



Purple Software

PowerBase

(Personal Edition)

Navigate this online document as follows:

To see bookmarks,
type:

Command-7 (Mac OS)
Ctrl-7 (Windows)

To navigate, click on:

any blue hypertext link
any [Table of Contents](#) entry
arrows in the toolbar

This page is intentionally blank



Purple Software

The PowerBase User Guide

© Copyright 1998 Purple Software Limited

All rights reserved. This documentation and the software referred to are copyrighted works of Purple Software Limited, London, England. Reproduction in whole or part by any means whatsoever without the express written permission of Purple Software Limited is prohibited. Reverse engineering is also prohibited.

The information in this document is subject to change without notice.

The copyright owners accept no liability for loss or consequential loss from misuse or any defect or failure of the programs howsoever arising.

Psion Series 5, and EPOC32 are trademarks of Psion plc. Purple Software Limited acknowledges other references may be registered trademarks belonging to third parties.

Under the terms of the license agreement, the purchaser is allowed to copy applications to combine applications onto one disk, provided the software is installed on only one computer at any one time.

Contact information:

Purple Software is one of Europe's most experienced software developers for handheld computers. With more than a decade's experience of programming, the Purple Software name has become synonymous with advanced solutions for Psion platforms and the Windows® CE operating system.

Our products are available world-wide and most are published in languages other than English. For more information, contact us by:

Phone: +44 (0) 20 7387 7777

Fax: +44 (0) 20 7387 1188

e-Mail: info@purplesoft.com

Web: <http://www.purplesoft.com>

Table of Contents

1 Introduction	1
Welcome to PowerBase	1
Database management with PowerBase	1
About this document	3
What This Guide Contains.....	3
Help.....	4
2 Starting PowerBase	5
Key concepts	5
Tables, Records and Fields.....	5
Expressions.....	6
Groups and Filters.....	6
Views	7
The PowerBase Desktop	7
Floating windows.....	8
The Schema view	9
The Current Window.....	10
Properties and Preferences.....	10
Your first database	11
Importing your Address book	11
Templates	13
Creating a new database from a Template.....	13
The supplied templates.....	14
Closing PowerBase	16
3 Using PowerBase.....	17
Creating and evaluating expressions.....	17
Databases, Tables and Fields.....	20
Opening and creating databases	20
Adding tables to a database	20
Deleting and duplicating tables.....	21
Adding, amending and deleting fields.....	21
Importing and exporting data.....	22
Compressing the database	23
Views and Groups.....	24
Working with groups.....	24

Table of Contents

Working with views	25
Using views	28
Sorting.....	29
The Desktop toolbar.....	30
Using the Find editor.....	30
How finding works.....	30
Find properties.....	31
Using the record navigation buttons.....	32
Filtering.....	32
Dialling	33
Printing.....	33
Creating, updating and deleting records.....	34
The "In place" editor	36
Specialised editors	37
Related fields.....	37
Unique value lists.....	37
The location of updated records'.....	37
Deleting records.....	38

4 Technical Reference39

General information.....	39
Field types	40
Field attributes.....	41
Field display formats.....	42
Text fields.....	42
Boolean fields.....	42
Integer fields.....	42
Floating point number fields.....	43
Currency.....	43
Date.....	43
Memo and Embedded fields	44
Relational and Calculated fields.....	44
Expressions.....	44
Arithmetic operators.....	45
Relational and logical operators.....	45
Mathematical functions	46
Trigonometric functions.....	46
String functions.....	46
Date and time functions.....	47

1

Introduction

Welcome to PowerBase

PowerBase is a powerful relational database management application for your handheld computer. PowerBase is compatible with devices that use the EPOC32 operating system, including the Psion Series 5 and the GeoFox One computers.

Rather than referring to any specific 'EPOC32 device', this user guide will simply refer to 'your handheld computer'. Readers should note that, unless otherwise stated, the same functionality is supported by all EPOC32 handheld computers available at the time of writing.

Database management with PowerBase

A database management system allows you to capture, store and retrieve information about similar items.

In the simplest case, a database may be a single list of items of data. This type of database is often referred to as a "flat file" or "card file" database. Your address book is a good example of a "flat file" database.

Although suitable for many purposes, "flat-file" databases present a number of problems. Typically, these begin to emerge when the "real world" things about which you wish to record information cannot be described in such a two dimensional way.

For example, consider the case where you have several contacts within the same organisation. Each contact has a direct dial telephone number but within an organisation they all share the same fax machine.

Using a "flat-file" approach one might record the name and number for each contact and then replicate the shared fax number across all members of the same organisation. An alternative approach might be to make a single entry

Introduction

for the organisation and then a short note referring to this against each contact. Both of these have significant disadvantages.

Using the first approach there is significant duplication of data that requires additional time to input and consumes space in the database. These problems rise rapidly as the amount of shared data increases - consider the case where the postal address of the organisation is required.

There is also the need to maintain the data over time - suppose the organisation changes its fax number, all the contact records will require amendment.

While the second approach may be sufficient if you simply want to lookup the fax number, this is tedious and not suitable for achieving effects such as standard letters addressed to each contact at their organisations postal address.

To resolve these issues, the ability to link entries is required. One might imagine two lists, one containing the details of an organisation, the other details for individual contacts. An entry in the contacts' list can be linked to an entry in the list of organisations such that when the list of contacts is being viewed, details of the linked organisation are shown against each contact.

The database management systems providing the ability to link items of information in this way are often referred to as "relational" database management systems (RDBMS).

The power and flexibility offered by a RDBMS unsurprisingly comes at a cost. Typically, this may involve:

- The need to give greater consideration to the way in which the database is designed. How should data be organised into multiple lists? How should these lists be related? Different designs may be optimal for different purposes - in the example above the value of the relational approach derives from their being multiple contacts within an organisation.
- Complex relational designs may require sophisticated form design tools and programming languages to present a suitable user interface for editing or viewing the data.
- Significant processing power and/or memory to manage complex data relationships.

The PowerBase family of products from Purple Software is designed to provide a set of RDBMS tools covering a wide range of database and application design needs.

PowerBase (Personal Edition) is the first product in the PowerBase range. It is designed to provide the individual user with a single set of tools enabling them quickly and easily to realise the benefits of a RDBMS for storing and retrieving their personal data without the need for complex programming.

None the less, PowerBase "Personal" offers the power and depth to handle considerable complexity and is supplied with a number of database "Templates" that demonstrate what can be achieved.

Even, for those who do not need or wish to use its relational capabilities, PowerBase "Personal" provides, as standard, a range of high performance tools for managing traditional "flat-file" databases.

About this document

The PowerBase User Guide explains how to install and use the PowerBase application. In so doing, the Guide assumes that the reader is familiar with the standard features of their handheld computer, such as the use of menu bars, scroll bars, buttons, dialog boxes and the help system.

Although the Guide provides examples and suggestions as to appropriate designs for the structure of relational database, this Guide does not seek to be a comprehensive tutorial in this area.

The Guide is also supplied in electronic form as a Portable Document Format (.pdf) file. Software to display .pdf files is freely available from Adobe Systems Incorporated and can be acquired from their World Wide Web site at <http://www.adobe.com/acrobat/>. This software may also be found on your PsiWin CD-ROM in the \Extras\Acrobat\ directory.

What This Guide Contains

Following this introduction:

- [Chapter 2](#) sets out the key concepts that should be borne in mind while using PowerBase, describes the key features of the PowerBase User Interface and presents a number of example databases.
- [Chapter 3](#) describes the main features of the PowerBase application and how they are used.
- [Chapter 4](#) provides a Technical Reference to key PowerBase components.

Introduction

Help

The PowerBase application also provides extensive on-line help, which can be accessed at any time by selecting the **Help** menu command.

2

Starting PowerBase

This chapter sets out the key concepts that should be borne in mind when using PowerBase, describes the key features of the PowerBase User Interface and presents a number of example databases. It is strongly recommended that you read this section to obtain maximum benefit from PowerBase.

Key concepts

PowerBase works with a single **Database** at a time. A database defines how items of information are stored together with the actual data recorded for these items.

Tables, Records and Fields

Items of information within a database are organised as **Tables**. A database can contain multiple tables. A Table can contain multiple entries (or **Records**) and typically, each record will describe some unique "real world" entity such as a person or a product.

A table also defines a number of **Fields**. Each field represents some aspect or attribute of the entity about which information is to be recorded such as the name or date of birth of a person. The definition of a field includes the name of the field, the type of data to be recorded and how it is to be displayed.

PowerBase supports a wide range of field types, including text strings, integers, Boolean numbers, floating point numbers, currency fields, memos (up to 64k of text) and embedded objects associated with other applications on your handheld computer.

Full details of the field types supported their attributes and their format options are provided in the [Technical Reference Guide](#).

Starting PowerBase

There are two further field types worthy of note. These are **Calculated** and **Relational** fields. Unlike other types, the information displayed is not actually stored in the record for these fields but rather, it is derived from other fields.

For a given record, the result of a calculated field derives entirely from the value of other fields in the same record. For this reason no value needs to be (or can be) entered for calculated fields when creating or amending the record.

For relational fields, the result derives from the value of fields in some other record to which a link has been established. The linked record can be in any table in the database - the specific table is defined as part of the definition of the relational field.

When creating or amending a record containing a relational field, the information entered for the relational field is used solely to identify the record to which the link is to be established.

Once the link has been established, the data in the linked record can be changed at will and the link will be preserved.

