there are any more lines to be printed, the report generator returns to the start of the line group logic.
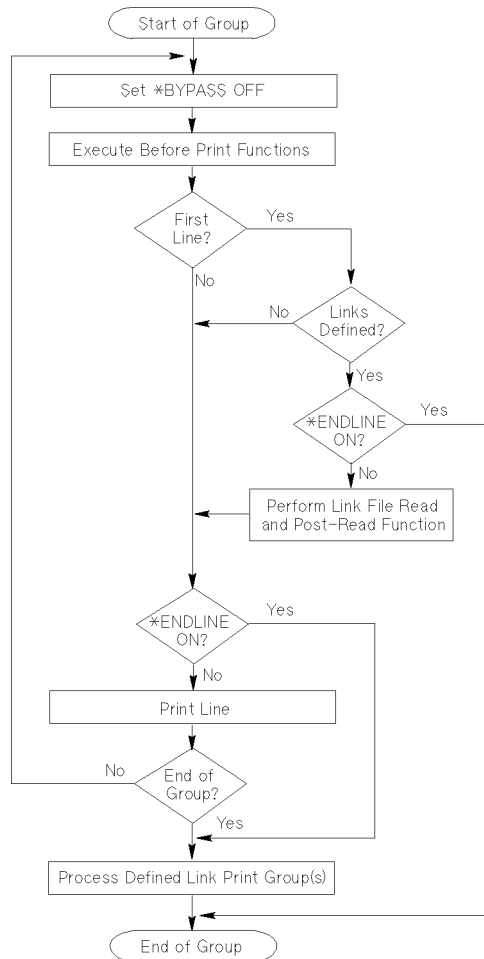
Figure 7-4. Line Group Logic

If all lines for the line group have been printed, the report generator processes any link print groups that are to be printed. For any file linkage that has an associated link print line group or an end-of-link print line group, the report generator continues to read records from the link file and print link line groups until one of the following conditions occurs:

■ The end of the link file is reached.

7-20   Reports

FINAL TRIM SIZE : 7.0 in x 8.5 in

- The value in the key field for the record read from the link file does not match the value of *KEY specified for the file linkage.

- *ENDLINE is set *on* in the post link read function.

At the end of each link, the report generator prints the end-of-link line group if one is defined and then processes any remaining linkages. If there are no remaining unprocessed linkages, the report generator returns to the main report processing logic and executes the step following the one that initiated printing of the line group.

Line Printing Logic

The line printing logic controls the printing of each physical line on the report page. Figure 7-5 shows the line printing logic.

For each line within a line group, the report generator checks the status of *BYPASS. If *BYPASS is *on*, the report generator immediately executes the line's after-print function and returns to the line group logic without printing the line on the page.

If *BYPASS is *off*, the report generator updates the totalling fields *TOTALS and *CROSS, and the line counter fields *COUNT as required. It then checks that there is enough space on the physical page to print the line. If there is sufficient space, the line is printed immediately.

If there is insufficient space on the page, the following actions occur before the line is printed:

- The bottom-of-page line group is printed.

- The paper is positioned at the top of the next page.

- The top-of-page heading line group and the column heading line group are printed.

Then the line itself is printed.

After printing the line, the report generator resets the *CROSS fields and then executes the line's after-print function. Control then returns to the line group logic.
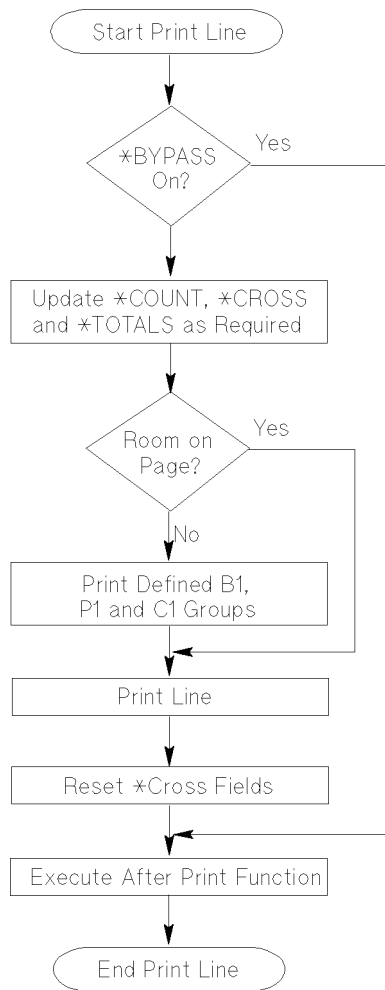
Figure 7-5. Line Printing Logic

FINAL TRIM SIZE : 7.0 in x 8.5 in

Reports Menu Screens

The following subsections describe the screens in the reports menu. Each subsection describes screens under the following subheadings:

- Screen Image.

- Field Descriptions.

- Function Keys.

- Additional Information.

Reports Menu

The reports menu allows you to access the HP ALLBASE/4GL report definition facilities and the report painter.

Screen Image

```
                          HP ALLBASE/4GL Developer
      Dictionary   Screens   Logic   Reports   Module Builder   Utilities   Generate   Test

                              ┌─────────────────────┐
                              │ Header              │
                              │ Sorting             │
                              │ Record Selection    │
                              │ Line Header         │
                              │ File Linkages       │
                              │ Painter             │
                              │                     │
                              │ Catalog Display     │
                              └─────────────────────┘









      Dict.    Screens   Logic    Utility     5* 32    System   Activate   Help   Previous
      Menu     Menu      Menu     Menu                 Keys     Item              Menu
```
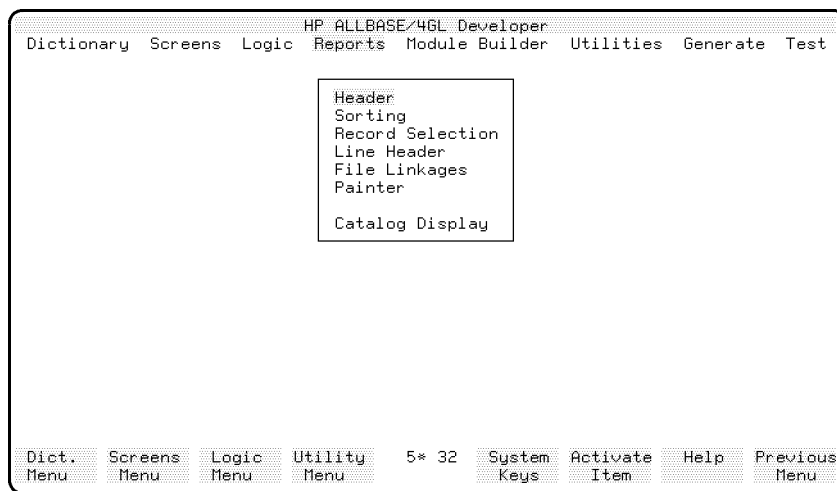
Menu Selections

Header   This selection displays the report header screen, which allows you to name a report, define the primary file for the report, define the output formal file designator, specify the report destination, specify the physical parameters for the printed output, and name the start- and end-of-report functions.

Sorting   This selection displays the report sorting screen, which allows you to define the fields on the primary file that are used as sorting fields and define the sorting order.

Record Selection   This selection displays the selection criteria screen, allowing you to define the criteria for selecting records from the primary report file or link files.

Line Header   This selection displays the report line header screen, which allows you to define the line types, line groups, and lines within each group.

File Linkages

This selection displays the file linkages screen, which allows you to define links to additional KSAM data files or HP TurboIMAGE/iX data sets that the report generator reads as it prints report line groups.

Painter   This selection runs the report painter. The report painter is a program that allows you to "paint" the lines of report as they will appear on the final printed report.

Catalog Display   This selection displays the catalog display screen, allowing you to display the names of existing application components. Refer to chapter 13 for details.

Function Keys

[f1]   `Dict. Menu`   Cancels the current screen and displays the dictionary menu.

[f2]   `Screens Menu`   Cancels the current screen and displays the screens menu.

[f3]   `Logic Menu`   Cancels the current screen and displays the logic menu.

[f4]   `Utility Menu`   Cancels the current screen and displays the utilities menu.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Report Header Screen

The report header screen allows you to define the name of a report. It also allows you to define the primary file for the report, the name of the output file designator, the report destination, the type of stationery, the parameters for the printed output, and the names of any start and end of report functions.

Screen Image

```
 _____
/ Developer                     Report Header            report_header  \
|                                                                        |
| Report Name              ██████████████        Secured      (Y/N)      |
| Report File Designator                                                 |
| Primary File[.Record]                          Index                   |
|                                                                        |
| Printer                  (F/L/D)                                       |
| Type of Stationery                     Number of Copies                |
|                                                                        |
| Characters per Line (maximum)          Actual Page Size in Lines       |
| Print Lines Used per Page              Formfeed Skip to Next Page (Y/N) |
|                                                                        |
| Start of Report Function Name                                          |
| End of Report Function Name                                            |
|        Description                                                     |
|                                                                        |
|                                                                        |
|        Last Modification:      Date            Time                    |
|                                                                        |
| Sorting   Record    Line    Generate  3* 36  System  Commit  Help  Previous |
|           Select   Header    Report          Keys    Data          Menu |
_____/
```

Field Descriptions

Report Name   Enter the name of the report (up to 16 characters).

Secured   To protect this report definition from unauthorized modification, enter Y after you have completed the report definition. Otherwise, enter N. The entry in this field defaults to N.

If you enter Y, the report can only be modified by a developer who has signed on to the application under the correct development security code.

Report File Designator   Enter the MPE/iX name (up to six characters) of the formal file designator to which the formatted report is to be sent.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Any options specified in a file equation override the corresponding options supplied by HP ALLBASE/4GL. This table lists the options that HP ALLBASE/4GL sets, if they are not overridden in a file equation.

Option	Comment
COPIES	Report header, number of copies field
DEV	Default value is "LP"
FORMS	Report header, type of stationery field if not "Standard"

By using a file equation, you can override these defaults. The following examples show the use of file equations.

    FILE MYREPT=PAYSLIPS;DEV=LASER,12,3

This file equation passes the report written to the formal file designator *MYREPT* to the MPE/iX spooler with the name *PAYSLIPS*. Three copies of the report are printed on the device class *LASER* with an output priority of 12.

    FILE MYREPT;DEV=DISC

This file equation writes the report to a disk file *MYREPT*. Any existing file *MYREPT* will be overwritten. The disk file will have embedded carriage control codes so it could be spooled later if required.

Primary File [.Record].   Enter the HP ALLBASE/4GL name of the application data file to be reported. If a serial data file, KSAM data file, or HP TurboIMAGE/iX data set uses more than one record layout, you can specify the required record layout. The report generator uses the default record layout for a data file or data set if you don't enter a record layout name. The report generator reads records from this file for selection and sorting. (You can specify other files for report linkages or for use from within functions.)

Index   The entry in this field defaults to 1, specifying that the report generator uses the primary index for reading the file.

If you want to use a different index to read the file, enter the index number in this field.

Printer   Enter `F` to have the report written to the formal file designator named in the *Report File Designator* field. Any options specified in a file equation override the corresponding setting on the report header screen. This enables the end user to customize some of the report parameters at run time.

Enter `L` to print the report on a slave printer attached to the terminal, or enter `D` to display the report on the terminal screen. (If you want to display a report on the terminal screen, use a line length of 80 characters or less.)

If you display a report on the terminal, set the physical page length to 22 lines so each page will fit on the screen. When HP ALLBASE/4GL displays a report on the screen, the display pauses at the end of each page. Pressing $\boxed{\text{Return}}$ displays the next page.

Type of Stationery   Enter the stationery type for the report (up to 12 characters).

If the value in this field is other than the default value "Standard", HP ALLBASE/4GL opens the file with the option FORMS=*stationery*, where *stationery* is the value entered in this field. If a FORMS option is specified in a file equation, this field has no effect.

Number of Copies   Enter a number in the range 0 to 99 specifying the number of copies of the report to be printed. This field defaults to 1 and is only applicable if the output is directed to a formal file designator or a slave printer.

HP ALLBASE/4GL opens the file with a COPIES=*number* option, where *number* is the value entered in this field. If a COPIES option is specified in a file equation, this field has no effect.

Characters per Line   Enter the maximum number of characters that can be printed on a report line. (From 70 to 255.)

Actual Page Size in Lines   Enter the number of lines per page. The default is 66 (11 inch deep paper at 6 lines per inch).

Print Lines Used per Page   Enter the maximum number of lines actually printed on each page. Don't include the B1 (bottom of page) line group in this figure. If you define a B1 line group, it is printed after this maximum number of lines have been printed.

The default is 60.

Formfeed Skip to Next Page   If you enter `Y`, the report generator uses a page feed command to skip to a new page. The default is `Y`.

If you enter `N`, the report generator uses the physical page length and skips the appropriate number of lines to start the next page. Typically, you would use this option with special paper with a page length other than 66 lines.

Start of Report Function Name   Enter the name of a function to be executed before the report is executed. The function can execute a screen to obtain report selection values from the end user. If the function executes a process, the report is terminated. The function cannot contain a REPORT logic command.

End of Report Function Name   Enter the name of a function to be executed after the report has completed. The function cannot contain a REPORT logic command.

Description   Documentation fields. Enter a description of the report for documentation purposes. HP ALLBASE/4GL automatically updates the date and time fields to show the date and time of the last modification.

Function Keys

(f1)   Sorting   Cancels the current screen and displays the report sorting screen.

(f2)   Record Select   Cancels the current screen and displays the selection criteria screen.

(f3)   Line Header   Cancels the current screen and displays the report line header screen.

(f4)   Generate Report   Generates the report.

Additional Information

You must define each report with this screen before you complete any of the other reports definition screens.

Report Stationery

The report generator allows you to specify the type of stationery to be used for a report. This allows you to specify special stationery for printing invoices or other special reports.

If the stationery name specified differs from the default stationery name "Standard", HP ALLBASE/4GL opens the output file with FORMS=*stationery*, where *stationery* is the value entered in the stationery field. A FORMS specifier in a file equation always overrides the setting made from HP ALLBASE/4GL.

When the MPE/iX spooler is ready to print a file with a FORMS specifier, it displays a message on the system console requesting the system operator to load the required stationery.

If the report is written to a disk file using the formal file designator, the stationery specification is ignored.

Note	HP ALLBASE/4GL is case sensitive with respect to stationery names.

Report Sorting Screen

This screen allows you to define the fields on the primary report file used as sort fields, and to define the report sort order.

Screen Image

```
┌─────────────────────────────────────────────────────────────────┐
│ Developer              Report Sorting              report_sort   │
│                                                                  │
│      Report name        ████████████████        Secured   (Y/N)  │
│                                                                  │
│                                                ORDER             │
│                                                Ascending or      │
│                                Field Name      Descending        │
│                                                                  │
│           1st          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           2nd          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           3rd          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           4th          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           5th          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           6th          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           7th          ░░░░░░░░░░░░░░░░░         (A/D)            │
│           8th          ░░░░░░░░░░░░░░░░░         (A/D)            │
│                                                                  │
│        Is File already in sequence?    (Y/N)                     │
│                                                                  │
│                                                                  │
│  Header   Record    Line    Generate  4* 27  System   Commit  Help  Previous │
│           Select   Header    Report          Keys     Data          Menu     │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

Report Name   Enter the name of a report defined on the report header screen.

Secured   Display-only field indicating the status of the secured field as defined on the report header screen.

Field Name and Order   These two fields define the fields on the primary report file that are used for sorting records and their sorting order. You can specify up to eight sort fields. The sequence of definition of the sort fields defines the sort levels. Level 1 is the highest, or primary, sort level and level 8 is the lowest sort level.

For each sort field, you must specify the field name and the order of sorting.

Enter the dictionary field specification name of a field on the primary report file in the *Field Name* field. You can use subscript referencing if needed, but you cannot use substring referencing.

Enter the sequence in which the field is to be sorted in the *ORDER* field. The options are:

  A   ascending sequence (lowest to highest)

  D   descending sequence (highest to lowest)

The sort sequence used for each sort field depends on the edit code of the field. For numeric fields, the report generator sorts records in numeric sequence. For alphanumeric fields, the report generator sorts records according to the system collating sequence (refer to appendix A).

The order and number of sort fields determines the order and number of control break levels for the report. A control break occurs when the value in a sort field changes from one record to the next. At each control break, the report generator prints the subtotal line groups and subheading line groups for the initiating control break and all lower level control breaks (if they exist). At control breaks, the report generator prints subtotal line groups for the old value of the field initiating the control break, and prints subheadings for the new value of the field initiating the control break.

Is File already in Sequence?   If you enter `Y`, the report generator does not perform any sorting on the file. For KSAM data files, the report generator reads the records in the sequence defined by the file index specified on the report header screen.

For HP TurboIMAGE/iX data sets, the report generator reads the records according to the sort order specified in the HP TurboIMAGE/iX schema definition of the data set. Refer to the *HP TurboIMAGE/iX Database Management System Reference Manual* for further details.

If you enter `N`, the report generator sorts records read from the primary report file into the sequence determined by the sort fields specified on this screen.

You must still specify the sort fields even if you do enter Y in this field. The report generator uses these fields to determine when a control break occurs.

7-32   Reports

Function Keys

[f1]   Header   Cancels the current screen and displays the report header screen.

[f2]   Record Select   Cancels the current screen and displays the selection criteria screen.

[f3]   Line Header   Cancels the current screen and displays the report line header screen.

[f4]   Generate Report   Generates the report.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Selection Criteria Screen

This screen allows you to define optional criteria for selecting records from the primary report file or link files for reporting. The report generator sets the switch *BYPASS *on* if a record read from the file fails the selection criteria. If *BYPASS is *on*, the report generator does not print the contents of the record.

Screen Image

