ReportPro

Report Designer Guide

Overview	Topics	
Design Interface	Bands	Configuring Bands
Menu System	Configuring a Report	Grouping
Creating a Report	Configuring Bands	Object Inspector
Report Wizards	Configuring Local Tables	Opening a Report
	Configuring Report Objects	Print
	Configuring Sections	Print Preview
	Configuring SQL Sources	ReportPro Configuration
	Configuring the Printer	Saving a Report
	Drawing Tools	Variables
	Expression Builder	
Light Lib	Field Wizard	Appendices

Copyright

DataPro, Inc.

PO Box 6663

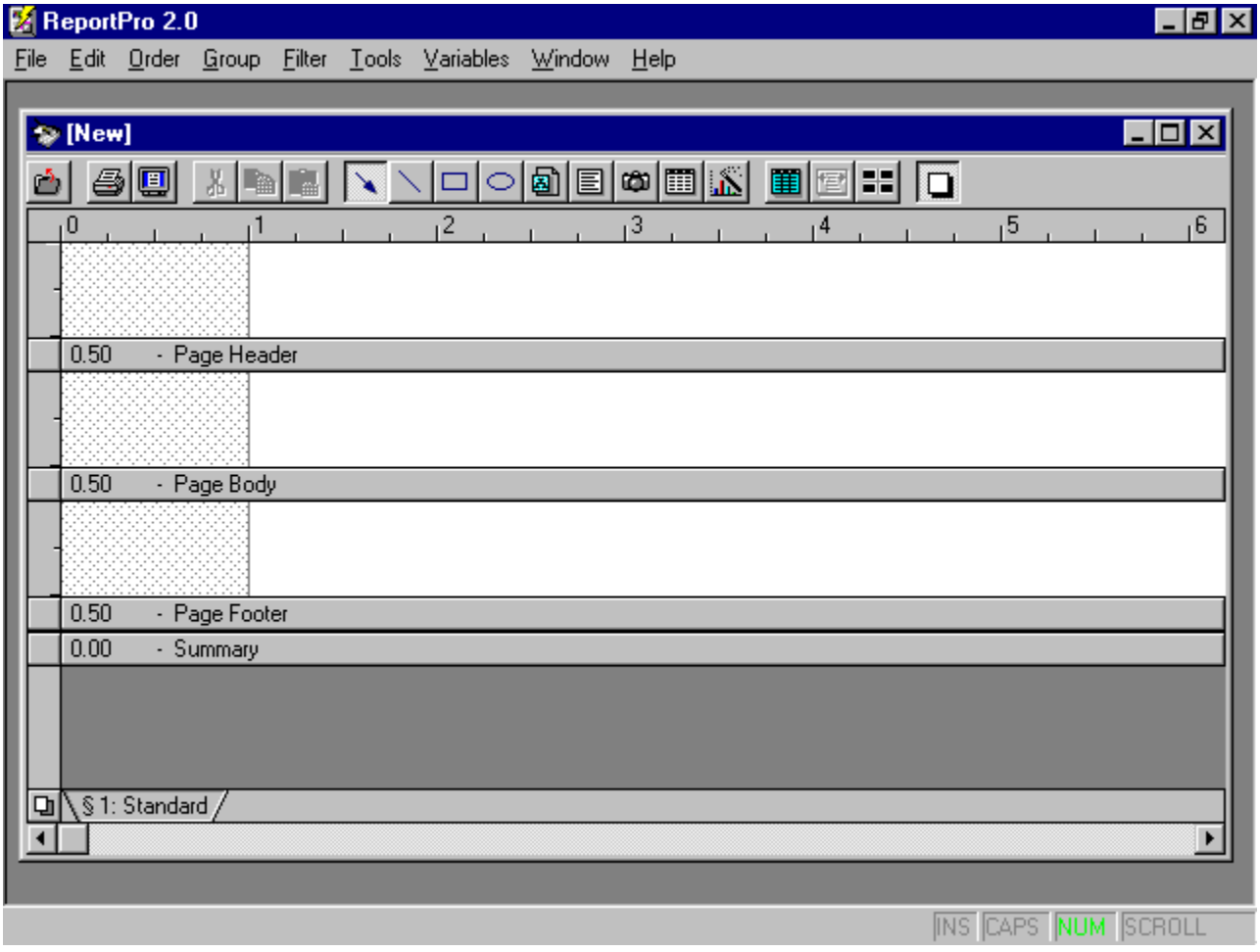
Warner Robins, GA 31095-6663

United States of America

ReportPro's Design Interface

This chapter provides a brief overview of ReportPro's design interface. It's purpose is to familiarize you with the key components that are used to create a report.

The figure below shows ReportPro's design interface as it appears when creating a new report. The large outer window is ReportPro's shell window. The inner window is ReportPro's work window. You can open as many work windows as you have available memory. The advantage of having multiple work windows open is that you can copy objects from one work window to another thus simplifying the report creation process.



At the top of the shell window is the title bar which contains the text "ReportPro 2". Directly below the title bar is ReportPro's main menu. The menu is used to access the various options available in ReportPro.