Expressions

The information displayed for calculated and relational fields is defined as an **Expression**. PowerBase incorporates a powerful expression evaluator that supports over 60 functions for the manipulation of numbers, text, dates and logical values. For example the expression:

[Title]+" "+[Initials]+" "+[Name] might be used to produce the formatted string "Mrs JA Smith" for display.

Alternatively, the expression "[salary]<20000.00" would be evaluated as "True" for all records in which the salary field contains a value of less than 20,000. Full details of the expression evaluator are provided in the [Technical Reference Guide](#).

Groups and Filters

Expressions like the second example are also used to define **Groups** and **Filters**. A group is simply a subset of records within a given table that match some criteria.

Criteria are defined as an expression that can be evaluated as "True" or "False". Additionally you can define a sort order on up to three fields for the group.

PowerBase allows you to define any number of groups associated with a given table and allows you to rapidly switch between them, presenting different

selections and orderings of the records. Group definitions are permanently stored as part of the database definition.

A Filter is simply an additional set of criteria that can be temporarily applied to the records in a group. Filter criteria are expressed in the same way, but they are not saved when the database is closed.

By default, PowerBase will always create a group for each table that has no criteria and so contains all the records in the table.

Views

A **View** defines how the records and fields of a table are presented for display and amendment. There are three types of View:

1. **List View.** This presents a grid in which each record represents a row and the fields selected for display are shown in the columns. Multiple records can be shown simultaneously
2. **Card View.** This presents one record at a time. The name of each field is shown in a column to the left of the screen and the value of the field is shown to the right.
3. **Page View.** This presents one record at a time. However, the size of the page and the positioning of field values and any text labels are entirely user configurable. Page views can be used to produce simple "input forms" or page based printed reports such as standard letters.

By default, PowerBase will produce a List and Card view for each table. The default views display a single copy of each field in the table using default settings for font size, style, colour and alignment.

PowerBase allows you to define any number of views for a given table. Views need not contain all the fields of a table and can contain multiple instances of the same field.

The order in which fields are displayed within a view is entirely user configurable.

The PowerBase Desktop

Once you have installed PowerBase, launch the application by tapping its icon shown in the "Extras Bar".

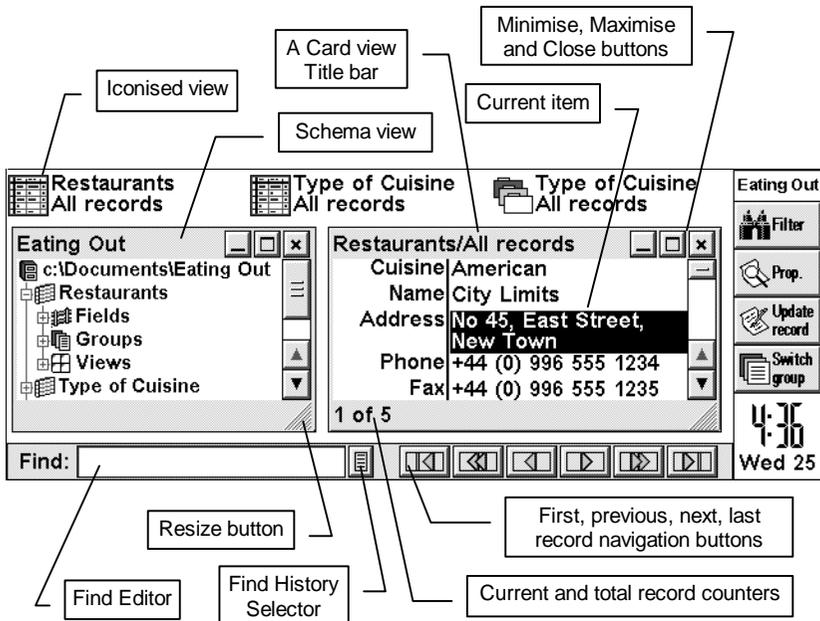
The first time you launch PowerBase, you will be asked to complete a simple registration process. Once this is complete, an empty database file is created

Starting PowerBase

(typically in **C:\Documents**) and the main PowerBase screen is displayed, ready for you to begin the process of setting up your first database.

PowerBase presents the various views of your database in ‘floating windows’. These windows are shown on the **Desktop**. The desktop can also contain a toolbar with tools that allow you to navigate and search for records in the current view.

The figure below provides an annotated example of the Desktop showing these items.



Floating windows

PowerBase windows all offer the following features:

- A Title bar. The title bar displays some informative text as to the contents of the window. Dragging the title bar region will move the window around the screen.

- A Resize button. Dragging the resize button will change the size of the window. The contents will adjust to best fit the available size.
- A Close button. Tapping the close button will close the window.
- A Maximise button. Tapping the maximise button will change the size of the window to cover the entire available screen. When a window is maximised, the button changes to a restore button. Tapping the restore button causes the window to reassume its previous size and position.
- Minimise button. Tapping the minimise button will hide the window and add an iconic representation of the view it contained to the desktop. Selecting and tapping an iconised view on the desktop will restore the view to its previous size and position.

A database may use several tables, some of which contain infrequently changing reference information while others are frequently used for data entry and retrieval. Typically, in such a case, you would create views for the commonly used tables and retain them as icons on the desktop for ready access. Views for the other tables can be created as required using the Schema view as described below.

The Schema view

In addition to the List, Card and Page views previously mentioned, PowerBase has one further type of view - the **Schema View**. The Schema view displays the various components of the database that have been defined (the database schema) as a tree structure.

Where appropriate, an item in the tree can be expanded to show lower levels of detail or contracted to hide these details. Each item in the tree shows a + or - icon indicating whether it can be expanded or not, a further icon indicating the type of database component and the name of the component.

To select an item in the schema view, tap it or use the up and down cursor keys to highlight it. To expand or contract an item, tap on the +/- icon or use the left cursor to contract or the right cursor to expand the item. The **Tab** and **Enter** keys can also be used to expand an item with the additional behaviour that if an item cannot be expanded further, its properties dialog will be displayed. **Shift+Tab** and **Shift+Enter** will also contract an item

At the top level of the schema is the database itself. When the database item is expanded, the tables within the database are shown in alphabetical order. When a table item is expanded, the items for all **Fields**, **Groups** and **Views** associated with the table are shown.

Starting PowerBase

Expanding a **Fields** item will display an alphabetically ordered list of the fields defined for the table. You may notice that each of the fields has an icon before the field name. The icons represent the different field types. Expanding a particular field item will present the properties dialog for the field.

Expanding a **Groups** item will display a list of groups defined for the table. Expanding a particular group item will present the properties dialog for the group.

Expanding a **Views** item will display a list of views defined for the table. Expanding a particular view item causes the view window to be displayed, showing the table records in accordance with the definition of the view and the current group criteria.

Like any other window, the schema view can be minimised to the desktop or closed altogether. If you close the schema view, to reopen it, select the **Create new** menu command from the **File** menu and then select the **Schema view** option.

The Current Window

PowerBase functions will typically apply to the items being displayed in the **Current Window**. The current window and item within it is indicated by a highlight or cursor. You can select the current window directly by tapping it. Alternatively, you can select the **Switch view** command from the **View** menu to cycle through the available views. The **Ctrl+Tab** and **Shift+Ctrl+Tab** hotkeys have an equivalent effect. Within the current window, standard cursor keys or taps are used to select the current item. The **Find** text editor provided by the desktop toolbar can also be the "current item". Pressing the **Esc** key will move the cursor to the Find text editor.

Properties and Preferences

All PowerBase components have "properties" that describe how they are displayed and how they operate. These properties can be reviewed and configured to your personal taste.

To display the properties of the current *item* within the current window (such as the font used for a particular field in a List view), select the **Properties** command from the **File** menu. Alternatively, you can tap the **Properties** button in the main toolbar.

To display the properties of the current *window* or view (such as whether column titles should be shown in the List view), select the **Preferences** command from the **Tools** menu.

To display the overall properties of the Desktop, select the **Desktop preferences** command from the **Tools** menu.

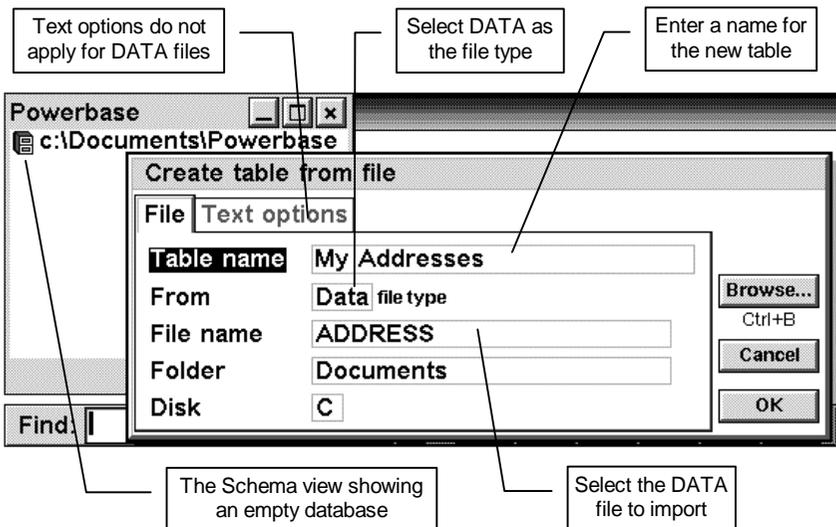
Your first database

To explore PowerBase further, some example data is required. One source of such data is your existing set of files associated with the DATA application, for example your address book. These files are "flat files" so each DATA file will become a single PowerBase table. The second source of example data is the set of database "templates" that are supplied with PowerBase. This section takes you through the steps required to set-up databases from either of these sources.