```
Developer                    Selection Criteria                report_select

Report Name           ████████████          Secured      (Y/N)
After Selection Function

Selection Number              Action    (A/C/I/D)

Link    Link File[.record]        Field Spec. Name

Selection Criteria: Values FROM
                           TO
-----------------------------------------------------------------------------

Number Link  Field              'FROM' selection criteria

Header   Sorting    Line    Generate   3* 30   System   Commit   Help  Previous
                    Header   Report             Keys     Data           Menu
```

Field Descriptions

Report Name   Enter the name of a report defined on the report header screen.

Secured   Display-only field indicating status of the secured field as defined on the report header screen.

After Selection Function   Enter the name of the function that is executed after the report generator has tested the record read from the file against the selection criteria. You can use this function to perform additional selection or checking.

The function has access to each record as the report generator reads it from the file. The function can also test the status of *BYPASS after the record

7-34   Reports

has been tested. You can use this function to accept rejected records, or reject accepted records by setting the value of *BYPASS appropriately.

Selection Number Enter a number between 1 and 99 to identify this selection. The selection number does not impose any sequence of resolution of the selection criteria. If you are defining a new set of selection criteria, the display in this field becomes the number of the next available selection, even if you enter a larger number.

Action Enter the code of the action to be performed on this selection. The action can be any one of the following:

> A   add new selection
>
> C   change this selection
>
> I   insert a new selection before this one
>
> D   delete this selection

Link Enter 0 to specify that the selection applies to a field on the primary report file.

Enter a number in the range 1 to 999 to specify the link number to which this selection applies. The link number must be defined on the file linkages screen.

Link File [.record] A display-only field that shows the file (and record) on which this selection criteria operates, as defined on the file linkages screen.

Field Spec. Name Enter the name of a field in the previously specified file (and record). Enter the dictionary field specification name only, without the name of the file and record.

Selection Criteria: Values FROM Enter one of the following containing the starting value of the range acceptable for the field:

- Integer number.

- Literal.

- Numeric or alphanumeric constant.

- Variable or calculated item.

- Screen field reference.

- Scratch-pad field reference.

- File record field reference.

- Work area field reference.

- Communication area field.

- *RANGE = *range_name*.

- *TABLE = *table_name*.

If you specify a *range_name* or *table_name*, you must use a validation table or validation range name that has been defined on the appropriate dictionary screen.

You cannot use substring references in this field.

Selection Criteria: Values TO   Enter the ending value of the range of acceptable values for the field. You can specify any of the values allowed for the *values from* field except *RANGE= and *TABLE=.

Leave this field blank if you have specified a validation range or table name in the *values from* field.

Function Keys

[f1]   `Header`   Cancels the current screen and displays the report header screen.

[f2]   `Sorting`   Cancels the current screen and displays the report sorting screen.

[f3]   `Line Header`   Cancels the current screen and displays the report line header screen.

[f4]   `Generate Report`   Generates the report.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Additional Information

The report generator allows you to specify up to 99 selection criteria per report. If you don't specify any selection criteria, the report generator reports every record in the primary report file.

If you specify more than one selection criteria for a record, (for example, selection on two different fields in the same record) the report generator only reports the record if all selection criteria are satisfied.

If you define selection criteria for a link file, the report generator only prints the link print line group or groups if all the selection criteria on the link records are satisfied.

If you define one or more critical links from any particular report line group, and you specify selection criteria for these linkages, the initiating report line group is only printed if all the link files contain records that pass selection. If all the link files do not contain records that pass selection, the report generator sets the switch *ENDLINE *on* and does not print the line group that initiated the linkages.

The selection criteria for link files are only defined for one file linkage, which in turn is only defined for one report line group. If you have more than one linkage defined for a link file (that is, links to the same file are defined for more than one report line group) you must specify selection criteria separately for each linkage. The selection criteria do not need to be the same for different linkages to the same file.

Report Line Header Screen

This screen allows you to define a report print line and specify the line type, the line group, and the lines within the group.

Screen Image

```
Developer                    Report Line Header         report_line_head

        Report Name    ████████████████        Secured      (Y/N)
        Line Group         (P1/C1/B1/Hn/Dn/En/Tn/TF)
        Line Number
        Action         (A/C/D)

Skip Lines/Page:       Before Print           (0-9/P)
                       After Print            (0-9/P)

Suppress Line If:      All Values Are Zero    (Y/N)
                       All Values Are Blank   (Y/N)

Count Lines Printed Into Counter Number       (1-5,Blank)

Underline Numbers                             (P/A/B)

Before Print Function
After Print Function
Grp #  Act Skip Suppress Count Underline  Function        Function




Header   Record   File     Generate  3* 22   System   Commit   Help   Previous
         Select   Linkage  Report            Keys     Data            Menu
```

Field Descriptions

Report Name   Enter the name of a report defined on the report header screen.

Secured   Display-only field indicating the status of the secured field, as defined on the report header screen.

Line Group   The entry in this field defines the type of report line group. A two character code identifies line groups. The first character identifies the line type, and the second character identifies the line group within the type. Enter one of the line group codes from the following table.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Line Group	Purpose
P1	Top of page heading
C1	Column headings
B1	Bottom of page footings
D1 to D9	Detail lines
E1 to E9	Extra lines
H1 to H8	Subheading lines
T1 to T8	Subtotal lines
TF	Final total line

Line Number   The entry in this field identifies the number of this line within the line group. The range of valid entries depends on the line type. Enter one of the following line numbers:

Line Group	Purpose
P1	1 to 99
C1	1 to 3
B1	1 to 9
D1 to D9	1 to 99
E1 to E9	1 to 99
H1 to H8	1 to 99
T1 to T8	1 to 99
TF	1 to 99

Action   Enter one of the following:

A	-	Add new line
C	-	Change this line
D	-	Delete this line

Skip Lines/Page   These two fields define the number of physical lines to be skipped before and after printing this line.

Reports   7-39

You can specify the following values for these fields:

0 - Don't skip any lines

1 to 9 - Skip 1 to 9 lines respectively

P - Skip to the next page

Normally the report generator prints each line immediately and then clears the report buffer. Depending on when you specify it, the 0 option in this field suppresses printing or clearing of the buffer as follows:

- Before print: Any previous contents of the report buffer are not cleared.

- After print: This line remains in the buffer and is not printed immediately.

You can merge two report lines in the buffer, and print them on the same physical report line by specifying 0 lines skip after print for the first line, and 0 lines skip before print for the second line.

Suppress Line If The next two fields allow you to automatically suppress the printing of a line. You can specify that the line is suppressed if the values in all numeric fields values are zero, or if all alphanumeric fields contain blanks.

Count Lines Printed Optional. Enter a number from 1 to 5 to specify that one of the fields *COUNT(1) to *COUNT(5) is incremented each time the line is printed. The report generator does not increment the counter for this line if *BYPASS is set *on* in the before print function and the line is not printed.

Underline Numbers This field allows you to specify that numeric values are underlined or highlighted when they are printed. The report generator can print a line of hyphens on the physical line before (*prior to*) the number, on the physical line below (*after*) the number, or both before and after the number. Enter one of the following options:

P - Print hyphens above numbers (*prior to*)

A - Print hyphens below numbers (*after*)

B - Print hyphens above and below numbers (*both*)

The underlining or overlining only applies to numeric fields in the print line.

7-40   Reports

Before Print Function   Enter the name of a function to be executed immediately before the current line is printed. The function may change the contents of any fields or accumulators, or print other lines.

You can use the before-print function to suppress printing of the line by setting the switch *BYPASS *on*.

You can terminate printing of this line and all remaining components for the line group by setting the switch *ENDLINE *on* in the before-print function.

After Print Function   Enter the name of a function to be executed immediately after the line has been printed.

Function Keys

(f1)   `Header`   Cancels the current screen and displays the report header screen.

(f2)   `Record Select`   Cancels the current screen and displays the selection criteria screen.

(f3)   `File Linkage`   Cancels the current screen and displays the file linkages screen.

(f4)   `Generate Report`   Generates the report.

FINAL TRIM SIZE : 7.0 in x 8.5 in

File Linkages Screen

This screen allows you to define linkages to KSAM files and HP TurboIMAGE/iX data sets other than the primary report file.

These link files are accessed when the report generator prints a report line group. You can optionally specify that the records read from these files are reported.

Screen Image

```
 _____
/                                                                   \
|  Developer                    File Linkages             report_link |
|                                                                     |
|       Report Name  ████████████████████       Secured     (Y/N)     |
|                                                                     |
|       Link Number  ▒▒                                               |
|       Action           (A/C/I/D)                                    |
|                                                                     |
|       Link From        Print Line      End of Link   Post-Read Function |
|       Line Group         Group         Line Group                    |
|                                                                     |
|          ▒▒               ▒▒              ▒▒          ▒▒▒▒▒▒▒▒▒▒      |
|                                                                     |
|  Link File[.Record]      ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒    Index Number  ▒▒▒▒▒▒   |
|  *KEY=                                                              |
|  Must the Record be Present        ▒▒    (Y/N)                      |
|                                                                     |
|  Number Act Link-From  Print  End  Function       Link File[.Record]|
|                                                                     |
|                                                                     |
|                                                                     |
|                                                                     |
|   Header    Record    Line    Generate   3* 21   System   Commit   Help   Previous |
|             Select    Header   Report             Keys    Data             Menu    |
_____/
```

Field Descriptions

Report Name Enter the name of a report defined on the report header screen.

Secured Display-only field indicating the status of the secured field on the report header screen.

Link Number A number in the range 1 to 999 that identifies this link. If you are defining a new file linkage, the number in this field becomes the number of the next available linkage, even if you enter a larger number.

Action   Enter the action to be performed on this link number. It can be one of the following:

> A   add new linkage
>
> C   change this linkage
>
> I   insert a new linkage before
>   this one
>
> D   delete this linkage

Link From Line Group   Enter the report print line group that initiates the file linkage. It can be any report line group including link print line groups.

Print Line Group   Optional. Enter the line group to be printed if the linkage is successful.

If you specify a line group in this field, the report generator continues to read the link file and print this line group until one of the following occurs:

■ The end of the link file is detected.

■ A post-read function sets the switch *ENDLINE *on* to indicate that no more records are to be read from this link file.

■ The value contained in the key field of the record just read no longer matches the value specified by *KEY=.

If you specify a line group in this field, you must not specify the same line group as that specified in the *Link From Line Group* field.

End of Link Line Group   Optional. Enter a line group to be printed when one of the following occurs:

■ The end of the link file is reached.

■ You specify that no more records are to be read from the link file by setting the switch *ENDLINE *on*.

■ The value contained in the key field of the record just read changes from the value specified by *KEY=.

If you specify a line group in this field, it must not be the same line type as that specified in the *Link From Line Group* field.

Reports   7-43

Post Read Function   Optional. Enter the name of a function that is executed immediately after each record is read from the link file. The function can set the switch *ENDLINE *on* to indicate that no more records are to be read.

This function can also set the switch *BYPASS to *on* to indicate that the record is to be bypassed. If you do bypass a record read from the link file, the report generator does not perform any selection criteria tests (if defined) for the record. However, the after-selection function (if defined) is still executed.

Link File [.Record]   Enter the name of the KSAM file or HP TurboIMAGE/iX data set to be read for the link. The record layout name is optional. The report generator uses the default record layout for the file if you don't specify a record layout name.

Index Number   Enter the number of the index used to read the link file. The entry in this field defaults to 1, specifying that the report generator uses the primary index to read the file. If you do specify an index number, it must correspond to an index number declared for the default record layout for the link file.

*KEY=   This field defines the value of the key the report generator uses to access the link file. The entry in this field can be any of the following:

- Literal.

- Alphanumeric constant.

- Variable or calculated item.

- Screen field reference.

- Scratch-pad field reference.

- File record field reference.

- Work area field reference.

- Non-numeric communication area field.

Must the Record be Present   Enter Y if the first read of this link file must be successful. That is, the link is critical and a record with the specified key value must exist on the link file.

Enter N if the existence of a link file record is optional.

7-44   Reports

Refer to Additional Information below for a description of the actions taken by the report generator if the link file record is not found.

Function Keys

f1   **Header**   Cancels the current screen and displays the report header screen.

f2   **Record Select**   Cancels the current screen and displays the selection criteria screen.

f3   **Line Header**   Cancels the current screen and displays the report line header screen.

f4   **Generate Report**   Generates the report.

Additional Information

The report generator allows you to specify up to 999 file linkages for each report.

Critical Linkages

You can specify that a link is critical by entering Y in the *Must Record be Present* field. This action specifies that the report generator must find a record that matches the required key value the first time the file is read. If you specify that the link is critical, and the report generator does not find the specified record in the link file or reaches the end of the link file, the following actions occur:

■ The value in the communication area field *IOSTATUS is set to the value of the file error condition encountered. (Refer to appendix C for file error codes.)

■ The buffer for the file is cleared.

■ The switch *ENDLINE is set *on* and the report generator does not print the line group that initiated the file linkage.

If the link file read failure is caused by any error condition other than a failure to find the record or reaching the end of the link file, HP ALLBASE/4GL aborts the report.

Reports   7-45

Non-Critical Linkages

If you specify that the link is not critical, and the report generator fails to find a record in a link file, the following actions occur:

- The communication area field *IOSTATUS is set to the value of the file error condition encountered. (Refer to appendix C for file error codes.)

- The file record buffer is cleared.

- The report generator executes the post-read function associated with this link. The report generator then prints the end of link print line group if you have specified one.

Limitations

The report can use any type of file as the primary report file, but you can only define linkages to KSAM data files or HP TurboIMAGE/iX data sets. Linkages to SQL tables, select lists, or serial files cannot be defined.

Report Painter

The report painter allows you to define the format of the report lines and the fields to be printed in each line on a report.

To run the report painter, select the *Painter* option on the reports menu. This starts the report painter and displays a prompt to allow you to enter the report name.

The report painter then displays the report painter screen.

Screen Image

```
┌──────────────────────────────────────────────────────────────────────┐
│ Developer                    Report Painter              report_fields │
│         0....:....1....:....2....:....3....:....4....:....5....:....6....:....7 │
│   ┌────────────────────────────────────────────────────────────────┐   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   │                                                                │   │
│   └────────────────────────────────────────────────────────────────┘   │
│ Current Field Length    000                 Current Cursor Position  001 │
│ Field Name/Literal                                                     │
│ Edit Code               (X/A/U/K/N/S/Q/D/T)                            │
│ Justification           (L/R/C/N)            Pad Character             │
│ Underline               (P/A/B)              Print Field Control  (A/C) │
│ Total Number            (1-16)               Cross Add Number     (1-5) │
│ Report Painter  --  report          [Application      ]               │
│ ┌──────┬──────┬──────┬──────┬────────┬──────┬──────┬──────┬──────┐     │
│ │ Move │ Copy │ End  │Delete│  4* 10 │Window│Field │ More │ Exit │     │
│ │      │      │Function│      │        │ Keys │ Keys │ Keys │      │     │
│ └──────┴──────┴──────┴──────┴────────┴──────┴──────┴──────┴──────┘     │
└──────────────────────────────────────────────────────────────────────┘
```
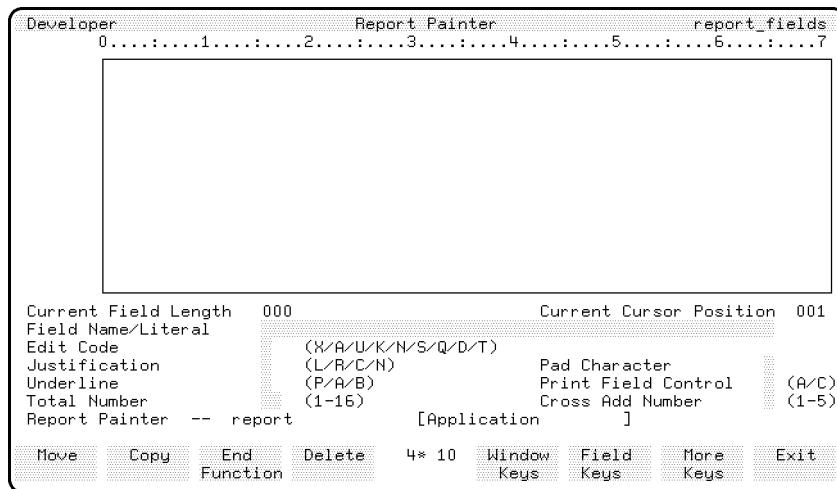
The report painter screen is divided into two parts. The upper part of the screen is a window that shows you part of the layout of the report. You can use the cursor control keys and the report painter function keys to paint report lines in this area. The display in the window can be scrolled horizontally and vertically. This allows you to see report lines, or parts of report lines that are outside the current window area.

A ruler line above the window indicates the column numbers on the report page.

A display-only field to the left of the painter window lists the report lines that you have defined with the report line header screen.

The lower part of the report painter screen contains a number of data entry fields. You can enter the specification of the various report line fields in these screen fields. Refer to "Screen Descriptions" for the details of these fields.

Two display-only fields in the lower part of the screen show the current cursor position and the length of the current field.

The last two lines on the report painter screen are prompt lines. The display in this area prompts you for an input where appropriate, or indicates the next appropriate action. The report painter also displays error messages in this area of the screen.

Cursor Movement

You can move the cursor to any position on the defined report lines within the report painter window using the terminal keyboard cursor control keys (cursor right), (cursor left), (cursor up), and (cursor down).

The new cursor position is only recognized when you press (Return), (Tab), or a function key. If you move the cursor to a position outside of the report painter window, the painter issues a warning message and places the cursor at the closest valid position within the window.

Pressing (cursor home), followed by (Return), moves the cursor to the top left corner of the report.

Pressing (Shift) and (cursor home) together, followed by (Return), moves the cursor to the end of the last line defined on the report.

Pressing (Tab) moves the cursor from field to field in the window. Pressing (Shift) and (Tab) together, followed by (Return), moves the cursor from field to field in reverse order.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Display Scrolling

If you have defined more report lines than the window can accommodate vertically, or the report lines are longer than the window, you can scroll the window display vertically or horizontally.

The *Window Keys* set of function keys control the scrolling of the report painter window. To call up the *Window Keys* function keys set, press the `Window Keys` function key. The painter responds by displaying the following function keys.