Importing your Address book

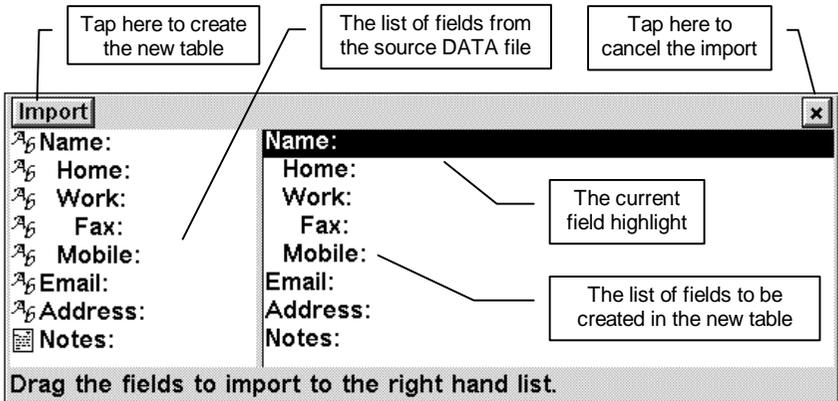
Assuming you have just started PowerBase for the first time, you will have an empty database. To import your address book (or an alternative DATA file):

1. Ensure that the file to be imported is not opened by the DATA application.
2. Select the **More (Import New table)** command from the **File** menu to display the Create table from file dialog. Complete the dialog as indicated in the following figure.



Starting PowerBase

- Once the dialog has been completed, select the individual fields from the DATA file to be imported. The fields (labels) in the DATA file are shown to the left of the field selector together with an icon indicating their field type. The fields, to be generated in the new table are shown to the right of the screen. By default, all fields will be included in the new table.



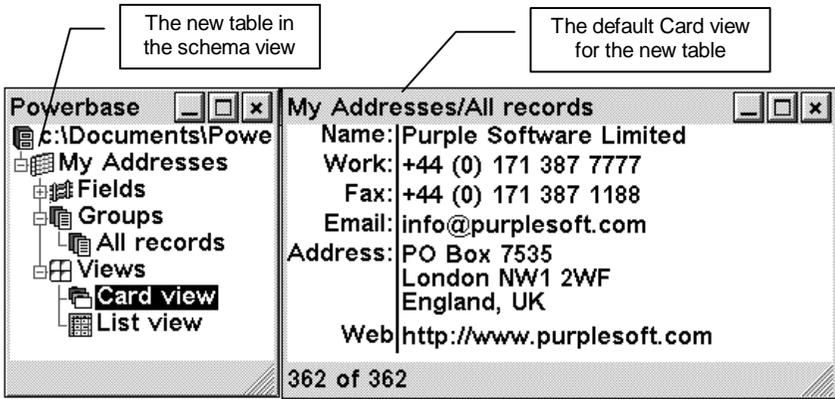
To remove a field from the table, select and drag it out of the list of table fields.

To add (or duplicate) a field, select and drag its name from the list of fields in the DATA file into the list of table fields.

To display or amend the properties of the fields to be created in the new table, use the up and down cursor keys to highlight the field and then select the **Properties** command from the **File** menu.

The **Enter** and **Tab** keys will also cause the properties to be displayed, as will tapping the **Properties** toolbar icon.

- Once all the fields have been defined in the new table, tap the **Import** button to create the new table. To cancel the process, tap the **Close** button or press the **Esc** key.
- Once the new table has been created, it will be shown in the schema view. You can now expand the items associated with the new table. Expand the default Card or List view item to display the imported data as shown in the following example.

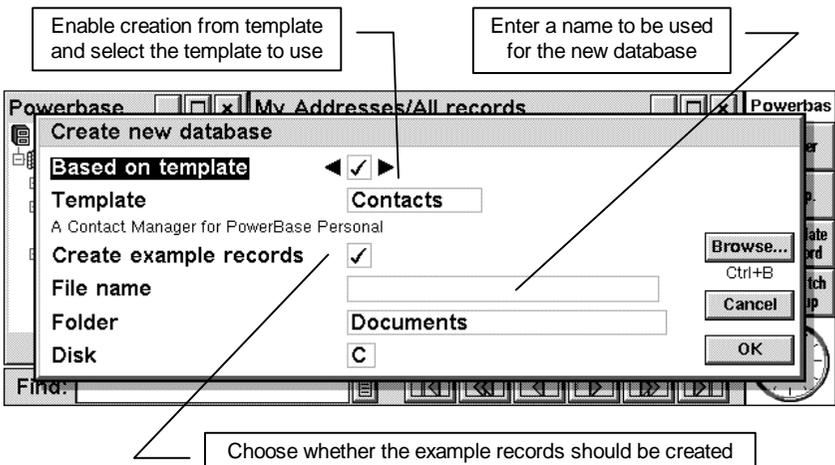


Templates

PowerBase is supplied with six "template" databases. A template comprises the definition of one or more tables within a database plus example data with which to populate the database tables. The supplied templates are designed to demonstrate a range of PowerBase applications. They also provide examples of some of the more sophisticated uses of the expression evaluator and offer a starting point for the design and development of your own databases.

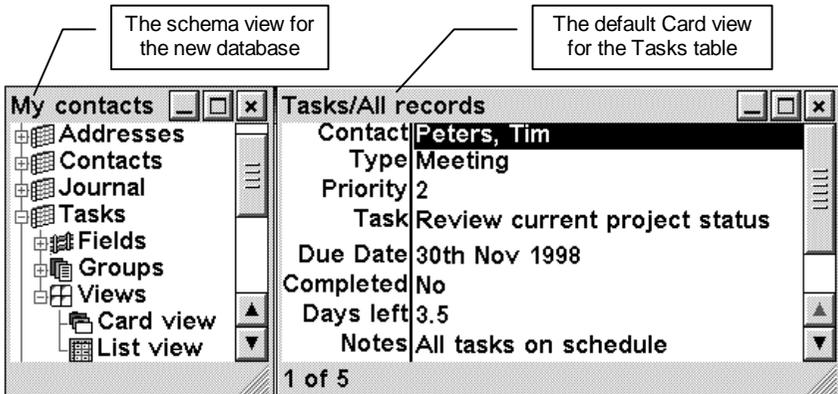
Creating a new database from a Template

To create a new database from a template, select the **Create new (Database)** command from the **File** menu to display the following dialog.



Starting PowerBase

Confirm the dialog to create the new database. Once the new database has been created, the new schema view will be displayed. You can now expand the items associated with the tables in the database. Expand Card or List view item to display the data as shown in the example below.



The supplied templates

PowerBase comes with six templates, covering a range of potential uses. These are:

1. Eating out

The main table in the Eating Out database is the Restaurants table. This provides a list of restaurants for which you can store the name and address details of places to eat out. Restaurants can then be categorised by type of cuisine, price range and various ratings. The possible values for these categories are contained in the other tables. A given category for a restaurant is established using a relational field.

The restaurant table also demonstrates the use of Boolean fields to indicate such items as whether certain credit cards are honoured. You may also like to look at the definition of the rating fields to see the use of an expression that converts a numeric rating into a visual "starred" display.

2. My Library

The main table in the My Library database is the Collection table. This provides a consolidated list of books, videos and music that you might own. Items in the collection can be categorised in various ways using relational fields.

Note in particular the relationship to the "Artists" table. Instead of directly storing the Author of a book or the Star of a film as a text field, a link is established to the Artists table. This means that when you want to add a new book by a new author, you must first create an entry for the author in the Artists table. However, adding any subsequent book by the same author only needs a reference to the existing record in the table of artists. The same approach would apply for Actors who may star in several films. The details for the Publishers are handled in a similar way by reference to the Publishers table.

You may like to look at the definition of the Artists field in the Collection table. The value displayed for the related artist is in fact an expression that produces a concatenation of two fields in the Artist table - the name and type of artist.

3. World Cities

The prime purpose of the World Cities database is to demonstrate sophisticated use of the expression evaluator. The database contains four tables:

TimeZones lists a number of zones in which the effect of Daylight Saving Time can be enabled or disabled.

Countries lists a number of countries together with their ISDN details.

Cities lists a number of cities together with items such as their latitude/longitude, whether they are a capital city, ISDN details and time difference from GMT. Country and TimeZone details are supplied using relational fields.

The Calculator table allows you to select a "home" city and a second city using links to the Cities table. The remaining fields are calculations of such items as the time at the two cities and the distance between them in Kilometres.

4. Price List

The Price List database demonstrates the use of calculated fields. In this case, it shows the list price of products being derived from the cost of the product with percentage mark-up, discount and tax rates.

Supplier and Product details are maintained in separate tables. The main Prices table also contains some simple stock information and indicates whether the product is available ex-Stock.

Starting PowerBase

5. Wines

The Wine List database demonstrates how some of the techniques used in Price list and My Library might be applied to store details about your wine collection.

6. Contacts

The Contacts database implements a modest contact management application. There are four main tables with other tables providing various categorisations used by the main tables.

The Contacts table simply provides the means to record details about a unique individual or organisation.

The Addresses table provides the means to store *multiple* addresses for a Contact. This means that any number of telephone numbers, postal addresses or email addresses can be entered for a Contact. You may like to note the use of the evaluator string function chr\$(6) used in the expression that defines the information displayed for a Contact. This indicates a new paragraph producing a string that is displayed over multiple lines. The display height of a row in the any of the views can be varied to hide or reveal such multi-line information. Additionally, an address record can be related to any other address. In this way, a single address can be created for a company head office and linked to the personal address details for contacts within that company.