```
┌────────┬────────┬────────┬────────┬───────┬────────┬────────┬────────┬───────┐
│   Up   │  Down  │  Left  │ Right  │ 4* 10 │ Centre │Set Horz│Set Vert│ Main  │
│ Window │ Window │ Window │ Window │       │ Window │ Scroll │ Scroll │ Keys  │
```

The `Up Window` , `Down Window` , `Left Window` , and `Right Window` keys scroll the display in the selected direction. You can change the size of the scroll steps with the `Set Horz. Scroll` and `Set Vert. Scroll` function keys.

To change the size of the scroll steps, press the appropriate set scroll function key. The painter responds by displaying a message requesting you to enter the new step size. Enter the required step size, and then press (Return). The default scroll step sizes are two lines for vertical scrolling, and 10 characters for horizontal scrolling.

The `Centre Window` function key centers the window vertically with respect to the number of report lines you have defined, and horizontally with respect to the length of the report lines.

Report Painter Function Keys

The report painter has a number of sets of function keys. The *Main Keys* function key set appears when the painter starts. Pressing the `Window Keys` , `Field Keys` , or `More Keys` function keys displays the other sets of function keys. The function keys and their actions are described below.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Main Function Key Set

f1   `Move` Indicates to the report painter that you want to move the current field to a different location in the window. (Refer to Moving, Copying, and Deleting Fields.)

f2   `Copy` Indicates to the report painter that you want to copy the current field to a different location in the window. (Refer to Moving, Copying, and Deleting Fields.)

f3   `End Function` Indicates to the report painter that you want to terminate the current move or copy operation. (Refer to Moving, Copying, and Deleting Fields.)

f4   `Delete` Indicates to the report painter that you want to delete the current field in the window. (Refer to Moving, Copying, and Deleting Fields.)

f5   `Window Keys` Displays the *Window Keys* function keys, allowing you to scroll the display in the report painter window.

f6   `Field Keys` Displays the *Field Keys* function keys, allowing you to create or modify output fields or dictionary fields on report lines.

f7   `More Keys` Displays the *More Keys* function key set.

f8   `Exit` Saves the current report line layout and terminates the report painter program.

Window Keys Function Key Set

f1   `Up Window` Scrolls the report painter window up by the vertical scroll distance. (Default is two lines.)

f2   `Down Window` Scrolls the report painter window down by the vertical scroll distance. (Default is two lines.)

f3   `Left Window` Scrolls the report painter window to the left by the horizontal scroll distance. (Default is 10 characters.)

f4   `Right Window` Scrolls the report painter window to the right by the horizontal scroll distance. (Default is 10 characters.)

7-50 Reports

f5   `Centre Window` Centres the report painter window display vertically with respect to the number of report line types you have defined, and horizontally with respect to the defined line length.

f6   `Set Horz Scroll` Allows you to change the size of the horizontal scrolling step.

f7   `Set Vert Scroll` Allows you to change the size of the vertical scrolling step.

f8   `Main Keys` Restores the report painter main keys function key set.

Field Keys Function Key Set

f1   `Create Field` Indicates to the report painter that you want to create an output field at the current cursor location.

f2   `Modify Field` Indicates to the report painter that you want to change the characteristics of the field at the current cursor position.

f3   `Dict. Field` Indicates to the report painter that you want to create an output field using a dictionary field specification name at the current cursor location.

f5   `Enter Details` Indicates to the report painter that you want to enter the details for the current report line field into the detail fields on the lower part of the report painter screen. When you press this key, the cursor moves to the *Field Name/Literal* screen field.

f6   `Commit Details` Completes data entry in the detail fields and returns the cursor to the report painter window.

f7   `Original Details` Restores the original field details in the detail fields, provided you haven't committed any changes for the details in these fields.

f8   `Main Keys` Restores the report painter main keys function key set.

More Keys Function Key Set

f1   `Abort Painter` Abandons the current report line layout, and terminates the report painter program.

Reports 7-51

f4 **Another Report** Optionally allows you to save the current report layout, and then reinitializes the report painter to allow you to define another report layout. The painter prompts you for the name of the next report.

f5 **Identify Report** Displays the name of the current report.

f6 **Refresh Window** Repaints the image in the report painter window.

f7 **Avail. Space** Displays the amount of space (in characters) between the current cursor position and the beginning of the next field on the same line, or the end of the report line if there are no other fields on the line.

f8 **Main Keys** Returns you to the main function key set.

Painting Report Lines

All report lines except type C lines can contain literals, output fields, and dictionary fields. The procedure for defining a field depends on the type of field.

Literal Fields

To define a literal field on a report line, move the cursor to the starting position for the literal, press (Return), and type in the text for the literal. The report painter echoes your input on the screen. You can use the normal terminal keyboard control keys to insert characters, delete characters, or backspace in a literal field. Press (Return) to terminate the entry.

When you're defining literal fields on a report line, you can only enter text up to the boundary of the window. If you want to enter a field that goes beyond the current window boundary, use the **Window Keys** function keys to set the position of the window before you define the field.

Output Fields

To define an output field on a report line, move the cursor to the starting position of the field, and then press the **Create Field** function key.

You can now specify the length and type of the field by entering a string of one of the characters shown in the following table. The number of characters you enter determines the length of the field and the actual character determines the type of field. Press (Return) to terminate the entry.

Character	Meaning
A	Alphanumeric field
N	Numeric field
Z	Suppressed zero numeric field

For numeric output fields you can use the following additional codes to specify the way numbers are printed.

Character	Where Placed	Meaning
$	Start of field	Print system currency symbol before the value
.,	Where needed	Numeric punctuation
( )	Surrounding field	Print brackets around the value if it is negative
CR	End of field	Print CR after the value if it is negative
DR	End of field	Print DR after the value if it is positive
-	Start of field	Print - before first digit if the value is negative
-	End of field	Print - after last digit is the value is negative

If numeric overflow occurs while a report is being printed, the field is filled with asterisks (****).

Press the `Enter Details` function key to enter the details for the report line output field. This moves the cursor to the *Field Name/Literal* screen field where you can enter the name of the HP ALLBASE/4GL entity to be printed in the report line field. Refer to Screen Field Descriptions for details of the acceptable entries in this field and the other fields in this part of the screen.

Press the `Commit Details` function key when you have completed your entries in the detail fields. This action commits the data that you have entered

for this report line field, and returns the cursor to the window to allow you to define further report line fields.

Dictionary Fields

To create an output field using a dictionary field specification name, move the cursor to the starting point for the field and press the `Dict. Field` function key. The painter prompts you for the field specification name. When you enter the name, the report painter retrieves the field specification from the dictionary and creates a field of the appropriate length and type at the cursor location.

The painter assumes that you want to reference the field on the default record of the primary report file, and creates the appropriate file record field reference in the *Field Name/Literal* screen field. You can alter this reference if you require.

The painter also retrieves the edit code and justification code for the field specification from the dictionary.

Moving, Copying and Deleting Fields

The report painter allows you to move fields, copy fields, or delete fields in the window area. To move or copy a field, move the cursor to the field, and then press the `Move` or `Copy` function key. The painter then asks you to move the cursor to the desired location for the new field. Press (Return) to complete the move.

You cannot move or copy an output field to a type C line group.

To delete a field, move the cursor to the field, and press the `Delete` function key. The painter responds with a message asking you to confirm the deletion. Enter Y to confirm the deletion. Any other entry cancels the delete command.

Saving Reports

When you have defined all the fields for the report, press the `Exit` function key to save the report layout and terminate the painter program.

Pressing the `Another Report` function key allows you to optionally save the current report, and then continue with another report.

7-54 Reports

Screen Field Descriptions

The following paragraphs describe the data entry fields on the lower part of the report painter screen. Refer to Painting Report Lines for details of the procedure for creating report line fields.

Current Field Length   A display-only field that shows the length of the current field in characters.

Current Cursor Position   A display-only field that shows the position of the cursor on the current report line.

Field Name/Literal   For an output field on the report line, this field allows you to enter the name of the item to be printed. If you have already defined the output field on the report line, this field displays the item name.

The permissible entries in this field are:

- Title name.

- Numeric or alphanumeric constant.

- Variable or calculated item.

- screen field reference.

- scratch-pad field reference.

- file record field reference.

- work area field reference.

- communication area field.

You cannot use synonyms for communication area field names.

If the report line field is a literal, this screen field displays the message:

`Literal - See line format.`

Edit Code. Enter the edit code for an output field on the report line. Refer to chapter 3 for details of the edit codes. The entry in this field defaults to the following values:

X	for alphanumeric report line fields
N	for unsigned numeric report line fields
S	for signed numeric report line fields

Justification. Enter the justification code for the report line field. Refer to chapter 3 for an explanation of the justification codes.

Underline. The entry in this field specifies the print line highlighting applied to the field. The report generator highlights report line fields by printing a line of hyphens above (Prior to) the field, below the field (After), or both above and below the field.

The entries for this field are:

P	Print hyphens above field
A	Print hyphens below field
B	Print hyphens above and below field

Total Number. The entry in this field specifies which of the communication area fields *TOTALS(1) to *TOTALS(16) is used to accumulate the total for this report line field (numeric fields only). Leave this screen field blank if you don't want to accumulate totals for the report line field.

You cannot accumulate totals for fields that are on type T lines.

Pad Character. Enter a character used to replace spaces at the beginning or end of the report line field. This field defaults to a blank.

Print Field Control. The entry in this field controls when the report line field is printed. Enter A to specify that the field is always printed. Enter C to specify that the field is only printed when its value changes or at the start of a new page.

Cross Add Number. Optional. Enter a number between 1 and 5 to specify the *CROSS field used to accumulate the contents of this report line field (numeric fields only).

The cross-add fields accumulate totals across one physical print line only. They are reset immediately after the line is printed.

8

Module Builder

This chapter describes the HP ALLBASE/4GL module builder, its capabilities, and the developer screens used to build an application module. The HP ALLBASE/4GL module builder facility creates a data or resultant screen and a number of HP ALLBASE/4GL items to provide a full maintenance or inquiry screen for an HP ALLBASE/4GL file. This screen allows the user to view or update the records within the file. The module builder uses a number of templates when creating a module. These templates are stored in library applications.

The first part of this section describes the operation of an application module built by the module builder. Module operation is described from the user's perspective. Described next are two developer screens: the module builder screen and the module details screen, which you use to create an application module. The final part of the section describes how HP ALLBASE/4GL constructs a module and the templates it uses to do this.

Module Operation

Module operation complies with an application structure defined in the templates. This application structure may not match the appearance or structure of your application. However, you can modify the templates if you want to change the way the module operates. This is covered in the subsection Module Construction and Templates.

Module Capabilities

The resultant screen and logic that HP ALLBASE/4GL creates for the module allows the user to retrieve records from the file. Scanning the file allows the user to retrieve records on a first, last, next, or previous record basis.

With HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, and KSAM data files, keyed access is also provided. Keyed access allows the user to retrieve records based on full-key or partial-key reads. A full-key read retrieves a record with a value in its key field that matches the value the user entered. A partial-key read retrieves a record with a value in its key field that is equal or greater than the value the user entered.

Note	Within this section, the term **keyed files** will refer to the three file types that offer keyed access: HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, and KSAM data files. Keyed files must contain a key. Any HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, or KSAM data files that do not contain a key field cannot be used to build a module.

Record retrieval capabilities are available in both inquiry (I type) modules and maintenance (M type) modules. In maintenance modules, the screen and logic created by the module builder also allow the user to add records, delete records, and modify existing records.

The following module capabilities table summarizes the capabilities of a module, including how they are influenced by module type and file type.

Table 8-1. Module Capabilities

	File Type			
	Keyed File		Serial	
Capabilities	Unique Key	Duplicate Key	Fixed Length	Variable Length
File Scanning	Yes	Yes	Yes	Yes
Full-Key Read	Yes	Yes	No	No
Partial-Key Read	Yes	Yes	No	No
Add Records	Yes	Yes	Yes	Yes
Insert Records	No	Yes	No	No
Modify Records	Yes	Yes	Yes	No
Delete Records	Yes	Yes	No	No

It is important to realize that many aspects of the module builder, including the capabilities of the module, are influenced by both the type of HP ALLBASE/4GL file accessed and the module type.

Another capability you can include in a module is link fields in the data screen. With link fields the user can access data from up to twenty other keyed files in addition to the main file for the module. The module can only read information from these *secondary* files. In maintenance and inquiry modules, this information can be displayed on the data screen. In maintenance modules, this information can also be used for data validation.

Process

Users access the module through a menu item. The menu item executes the main process, which initializes the access mode and displays the resultant screen. This transfers control of the application to the resultant screen. Control returns to the main process when the user presses the `Commit Data` function key. The process then performs the appropriate file transaction and returns control to the data screen. The process terminates when the user presses the `Previous Menu` function key.

Module Builder 8-3

Data Screen

In an inquiry module, all fields on the resultant screen, except the first field, are display-only. In a maintenance module, all fields from the main file are input fields on the resultant screen. Fields displaying data from secondary files are always display-only fields.

The first line of the resultant screen is the banner line. It consists of the application name at the left, a screen title in the center, and the screen name at the right. The rest of the screen is used for label and field pairs. The label provides an on-screen name for the field.

The first pair is the input key field and its label, and is normally separated from the remaining screen fields. When the main file is a keyed file, the user uses this field for keyed access to the main file. In resultant screens that access serial files, this label and field pair is still separate from the remaining fields.

The format of the screen depends on the length, type, and number of fields placed on the screen. When creating the resultant screen, the module builder first attempts to place all the label and field pairs in one column down the center of the screen. If there are too many fields for this layout, the module builder then tries to place all the label and field pairs in two columns. Finally, if there are too many label and field pairs for a two column layout, the module builder tries to fit as many label and field pairs as possible onto the resultant screen.

The tabbing sequence through the screen fields is from top left to bottom right. The cursor moves across the fields in each line from left to right, and then down to the left-hand field of the next line.

Refer to Module Construction and Templates for further information on how the module builder creates the resultant screen.

File Scanning

File scanning is the only access method that is common to all HP ALLBASE/4GL file types. The user can use file scanning in both inquiry and maintenance modules. File scanning allows the user to select the first, next, previous, or last record from the file. All scanning is controlled with the function key sets. Four function keys are available to the user for

scanning: `Previous Record` , `Next Record` , `First Record` , and `Last Record` .

The function keys for scanning can be placed in one or more function key sets, depending on the file type and module type. Inquiry modules contain all the available scanning function keys in a single function key set, which is always displayed. Maintenance modules generally require two to six sets of function keys to display the available scanning keys. However, from the user's perspective, only two sets of function keys are used to display the function keys for file scanning. The first set displays the `Previous Record` and `Next Record` function keys. The second set displays the `First Record` and `Last Record` function keys. The `More Keys` function key in the first set, and the `Main Keys` function key in the second set, are used to switch between the two sets of function keys.

The following table shows the scanning operations that are available for the various file types.

Table 8-2. Scanning Capabilities by File Type

| | File Type | | | | |
| | HP ALLBASE/SQL | | KSAM & HP TurboIMAGE/iX | SERIAL | |
Scan Type	Unique Key	Duplicate Key	Unique & Duplicate	Fixed Length	Variable Length
First Record	Yes	Yes	Yes	Yes	Yes
Next Record	Yes	Yes	Yes	Yes	Yes
Previous Record	Yes	No	Yes	Yes	No
Previous Key Value	No	Yes	No	No	No
Last Record	Yes	No	Yes	Yes	No
Last Key Value	No	Yes	No	No	No

The user can always step forward through a file, either from the current record or from the first record in the file. For keyed files, the order in which successive

records are retrieved is ascending key field sequence. Records in serial files are retrieved in the physical order in which they are stored in the file, which is normally the order they are created. There is no ascending or descending key value sequence.

Keyed Access

Keyed access is available for keyed files, that is, HP ALLBASE/SQL tables, HP TurboIMAGE/iX data sets, and KSAM data files. Both inquiry and maintenance modules allow keyed access.

Where the record layout allows unique key values, the user can use keyed access to retrieve a specific record. This is done by entering a value in the key field and pressing `Return`. HP ALLBASE/4GL attempts to locate and display a record with a matching key value. This is a full key read of the main file, and is available whenever the cursor is in the key field.

If a record is located, the user can access records with key values adjacent to that entered by pressing the `Previous Record` and `Next Record` function keys. In a maintenance module, the user can modify the retrieved record or add a new record if no record is found.

Where the record layout allows duplicate key values, keyed access retrieves only the first record with a matching value. The user then presses the `Next Record` function key to step through successive records with the same key value. For HP ALLBASE/SQL tables with duplicate keys allowed, the `Previous Record` function key is replaced with the `Prev Key Value` function key. When the user presses this function key, HP ALLBASE/4GL searches back through the table for the key value preceding the value displayed. It then retrieves the first record in the table with this key value.

If after entering a value, the user presses the `Next Record` function key instead of `Return`, the partial key read function is invoked. This function attempts to retrieve the first record that matches or is greater than the contents of the key field. In all field types, except numerical and date, records are ordered by ASCII coding for retrieval.

Maintenance Modules

A maintenance module allows the user to add new records, and delete or modify existing records in the file. Each action is performed when the module is in the corresponding mode for the action. The current module mode is displayed in the label for the `Mode` function key, located in the bottom left corner of the screen. To change the module mode, the user presses the `Mode` function key until the required mode is attained.

The mode indicates the type of change that will be made to the file when the user presses the `Commit Data` function key.

HP ALLBASE/4GL automatically sets the mode based upon the previous action by the user. If required, the user can then change the mode as required to another available mode by pressing the `Mode` function key.

Files with Unique Keys

When working with keyed files with unique keys, the module has three possible modes: add, modify, and delete. When the user first enters the resultant screen, the mode is set to add.

In add mode, if the user enters a value into the key field and successfully retrieves a record, HP ALLBASE/4GL changes the mode to modify.

In modify mode, if the user enters a key value that doesn't retrieve a record, HP ALLBASE/4GL changes the mode to add, holding the entered value in the key field.

From modify mode, the user can toggle to delete mode if required.

Delete mode remains the current mode until the user changes it. When the user presses the `Mode` function key, HP ALLBASE/4GL changes the mode to modify if an existing record is displayed, otherwise it changes the mode to add.

Files with Duplicate Keys

When working with keyed files where duplicate keys are allowed, the module has three possible modes: add, modify, and delete. When the user first enters the resultant screen, the mode is set to add.

Module Builder   8-7

Add mode remains the current mode until the user changes it to modify or delete mode. While in add mode, if the user presses the `Commit Data` function key with an existing key displayed, another record with the same key value is inserted into the file.

From modify mode, if the user enters a key value that exists, HP ALLBASE/4GL retrieves the first record with that key and holds modify mode. If no record is retrieved, HP ALLBASE/4GL changes to add mode. While in modify mode, the user can change to add or delete mode.

Delete mode remains the current mode until the user changes it. When the user presses the `Mode` function key, HP ALLBASE/4GL changes the mode to modify if an existing record is displayed, otherwise it changes the mode to add.

Serial Files

When working with fixed-length serial files, the user has two available modes: add and modify. The mode is set to add when the user first enters the resultant screen.

From add mode, if the user uses any of the four function keys for scanning, HP ALLBASE/4GL changes the mode to modify. The module remains in modify mode until the user changes it to add.

With variable length serial files, the user has two available modes: add and review. Add is only available when the user is at the end of the file.

In add mode, if the user presses the `Mode`, `First Record`, or `Next Record` function key, HP ALLBASE/4GL moves to the beginning of the file and changes to review mode.

In review mode, if the user presses the `Mode` function key, HP ALLBASE/4GL moves to the end of the file and changes the mode to add.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Secondary Files

With module builder you can specify that a field in the resultant screen is linked to one of the following:

- A key field in an indexed KSAM file record.

- A key field in an HP TurboIMAGE/iX file record.

- Any field in an HP ALLBASE/SQL table record.

When creating the module, HP ALLBASE/4GL links this field to a secondary file. The secondary file can then be used to display additional information and to validate user data entry.

Secondary File Functions

For each field in the resultant screen that is linked to a secondary file, the module builder creates an after function to retrieve the appropriate record from the secondary file. In maintenance modules, the after function can also be used to validate the data entered. If the link field is an input field, the function is executed as an after function. In addition, regardless of the field type, the function is always executed as a result of the SHOW command.

Additional Information Display

After the user enters data into the link field, the after function uses that value as the key to retrieve a record from the specified secondary file. If a record cannot be retrieved, a warning message is displayed. If a record is retrieved, this data is available in the file record buffer and can be displayed on the resultant screen. Any fields displaying data from the retrieved record must appear on the screen after the link field. That is, the display fields must have a higher sequence number.

Data Validation

When you specify the link between a screen field and a secondary file, you can also specify whether that field is validated against the secondary file. When validation is specified, the secondary file record must be retrieved before the user can commit data in the link field and continue entering data into further fields. If the validation fails, an error message is displayed and the field cannot be committed until the data is validated, or the field is blank.

Module Builder   8-9

Limitations

All keyed file types can be used as secondary files. Serial files cannot be used as secondary files.

If the index field for the secondary file allows duplicate values, the first record found will be retrieved.

A file cannot be used more than once in a module. Therefore, the main file cannot be used as a secondary file within the same module, and a secondary file can only be linked to one screen field. The only limit to the number of fields that can be displayed from a secondary file is the number of fields that can fit on the screen.

A maximum of twenty secondary files per module is allowed.

Building a Module

The procedure to create a module eliminates the need to create individual items. You do not need to create a data screen, or any logic blocks to obtain a working application module. This is all done by the module builder.

The procedure required to create a module involves the following steps:

■ **Create the required dictionary items.** You need to create the field specifications, record layout, and file that the module builder is going to use for the module.

■ **Complete the module builder screen.** In this screen you define the following:

   - The module name.

   - The module type, either maintenance or inquiry.

   - The name of the main file for the module.

   - The record layout and index field for the main file.

   - Whether all the fields from the record layout are automatically included in the resultant screen.

■ **Complete the module details screen.** This screen is optional, if you specified that all fields from the record layout are placed in the resultant screen. If

you specified that not all fields will be placed on the resultant screen, you must complete this screen to specify the following details:

- Which fields from the record layout are included in the resultant screen.

- Which fields are required fields.

- Which fields are display fields.

- Links between screen fields and secondary files. These links can be used to display data from the secondary file on the resultant screen, or to validate data entered into the link field.

- **Generate the module.** By pressing the `Generate Module` function key, you execute the module builder process to construct the module.

- **Join the module to a menu.** You add a menu item which executes the module.

Module Builder Screen

This screen allows you to specify the details to create a resultant screen and associated logic blocks with the module builder, providing you with a module to inquire upon or maintain an HP ALLBASE/4GL file.

Screen Image