The Tasks table provides a list of categorised tasks with priorities, due dates and so on. Tasks can optionally linked to a Contact.

The Journal table demonstrates how PowerBase can "date stamp" records. A new entry in the Journal table automatically stores the date and time of its creation in the "Date" field. Journal entries provide a memo field for recording up to 64k of text and an embedded field for recording documents. A Journal entry can optionally be linked to a Contact.

Closing PowerBase

To close the PowerBase application, select the **Close** command from the **File** menu. Note that if you are in the process of creating or amending a record, menu commands are not available. You must save or cancel any changes made to the record before performing any other action.

3

Using PowerBase

This chapter describes the main features of the PowerBase application and how they are used to manage your data.

Creating and evaluating expressions

PowerBase incorporates a powerful expression evaluator. An expression can be simply a number or a complex formula. You can use expressions to:

- Define the default value for a field.
- Define the value of a calculated field.
- Define the information displayed for a related field.
- Define logical criteria by which to filter a view or define a group.

The expression evaluator is therefore central to the PowerBase application. An understanding of the basics is required to obtain some of the more significant benefits of the application.

The full range of parameter types, operators and functions supported by the expression evaluator are detailed in the [Technical Reference](#) guide.

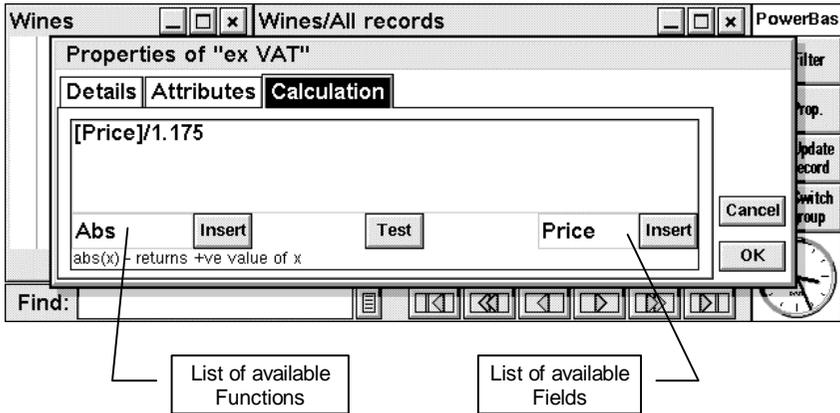
The purpose of an expression evaluator is to calculate the result of an expression. A simple expression such as '1+2' will evaluate to a numerical result of '3'. A slightly more complex expression such as "'Mr'+ ' Jones'" will evaluate to a textual result of 'Mr Jones'.

In these examples, both expressions used fixed parameters. PowerBase extends this approach such that you can also use the names of fields within a table. When such an expression is evaluated, it will take the content of the indicated field its source of data.

Using PowerBase

Since the expression evaluator is so important, PowerBase provides an 'Expression evaluation wizard' to help you construct expressions more easily.

The figure below shows the expression wizard being used to define the value of a calculated field. The table in question contains a currency field called 'Price' and a calculated field called 'ex VAT'. This second field is defined by the expression that calculates the value for 'ex VAT' from the 'Price' field.



Note that the 'Price' field name was enclosed in square brackets within the expression. The expression evaluator needs to be able to distinguish between functions such as 'Sin' and 'Cos', textual values and field names. It does this by applying the following rules:

- Any item enclosed in square brackets is assumed to be a field name and must match the name of a field.
- Items enclosed in double quote marks are assumed to be text values.
- Any item that does not obey either of the above is assumed to be an operator, operand or function name.

The expression evaluator can also use the result of one expression in another. For example, if you had a field called 'VAT amount' you could define the expression '[Price]-[Price]/1.175' to calculate the values for the 'VAT amount' field. Alternatively, you could define the expression to be '[Price]-[ex VAT]'.

Using the first expression will be slightly faster as only one expression has to be evaluated. However, it is unlikely that you will notice any difference as PowerBase evaluates expressions very rapidly.

The expression evaluation wizard contains three major components. Firstly, there is the large area in which you may type in your expression.

To help you construct expressions the wizard contains a list of the available functions to the left and a list of the available fields to the right.

Tapping on the list of functions will display a choice list of the functions available. Choosing a function from that list will close the list and update the descriptive text displayed below the function name that provides a brief description of operation of the function and its parameters.

Tap the **Insert** button situated immediately to the right of the function name list to insert the function name at the current cursor position in the editor.

The function name list only contains a list of the functions available, it does not list the various operators or constants supported by PowerBase.

Tapping on the list of field names will display a choice list of the available fields.

Tap the **Insert** button to the right of the field name list to insert the field name, enclosed in square brackets, at the current cursor position.

Tap the **Test** button to check the expression. If PowerBase detects an error, it will move the cursor to the source of the error.

The most common errors are:

- **Invalid variable name.** It is likely that you have entered the name for a field that PowerBase cannot find or have forgotten to add the trailing ‘]’ character.
- **Incorrect number of arguments.** You have not entered enough parameters for the function.
- **Invalid argument.** You entered a parameter that does not have the correct data type. For example, a function requires an integer parameter but you have given a text parameter.

Databases, Tables and Fields

The nature and structure of information recorded in a database is defined by the tables and fields it contains.

Opening and creating databases

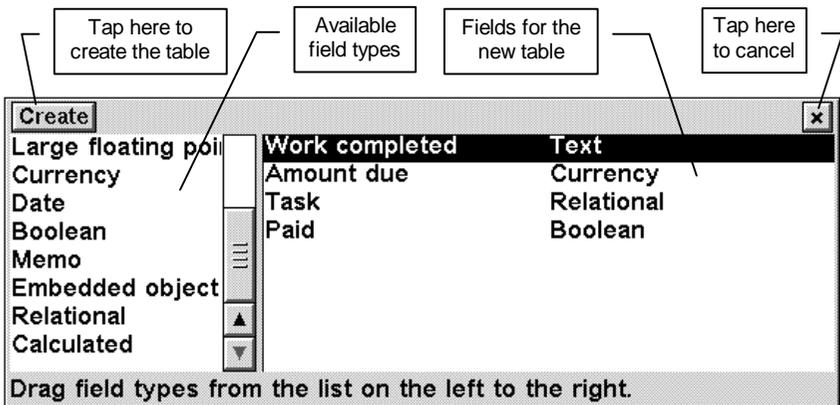
To open an existing database, select the **Open** command from the **File** menu.

To create a new database, select the **Create new (Database)** command from the **File** menu. The **Create new database** dialog will be presented. This enables you to enter the name of the database file and choose whether the new database should be based on an existing template and if so whether it should be populated with example records.

Adding tables to a database

To add a new table to a database, select the **Create new (Table)** command from the **File** menu. The **Create table** dialog will be presented, allowing you to enter a name for the new table. Once you accept this dialog, the process of defining the fields within the table begins.

The field definition window presents a list of all possible field types to the left of the screen. The fields defined in the table are listed to the right of the screen - initially this list is empty.



To add a field to the table, select and drag the desired type, dropping it into the list of table fields. A series of dialogs will be presented, allowing you to enter a name for the field and set-up its attributes and display format. The range of

available field types, their attributes and display formats are detailed in the [Technical Reference](#) guide. Once all the desired fields have been defined, to create the table, tap on the **Create** button in the top left hand of the display. The table will automatically be assigned an "All records" group and the default card and list views.

To cancel creation of the table, tap the **Close** button in the top right hands corner of the display.

Once the table has been created, it is shown in the Schema view. If you have closed the Schema view, a new one can be created by selecting the **Create new (Schema view)** command from the **File** menu.

Deleting and duplicating tables

To delete a table, highlight its name in the schema view and select the **Delete** command from the **File** menu. Deleting a table also deletes all other database items associated with it. This includes all data in any table record and any views or groups associated with the table. However, if PowerBase detects that records in *other* tables depend on information in the table being deleted (via a relational link), the table will not be deleted and PowerBase will report an "In use" error. In this way, the "referential integrity" of the database is preserved.

To duplicate a table (in terms of its field definitions), highlight its name in the schema view and select the **Duplicate** command from the **File** menu. You will be prompted for a new table name and then the field definition process (described above) will begin with the new table having an exact copy of the field definitions in the original table. Note that groups, views and data records of the original table are not duplicated.

Adding, amending and deleting fields

To add a field to an existing table, highlight the table name (or any of its components) in the schema view. Alternatively, make a view window associated with the table the current window. Select the **Create new (Field)** command from the **File** menu. A series of dialogs will be presented, allowing you to enter a name for the field, select its type and set-up its attributes and display format. Once the new field has been created, it will be shown in the schema view. Note that new fields are not *automatically* added to existing views.

To change the name of a field, its display format or attributes highlight the field name in the schema view and select the **Properties** command from the **File** menu or simply press the **Enter** or **Tab** keys. Note that the field type can only be assigned when the field is created and cannot be changed at a later

Using PowerBase

point. Note that changes to field attributes that have implication for the field content (such as the length of a text field or whether a field can have a zero value) do not affect existing records. The new constraints will only be applied when a new record is created or an existing record is amended.

To delete a field, highlight its name in the schema view and select the **Delete** command from the **File** menu. Deleting a field also deletes all data associated with the field in all records. If "referential integrity" is compromised, the field will not be deleted and the "In Use" error will be reported.

To duplicate a field definition, highlight its name in the schema view and select the **Duplicate** command from the **File** menu. The dialogs for adding a new field will be presented pre-loaded with a copy of the settings for the original field.