```
 _____
| Developer                      Module Builder                 module_builder |
| Module Name        ██████    Type     M (I/M)                                |
|                                                                              |
| Main Access:  File ░░░░░░   Record ░░░░░░░░░░░░░░░    Index ░░░░░░░░░░░░░░    |
|                                                                              |
| Include all Fields  Y (Y/N)                                                  |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|  Dict.    Screens   Logic   Reports   3* 21   System   Commit   Help  Previous|
|  Menu     Menu      Menu    Menu               Keys     Data           Menu   |
|_____|
```

Field Descriptions

Module Name.   Enter the name of the module (up to eight characters).

The module builder uses the module name, with an *mb_* prefix, to form the
name of many items within the module. For example, the resultant screen is
called *mb_name*, where *name* is the module name you enter.

Caution

✋

If you use a module name that already exists, HP
ALLBASE/4GL assumes that you want to replace the existing
module, and displays a message asking you to confirm this.
If you want to keep the module items that currently exist
under this name, you must use a different module name. If
you continue, all existing module items, such as functions,
processes, and SQL logic blocks, are overwritten.

Type   Enter I or M to specify the type of module. The default is M.

■ Enter I to allow inquiry access. The user is only able to retrieve data.

■ Enter M to allow maintenance access. The user is able to retrieve, add, delete, or modify data in the file.

Main Access: File.   Enter the HP ALLBASE/4GL name of the main file for the module. If the file is an HP ALLBASE/SQL table, HP TurboIMAGE/iX data set, or KSAM data file, the file must contain a key field. If the file does not contain a key field, the error message *"Invalid index."* is displayed.

Main Access: Record.   Enter the name of a record layout for the main access file. The module builder uses this record layout as a source for fields that will appear on the resultant screen. The entry in this field defaults to the default record layout for the file. This field is bypassed for HP ALLBASE/SQL files.

Main Access: Index.   Enter the index number or name of the field in the record layout to be used for keyed access. The following values are allowed, depending on the file type:

■ **HP ALLBASE/SQL.** Enter the name of any field in the record layout or the number of any key field on the record layout. If the field is not a key field in the record layout, the module builder assumes that duplicate values are allowed. This entry defaults to the primary key field for the record, or the first field if no keys exist. If you specify an index field that is not a key field in the record layout, the file scanning and record retrieval performance of the module will be slow.

■ **KSAM or HP TurboIMAGE/iX.** Enter the name or number of any field that is defined as a key field in the record layout. If the field is not a key field in the record layout, an error message is displayed. This entry defaults to the primary key field for the record.

■ **SERIAL.** This field is bypassed.

Include All Fields.   Enter Y or N. If you enter Y, all fields from the record layout are placed on the resultant screen. Each field will be given a default label, and a primary data movement specifier relating to its corresponding field in the main file. You can then use the module details screen to modify or delete these screen fields. If you enter N, only the first field is automatically placed on the screen. You must then use the module details screen to specify each subsequent field that you want to include on the resultant screen. The default is Y.

Function Keys

[f1]   `Dict. Menu` Exits the current screen and displays the dictionary menu.

[f2]   `Screens Menu` Exits the current screen and displays the screens menu.

[f3]   `Logic Menu` Exits the current screen and displays the logic menu.

[f4]   `Reports Menu` Exits the current screen and displays the reports menu.

[f8]   `Previous Menu` Returns you to the last menu displayed by the system. Once you have committed a field you will be asked to confirm your request to exit from the module builder. If you do exit before generating the module, all details entered so far will be lost.

After you press the `Commit Data` function key, the first four function keys are replaced with the following two:

[f1]   `Specify Details` Exits the current screen and takes you to the module details screen.

[f4]   `Generate Module` Accepts the module definition as entered so far and creates and generates all the HP ALLBASE/4GL items necessary.

Additional Information

When you press the `Commit Data` function key, the details you have entered are saved and two new function keys are displayed.

Press the `Generate Module` function key to create and generate all the items necessary for the module as defined so far. Only use this function key if you have entered *Y* in the *Include All Fields* field. Alternatively, press the `Specify Details` function key to display the module details screen. This screen allows you to specify the field details for the resultant screen on a field-by-field basis.

Module Details Screen

The module details screen allows you to specify further details for each field on the resultant screen. This screen also allows you to add additional fields to the screen, or delete fields, and to link screen fields with secondary files.

Screen Image

Screen Image