Importing and exporting data

PowerBase provides functions to import and export records within a table. The external file formats supported by PowerBase "Personal" are currently those of simple text files and the DATA application.

To import data into a *new* table, select the **More (Import new table)** command from the **File** menu. A detailed description of the subsequent process is given in the [Importing your Address book](#) section above.

To import data into an existing table, either highlight the table name (or one of its components) in the schema view or make a view associated with the table the current window. Select the **More (Merge into table)** command from the **File** menu. The subsequent process is very similar to importing a new table. The essential difference is that instead of selecting any number of fields from the source file, you merely drag fields onto a fixed list of existing fields in the target table. In this way, you establish the mapping between source and target fields.

Exporting data from a table is subject to a Group criteria. This allows you to selectively export table records. To export data, highlight the group name in the schema view or make a view associated with the group the current window. Select the **More (Export group)** command from the **File** menu.

The subsequent process is again very similar to that for importing a new table. Any number of fields in the existing table can be dragged into the list of fields to export. Note that dragging fields within the list of fields to export allows you to control the physical order in which they are stored in the export file.

As always when converting between file formats, the success of field conversion is limited by the data types supported by the target file format.

Thus while import and export of simple numbers or text fields is generally unproblematic, limitations may apply to the handling of more complex field types. Several of the important effects to bear in mind are:

- DATA and text files do not support calculated or related fields so these are exported as the result of the underlying expression rather than the underlying calculation.
- DATA supports embedded objects *within* text or memo fields whereas PowerBase supports these as *separate* field types. Thus PowerBase can export an embedded field to DATA but importing objects within text fields is not supported. Import and export of embedded objects from and to text files is not supported.
- Memo fields exported to text files will be truncated at 255 characters. Import of memo fields from text files is not supported.
- The new paragraph character chr\$(6) within a text field will be expanded to two characters (CR and LF) on export to a text file. This may result in truncation of the field in the text file if the expansion causes the 255-character limit to be exceeded.
- Import of relational fields from a text file is supported. The value in the text file being that of the related field identifying a record in the related table. Import of relational fields from DATA is not supported.
- Date/time fields are exported to text files in a fixed format independent of their display format. This is to provide compatibility with desktop applications such as Access. Date/time fields can be imported from text files provided that, their format is recognised as a date/time value by PowerBase.

Compressing the database

The underlying filing system of your handheld computer used by PowerBase to store the database generates additional information to aid the efficiency of its low-level data retrieval. This information adds to the overall size of the database file on disk. To ensure this information is optimally organised and potentially reduce the overall size of the file, you can "compress" the database.

To compress the database, select the **Compress** command from the **Tools** menu. Note that compressing a database can take an extended time and that if the size of a database exceeds a certain threshold, PowerBase will offer the option to compress the database when it is opened.

Views and Groups

While the definitions of tables and fields determine how your data is stored and organised, PowerBase offers a wide range of features allowing you to control how this data is presented.

Groups allow you to specify a subset of records within a table and the sort order in which they are displayed. These settings are stored within the database and can be restored at any time. You can also temporarily refine the subset of records by using the Find or Filter functions and use the Sort function to change the default sort order.

Views allow you to specify a subset of the fields within a table and how they are to be displayed on screen. View windows provide the means to display and print your data. They also provide the user interface for creating, amending and deleting table records.

A view displays the subset of records defined by a group and reflects any subsequent use of the Find, Filter or Sort functions. A view can switch between any group defined for the table.

Working with groups

To create a new group, highlight the table name (or any of its components) in the schema view. Alternatively, make a view window associated with the table the current window. Select the **Create new (Group)** command from the **File** menu. The **Create new group** dialog will be displayed. The **Details** page allows you to enter a name for the group and an expression that will be evaluated as true or false for all records in the table. Only those records for which the evaluated result is true will be displayed when the group is in effect. The **Sort order** page allows you to define up to three fields that determine the sort order in which records will be displayed when the group is in effect.

To amend the definition of a group, highlight its name in the schema view and select the **Properties** command from the **File** menu.

To delete a group, highlight its name in the schema view and select the **Delete** command from the **File** menu. Note that deleting a group has no impact on the actual records in the table.

To duplicate a group, highlight its name in the schema view and select the **Duplicate** command from the **File** menu. This will present the **Create new group** dialog initialised with the properties of the original group.

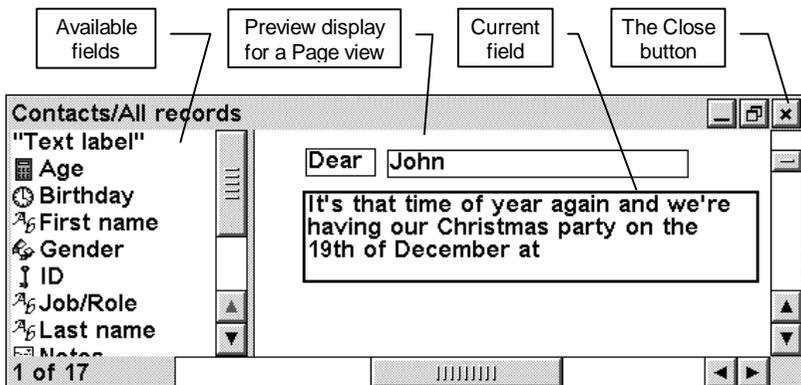
Note that PowerBase will always ensure that a record in a view obeys the group definition expression. For example, assume you have a group whose definition is “wines whose price is less than £3.99”. If you update a record in this group because the price has changed to £4.39, this record will be removed from the group since it no longer obeys the group definition. Similarly, if a record is updated such that the price changes from £4.99 to £3.79 then PowerBase will add this record to the group since it now obeys the group definition.

One exception to the above rule is the case where the group criteria expression references a relational field. If *in the related table* you change some component of a field that would cause evaluation of the group criteria in the original table to change, this is not automatically reflected in views of the group. To handle this situation, should it arise, select the **Refresh group** command from the **View** menu to force a full regeneration of the group information for each record.

Working with views

To create a new view, highlight the table name (or any of its components) in the schema view. Alternatively, make another view window associated with the table the current window. Select the **Create new (View)** command from the **File** menu. The **Create new view** dialog will be displayed. The **Details** page allows you to enter a name for the view and to select the type of view (List Card or Page). If the Page type is selected you can also define the page set-up in terms of **Page size** and **Margins**.

On confirming the dialog, the view layout editor is presented. The layout editor comprises a list of the available fields in the table to the left of the screen and a preview display of the view to the right of the screen.



Using PowerBase

To add a field to the display, simply drag it into the preview display and place it in the desired location. To remove a field from the view, simply drag it out of the preview display. Note that removing a field has no impact on the table records.

Existing fields in the preview display can be repositioned in the same way. In this way you can control:

1. The column order of fields in a list view.
2. The row order of fields in a card view
3. The absolute position of a field in a page view

For a page view, an addition item is shown in the list of available fields - a **Text label**. Text labels can contain any fixed text. You might use this to display the name of a field or a standard paragraph in a form letter. Text labels are otherwise treated in the same way as other fields.

Note: Text labels alone may be suitable for some simple standard forms. However, they cannot achieve the effect traditionally associated with a "mail merge" operation in which a field value is inserted into a paragraph.

Although PowerBase offers no direct support for mail merge, this effect can be readily achieved using a calculated field. For example, consider the expression:

"The outstanding sum of £"+str\$([Amount due])+" is now due within "+str\$([Period])+" days."

This would produce a sentence of the form: "The outstanding sum of £1559.95 is now due within 14 days." This could be intermingled with text labels providing standard paragraphs in a page view.

In page view, the explicit size and position of the fields must be set. Fields are positioned by tapping their centre and dragging then to the desired location. Fields are sized by dragging their borders.

Alternatively, you can use the **cursor** keys to move an item or use the **cursor** keys in combination with the **Shift** or **Ctrl** keys to change the size of the item.

Fields within a view have individual display properties. These include such items as:

- The font colour, style, and size
- The alignment of the information to be displayed
- Whether page view items should be bordered.

To review or set display properties for a field, simply highlight it in the preview display and select the **Properties** command from the **File** menu.

The view also has a number of general preference settings. The review or set these, select the **Preferences** command from the **Tools** menu. The range of preferences varies with the type of view. These cover items such as the font to be used for field labels in a card view, how grid lines should be drawn in a list view and the page set-up details for a page view. Two particular view preferences are common to all view types and worth of special note:

1. **Track current record.** This determines whether "current record" shown in the view should be kept in line with the "current record" of other views of the same group.

For example, with tracking enabled, you can have a card and a list view on screen and as you move from record to record in the list view, the card view will update to show the same record.

2. **Display field properties.** This option determines whether, the display properties of a field should be shown when you tap on it in a view window.

Once the definition of the new view is complete, select the **Finish editing view** command from the **View** menu or tap the **Close** button.

To amend the field layout for a view, make the appropriate view window the current window and select the **Edit view layout** command from the **View** menu. The view layout editor described above will be displayed.

To review or amend the general preferences for a view, make the appropriate view window the current window and select the **Preferences** command from the **Tools** menu.

To review or amend the display properties for a field within a view, make the field the current item in a view window and select the **Properties** command from the **File** menu. Alternatively, if the option is enabled, you can simply tap the field.

To delete a view, highlight its name in the schema view and select the **Delete** command from the **File** menu. Note that deleting a view has no impact on the associated group or any records in the table.

To duplicate a view, highlight its name in the schema view and select the **Duplicate** command from the **File** menu. This will present the **Create new view** dialog initialised with the properties of the original view.