```
┌─────────────────────────────────────────────────────────────────┐
│ Developer                    Module Builder           module_details │
│                                                                   │
│ Module Name        product   Type    I (I/M)                      │
│                                                                   │
│ Main Access:  File product   Record  product_rcrd     Index  product_no │
│                                                                   │
│ Include all Fields Y (Y/N)                                        │
│ ────────────────────────── Screen Field Details ──────────────── │
│                                                                   │
│   Sequence Number    ▐1▌      Action     (A/C/D/I/L)              │
│                                                                   │
│   File                                                            │
│   Field Spec. Name                       (   )                    │
│   On Screen Label                                                 │
│   Type             (I/D)                                          │
│   Required         (Y/N)                                          │
│                                                                   │
│   Link             (Y/N)                                          │
│   Validate         (Y/N)                                          │
│                                                                   │
│   Link to:    File           Record               Index          │
│                                                                   │
│   List                    Generate  11* 21   System  Commit  Help  Previous │
│   Record                  Module             Keys    Data          Menu │
└─────────────────────────────────────────────────────────────────┘
```

Field Descriptions

The first group of fields on this screen are display only fields that show the information you specified in the module builder screen.

Sequence Number.   Enter the sequence number of the field you want to add, change, insert, or delete. The sequence number indicates position of the field in the tab sequence on the resultant screen. HP ALLBASE/4GL automatically increments this value each time the cursor enters the field. You can then overtype this value as required.

Action.   Enter the code of the action to be performed on the specified field.

Choose from one of the following:

Code	Action
A	Add a new field.
C	Change current field details.
D	Delete current field from screen.
I	Insert a field before this field.
L	List screen field details.

File.   Enter the name of the file included in the primary data movement field name. The entry in this field first defaults to the name of the main file. If all the fields from the main record layout have been placed, it defaults to the most recently specified secondary file.

Field Spec. Name.   Enter the name of a dictionary field specification. The module builder uses this name to complete the definition of the primary data movement field for the screen field. You can only specify the primary data movement path for a screen field with the module builder.

When the cursor enters the *Field Spec. Name* or *( )* fields, two function keys are displayed: `Previous Field` and `Next Field` . These function keys allow you to scan all the fields in the record layout for the file specified.

( ).   Enter the occurrence number of the field. If the dictionary field specification has not defined multiple occurrences, this field is bypassed.

On Screen Label.   Enter the label text to be placed on the screen next to or before the data field. This defaults to the text in the short description field from the field specifications screen. If the field is one of a series of multiple occurring fields, only the first occurrence defaults to this value. The others default to blank.

Type.   Enter I to make the field an input field or D to make the field display only. HP ALLBASE/4GL sets and bypasses this field in the following circumstances:

- The first field on the screen is always an input field.

- All other screen fields in inquiry modules are display-only fields.

8-16   Module Builder

■ All screen fields retrieved from secondary files are display-only fields.

The remaining fields on the module details screen are bypassed if the *Type* field contains `D`.

Required.   Enter `Y` if the user must enter data into the screen field being specified. The default is N. HP ALLBASE/4GL sets and bypasses this field in the following circumstances:

■ The first field on the screen is always a required field.

■ Any display-only field is a non-required field.

Link.   Enter `Y` if you wish to link the screen field to a secondary file. If you enter `N`, all following fields on the module details screen are bypassed. You cannot link secondary file screen fields with other secondary files, and you can only link twenty main file screen fields with secondary files. You cannot link a main file screen field that has multiple occurrences with a HP ALLBASE/SQL table.

If the link succeeds in retrieving a record in the secondary file, fields in the record retrieved from the secondary file can be displayed on the resultant screen. If you want to do this, the resultant screen fields used to display data from the retrieved record must have higher sequence numbers (that is, occur later in the resultant screen) than the field used to make the link.

The retrieval specified by the link is executed as an after function when the user commits the field. When a record from the main file is retrieved using keyed access or file scanning, the secondary file retrieval results from execution of a SHOW command.

Validate.   Enter `Y` if a record must be retrieved from the secondary file before the user can commit the link field with data entered. Setting *Validate* to *Y* means the link field is validated by data in the index field of the secondary file in the same way a validation range or validation table operates.

If the record does not exist and this field is set to Y, then an error type message is displayed. If this field is set to N, then a warning type message is displayed. The default is N.

Link To: File.   Enter the name of the secondary file with which the screen field is linked. This must not be a serial file.

Link To: Record. Enter the name of the record layout to be used to access the secondary file. The entry in this field defaults to the default record layout for the file. This field is bypassed for HP ALLBASE/SQL files.

Link To: Index. Enter the index number or name of the field in the record layout to be used for the link. The following values are allowed, depending on the file type:

- **HP ALLBASE/SQL.** Enter the name of any field on the record layout or the number of any key field on the record layout. This entry defaults to the primary key field for the record or the first field if no keys exist.

- **KSAM or HP TurboIMAGE/iX.** Enter the name or number of any field which is defined as a key field on the record layout. If the field is not a key field in the record layout, an error message is displayed. This entry defaults to the primary key field for the record.

- **SERIAL.** A serial file cannot be a secondary file.

Function Keys

(f1)   `List Record`   Displays a list of the fields in the current file's record layout. One column of the list shows whether the field is already included on the current screen.

(f2)   `Previous Field`   This function key is only displayed if the cursor is in the *Field Spec. Name* field or the *( )* field. Pressing the key displays the details for the previous field in the record layout.

(f3)   `Next Field`   This function key is only displayed if the cursor is in the *Field Spec. Name* field or the *( )* field. Pressing the key displays the details for the next field in the record layout.

(f4)   `Generate Module`   Accepts the module definition as entered so far, and creates and generates all the HP ALLBASE/4GL items necessary for the module.

(f8)   `Previous Menu`   Returns you to the last menu displayed by the system. You will be asked to confirm your request to exit. If you exit before generating the module, all details entered so far will be lost.

8-18 Module Builder

Module Construction and Templates

The module builder constructs a number of HP ALLBASE/4GL items to create a module. The following list shows the items that can be built for a module:

- A process.

- A screen.

- Function key sets.

- Functions.

- SQL logic blocks.

- Constants.

- Variables.

- Messages.

- Help screens.

The module builder creates items by copying templates, which are stored in library applications, into the current application and making the necessary modifications. A template holds all the required information to create a single item in the resultant module. Together the templates specify the module operation described at the beginning of this section. You can modify the templates to obtain module operation that suits your own needs. Refer to the Templates and Item Naming section for lists of all the templates, and refer to Template Libraries for further information on the library applications.

Module Construction

Module construction involves five steps:

1. Definition of the template type.

2. Creation of the resultant screen.

3. Creation of the main process.

4. Processing the to do list.

5. Generation of all items on the done list.

Module Builder 8-19

The to do list is an internal system list that contains the names of all the templates required to create the items for the complete module. The done list is an internal system list that contains the names of all items that module builder creates for the module.

Template Type

The templates that the module builder uses to create a module are influenced by file type and module type. These two parameters have been combined into template type. Template type is represented by a two letter code. The first letter depends on the file type, the second letter on module type.

The first letter of the template type is one of those listed below:

Code Letter	Usage
A	All file types.
D	Duplicate keyed KSAM file.
F	Fixed-length serial file.
I	Unique keyed KSAM file.
S	Unique keyed HP ALLBASE/SQL table.
T	Duplicate keyed HP TurboIMAGE/iX data set.
U	Unique keyed HP TurboIMAGE/iX data set.
V	Variable-length serial file.
Z	Duplicate keyed HP ALLBASE/SQL table.

The second letter of the template type is one of those listed below:

Code Letter	Usage
A	All modules.
E	Secondary file links with validation, maintenance module.
I	Inquiry module.
M	Maintenance module.
W	Secondary file links without validation.

Many template names include the template type in the name character string. For example, all templates with the template type ZM would have the character string "_ZM_" included in their template names, such as the process template "HP_mb_ZM_proc".

Resultant Screen

After defining the template type, the module builder searches through the library applications for the screen template that complies with the template type. The module builder then creates the resultant screen.

The layout of the resultant data screen depends on whether you specified *Y* or *N* at the *Include All Fields* prompt in the module builder screen. If you specified *Y*, HP ALLBASE/4GL attempts to fit all the fields in the record layout for the main file onto the resultant screen. If you specified *N*, HP ALLBASE/4GL only places one field in the resultant screen. You must specify the remaining screen fields with the module details screen.

The first line of the resultant screen is the banner line. It consists of the application name (communication area field *SUITE) at the left, a screen title in the center, and the screen name (communication area field *SCREEN) at the right. The screen title consists of the short description field from the dictionary definition of the main file and some literal text.

The rest of the resultant screen is used for label and field pairs. The text for the label is the text in the short description field on the field specification screen. The first pair is the input key field and its label. This label and field

Module Builder   8-21

pair are placed at the top of the resultant screen, separate from the remaining fields. Modules accessing serial files do not have keyed access, but the module builder still places this first field and label pair separate from the remaining fields.

The format of the resultant screen depends on the length, type, and number of fields placed on the screen. If there are only a small number of fields to be placed on the screen, the field and label pairs are in one column down the center of the screen. The alignment column is determined by the center point of the longest label field pair. The vertical spacing is adjusted so that the fields remain as one block in the center of the screen.

If a field and its label don't fit on one line, the field appears on the line immediately under the label. If a field is too large to fit on one line (that is, greater than 78 characters in length), it is omitted from the screen and a message is displayed. This message is written to the trace log file.

If there are more than 18 fields, a two column format is used. Where a label and field pair will not fit in one of these two columns, it occupies the entire line. If there is insufficient space for a two column format the module builder packs as many fields as possible onto the screen. Starting at the second line of the screen, it places label and field pairs, separated by a single space, from top left to bottom right.

The tabbing sequence through the screen fields is from left to right, top to bottom, with screen wrap-around from bottom right to top left.

The module builder places a higher priority on placing all fields on the screen than on the appearance of the screen. Where there is sufficient space for all fields, it attempts to make the screen image as intelligible as possible. You can modify the layout of the screen, once it has been built, using the screen painter.

If there is insufficient space to fit all fields on the screen, HP ALLBASE/4GL displays a message for each field that is omitted. These messages are also written to the trace log file.

When the module builder has completed creating the resultant screen, it generates the screen, and places the template names of any function key sets, help screens, or after functions on the to do list. The to do list specifies the items required for the module, listing them as template names. The module builder then creates the after functions for linking fields in the resultant screen with secondary files.

8-22   Module Builder

If the screen template does not define a required set of function keys, the module builder adds the template name of a standard set to the to do list. The template name for the standard set has a suffix: _key1 .

Main Process

After creating the resultant screen, the module builder then searches for the main process template. It copies this template in memory, performs any macro substitution and name substitution required, renames it, and then copies it into the current application. The process name is then added to the done list for subsequent generation. Refer to Macro and Name Substitution and to Naming Templates and Items for further information.

To Do List Processing

Each template on the to do list is processed individually following the sequential procedure below:

1. The module builder searches through the application hierarchy for a template where the name of the template matches the name on the to do list. If it cannot find the required template, the module builder substitutes the second letter of the template type specified in the to do list name with "*A*", and searches again for the template. If it still cannot find a matching template, the module builder replaces the template type contained in the to do list name with "*AA*" and searches again. If it doesn't locate a template this time, the module builder displays an error message and moves to the next item on the to do list. The application hierarchy is described in Template Libraries.

2. When the template is found, the module builder copies it into memory.

3. The module builder then performs any macro substitution required for the template. Macro substitution is followed by name substitution, if it is required, which can result in the addition of template names to the to do list. Refer to Macro and Name Substitution for further information.

4. The completed item is copied into the current application with the correct name based on the item naming rules, which are covered in the subsection Naming Templates and Items.

5. The name of the item is added to the done list.

6. The module builder moves on to the next item in the to do list.

To do list processing finishes when there are no further items on the to do list.

Item Generation

When the module builder has completed the to do list, it generates all the items on the done list that require generation. This completes the construction of the module. If the construction proceeded without errors, HP ALLBASE/4GL displays a message stating that the module was successfully created; otherwise, it displays a message stating that errors occurred and that the module may not function correctly.

Template and Item Naming

Template Naming

Template names follow a set of rules that allow easy specification of those templates required for a particular module. These naming rules also allow systematic searching through the library applications for a required template. Unlike item naming, template naming is not an automatic process. If you create templates for the module builder to use, you must give them names that follow the naming rules.

Template names must comply with the following rules:

■ All template names begin with *HP_mb_*.

■ If use of the template by the module builder is dependent on the file type and module type, the remaining characters in the name follow these rules:

  - Next in the template name is the two letter template type code. The template type code is described in the section called Module Construction.

  - The final part of the template name is an underscore (_) character and a four-character suffix indicating the function of the item created from the template in the resultant module.

  - Names for the resultant screen templates have the suffix *_scrn*.

  - Names for the main process templates have the suffix *_proc*.

  - Name suffixes for functions are listed in table 8-4.

8-24   Module Builder

- Name suffixes for SQL logic blocks are listed in table 8-5.

■ As the use of variables, constants, and messages in the resultant module is not dependent on the file type, the final part of the template names for these items is a 10-character suffix. This suffix indicates the function of the item in the resultant module. Refer to table 8-6 for a list of templates for variables and constants. Refer to table 8-7 for a list of templates for messages.

Item Naming

When the module builder copies a template into the current application to create an item, it renames it. The name given also follows a set of rules. These rules result in item names that are influenced by the name of the template that created it and the module name you specify.

Item names comply with the following rules:

■ The names of **all** items created by the module builder begin with $mb\_$.

■ The resultant screen and the main process are both named mb_name, where $name$ is the module name you specify.

■ Names for functions, SQL logic blocks, and other processes are called mb_name_xxxx, where $name$ is the module name and $xxxx$ indicates the items function in the resultant module. The suffixes for functions are shown in table 8-4. The suffixes for SQL logic blocks are shown in Table 8-5.

■ After functions for link fields are named $mb\_name\_l\_yy$, where $name$ is the module name and yy is the sequence number of the field in the resultant screen.

■ The messages, constants, and variables created all have fixed names. Names for messages are listed in table 8-7. Names for variables and constants are listed in table 8-6. These tables also list the names of the templates from which the items come.

■ Help screens have the name mb_XX_scrY, where XX is the template type and Y is the help screen number for that template type. There may be more than one help screen for each template type.

Template Libraries

The module builder searches for a required template through a hierarchy of four applications used to store all templates used for module building. First in the hierarchy is the current application, followed by the three library applications. When the search for a template fails at one level, the module builder searches the next level in turn.

Library applications are only used to store templates for the module builder, they cannot be run as stand alone applications.

The order in which these applications are searched is as follows:
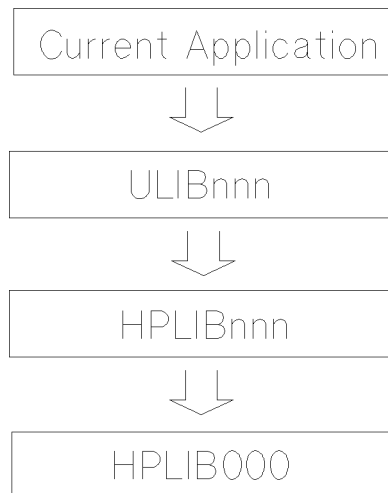
Current Application

The module builder first searches for templates in the application you are currently developing. This means that you can define templates that specifically match the requirements of your current application, and place them in the current application.

Site Customizable Template Library

The module builder next searches a user library application. This user library application is called *ULIBnnn*, where *nnn* is the three digit code referring to

your language identifier. In this library you store templates that you have created for use in several applications.

Native Language Template Library

Your localization center may have prepared local language versions of the templates. If this is true, you will also have an application called *HPLIBnnn*, where *nnn* is the three digit code referring to your language identifier. Refer to the Native Language Support (NLS) documentation supplied with you operating system for more information. The HP ALLBASE/4GL system administrator can use the administrator printing facility to see which HPLIB applications exist on your system.

Standard Template Library

By default, all required templates exist in an application called *HPLIB000*. This application is a part of the HP ALLBASE/4GL product.

Once you have access, use the developer printing utilities to print the contents of the HPLIB applications.

Note	Do not modify the templates in the HPLIB library applications. If you want to modify the templates, copy them to the ULIB application or the current application and modify them there.
	If you modify the HPLIB application, you will need to re-install a new set of S-files or reload the HPLIB application from a clean set of S-files.

Macro and Name Substitution

Within application items there are cross references to fields, record layouts, files, SQL table items, and other application items. The names of all these are unique for each module. To ensure that all references are resolved, there are macros and template names in the templates that are substituted with unique names during the module building process. Macros refer to files and their components, and SQL table items, while template names refer to other application items.

Processes and Functions

After the module builder copies the templates for processes and functions into the current application, it executes macro substitution as follows:

@DATABASE@
Replaced by the name of the HP TurboIMAGE/iX database associated with the main data set, or the name of the HP TurboIMAGE/iX database associated with the secondary data set when processing a _link_ template.

@FILE@
Replaced by the name of the main file, unless processing a _link_ template, in which case it is replaced by the name of the secondary file.

@RECORD@
Replaced by the name of the record layout for the main file, or the secondary file when processing a _link_ template.

@INDEX@
Replaced by the name of the index field for the main file, or the secondary file when processing a _link_ template.

@LINK@
Only valid when processing a _link_ template function. It is replaced by the name of the SQL logic block associated with this function.

@SCREEN@
Replaced by the name of the resultant screen item.

Macro substitution is performed line-by-line through the logic block. Macro substitution is done before name substitution. All valid substitutions are made, including those within quote strings and comments. You can quote a single @ by using @@.

Macros can be placed within the DEFINE command. Thus this line would work in a template for a logic block:

```
DEFINE %KEY% F-@INDEX@.@FILE@.@RECORD@.
```

In the resultant item, macro substitution would produce the following line:

```
DEFINE %KEY% F-index_field.file_name.record_layout.
```

When performing name substitution, the module builder searches through the logic block and replaces all template names with the corresponding application item name. Name substitution follows the naming rules detailed in the section on Item Naming earlier in this chapter. When the module builder finds a

template name, it checks whether the name is on the to do or done lists, and if not, places the name on the to do list.

If the module builder finds an unknown macro or an unmatched @ during the construction process, it displays a warning message.

SQL Logic Blocks

There are a number of additional macros used in SQL logic blocks. These are:

@FIELD@	Replaced by the field specification name of the current screen field when processing a *_link* template.
@TABLE@	Replaced by the name of the current SQL table.
@COLUMNS@	Replaced by a list of SQL table column names to update, excluding the index field.
@VALUES@	Replaced by a list of screen fields that correspond to SQL table column names.
@UPDATECLAUSE@	Replaced by a list of fields to update. Formatting for the list is "column name = screen field."

When the module builder is processing a *_link* template the values relevant to the secondary file are substituted. Name substitution is not relevant to SQL logic blocks.

Function Key Sets

When the module builder creates the function key sets in the current application it performs name substitution for all the functions called.

Screen Titles

Two templates are used for screen titles: one for maintenance modules and one for inquiry modules. The @FILE@ macro is used in these titles, and is replaced with the main file name. The module builder shifts the first character of the name to uppercase when it used in a screen banner line. These templates are not copied into the module. After substitution, the text becomes a literal on the resultant screen.

Module Builder 8-29

Templates

This subsection details the templates that exist in the *HPLIB000* library application. The purpose of making these templates available to you as a developer is to enable you to copy the templates into ULIB*nnn* or your current application, and modify their logic and layout to suit your needs. You can use the developer utilities to list the items within template library applications and copy them from the HPLIB library applications.

Note

Do not modify the templates in the HPLIB library applications. If you want to modify the templates, copy them to the ULIB application or the current application and modify them there.

If you modify an HPLIB application, you will need to re-install a new set of S-files, or reload the unmodified HPLIB application from a clean set of S-files.

Screens

There is a screen template for each template type. The entire screen is overwritten when the module builder creates a new screen. Any changes you make to the screen layout template are overwritten when a new screen is created. You can modify the screen description, and you can modify the screen that is produced for a module. Many screens have sets of function keys associated with them. These function key sets partly define the operation of the resultant module.

All screen templates have the name *HP_mb_XX_scrn*, where *XX* is the template type code. The name of the resultant screen in the module is *mb_name*, where *name* is the module name you specify.

Function Key Sets

There are a number of templates for function key sets. Some are associated with particular screen templates, and some are sets of standard function keys. The functionality of the resultant module revolves around the functions called from the function key sets. If you want to customize modules, you must modify these application items.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Names for function key set templates all begin with $HP\_mb\_XX\_$, where XX is the template type. The name is completed with a four-character suffix from the following table. The name of the resultant application item begins with mb_name, where $name$ is the module name, and ends with the same four-character suffix.

Table 8-3. Function Key Template and Resultant Item Suffixes

Suffixes	Comment
$\_key1$	Function key set for scanning in inquiry modules.
$\_add1$	First set of function keys used when in add mode.
$\_add2$	Second set of function keys used when in add mode.
$\_mod1$	First set of function keys used when in modify mode.
$\_mod2$	Second set of function keys used when in modify mode.
$\_del1$	First set of function keys used when in delete mode.
$\_del2$	Second set of function keys used when in delete mode.
$\_rev$	Set of function keys used when a module accessing a variable length serial file is in review mode.

Maintenance modules often require a pair of function key sets for each suffix because the function keys for file scanning cannot be presented in one set.

Processes and Functions

There is a separate process template for each template type. There are several function templates. Each resultant module contains a single main process and a number of functions. The functions required depend on the template type, the module functionality, and any linking to secondary files. For example, some function templates are used for screen field after functions and some for functions called by function key sets.

You can use all the logic commands available in HP ALLBASE/4GL within process and function templates. However, you cannot place logic command synonyms within templates. The module builder doesn't support all HP

ALLBASE/4GL items. The items not supported are listed at the end of this section.

All main process templates have the name *HP_mb_XX_proc*, where *XX* is the template type code. The name of the resultant process in the module is *mb_name*, where *name* is the module name you specify.

Names for function templates all begin with *HP_mb_XX_*, where *XX* is the template type. The name is completed with a four-character suffix from the following table. The name of the resultant item begins with *mb_name_*, where *name* is the module name, and ends with a four-character suffix.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Table 8-4. Function Template and Resultant Item Suffixes

Template Suffix	Item Suffix	Comment
_addm	_addm	Changes the module to add mode after clearing the screen and buffers.
_curs	_curs	Opens a cursor in a SQL table.
_del	_del	Deletes the currently displayed record.
_err	_err	Displays errors reported by SQL.
_find	_find	Performs a generic read.
_frst	_frst	Retrieves first record in the file.
_last	_last	Retrieves last record in the file.
_link	_l_yy	Used to retrieve record in secondary file with index value from field *yy*.
_lock	_lock	Reads the current record with lock.
_mode	_mode	Changes the current module mode.
_modi	_modi	Modifies the currently displayed record.
_modm	_modm	Changes the module to modify mode.
_next	_next	Retrieves the next record from the file.
_prev	_prev	Retrieves the previous record from the file.
_quit	_quit	Quits the module and exits to the previous menu.
_read	_read	Performs an absolute read.

SQL Logic Blocks

SQL logic blocks are required when either the main file or a secondary file is an SQL table. Templates for SQL logic blocks only require macro substitution when the template is copied into the current application.

Names for SQL logic block templates all begin with $HP\_mb\_XX\_$, where XX is the template type. The name is completed with a four-character suffix from the following table. The name of the resultant item begins with $mb\_name\_$, where $name$ is the module name, and ends with a four-character suffix.

Table 8-5.
SQL Logic Block Template and Resultant Item Suffixes

Template Suffix	Item Suffix	Comment
$\_curs$	$\_curs$	Selects records with the current key for inquiry or update.
$\_del$	$\_del$	Deletes the currently displayed record.
$\_expl$	$\_expl$	Retrieves the next SQL error.
$\_find$	$\_find$	Selects all records with index values greater than, or equal to, the displayed key value, in index order.
$\_frst$	$\_frst$	Selects all records from the table in index order.
$\_ins$	$\_ins$	Inserts a new record in the table.
$\_last$	$\_last$	Selects all records from the table in reverse index order.
$\_link$	$\_l\_yy$	Selects record in secondary file with index value from field yy.
$\_modi$	$\_modi$	Updates the currently displayed record.
$\_next$	$\_next$	Selects all records after the current index value, in index order.
$\_prev$	$\_prev$	Selects all the records before the current index value in reverse index order.
$\_read$	$\_read$	Selects the current record for inquiry or update.

Variables and Constants

There are a number of templates for variables and constants. Templates for variables and constants do not have the template type code in their names.

The following table lists all the templates for variables and constants, and shows the names of resultant items.

The following table uses these conventions:

* Used in this module.

D Used if the main file index allows duplicates.

U Used if the main file index is unique.

Table 8-6. Templates for Variables and Constants

Item Name	Template	HP ALL-BASE /SQL		KSAM & HP Turbo-IMAGE /XL		Serial (F)		Serial (V)	
Variables		M	I	M	I	M	I	M	I
mb_curr_mode	HP_mb_curr_mode	*		*		*		*	
mb_got_cursor	HP_mb_got_cursor	D	D						
mb_got_record	HP_mb_got_record	U	U	*	*		*		*
mb_quit_flag	HP_mb_quit_flag	D							
mb_undo_done	HP_mb_undo_done	*	*						
Alpha-numeric Constants									
mb_add	HP_mb_add	*		*		*		*	
mb_delete	HP_mb_delete	*		*					
mb_modify	HP_mb_modify	*		*		*			
mb_no	HP_mb_no	*	*	*	*		*		*
mb_review	HP_mb_review							*	
mb_yes	HP_mb_yes	*	*	*	*		*		*
Numeric Constants									
mb_bad_insert	HP_mb_bad_insert			U					
mb_eof	HP_mb_eof			*	*	*	*	*	*
mb_no_file	HP_mb_no_file						*		*
mb_no_record	HP_mb_no_record			*	*				
mb_rec_lock	HP_mb_rec_lock			*					
mb_sql_badins	HP_mb_sql_badins	U							
mb_sql_eof	HP_mb_sql_eof	*	*						
mb_sql_no_rec	HP_mb_sql_no_rec	D							

Messages

The messages are created only once in each application. You can modify the text of the messages in the template applications. It is best not to change the message type as this may affect the flow of control when the message is displayed. The message names in the resultant module and the message template names are listed in the table of message usage. This table also details where module builder uses each message.

The following tables use these conventions:

* Message used in this module.

D Message used if main file index allows duplicates.

U Message used if main file index is unique.

Table 8-7. Message Usage by Template Type

Message		File Type and Module Type							
		HP ALL-BASE SQL		KSAM & HP Turbo-IMAGE /XL		Serial (F)		Serial (V)	
Name	**Template**	**M**	**I**	**M**	**I**	**M**	**I**	**M**	**I**
mb_add_ok	HP_mb_add_ok	*		*		*		*	
mb_avail_rec	HP_mb_avail_rec			D					
mb_bad_insert	HP_mb_bad_insert	U			U				
mb_begin	HP_mb_begin	*	*	*	*	*	*		
mb_blank_key	HP_mb_blank_key	D	D						
mb_blank_mess	HP_mb_blank_mess	*	*	*	*	*	*	*	*
mb_chg_mode	HP_mb_chg_mode	*		*		*		*	
mb_del_ok	HP_mb_del_ok	*		*					
mb_empty_file	HP_mb_empty_file	*	*	*	*	*	*	*	*
mb_end	HP_mb_end	U	U	*	*	*	*		
mb_end_chgmde	HP_mb_end_chgmde							*	
mb_eof	HP_mb_eof								*
mb_io_status	HP_mb_io_status			*	*	*	*	*	*
mb_last_key	HP_mb_last_key	D	D						
mb_mod_ok	HP_mb_mod_ok	*		*		*			
mb_more_rec	HP_mb_more_rec							*	*
mb_no_file	HP_mb_no_file					*		*	
mb_no_modechg	HP_mb_no_modechg	*		*					

FINAL TRIM SIZE : 7.0 in x 8.5 in

Table 8-7. Message Usage by Template Type (continued)

Message		File Type and Module Type							
		HP ALL-BASE SQL		KSAM & HP Turbo-IMAGE /XL		Serial (F)		Serial (V)	
Name	Template	M	I	M	I	M	I	M	I
mb_no_modify	HP_mb_no_modify						*		*
mb_no_record	HP_mb_no_record	*	*	*	*				
mb_rec_lock	HP_mb_rec_lock			*					
mb_sql_status	HP_mb_sql_status	*	*						
mb_link_error	HP_mb_link_error	Included when a field is linked to a secondary file, with *Validate* = Y.							
mb_link_warn	HP_mb_link_warn	Included when a field is linked to a secondary file, with *Validate* = N.							

Help Screens

Each template type has a number of help screens that are copied into the module for the user to access while running the module. The help screens are only copied into the current application if they don't already exist. Information in the help screens assumes that the module will operate in the manner defined by the templates in the *HPLIB* library applications. If you change the functionality of the module, you will need to change the relevant help screens.

Help screen templates have the name *HP_mb_XX_scrY*, where *XX* is the template type code and *Y* is the number of the help screen required for the template type. Some template types require more than one help screen in the module. The module help screens item names are "*mb_XX_scrY*, where *XX* and *Y* are as defined above.

Application Titles

These are a special case. They are not created in the resultant application, but are used to create the text for the banner line in the resultant screen. The module builder searches for and replaces the string :FILE: within the application title with the short description of the module's main file. The two application titles are as follows:

HP_mb_inq_banner. This text is centered at the top of the resultant screen for all inquiry modules. The default value for *HP_mb_inq_banner* is:

                    @FILE@ Inquiry Screen

HP_mb_mnt_banner. This text is centered at the top of the resultant screen for all maintenance modules. The default value for *HP_mb_mnt_banner* is:

                    @FILE@ Maintenance Screen

Items Not Supported

There are a number of application items that the module builder cannot include in the module. This means that the functionality associated with these items cannot be automatically included in the module; however, you can modify a resultant module to include these application items. You may have to regenerate some or all of the module after you make modifications. Those items not supported by the module builder are:

■ Application titles (except in the special case defined above).

8-40   Module Builder

- Calculated items.

- Decision tables.

- Master titles.

- More then one resultant screen.

- Reports.

- Scratch pad field names.

- SQL select lists.

- Validation ranges.

- Validation tables.

- Work areas.

Also, the module builder cannot support synonyms for logic commands.

9

HP ALLBASE/SQL Interface

Introduction

This section describes the facilities for accessing and using HP ALLBASE/SQL databases from HP ALLBASE/4GL applications.

The interface between HP ALLBASE/4GL and HP ALLBASE/SQL uses a number of components in the HP ALLBASE/4GL system. The components are:

- Dictionary base table definitions and select list definitions.

- SQL logic blocks.

- The FILE, SQL and TRANSACT logic commands.

This section concentrates on SQL logic blocks and the techniques for using HP ALLBASE/SQL data in HP ALLBASE/4GL.

Refer to chapter 5 for information about dictionary base table definitions and select list definitions.

Refer to chapter 12 for information about HP ALLBASE/4GL logic commands.

Database Environments

HP ALLBASE/4GL can access HP ALLBASE/SQL database environments anywhere in the host file system. The name and location of a database is determined by the name and location of the DBECon file for the database.

The database for an application is specified on the application definition screen in the HP ALLBASE/4GL administrator.

The application definition screen in the HP ALLBASE/4GL administrator also allows the administrator to specify the name of the SQL owner group for the application. HP ALLBASE/4GL transfers ownership of all tables created from HP ALLBASE/4GL to this owner group. HP ALLBASE/4GL also transfers ownership of the module for an application or version to this owner group.

The database environment for an application must exist, and must contain at least one DBEfileset and at least one DBEfile before you can start developing the application.

HP ALLBASE/4GL uses the MPE/iX variable HP4SQLPATH to determine the location of the database DBECon file.

The HP4SQLPATH variable can specify an account name and a group name for the DBECon file.

If the DBECon file name specified on the application definition screen is a fully qualified name (that is, a file name in the form FILE.GROUP.ACCOUNT), the HP4SQLPATH variable is ignored. If the DBECon file specification on the application definition screen does not include a group and account name, HP ALLBASE/4GL appends the current value of the HP4SQLPATH variable to the file name to determine the name and location of the database environment for the application.

If HP4SQLPATH is not defined, HP ALLBASE/4GL expects to find the DBECon file in the user's current group and account.

Multi-User Database Environments

Databases used with HP ALLBASE/4GL should be configured as multi-user databases during application development.

HP ALLBASE/4GL connects to the database environment when you generate an SQL logic block or create a table from HP ALLBASE/4GL. The connection is maintained for the remainder of the HP ALLBASE/4GL session.

If you use the application testing option to run the application during development, HP ALLBASE/4GL attempts to connect to the database environment a second time. This causes an error condition if the database environment is configured as a single user database.

In the HP ALLBASE/4GL run-time environment, database environments can be configured for single user operation if required.

SQL Logic Blocks

SQL logic blocks are the mechanism for passing SQL commands to HP ALLBASE/SQL from HP ALLBASE/4GL. SQL logic blocks consist of a header and the SQL logic block details. The header defines the name of the SQL logic block, and the SQL logic block details contains the SQL commands. The commands contained in an SQL logic block are invoked by a command (the SQL command) in an HP ALLBASE/4GL logic block.

You must generate an SQL logic block before it can be used in an application.

The following descriptions of HP ALLBASE/4GL SQL logic blocks assume that you are familiar with the basic syntax and usage of SQL commands. Refer to the *HP ALLBASE/SQL Reference Manual* for more information about SQL commands.

SQL Block Commands

HP ALLBASE/4GL imposes some limitations on the SQL commands you can use in an SQL logic block.

You cannot use the following commands in an HP ALLBASE/4GL SQL logic block:

> BEGIN DECLARE SECTION and END DECLARE SECTION
> BULK
> CLOSE *cursorname*
> CONNECT TO *DBEnvironmentname*
> DECLARE *cursorname*
> DESCRIBE *commandname* INTO *areaname*
> EXECUTE IMMEDIATE
> FETCH
> INCLUDE SQLCA or SQLDA
> INTO clause with SELECT command
> OPEN *cursorname*
> PREPARE
> RELEASE
> START DBE and STOP DBE
> WHENEVER

If an SQL logic block contains a SELECT command, it must be the only command in the SQL logic block.

In general, you can use any other valid SQL commands in an SQL logic block, subject to the limitations of SQL itself. If an SQL logic block does not contain a SELECT command, it can contain up to eight SQL commands that generate into stored database sections. The SQL logic block can also contain additional SQL commands that do not generate into stored database sections. The SQL commands that do not generate into stored database sections are:

BEGIN WORK.

COMMIT WORK.

ROLLBACK WORK.

SAVEPOINT.

Host Variable Referencing

SQL logic blocks can contain references to HP ALLBASE/4GL file record fields, work area fields, variables and calculated items, and constants.

Note	You cannot use references to scratch-pad fields or communication area fields in SQL logic blocks, and you cannot use host variable referencing to replace the name of a table.

SQL logic blocks can also contain references to screen fields, provided the reference is a fully qualified reference by name. References to screen fields must be in the form:

S-field_name.screen_name

You cannot use substrings in references to screen field names.

You must precede a reference to an HP ALLBASE/4GL item with a colon (:). That is, you must express the reference in the format:

:data_ref

where *data_ref* is the HP ALLBASE/4GL name, including the item type prefix, for the HP ALLBASE/4GL item.

References to host variables cannot include null indicator variables. However, HP ALLBASE/4GL does support reading and writing null values, so you can use the *NULL argument for the IF command to test for null status in HP ALLBASE/4GL. You can also use the *NULL argument for the MOVE command to set the HP ALLBASE/4GL indicator variable associated with an HP ALLBASE/SQL table or select list buffer field to the null status.

The SELECT Command

HP ALLBASE/4GL requires a slightly modified format for the SELECT command in SQL logic blocks. You must use the SELECT command in the following format:

SELECT $\left\{ \begin{array}{l} :table\_name \\ :select\_list \end{array} \right\}$ FROM clauses [other clauses]
[FOR UPDATE OF *column_name* [, *column_name* ...]];

In this expression, *:table_name* is an HP ALLBASE/SQL table name that has been defined on the file/SQL table definition screen in the dictionary, and *:select_list* is a select list name that has been defined in the dictionary. A host variable reference cannot be used to specify the name of the table or select list.

You cannot use the SELECT * FROM ... form of the SELECT command in an SQL logic block, and you cannot use the INTO clause in a SELECT command.

At generate time, HP ALLBASE/4GL converts this command to a DECLARE CURSOR command in the format:

DECLARE *SQL_block_name* `CURSOR FOR` *select_command*
    `FOR UPDATE of ...;`

where *SQL_block_name* is the name of the SQL logic block containing the SELECT command.

If a SELECT command contains a FOR UPDATE OF clause, it cannot contain a GROUP BY clause, or an ORDER BY clause.

Implicit SQL Commands

HP ALLBASE/4GL issues some SQL commands automatically as they are required. These commands are:

- CONNECT and RELEASE
- Cursor commands such as DECLARE CURSOR, OPEN CURSOR, and CLOSE CURSOR
- BEGIN WORK and COMMIT WORK

CONNECT and RELEASE

In the developer environment, HP ALLBASE/4GL automatically connects to the database environment specified for the application when you create or delete a table from within HP ALLBASE/4GL, or generate an SQL logic block.

HP ALLBASE/4GL also connects to the application database environment when you run an application under the application testing mode.

HP ALLBASE/4GL releases from the database environment automatically at the end of the session.

At run-time, HP ALLBASE/4GL automatically connects to the database environment when the user executes a command that accesses the database, and releases from the database environment at the end of the session.

Cursor Commands

HP ALLBASE/4GL issues OPEN CURSOR and CLOSE CURSOR commands automatically during run-time execution of SQL logic blocks.

When HP ALLBASE/4GL generates a SELECT command in an SQL logic block, the command is converted to a DECLARE CURSOR for SELECT command. At run-time the SQL logic block SELECT command initiates an OPEN CURSOR command.

Executing an SQL logic block that opens a cursor on a base table or a select list automatically closes any cursor that is already open on the same base table or select list.

BEGIN WORK and COMMIT WORK

HP ALLBASE/4GL issues an implicit BEGIN WORK command whenever an SQL logic block is executed or it encounters a TRANSACT *BEGIN command. In general, you should include an explicit COMMIT WORK command in an SQL logic block to commit any work performed in an HP ALLBASE/4GL logic block.

Any implicit BEGIN WORK issued by HP ALLBASE/4GL invokes the default repeatable read locking mode. If you want the transaction to use the cursor stability locking mode, you must explicitly issue the BEGIN WORK CS command in an SQL logic block.

HP ALLBASE/4GL issues an implicit COMMIT WORK command whenever it encounters a TRANSACT *END command in an HP ALLBASE/4GL logic block.

HP ALLBASE/4GL issues an implicit ROLLBACK WORK command whenever it encounters a TRANSACT *UNDO command in a logic block.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Refer to HP ALLBASE/SQL Transaction Management for more information about transaction management with HP ALLBASE/SQL.

Using HP ALLBASE/SQL Data

In combination, the FILE command and HP ALLBASE/SQL commands in SQL logic blocks allow you to retrieve records from HP ALLBASE/SQL tables and views, insert new records in base tables, modify existing base table records, and delete base table records.

The following paragraphs show some techniques for manipulating HP ALLBASE/SQL data in HP ALLBASE/4GL applications.

Retrieving HP ALLBASE/SQL Data

HP ALLBASE/4GL performs all data retrieval from HP ALLBASE/SQL databases via cursors.

To retrieve data from an HP ALLBASE/SQL database, you must use a SELECT command in an SQL logic block, followed by a FILE *NEXT command in an HP ALLBASE/4GL logic block. These commands take the form:

> **SQL** *sql_block_name* [;command]
> **FILE** *NEXT *file_name* [;command]
>   .
>   .
>   .

The SQL command calls the SQL logic block *sql_block_name*. To retrieve data from an HP ALLBASE/SQL database, this SQL logic block must contain a SELECT command as the only command.

The SELECT command must be in the following format:

$$\text{SELECT } \left\{ \begin{array}{l} :table\_name \\ :select\_list \end{array} \right\} \text{FROM } clauses \left[other\ clauses \right]$$
$$\left[\text{FOR UPDATE OF } column\_name \left[, column\_name \ldots \right] \right];$$

In this expression, *:table_name* is an HP ALLBASE/SQL table name that has been defined on the file/SQL table definition screen in the dictionary, and *:select_list* is a select list name that has been defined in the dictionary. A host variable reference cannot be used to specify the name of the table or select list. If later commands in the HP ALLBASE/4GL logic block update the HP ALLBASE/SQL table or select list, you must include the FOR UPDATE OF clause in the SELECT command.

At generate time, HP ALLBASE/4GL converts the SELECT command to a DECLARE CURSOR for SELECT command. At run-time, the SELECT command performs the same function as an OPEN CURSOR command in HP ALLBASE/SQL.

The FILE *NEXT command retrieves the first record in the active set for the cursor defined by the SELECT command. In this command, *file_name* is the name of the table or select list in the SELECT command. The record is retrieved into the record buffer for the select list or table. The FILE *NEXT command performs the same function as an SQL FETCH command.

Further commands in the HP ALLBASE/4GL logic block can then use the retrieved data as required. Subsequent FILE *NEXT commands retrieve records in sequence from the active set for the cursor.

Example 1 - Retrieving HP ALLBASE/SQL Data

Function *retrieve_titles*

```
       .
       .
 6 SQL select_titles
 7 FILE *NEXT titles
 8 SHOW *REFRESH
       .
       .
```

SQL Block *select_titles*

```
SELECT :titles FROM library.titles
WHERE index_no = :V- index_no;
```

This example shows the use of a SELECT command in an SQL logic block, followed by a FILE *NEXT command in an HP ALLBASE/4GL logic block, to retrieve a record from an HP ALLBASE/SQL table. In this example, the SQL logic block *select_titles* declares and opens a cursor on the HP ALLBASE/SQL base table *titles*. The FILE *NEXT command in the HP ALLBASE/4GL function *retrieve_titles* retrieves the first record in the active set for the SQL cursor. The SHOW *REFRESH command following the FILE command displays the details of the record retrieved by the FILE command.

Adding New Records

The FILE *INSERT command and the SQL INSERT command allow you to insert new records into an HP ALLBASE/SQL table.

FILE *INSERT Command

You can insert a row into an HP ALLBASE/SQL base table with the FILE *INSERT command in an HP ALLBASE/4GL logic block. The command has the format:

FILE *INSERT *file_name*

where *file_name* is the HP ALLBASE/4GL name of a base table. This command inserts the current record buffer contents into a base table as a new record. You must specify that the file is available for update by including a MODE *WRITE command in the same HP ALLBASE/4GL process.

Example 2 - Adding New HP ALLBASE/SQL Records

Process *add_titles*

1 **MODE** *WRITE titles

```
     .