Using PowerBase

Using views

When records are displayed in a view window, the following general features are available.

- Use the standard cursor keys to select the current field. In a list view, the current record can also be selected.
- The display can be zoomed in and out using the standard commands.
- **Ctrl+Tab** causes the next view window to become the current window. This is equivalent to selecting the **Switch view** command from the **View** menu.
- **Shift+Ctrl+Tab** causes the previous view window to become the current window
- **Tab** will open the current field in the current record for editing
- **Enter** will cause the current record to be advanced to the next record in the group.
- **Shift+Enter** will cause the current record to be reversed to the previous record in the group.
- The view can be applied to a different group directly by selecting the **Switch group** command from the **View** menu.
- The view can be cycled over all available groups using the **Shift+Ctrl+Q** hotkey.
- In a list view, the last row is empty. If this is the current row when you start editing a record, a new record will be created.
- In a list view, at the top of each column, a button displays the name of the field. Tap the button to sort the records in the displayed group. Tap the button a second time will reverse the order of the sort.
- In a list view, if the content of a field is too large to fit the width of the column, the text will wrap onto subsequent lines. To be able to see the text you can adjust the row height by dragging the row title region shown to the left of the first column. Alternatively use the **up** and **down** cursor keys in combination with the **Shift** or **Ctrl** keys. Individual row heights are not supported in list view, so all rows are adjusted to the same height.

- In a list view, column widths can be set independently for each column by dragging the region between two column titles. Alternatively, use the **left** and **right** arrow keys in combination with the **Shift** or **Ctrl** keys.
- In a card view, if the content of a field is too large to fit the width of the column, the text will wrap onto subsequent lines.

To be able to see the text you can adjust the individual row height for the field by dragging the row title region shown to the left of the view. Alternatively use the **up** and **down** cursor keys in combination with the **Shift** or **Ctrl** keys.

Sorting

Sorting is simply a definition of the order in which you wish to display records within a group. Numeric fields are sorted according to the underlying value of the field. Text fields are sorted in alphabetical order.

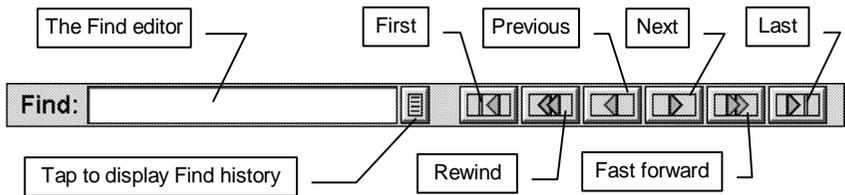
A sort can be specified on up to three fields in a table. The sort order for each field can be ascending or descending.

- To specify a sort order, make a view window associated with the group to be sorted the current window and select the **Sort** command from the **Records** menu. The Sort group dialog will be displayed. This can also be used to remove a previously applied sort order.
- In a list view, you can sort the group by tapping the column heading for a field. This is described below.
- Only the first 255 characters of a memo field will be used sort sorting.
- Sort an embedded field does not produce any meaningful sort order

PowerBase will always ensure that a group obeys its current sort definition. For example, assume you have a group that is sorted by 'Price'. If you update a record in this group and change the contents of the price field, PowerBase will ensure that the record is positioned correctly within the group after you save the record.

The Desktop toolbar

The desktop toolbar provides a number of tools for finding and navigating records. In all cases, the operation of these tools is directed at the current window or in the case of the Find editor, the window that was current directly before the Find editor itself. Note that should the previous window be the Schema view, Find will report “Unable to find records”.



Using the Find editor

To use the Find editor, make it the current window by tapping it or by pressing the **Esc** key. Type the text you wish to find and press the **Enter** key. PowerBase will match the text you have entered with the information displayed for each record in the current group and update all the associated views to show only those records for which a match is found.

To cancel the effect of the Find operation and cause the view to display all records within the current group, press the **Esc** key.

To recall text used for a previous Find operation, tap the Find history button. A popup list will show the texts most recently entered in the Find editor. Select one of these texts for it to replace any existing text in the Find editor.

How finding works

Finding will match the text you have entered with information that is displayed on screen. The displayed information is not necessarily what is stored in the field. For example, a date is stored as a series of eight bytes of information. The date “24th Sep 1998 5:24 pm” happens to be stored as the hexadecimal number 0x00e0164cccf337c0. When you use ‘Find:’ typing in “24th Sep” is far easier than attempting to determine the internal representation.

An important point to note is that if you change the display format for a field you change what is displayed on screen. For example changing the display

format of a date field from 'Date and time' to 'Short date' would change the display from "24th Sep 1998 5:24 pm" to 24/09/98. If you now attempted to find "24th Sep", you would fail since that text is no longer displayed on screen. You would need to find "24/09" to achieve the desired result.

Find properties

To display the Find properties dialog, make the Find editor the current window by tapping it or by pressing the **Esc** key and select the **Properties** command from the **File** menu. You can configure the following items.

1. Which fields to search when matching records
 - All displayed fields. Only those fields displayed in the current view are searched.
 - All fields. All fields in the table will be searched even if they are not currently displayed within the view. PowerBase uses the format that would be used if the field were to be displayed within the view.
 - Only the currently selected field.

In all cases, PowerBase will also obey the field attribute 'Search when finding'. See the section on field attributes in the [Technical Reference](#) guide for further information. PowerBase will never search embedded fields.

2. Which rule PowerBase applies to perform matching.
 - Contains. A record where any of the fields contain the text is considered to match.
 - Does not contain. A record where none of the fields contain the text is considered to match.
 - Starts with. A record where at least one of the fields starts with the text is considered to match.
 - Does not start with. A record where none of the fields start with the text is considered to match.

You can also configure whether matching is case sensitive, the time delay use by the Fast forward and Rewind buttons and the size of the find history list.

If a Find operation does not produce the result you expect, for example records that you know exist are not displayed, you should check that the Find properties are set appropriately.

Using PowerBase

Using the record navigation buttons

Associated with the Find editor, are a number of buttons that perform the following functions:

- **First** updates the display to show the first record in the current group.
- **Rewind** repeatedly moves to the previous record stopping when it reaches the first record. The rate of movement is one of the properties of the Find editor.
- **Previous** updates the display to show the previous record in the group.
- **Next** updates the display to show the next record in the group.
- **Fast forwards** repeatedly moves to the next record, stopping when it reaches the last record. The rate of movement is one of the properties of the Find editor.
- **Last** updates the display to show the last record in the group.

Note you can also move to the next or previous record by tapping the current or total record counters shown in the bottom left corner of the view window. To move to the previous record, tap the current record counter. To move to the next record, tap the total record counter.

Filtering

In contrast to Finding, Filtering operates by matching all records for which an expression evaluates to a result of 'True'.

In general, Filtering does not take any notice of the field formatting information and operates directly on the data stored within a record. A variation to this rule occurs in the case of relational fields where Filtering operates on the result of the display expression for the field.

This means that for simple queries, a more complex expression may be required. For example, to find the records that contain a date of 24th September you would use the expression:

Month([Date])=9 AND Day([Date]) = 24

This expression refers to a single field so matching will only involve the [Date] field rather than *all* fields in the record. To check just two date fields, the expression would need to be extended as follows:

(Month([DateA])=9 AND Day([DateA]) = 24) OR
(Month([DateB])=9 AND Day([DateB]) = 24)

In this example, finding is simpler and easier. However, filtering can achieve results that are not possible using finding. For example, suppose you wanted to find all the red wines from Australia that cost less than £3.99. This would be quite difficult to achieve using find but relatively simple for a filter expression such as:

[Colour]="Red" AND [Country]="Australia" AND [Price]<3.99

To create and apply a filter expression, select the **Filter** command from the **Records** menu.

To cancel the effect of the filter expression and cause the view to display all records within the current group, you can simply delete the expression.

Alternatively, press the **Esc** key from within the Find editor to cancel the effect of the filter but leave the filter expression in place.

Dialling

PowerBase supports the standard features for DTMF tone dialling of telephone numbers. To dial a number, press the **Fn+Menu** keys.

Any field in the current record, in the current view that has its "**Search for phone number**" attribute enabled and contains a valid number will be presented in the **Dial** dialog.

Printing

PowerBase provides the standard facilities for producing printed output. Select the **Printing** option from the **File** menu to use these facilities.

Printed output will contain the records shown in the current view with formatting based on the prevailing display properties.

Note that when printing a Page view, you should ensure that the page sizes defined for the printer and for the Page view itself are equivalent.

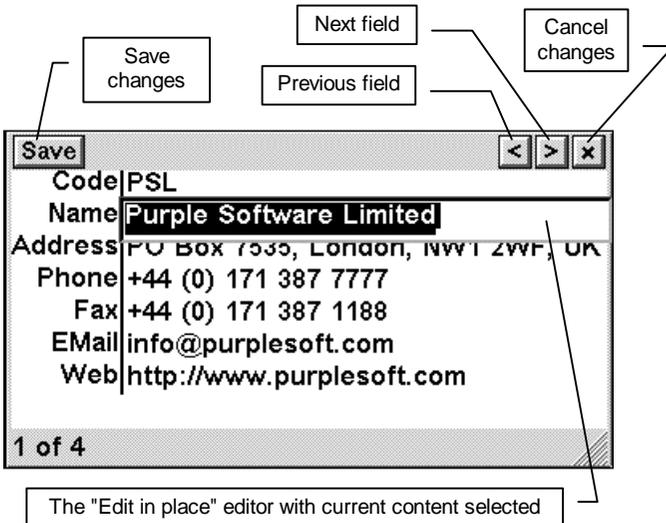
Creating, updating and deleting records

Irrespective of the view you are currently using, you may update existing records or add new records to the table associated with the view. PowerBase supports 'edit in place' as the primary means to edit records.

This means that rather than displaying a separate dialog in which you edit records PowerBase will display a text editor placed on top of the field you are editing.

- To create a new record in the current view, select the **Create new (Record)** command from the **File** menu or the **New record** command from the **Records** menu. These commands are equivalent.
- To duplicate the current record in the current view, select the **Duplicate** command from the **File** menu. This is equivalent to creating a new record with fields initialised to the values contained in the original record.
- To update the current record in the current view, select the **Update record** command from the **Records** menu or simply press the **Tab** key.

When creating or updating a record the current view window switches to "edit mode". This is shown for a Card view in the following figure.



Note that when you create a new record, PowerBase will set all the fields to contain their default values. For example, an auto number field will be displayed with the value that is assigned to that field.

If you duplicate a record, the content of the original field overrides any default values for fields in the new record.

Note also that to guarantee the uniqueness of auto-number fields, when duplicating a record, auto-number values are not duplicated and a new unique value is assigned as normal.

While you are editing a record, other PowerBase functions are disabled until you leave edit mode by saving or cancelling any changes to the record.

- To save changes to the record, tap the **Save** button or press the **Ctrl+S** hotkey.

When you save a record, PowerBase will ensure that all the fields that have their **Allow blank field** attribute disabled contain some data. If this condition is not met, PowerBase will report the error and not allow the record to be saved.

Note that record editing is based on the current view. The view may *not* include all the fields in the table. If the view excludes a field for which the **Allow blank field** attribute is disabled, new records cannot be created using this view as there is no way to enter the required information.

- To cancel any change to the record, tap the **Close** button or press the **Esc** key. If changes have been made to the record, you will be asked to confirm this action.

When you start editing a record, the "in place" editor will be opened on the current field. Note that as you move from field to field, PowerBase will automatically skip any field that cannot be amended.

Fields that cannot be amended include calculated fields, auto-number fields and any field for which the **Allowed to change** attribute is disabled.

- To move to the next field, tap the **Next field** button or press the **Enter** key.
- To move to the previous field, tap the **Previous field** button or press the **Shift+Enter** hotkey.

Using PowerBase

The "In place" editor

The "In place" editor is a text editor. This means that for field types that do not have text content, PowerBase will interpret, convert and validate the text that is entered to generate the actual value to be stored in the field.

If PowerBase cannot correctly determine a value for the field, or the value is outside a valid range, PowerBase will display an error message and will not allow you to move to another field or save the record.

The editor will perform a wide range of conversions. For example for a numeric field you could enter a value as "20000" or in engineering notation as "2e5".

Similarly, for a date/time field (assuming your system date format is set to the European day/month/year style) you could enter text such as "12 sep 98", "12/9/98" or "12 9 1998". PowerBase will correctly identify the 12th of September 1998.

The size of the editor is determined by the width and height of the associated row and column in the view. This means that if there is insufficient width to display all the field content, the content will wrap onto multiple lines.

Similarly, if a text or memo field contains the new paragraph character, this will be interpreted correctly as multi-line data.

In these cases, simply use the cursor keys to scroll the text displayed by the editor.

Note that if you wish to enter a new paragraph character, you need to press the **Ctrl+Enter** key combination as the **Enter** key takes you to the next field.

The editor supports the standard keystrokes for text selection and clipboard operations.

- Use the cursor keys in combination with the **Shift** key to select text.
- To cut the currently selected text, press **Ctrl+X**
- To copy the currently selected text, press **Ctrl+C**
- To paste text at the current insertion point, press **Ctrl+V**
- To restore deleted text, press **Ctrl+Z**

Specialised editors

Although, in most cases, the in-place editor provides a fast and convenient means for data entry, you may prefer a more structured form of input.

For each field type, PowerBase offers the appropriate type of editor, such as a number editor or date editor that contains a calendar. To activate these editors from within the in-place editor, simply press the **Tab** key.

Note that it is not possible to enter a null or "blank" field using a specialised editor and the in-line editor must be used to achieve this effect.

Note also that for memos or long text fields, the specialised text editor provides a larger display that you may find more convenient to use.

Related fields

When you edit a record containing a relational field, the in-line editor operates on the value of the field in the related table by which the link is established. This value is not necessarily the value displayed for the field in the main view.

The display value derives from the display expression evaluated for the related field. In contrast, the specialised editor for a related field presents a choice list containing the results of the display expression evaluated for all records in the related table. This allows you to select directly the record to which the link is to be established.

Unique value lists

In some cases, you may wish to set the value of a field to that of another record in the same table. To do this, press either **Ctrl+Tab** or **Shift+Tab** from within the in-place editor. PowerBase will display a choice list containing all the unique values of the field that are currently stored in the table. Selecting an item from this list will update the editor to contain the value chosen.

The location of updated records'

PowerBase will always ensure that only records belonging to a group are shown in views associated with the group and that records are maintained in the currently defined sort order.

When you create or update a record it may no longer meet the group definition criteria for the current view. In this case, the record will no longer appear in the current view. Similarly, a change to a field that determines the order of records in the view will cause the record to change position within the view.

Using PowerBase

Should you wish to avoid these effects, you can simply choose to edit records from a view in which all records are displayed and no sort order is applied.

Deleting records

To delete the current record in the current view, select the **Delete** command from the **File** menu or the **Delete record** command from the **Records** menu. These commands are equivalent.

PowerBase will not allow you to delete a record that is currently being used by a relational field in another table. Any attempt to do so will result in an 'In use' error.

4

Technical Reference

This chapter provides technical reference information for key PowerBase components.

General information

1. The maximum number of fields per table is 256.
2. The maximum size of data within a record is 4096 bytes. Note that this does not include data associated with the Memo or Embedded field types. Additionally each record has a fixed overhead of 34 bytes that PowerBase uses for internal data management.
3. The maximum number of records per table is in the order of $2^{*}30$
4. The maximum number of tables per database is $2^{*}32$
5. PowerBase can sort over 750 records per second, sorting a single text field, using a Psion Series 5, with an 18Mhz ARM 710 microprocessor.
6. Maximum recursive depth of the expression evaluator is 6.
7. Date and time values are supported until some 300,000 years in the future.
8. PowerBase stores general user preferences in the file:

C:\System\Apps\PowerBase\PowerBase.Ini

Field types

PowerBase supports the following field types:

1. **Text**, up to 255 characters can be stored in a text field. (Storage required = length of text + 1 byte)
2. **Boolean**, numbers that either have the value 0 or 1 can be stored in fields of this type. (Storage required = 1 byte)
3. **Small integer**, numbers in the range 0 to 255 can be stored in fields of this type. (Storage required = 1 byte)
4. **Medium integer**, numbers in the range -32768 to +32767 can be stored in fields of this type. (Storage required = 2 bytes)
5. **Large integers**, numbers in the range -2,147,483,648 to +2,147,483,647 can be stored in fields of this type. (Storage required = 4 bytes)
6. **Auto numbers**, unique numbers are automatically assigned to fields of this type. Auto number fields are often used to uniquely identify a record within a database. This feature can also be used by a relational field to uniquely identify a related record. (Storage required = 4 bytes)
7. **Small floating point**, numbers in the range -3.4e38 to +3.4e38 with a reasonable degree of accuracy can be stored in fields of this type. (Storage required = 4 bytes)
8. **Large floating point**, numbers in the range -1.78e308 to +1.78e308 with a high degree of accuracy can be stored in fields of this type. (Storage required = 8 bytes)
9. **Currency**, has the same number range as large floating point fields but with different display attributes. (Storage required = 8 bytes)
10. **Date**, date fields use a format that allow dates for approximately the next three hundred thousand years to be stored and processed correctly. Thus PowerBase will not fall foul of any “Millennium bug”. (Storage required = 8 bytes)
11. **Memo**, up to 64K of text can be stored in fields of this type. (Storage required = 4 bytes)
12. **Embedded object**, allows for an arbitrary amount of binary data belonging to other applications but stored within a database. (Storage required = 4 bytes)

13. **Relational**, relational field contains a links to other records in the database. (Storage required = 4 bytes)
14. **Calculated**, calculated fields do not store any data. However, they display the result of a calculation applied to the contents of other fields.

Field attributes

All fields have a set of attributes associated with them. The following attributes are possible, however not all fields can have all attributes.

1. **Allowed to change**. If set, PowerBase allows you to edit the field contents. It is reasonable to have a field where a default value is generated that you do not want to be easily edited. For example, you could set a date field to automatically have a default value of the time the record was created.
2. **Allow zero value**. Certain database uses require that a field must have a non-zero value. In this case if you clear the 'Allow zero value' attribute, PowerBase will not allow you to save a record if the field in question contains the value zero.
3. **Allow blank field**. In certain cases, it is very important to be able to distinguish between the value 0 and nothing. For example, you may have a list of drugs and the number of people cured by each drug. It would be important to know whether 0 people were cured or that the drug was not tried on any people. Setting the 'Allow blank field' attribute allows the field to contain no data. Clearing the attribute requires data to be entered.
4. **Default value from last record edited**. If this attribute is set, then when adding a new record to a table, PowerBase sets the initial value to that belonging to the last record you edited. If you edit a record belonging to a different table or delete the last record you edited, then PowerBase is not able to provide a default value for the field.
5. **Default value expression**. As an alternative to suggesting that PowerBase set the default value of a new record to that last entered against a field, you may define an expression that will be evaluated to provide the default value. You should not reference other fields in such an expression since other fields have no value at the time at which the default value is being calculated. Sensible uses include 'Now' 'Date', 'Time' and fixed quoted strings.
6. **Search when finding**. If set, it allows PowerBase to search this field when it is attempting to match records when 'Finding:' records.