     .

     .

 5 FILE *INSERT titles; ENTER 10
 6 SQL commit

     .

     .

10 VISIT sql_error
```

The FILE command in this HP ALLBASE/4GL logic block inserts the current
contents of the record buffer for the file *titles* into the base table whose name
is specified on the dictionary file definition screen for the HP ALLBASE/4GL
file *titles*. If the record already exists, the optional error command on the FILE
command is executed and control passes to step 10.

SQL INSERT Command

You can also insert a new row into an existing a base table by using the SQL
INSERT command in an SQL logic block. The INSERT command must be in
the following format:

INSERT INTO *table_name* [(*column_name* ...)]
           VALUES (*SingleRowValues*);

In this expression, *table_name* is the external name of the table. You can use a
fully qualified name of the form *owner.table_name* in this command.

Example 3 - Adding New HP ALLBASE/SQL Records

This example shows the use of the SQL INSERT command to insert a new
record into a base table.

SQL Logic Block *titles*

```
 INSERT INTO library.titles
  (ISBN, TITLE, AUTHOR)
  VALUES (:V-isbn_no,:S-title.addtitle, :S-author.addtitle );
COMMIT WORK;
```

This SQL logic block contains an INSERT command that inserts a new record into the table *library.titles*. This command inserts the current values of the HP ALLBASE/4GL variable *V-isbn_no*, and the current contents of the screen fields *title* and *author* on the screen *addtitle*, into the corresponding columns of the new record. Note that references to HP ALLBASE/4GL screen fields in SQL logic blocks must use the fully qualified screen field name. You cannot use references by number, or relative number.

Updating HP ALLBASE/SQL Data

To update an existing record in a base table, you must use the HP ALLBASE/SQL UPDATE command in an SQL logic block. You can perform the update via an SQL cursor, or use the UPDATE command without a cursor.

Updating Via a Cursor

To update a base table via a cursor, the UPDATE command must be in the following format:

> **UPDATE** *table_name* SET *column_name* = *value* [,...]
>   WHERE CURRENT OF *cursor_name*;

In this expression, *table_name* is the external name of an HP ALLBASE/SQL base table. The term *value* is an HP ALLBASE/4GL item containing the new value to be inserted in the column to be updated. It can be any of the following:

- A literal enclosed in single quotes (').

- A file record field reference.

- A work area field reference.

- A screen field name.

- A variable or calculated item.

- A numeric or alphanumeric constant.

The term *cursor_name* is the HP ALLBASE/4GL name of the SQL logic block containing the SELECT command that defined the current cursor.

To use this form of the UPDATE command, you must execute a SELECT command in an SQL logic block earlier in the same process to define and open an SQL cursor, and then execute a FILE *NEXT command in an HP ALLBASE/4GL logic block to position the cursor.

Example 4 - Updating HP ALLBASE/SQL Data via a Cursor

This example shows one way of updating records in an HP ALLBASE/SQL base table. The HP ALLBASE/4GL function and SQL logic blocks shown here operate under the process *emp_update*. This process firstly displays the screen *mod_employee*, then updates the base table *empgrp.employee* according to the data entered by the user.

Process *emp_updt*

```
1 MODE *WRITE employee
2 SCREEN mod_employee
3 SQL upd_employee; ENTER 6
4 SQL commit
5 ENTER 2
6 MESSAGE upd_error
7 SQL roll_back
8 EXIT
```

Function *mod_emp*

```
1 SQL sel_employee
2 FILE *NEXT employee; ENTER 5
3 SHOW *REFRESH
4 EXIT
5 IF *IOSTATUS <> N-eof THEN ENTER 10
6 MESSAGE no_rec
7 FILE *BUFFER employee
8 CLEAR *MAP S-emp_name S-spouse
```

FINAL TRIM SIZE : 7.0 in x 8.5 in

```
 9 EXIT
10 VISIT db_error
11 EXIT
```

SQL Logic Block *sel_employee*

```
SELECT :employee FROM empgrp.employee
  WHERE emp_no = :S-emp_no.mod_employee
  FOR UPDATE OF emp_name, address, spouse;
```

SQL Logic Block *upd_employee*

```
UPDATE empgrp.employee SET
  emp_name = :F-emp_name.employee,
  address  = :F-address.employee,
  spouse   = :F-spouse.employee
WHERE CURRENT OF sel_employee;
```

The function *mod_emp*, which is called as an after function on an employee number field on the screen *mod_employee*, calls the SQL logic block *sel_employee* and then executes a FILE *NEXT command. These two commands declare and open a cursor on the table *empgrp.employee* and retrieve the first record in the active set for the cursor. If a record is retrieved successfully, the SHOW *REFRESH command displays the record on the screen, and the function exits. If the FILE command does not retrieve a record, control passes to step 5 which tests the value in *IOSTATUS. If the SELECT command does not retrieve a record, the FILE command encounters an end of file condition without retrieving a record. In this case the *IOSTATUS value is 60110, (the value of the numeric constant N-eof), and a suitable message is displayed. The function then clears the file record buffer and the screen to allow the user to enter the details for a new record.

If the FILE command fails for any reason other than reaching the end of file, control passes to step 10. This step executes a further function *db_error*. (This function is not shown in this example.) Typically, such a function would use the SQLEXPLAIN command in a SQL logic block to determine the cause of the error, and then take the appropriate actions to rectify the situation.

The SQL logic block *sel_employee* contains a single SELECT command that opens a cursor on the table *empgrp.employee*, where *employee* is the HP ALLBASE/4GL name of an HP ALLBASE/SQL base table, and *empgrp.employee* is the external name of an existing base table. This SELECT command finds a record with the value in the field *emp_no* that matches the current value in the *emp_no* field on the screen *mod_employee*. Note that the SELECT command must include the FOR UPDATE OF clause to specify the columns of the table that are to be updated.

When the user exits from the screen by pressing the `Commit Data` function key, control returns to the process *emp_updt*. The process immediately calls the SQL logic block *upd_employee* to update the current record in the base table. If the update is successful, control passes to the SQL logic block *commit*. This logic block must contain an SQL COMMIT WORK command. If an error occurs during the update, control passes to step 6 which displays an error message. The SQL logic block *roll_back* then reverses the incomplete SQL transaction. This SQL logic block would contain an SQL ROLLBACK WORK command.

The SQL logic block *upd_employee* contains an SQL UPDATE command. This command must specify the new values for the columns in the base table row to be updated. It must also contain the WHERE CURRENT OF clause to specify the row to be updated. Note that the cursor name for this form of the UPDATE command is the name of the SQL logic block that contains the SELECT command that declares and opens the cursor. In this example the cursor name is *sel_employee*.

Updating Without a Cursor

You can update a base table without using a cursor by using the UPDATE command with an SQL WHERE *search_condition* clause in an SQL logic block. The UPDATE command has the format:

UPDATE *table_name* SET *column_name* = *value* [,...]
WHERE *search_condition*;

In this expression, *table_name* is the external name of the table. You can use a fully qualified name of the form *owner.table_name* in this command.

Example 5 - Updating HP ALLBASE/SQL Data Without a Cursor

This example demonstrates the procedure to update an HP ALLBASE/SQL table without using a cursor.

SQL Block *update*

```
UPDATE debtors
SET extend_credit = 'NO'
WHERE days_overdue >= :V-days_late;
COMMIT WORK;
```

This SQL logic block contains a single SQL UPDATE command that sets the *extend_credit* field to "NO" for each record where the value in the field *days_overdue* is equal to or greater than the value in the HP ALLBASE/4GL variable *days_late*.

Deleting Records

To delete a record from an HP ALLBASE/SQL base table you can use the FILE *DELETE command in an HP ALLBASE/4GL logic block, or the SQL DELETE or DELETE WHERE CURRENT command in an SQL logic block.

FILE *DELETE Command

To delete a row from a base table, the FILE *DELETE command has the format:

FILE *DELETE *file_name*

where *file_name* is the HP ALLBASE/4GL dictionary name of a base table. To use this command, you must declare and open a cursor with a SELECT command in an SQL logic block earlier in the same HP ALLBASE/4GL process, and position the cursor with a FILE *NEXT command.

The FILE *DELETE command performs the same function as the following SQL command:

DELETE FROM *table_name* WHERE CURRENT OF *cursor_name*;

The FILE *DELETE command deletes the table row identified by the current cursor position. After the row has been deleted, the cursor is positioned between the preceding and following rows of the active set. To reposition the cursor for further deletions or updates, use the FILE *NEXT command.

Example 6 - Using the FILE *DELETE Command

This example shows the use of the FILE *DELETE command to delete records from a base table.

Function *del_items*

```
 1 SQL find_items
 2 FILE *NEXT parts; ENTER 8
 3 SHOW *REFRESH
 4 MESSAGE confirm_del
 5 IF V-confirm <> "Y" MESSAGE no_delete; ENTER 2
 6 FILE *DELETE parts; ENTER 12
 7 ENTER 2
 8 IF *IOSTATUS <> N-eof ENTER 12
 9 MESSAGE all_done
10 SQL commit
11 EXIT
12 MESSAGE file_error
13 EXIT
```

SQL Logic Block *find_items*

```
SELECT :parts FROM Purchdb.parts
  WHERE part_no BETWEEN :V-first_no AND :V-last_no;
```

The first command in the function *del_items* calls the SQL logic block *find_items*. This SQL logic block opens a cursor on the base table *Purchdb.parts*. The FILE *NEXT command then positions the cursor on the first record to be deleted, and the SHOW *REFRESH command displays the record contents on the screens.

FINAL TRIM SIZE : 7.0 in x 8.5 in

If the user enters Y in response to the message, the FILE *DELETE command in step 6 deletes the record, and control returns to step 2 to retrieve the next record.

After the last selected record has been deleted, the FILE *NEXT command encounters the end of file, and returns an error condition. Control then passes to step 8 which tests the error return in the communication area field *IOSTATUS. If the value in *IOSTATUS is the same as the value of the numeric constant N-eof (60110, end of file), control passes to step 9. This step displays a message to confirm the deletions, and the following SQL command commits the changes.

Any other file error condition passes control to step 12 to display an error message, and exit from the function without committing the changes.

The SQL logic block *find_items* contains a SELECT command. This command selects all records with a value in the *part_no* field that is between the values contained in the HP ALLBASE/4GL variables *V-first_no* and *V-last_no*. In this SQL logic block, *parts* is the HP ALLBASE/4GL name of a base table, and *Purchdb.parts* is the external name for the table.

Deleting via a Cursor

You can use the SQL DELETE WHERE CURRENT command in an SQL logic block to delete a row from a base table via a cursor. The command has the format:

  DELETE FROM *table_name* WHERE CURRENT OF *cursor_name*;

In this command, *table_name* is the external name of the HP ALLBASE/SQL table. You can use a fully qualified name of the form *owner.table_name*. The term *cursor_name* is the name of the SQL logic block containing the SQL SELECT command that defined the cursor. You must execute the SELECT command to open the cursor, and use a FILE *NEXT command in an HP ALLBASE/4GL logic block to position the cursor before you can delete the row.

Example 7 - Deleting HP ALLBASE/SQL Records via a Cursor

This example shows the use of the SQL DELETE command in an SQL logic block.

9-18 HP ALLBASE/SQL Interface

Process *del_order_items*

```
1 MODE *WRITE ord_item
     .
     .
  SQL select_items
  FILE *NEXT ord_item
     .
     .
  SQL delete_item
     .
     .
     .
```

SQL Logic Block *select_items*

```
 SELECT :ord_item FROM Purchdb.ord_item
   WHERE order_no = :F-order_no.ord_item;
```

SQL Logic Block *delete_item*

```
 DELETE FROM Purchdb.ord_item
   WHERE CURRENT OF select_items;
```

This HP ALLBASE/4GL process calls two SQL logic blocks. The SQL logic block *select_items* opens a cursor on the base table *ord_item*. The FILE *NEXT command in the HP ALLBASE/4GL logic block positions the cursor on the first record in the active set for the cursor.

The SQL logic block *delete_item* deletes the current record. Note that you must specify the cursor name as the name of the SQL logic block containing the SELECT command that defines the cursor. This SQL DELETE command is equivalent to the HP ALLBASE/4GL command:

```
   FILE *DELETE ord_item
```

After the delete, the cursor is positioned between the preceding row and the following row in the table. You can use a further FILE *NEXT command to

position the cursor on the next row of the active set. You must also use an explicit COMMIT WORK command in an SQL logic block to terminate the SQL transaction.

Deleting Without a Cursor

You can use the SQL DELETE command in an SQL logic block to delete a row, or a number of rows, from an HP ALLBASE/SQL base table. The command has the format:

    DELETE FROM *table_name* [WHERE *search_condition*]

In this expression, *table_name* is the external name of the table. You can use a fully qualified table name of the form *owner.table_name* in this command.

Example 8 - Deleting Without a Cursor

This example shows the use of an SQL DELETE command in an SQL logic block.

> **SQL Logic Block** *del_old_orders*

> **DELETE** from purchdb.orders
>   WHERE order_number < :V-orders_old;

This SQL logic block deletes all records from the base table *purchdb.orders* where the value of the field *order_number* is less than the current value in the HP ALLBASE/4GL variable *orders_old*.

The SQLEXPLAIN Command

HP ALLBASE/4GL requires a slightly modified form of the SQLEXPLAIN command in SQL logic blocks. The SQLEXPLAIN command allows you to retrieve the text of HP ALLBASE/SQL error messages into the communication area field *ERROR. The format for the SQLEXPLAIN command is simply SQLEXPLAIN;

You cannot use a host variable reference with the SQLEXPLAIN command in an SQL logic block.

When HP ALLBASE/SQL detects an error at run-time, HP ALLBASE/4GL executes one SQLEXPLAIN command automatically. This command retrieves the text of the error message into *ERROR, and the error number, plus 100000, into the communication area field *IOSTATUS.

In some cases, a single SQL statement can cause multiple SQL error messages. You can use further SQLEXPLAIN commands to retrieve the text and number of subsequent error messages. After the last error message has been retrieved, a further SQLEXPLAIN command returns an empty string to *ERROR and sets *IOSTATUS to zero.

SQL Logic Block Generation

When HP ALLBASE/4GL generates an SQL logic block, it performs a syntax check on the command. A SELECT command in an SQL logic block is converted to a DECLARE CURSOR for SELECT command. At generate time, HP ALLBASE/4GL also replaces references to select lists or table record layouts with the dictionary definitions for these items.

The generated form of the SQL logic block is stored in the HP ALLBASE/4GL S-files. Depending on the SQL commands, the SQL logic block may also be stored in the HP ALLBASE/SQL database environment as a chapter in a database module. The name of the module for an application or version is *owner_group.name* where *owner_group* is the SQL owner group for the application as specified on the administrator application definition screen, and *name* is the name of the application or version.

When you generate all SQL logic blocks in an application, or generate an entire application, HP ALLBASE/4GL writes the definition of the module to a database module file. The name of the application is used as the name of this file. HP ALLBASE/4GL creates this file in the group and account specified by the variable HP4DBMPATH.

If the HP4DBMPATH variable does not exist, HP ALLBASE/4GL does not create the database module file.

By default, HP ALLBASE/4GL creates the database module file with a file limit of 4095 records. You can override this default size by setting the MPE/iX variable HP4DBMSIZE to the required file limit.

Note	When you generate all SQL logic blocks, HP ALLBASE/4GL only performs a COMMIT WORK when all logic blocks have been generated. It is possible for the SQL log file to become filled during the generate. This may happen if you have a small log file and a large number of SQL logic blocks, or archive logging is turned on for your database environment and the log file is nearly full.
	If the log file space is not sufficient, you will receive an SQL error number 14046 *"Log Full"*.
	If you receive this error, the database log files must be enlarged before you can generate all SQL logic blocks.

If HP ALLBASE/4GL detects any errors during SQL logic block generation, it displays an error message screen. This screen contains a row and column reference for the error. These values refer to the cursor row and column position, within the SQL logic block field, at or near the source of the error.

HP ALLBASE/SQL Transaction Management

HP ALLBASE/SQL requires an explicit COMMIT WORK command to terminate any transaction and make the transaction permanent in the database. If you don't commit a transaction, HP ALLBASE/SQL automatically rolls back the transaction.

If a transaction is not already open, HP ALLBASE/SQL automatically issues a BEGIN WORK RR command when you execute most HP ALLBASE/SQL commands. HP ALLBASE/SQL does not issue COMMIT WORK commands automatically. You must always commit a transaction explicitly.

You can use the SQL commands BEGIN WORK, COMMIT WORK, and ROLLBACK WORK in SQL logic blocks to control SQL transactions.

FINAL TRIM SIZE : 7.0 in x 8.5 in

TRANSACT Command

The HP ALLBASE/4GL TRANSACT command and the SQL BEGIN WORK, COMMIT WORK, and ROLLBACK WORK commands are interchangeable. The following table summarizes the equivalence:

BEGIN WORK [RR]	<==>	TRANSACT *BEGIN
BEGIN WORK CS	<==>	No TRANSACT equivalent
COMMIT WORK	<==>	TRANSACT *END
ROLLBACK WORK	<==>	TRANSACT *UNDO

The equivalent commands execute the same code. However, the TRANSACT command can only be used in HP ALLBASE/4GL logic blocks while the SQL commands can only be used in SQL logic blocks.

Table Locking

All access to HP ALLBASE/SQL tables is subject to the normal HP ALLBASE/SQL locking systems. Table locking is controlled by the table locking mode, the LOCK TABLE command, and the options used with the BEGIN WORK command. The following paragraphs provide a brief overview of these items. For further information on table locking, refer to the *HP ALLBASE/SQL Reference Manual*.

The Table Locking Mode

When you define an HP ALLBASE/SQL base table you specify the default locking mode to be used for all accesses to the table. This can be one of three values:

- **Public.** Allows multiple read access and multiple write access.

- **Publicread.** Allows multiple read access, but requires exclusive access for writing.

- **Private.** Requires exclusive access for reading or writing.

A *Public* table allows the highest amount of concurrency.

The LOCK TABLE Command

You can use the SQL LOCK TABLE command within an SQL logic block to override the default table locking mode. This command allows you to obtain a lock on an entire table before you begin manipulating the table.

Typically, you would use a LOCK TABLE when you know that a transaction is going to access a large portion of the table. By locking the table before you start the transaction, you reduce the possibility of encountering a deadlock part way through the transaction. Locking the table also removes the overhead involved in obtaining individual locks for the actions performed within the trans

FINAL TRIM SIZE : 7.0 in x 8.5 in

10

HP TurboIMAGE/iX Interface

Introduction

This section describes the facilities for accessing and using
HP TurboIMAGE/iX databases from HP ALLBASE/4GL applications.

The interface consists of:

- Dictionary data set definitions.

- Dictionary field specification (data item) definitions.

- HP ALLBASE/4GL logic blocks.

- A utility to upload database definitions.

- Database access parameters and definitions (within the administrator
  application).

You can add, update, and delete HP TurboIMAGE/iX records (data entries)
from within HP ALLBASE/4GL. However, you cannot physically create an
HP TurboIMAGE/iX database or data set from within HP ALLBASE/4GL.

This manual assumes that you are familiar with the HP TurboIMAGE/iX
database management system. For detailed information about
HP TurboIMAGE/iX, refer to the *HP TurboIMAGE/iX Database Management
System Reference Manual*.

Databases

HP ALLBASE/4GL allows you to define the components of an HP TurboIMAGE/iX database in HP ALLBASE/4GL and maintain an HP TurboIMAGE/iX database using HP ALLBASE/4GL applications.

You can use a maximum of 255 databases in a single HP ALLBASE/4GL application.

Each database must be defined and linked to the application by the HP ALLBASE/4GL administrator in the administrator application. The name and location of the database in the file system is part of the database definition.

Unless the HP ALLBASE/4GL administrator specifies a fully qualified database name (that is, a name in the form *FILE.GROUP.ACCOUNT*), HP ALLBASE/4GL appends the current value of the HP4TIPATH system variable to the external name of the database to identify the group and account of the database. If HP4TIPATH is not defined, HP ALLBASE/4GL searches for the database in the user's current group and account.

Commonly, application developers will create HP ALLBASE/4GL applications to access existing HP TurboIMAGE/iX databases. HP ALLBASE/4GL provides several utilities so that you may load data set, data item, and database definitions into an HP ALLBASE/4GL application. This means that your data set definitions, record layouts, and field specifications do not need to be redefined in HP ALLBASE/4GL. The upload utilities are described below.

Uploading Definitions

The HP4TUPLD utility converts information about data items, data entries and data sets contained in an HP TurboIMAGE/iX schema to HP ALLBASE/4GL field specifications, record layouts and data set (file) definitions. The converted definitions can be printed to the screen or redirected to a file.

Once the HP TurboIMAGE/iX definitions have been converted, the HP4ATOS utility can be used to write the information into a set of HP ALLBASE/4GL S-files. Note that you must define the application in HP ALLBASE/4GL

before the definitions can be uploaded into the S-files. For more information
about HP4ATOS, refer to chapter 13.

Running HP4TUPLD

The syntax for running HP4TUPLD from the MPE/iX prompt is:

HP4TUPLD "-a*application* -o*object_type* [*SCHEMA_FILE*] > [*ATOSFILE*]"

SCHEMA_FILE is the name of the schema file for the HP TurboIMAGE/iX
database that you wish to upload.

ATOSFILE is the name of the ASCII file produced by the upload utility. If
you do not specify a file, HP4TUPLD sends its output to $STDLIST. If you
redirect output to a file, you must create the file before running HP4TUPLD,
with an adequate record size and an adequate number of records specified.

Command Line Parameters

Parameter	Description
-a *application_name*	The name of the HP ALLBASE/4GL application that will access the HP TurboIMAGE/iX database. All definitions must be associated with an application. The application need not exist when you run HP4TUPLD.
-o *object_type*	This parameter is optional. You may specify either **x** or **d** as the object type. If you don't specify an object type, **-od** is assumed.
	If you wish to extract developer information from the schema file (data set and data item definitions), specify **d**.
	If you wish to extract administrator information from the schema file (the database definition and the database access parameter), specify **x**. No password is extracted from the schema file; the user class password is set to blank. Note that you may find it faster to log in to HP ALLBASE/4GL as *administ* and define this information within HP ALLBASE/4GL.

Naming Conventions

Naming rules in HP TurboIMAGE/iX differ somewhat to HP ALLBASE/4GL naming rules. The following characters may be used in HP TurboIMAGE/iX names, but may not be used in HP ALLBASE/4GL names:

> +, -, *, /, ?, ', #, %, &, and @.

HP4TUPLD transforms any of the above characters to underscores (_) in its output. For example, the HP TurboIMAGE/iX name *ADDRESS&ZIP-CODE* would be converted to *address_zip_code* by HP4TUPLD.

HP TurboIMAGE/iX data set names may be up to 16 characters long. Within HP ALLBASE/4GL, the international data set name may be no longer than 8 characters. HP4TUPLD truncates any data set names that are longer than 8 characters.

In some circumstances, truncation or conversion of characters by HP4TUPLD causes two HP TurboIMAGE/iX names to be converted to the same name. If this occurs, HP4TUPLD generates a unique name for the second name, and displays a message when this occurs.

You may edit the output file produced by HP4TUPLD to change any database, data set, or data item names.

An Example

An example of running HP4TUPLD is:

```
HP4TUPLD "-asupply -od STKCNTRL > STOCK"
```

This command uploads the developer definitions from an HP TurboIMAGE/iX schema file called *STKCNTRL* into an ASCII text file *STOCK*.

The Upload Procedure

To use HP4TUPLD and HP4STOA to upload definitions into HP ALLBASE/4GL, follow the procedure below.

Before you start, you must create the file to which HP4TUPLD output will be redirected. You must create the file before running HP4TUPLD, with an adequate record size and an adequate number of records specified.

FINAL TRIM SIZE : 7.0 in x 8.5 in

To upload definitions:

1. Run HP4TUPLD with the `-ox` option to upload the administrator system items from the HP TurboIMAGE/iX schema, or with the `-od` option to upload the developer definitions from the HP TurboIMAGE/iX schema. Direct the output to a file.

2. Make sure that the application you wish to load the definitions into is defined in the HP ALLBASE/4GL administrator application.

3. Run HP4ATOS to load the HP TurboIMAGE/iX details into your application by entering the following at the MPE/iX prompt:

   HP4ATOS "-u*username:password* -a*application_name:dev_sec_code* <ATOSFILE"

   If you are loading administrator system items, the application name is *administ*.

   If you are loading dictionary definitions, use the name of the application that will access HP TurboIMAGE/iX.

   You may also wish to specify that if the names of any objects being loaded match existing objects in the S-files, the existing objects are not overwritten. To do so, use the **-e** option when you run HP4ATOS. If you don't use the **-e** option, any existing objects are overwritten if an object in the ATOSFILE file has the same name. For details about this option, refer to chapter 13.

When you next run HP ALLBASE/4GL to develop the application, you will find that the field specifications, default record layouts, and data set definitions are defined for the application.

If any messages, warnings, or errors occur during the upload, an error listing is written to the HP4LIST file designator.

Upload Defaults

HP4TUPLD produces a field specification for each HP TurboIMAGE/iX data item that is uploaded. This section describes how the data item attributes are translated by HP4UPLD to HP ALLBASE/4GL field specification attributes.

The repeat count of the field specification is set to the sub-item count of the data item. The edit code, storage type, and display length are determined according to the HP TurboIMAGE/iX type designator.

Note that some legal HP TurboIMAGE/iX type designator and sub-item length combinations are not accepted by HP4TUPLD, as they have no corresponding HP ALLBASE/4GL edit code and storage type. If HP4TUPLD comes across such data items in the schema file, an error message is issued, and the data item is ignored. You may manually create a field specification within HP ALLBASE/4GL that best suits your needs, within the HP ALLBASE/4GL restrictions.

The table below describes the values that are set for each type designator.

HP TurboIMAGE	HP ALLBASE/4GL		
Type Designator	Edit Code	Storage Type	Display Length
I	S	I	6
I2	S	L	11
I4	S	P	20
In	ERROR if n $\neq$ 1, 2, or 4		
J	S	I	5
J2	S	L	10
J4	S	P	19
Jn	ERROR if n $\neq$ 1, 2, or 4		
K	N	I	5
K2	N	L	10
K4	S	P	19
Kn	ERROR if n $\neq$ 1, 2, or 4		
R2	S	F	10
R4	S	G	20
Rn	ERROR if n $\neq$ 1, 2, or 4		
Pn	S	P	$n + 1$
Un	U	C	n
Xn	X	C	n
Zn	S	C	$n + 2$

FINAL TRIM SIZE : 7.0 in x 8.5 in

The Interface to HP TurboIMAGE/iX

Opening and Closing Databases

HP TurboIMAGE/iX databases are automatically opened when a data
set within the database is accessed. Databases are opened with the
HP TurboIMAGE/iX database password that is specified by the HP
ALLBASE/4GL administrator within the administrator application. This
password identifies the read and write access available to all end users and
developers of the application.

HP TurboIMAGE/iX databases are only closed at the end of an HP
ALLBASE/4GL session or when an explicit DM IMAGE *CLOSE logic
command is executed.

Database Access Modes

HP TurboIMAGE/iX databases must be opened in an access mode. Within
HP ALLBASE/4GL, this mode is set by the DM IMAGE *MODE command.

The default database access mode is *MODE *READLOCK. This corresponds
to the HP TurboIMAGE/iX access mode number 5. If no database access
mode has been set for a database, it is opened in the default mode.

The database is accessed in the same access mode until the database is closed.
Any DM IMAGE *MODE commands executed for the database while it is
open will *not* reset the access mode.

In addition to these access restrictions, HP ALLBASE/4GL must use a user
class password to access HP TurboIMAGE/iX. This password, which is
specified by the HP ALLBASE/4GL administrator, defines the overall access
permissions of all developers and end users of the application for the particular
database. The DM IMAGE *MODE and the MODE command cannot provide
more capability than the access permissions allowed for the user class password.
For further details about this password, refer to the *HP ALLBASE/4GL
Developer Administration Manual*.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Opening and Closing Data Sets

Data sets are opened in the database access mode specified by the DM IMAGE *MODE logic command. The type of access to the data set is also controlled by the MODE command. Refer to the FILE (HP TurboIMAGE/iX) command in chapter 12 for more information about access modes.

All open data sets are automatically closed at the end of each process. In addition, developers may specify that the entire database is closed, in which case all data sets within the database are closed automatically.

HP ALLBASE/4GL FILE Logic Command

Each file operation calls one or more HP TurboIMAGE/iX library procedures. The table overleaf outlines the HP ALLBASE/4GL FILE logic command operations and the HP TurboIMAGE/iX library procedures called. For library procedures that specify a mode of operation, the mode is also listed in table 10-1.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Table 10-1. Master Data Set File Operations

File Operation	Library Procedure	Procedure Mode
*CLOSE	DBCLOSE	2
*DELETE	DBDELETE	1
*FIND	DBGET	7
*FIRST	DBCLOSE then DBGET	3 2
*FIRST *KEY	DBGET	7
*INSERT	DBPUT	1
*MODIFY	DBUPDATE or DBPUT the DBDELETE	1 1 1
*LAST	DBCLOSE the DBGET	3 3
*LAST *KEY	DBGET	7
*NEXT	DBGET	2, or 1 if migrating secondary not processed
*PREVIOUS	DBGET	3, or 1 if migrating secondary not processed
*READ	DBGET	7, using key indicated by *INDEXNO or *INDEX = file operation argument
*READ *KEY	DBGET	7
*READ *REC	DBGET	4
*WRITE	Refer to *INSERT or *MODIFY above.	

FINAL TRIM SIZE : 7.0 in x 8.5 in

Table 10-2. Detail Data Set File Operations

File Operation	Library Procedure	Procedure Mode
*CLOSE	DBCLOSE	2
*DELETE	DBDELETE	1
*FIND	DBFIND	1
*FIRST	DBCLOSE the DBGET	3 2
*FIRST *KEY	DBFIND the DBGET	1 5
*INSERT	DBPUT	1
*LAST	DBCLOSE the DBGET	3 3
*LAST *KEY	DBFIND the DBGET	1 6
*MODIFY	DBUPDATE or DBPUT the DBDELETE	1 1 1
*NEXT	DBGET	2, or 5 for chained read
*PREVIOUS	DBGET	3, or 6 for chained read
*READ	DBFIND then DBGET	1, using key specified by *INDEXNO or *INDEX = file operation argument. or 5 for chained read
*READ *KEY	DBFIND DBGET	1 5
*READ *REC	DBGET	4
*WRITE	Refer to *INSERT or *MODIFY above.	

When an HP TurboIMAGE/iX library procedure is executed,
HP TurboIMAGE/iX returns a status array. The first halfword (16 bits) of the
status array contains the error number returned by the library procedure. The
rest of the status array contains miscellaneous information about the status
of the library procedure. The HP ALLBASE/4GL communication area field
*IOSTATDM contains the error number returned by the most recent library

procedure executed, and the *IMSTAT communication area field contains the complete status array.

For details about the meaning of an error number or a status array, refer to details about the particular library procedure in the *HP TurboIMAGE/iX Database Management System Reference Manual*.

Transaction Management

Transaction management provided by HP ALLBASE/4GL consists of the TRANSACT *BEGIN, TRANSACT *END, and TRANSACT *MEMO logic commands. HP ALLBASE/4GL does not roll back or roll forward any incomplete transactions that occur. Refer to chapter 12 for details.

HP ALLBASE/4GL transaction management has no effect for HP TurboIMAGE/iX file transactions unless HP TurboIMAGE/iX logging is enabled. Refer to the *HP TurboIMAGE/iX Database Management System Reference Manual* for more information about HP TurboIMAGE/iX transaction management.

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-2

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-4

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-8

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-12

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-16

FINAL TRIM SIZE : 7.0 in x 8.5 in

Index-18

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in

FINAL TRIM SIZE : 7.0 in x 8.5 in