Technical Reference

7. **Search for phone number.** If set it allows PowerBase to search this field when attempting to provide telephone numbers.

Field display formats

Display format options depend on the type of field and describe the formatting applied to the contents of a field. The display format is applied to the field in all views. Apart from being able to change the way in which data is displayed, users should note that the display format has a significant effect on the usage of 'Find:'

The Display format options available are:

Text fields

1. Entered, text is displayed exactly as entered.
2. Lower case, all text is converted to lower case
3. Upper case, all text is converted to upper case
4. Capitalised, the first letter of each word is upper cased, all other letters are lower cased.
5. Proper, the first letter of the first word is upper cased, the rest is lower cased.

Boolean fields

1. '1' or '0'
2. 'Yes' or 'No'
3. 'True' or 'False'
4. 'On' or 'Off'
5. 'Male' or 'Female'

Integer fields

This includes small, medium and large integer field types and auto number field types.

1. Decimal
2. Binary

3. Octal
4. Hexadecimal

Floating point number fields

1. General, PowerBase chooses the best format given the number needed to be displayed.
2. Fixed point
3. Scientific
4. Currency

Currency

Currency fields are always displayed obeying the currently defined internationalisation rules for currency symbol, position and how negative amounts should be displayed. These parameters can be set up in the Control Panel of your handheld computer. PowerBase enables you to override the number of decimal places to be displayed since it is common to have items in a database whose individual value is very small.

Note that the currency symbol in particular is not stored as part of the field definition. If you anticipate using copies of a database on computers that may have different internationalisation rules, you should consider recording the currency type as part of the field name or indeed as a separate field in the table.

Date

- Date and time - e.g. 19th Jan 1997 07:32 am
- Long date - e.g. 19th January 1997
- Medium date - e.g. 19th Jan 1997
- Short date - e.g. 19/1/97
- Long time - e.g. 07:32:21 am
- Medium time - e.g. 07:32 am
- Short time - e.g. 07:32

Users may wish to note that date and time fields obey the currently defined internationalisation rules concerning date displays. These parameters can be set up in the Control Panel of your handheld computer.

Technical Reference

Memo and Embedded fields

Memo fields are displayed as entered. Embedded fields contain data associated with another application hence the display of that data is determined by the other application.

Relational and Calculated fields

The display of both relational and calculated fields is determined by the result an expression. Since this result is either textual or numeric, it is up to the user to choose an appropriate display format for these fields. If required, PowerBase will attempt to convert the result of an evaluation expression to the desired display format. However, there will be cases where this is not possible.

Expressions

PowerBase incorporates a powerful expression evaluator. An expression can be simply a number or a complex formula. You can use expressions to:

- Define the default value for a field.
- Define the value of a calculated field.
- Define the information displayed for a related field.
- Define logical criteria by which to filter a view or define a group.

Expressions can contain the following components:

Numbers. PowerBase uses IEEE 64 bit Floating point values to represent all numbers. Numbers can be entered conventionally or using engineering notation. The range of values supported is approximately from $2.2e-308$ to $1.8e+308$.

Strings. Strings can contain any alphanumeric character. Strings are delimited by the " character thus: "abc".

Field names. An expression can refer to the value of any field in the table in which the expression is being performed. The field is referenced by its name enclosed within the [and] characters. For example, given a field called "Salary" which has the value 25,000, the expression [Salary]*0.10 will evaluate to 2500.

Operators and Functions. PowerBase provides a wide range of operators and functions for use in expressions. These are described in detail in the following sections. When writing expressions, bear in mind that the names of Functions

and System variables are not case sensitive and that PowerBase applies the standard rules regarding operator precedence and the use of parentheses.

Arithmetic operators

The following arithmetic operators are available:

- +Addition. $3+4 = 7$
-Subtraction. $10-15 = -5$
- *Multiplication. $2*3 = 6$
- /Division. $10/5 = 2$
- ^Power. $3^2 = 9$
- ~Negation. $\sim(-1) = +1$
- MODModulus. $11 \text{ MOD } 3 = 2$
- <<.....Bitwise left shift. $1 \ll 8 = 256$
- >>.....Bitwise right shift. $256 \gg 4 = 16$

Relational and logical operators

Expressions that evaluate to a non-zero value are considered TRUE. Zero values are considered FALSE. The logical operators return 1 or 0 to indicate TRUE or FALSE. The following operators and constants are available:

- <.....Less than
- =Equal to
- >.....Greater than
- <=.....Less than or equal to
- <>.....Not equal to
- >=.....Greater than or equal to
- OR.....Logical OR. (a OR b) is TRUE if either a or b is true.
- AND.....Logical AND. (a AND b) is TRUE if both a and b are true.
- XOR.....Logical eXclusive OR. (a XOR b) is TRUE if only one of a and b is true and the other false.
- !.....Logical NOT. (!a) is TRUE if a is false and FALSE if a is true.
- False.....is a system constant with the value 0
- True.....is a system constant with the value 1

Technical Reference

Mathematical functions

The following mathematical functions are available. In all cases below x is a numeric value.

- abs(x).....returns the absolute value of x
- sign(x).....returns the sign of x . -1 if $x < 0$, 0 if $x = 0$ or 1 if $x > 0$
- int(x)returns the integer part of x
- frac(x).....returns the fractional part of x
- sqrt(x).....returns the square root of x
- log(x).....returns the logarithm to the base 10 of x
- ln(x)returns the natural logarithm of x
- exp(x)returns the inverse natural logarithm of x
- fact(x).....returns the factorial value of x (the product of all integers from 1 to x)

Trigonometric functions

The following trigonometric functions and constants are available. In all cases below x is a numeric value.

- cos(x).....returns the cosine of x , where x is in radians.
- sin(x).....returns the sine of x , where x is in radians.
- tan(x).....returns the tangent of x , where x is in radians.
- acos(x).....returns the arc-cosine of x in radians.
- asin(x)returns the arc-sine of x in radians.
- atan(x).....returns the arc-tangent of x in radians
- deg(x)returns the number of degrees represented by x radians.
- rad(x).....returns the number of radians represented by x degrees.
- Pi.....returns a system constant that is the value of pi

String functions

The following string functions are available. In all cases below x , y and z are numeric values and s represents a string of the form "abc".

- +..... Concatenation. "Hello"+"world" = "Hello world"
- chr\$(x)..... returns the string character with decimal ascii code x .
- asc(s) returns the decimal ascii code of the first character in s .
- str\$(x)..... returns a string representation of the value x .

`fix$(x,y,z)` returns up to z characters representing x, to y decimal places

`gen$(x,y)` returns up to y characters representing x

`sci$(x,y,z)` returns up to z characters representing x, to y decimal places

`hex$(x)` returns character representation of x in hexadecimal

`len(s$)` returns the number of characters in s\$.

`mid$(s$,x,y)`..... returns a string of length y starting from the character at position x in s\$ (the 1st character is at position 1).

`left$(s$,x)` returns a string of x characters taken from the start of s\$.

`right$(s$,x)`..... returns a string of x characters taken from the end of s\$.

`rept$(s$,x)` returns a string containing x concatenated copies of s\$.

`upper$(s$)` returns the string s\$ converted to upper case.

`lower$(s$)`..... returns the string s\$ converted to lower case.

`fold$(s$)` returns the string s\$ converted using the system fold table.

`capitalize$(s$)` ... returns the string s\$ with the 1st character converted to upper case.

`match(a$,b$)`..... returns position of the first occurrence of b\$ in a\$ using wildcards

`find(a$,b$)`..... returns position of the 1st occurrence of b\$ in a\$

`eval(s$)`..... returns the result of evaluating the string s\$ as an expression.

`val(s$)` returns a numeric value expressed by the string s\$.

Date and time functions

Date and time are expressed as a "serial" number. This is a floating point number. The date is represented by the integer part of the serial number and is the number of days since 30/12/1899. The time is represented by the fractional part of the serial number and is the number of seconds since 00:00:00 on the given day.

In all cases below:

`Yr`.....is a year number including the century

`Mo`.....is a month number from 1 to 12

`Da`.....is a day number from 1 to 31

`Hr`is an hour number from 0 to 23

`Mi`.....is a minute number from 0 to 59

`Se`is a second number from 0 to 59

Technical Reference

Sn.....is a date/time serial number

The available functions are:

dateserial(yr,mo,da) returns the serial number of the given date.
timeserial(hr,mi,se)..... returns the serial number of the given time.
week(yr,mo,da) returns the current week number in the year.
year(sn)..... returns the year number represented in sn.
month(sn) returns the month number represented in sn.
day(sn)..... returns the day number represented in sn.
hour(sn)..... returns the hour number represented in sn.
minute(sn) returns the minute number represented in sn.
second(sn)..... returns the second number represented in sn.

PowerBase also provides a range of system variables that derive from the current date and time. The date and time system variables are:

Time.....returns the current time as a serial number.
Date.....returns the current date as a serial number.
Now.....returns the current time and date as a serial number.
Year.....returns the current year number.
Month.....returns the current month number.
Day.....returns the current day number.
Hourreturns the current hour number.
Minute.....returns the current minute number.
Second.....returns the current second number.