At the top of the work window is another title bar. This title bar shows the file name of the report that is

opened in the work window. When you create a new report and have not saved it yet, the text "[New]" is displayed in the title bar.

Underneath the work window title bar is ReportPro's toolbar. The toolbar allows quick access to frequently used ReportPro menu options. The toolbar contains the following options (from left to right): Close, Print, Print Preview, Cut, Copy, Paste, Arrow Tool, Line Tool, Rectangle Tool, Ellipse Tool, Field Tool, Text Tool, Picture Tool, Embedded Section Tool, Graph Tool, Grid Display Toggle, Grid Snap Toggle, Center Object Command and Proof Mode Toggle.

Below the toolbar is the ruler. The ruler indicates where on the report surface you are drawing or editing. The ruler can be displayed in either inches or centimeters and corresponds to the location that objects will print from the left edge of the paper.

The white area below the ruler is the report surface. On this surface you will place the objects which constitute the report. The light gray area on the left side of the report surface represents the left margin of the report. Margin sizes are configurable and you can always place objects in the margin area if desired.

The report surface is divided into bands. When you initially create a report four bands are displayed. The Page Header allows you to specify objects that will be printed on the top of every page. The Page Body is used to print information that constitute the body of the report. The Page Footer allows you to specify objects that are printed on the bottom of every page. The Summary band allows you to print summary information at the end of the report.

On the left side of the work window you will find the band ruler. The band ruler indicates the size of the band and identifies the vertical placement of objects in the band. It also contains controls that allow you to change the height of each band.

At the bottom of the band ruler is the Background Toggle Button. This button places the ReportPro work window in the background edit mode. When in the background mode, the bands shown in figure 1 are hidden and the Background Band is displayed.

To the right of the Background Toggle Button is the Section Tab Bar. The tab bar displays one tab for each section defined in the report. By default a report has one section, but as you add sections to your report, additional tabs will appear on the tab bar. You select a section by clicking its tab. When you do so the current section is hidden and the selected section is shown.

To the right and bottom of the work window are scroll bars. The scroll bars allow you to scroll the work window to display portions of the work area which are hidden. This is useful since the report surface is generally larger than the work window.

At the bottom of the main ReportPro window is the status bar. The status bar is used to display help information about menu and toolbar options.

ReportPro's Menu System

ReportPro's menu system provides access the many features of ReportPro. There are two types of menus. The menu bar which appears on ReportPro's shell window and pop-up menus that are accessibly by clicking the right mouse button over a control or object.

You'll encounter two different menus on ReportPro's menu bar. The work window menu is covered in this chapter. The print preview menu is discussed in a separate chapter.

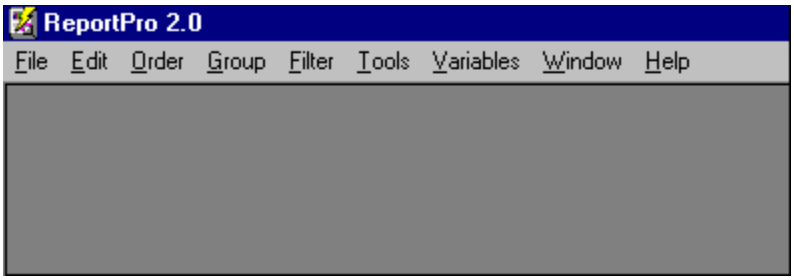
Each menu option on the menu bar has a accelerator which is identified by an underlined character. The accelerator allows you to use the keyboard to quickly access the menu option. To activate an accelerator, hold down the **Alt** key and at the same time press the underlined character. For example, to select the *File* menu, press **Alt** and at the same time press **F**. You will notice that when you select the *File* menu another menu is displayed. In many cases these additional menu options also have accelerator keys. The accelerator for these *pull-down menus* are activated with the **Ctrl** key. For example, to select the *File, Open* menu option, hold down the **Ctrl** key and press **O**.

ReportPro's Work Window Menu

ReportPro's Pop-up Menus

ReportPro's Work Window Menu

The second menu you'll encounter on ReportPro's menu bar is displayed after you have opened or created a report. This is the work window menu. It contains the options necessary to create spectacular reports.



Each of the menu options are identified below.

File

The work window *File* menu is very similar to the shell window *File* menu. It contains the options necessary to create, open and save a report, but it also contains options to configure and print the report.

Menu Option	Description
New, Standard Report	Creates a standard report.
New, Label Report	Creates a columnar or label report.
New, Cross-Tab Report	Creates a cross tab report.
New, Standard Report Wizard	Launches the standard report wizard.
New, Label Wizard	Launches the label report wizard.
New, Cross-Tab Wizard	Launches the cross-tab report wizard.
Open	Opens an existing ReportPro report.
Save	Saves the opened report.
Save As	Saves the open report with a different name.
Close	Closes the report.
Setup, Report	Displays a dialog to allow you to specify options specific to the report.
Setup, Section	Add, delete and configure sections, tables and SQL queries.
Setup, Printer	Displays the printer's setup dialog where you configure the printer.
Setup, ReportPro	Displays the ReportPro Setup dialog where you can specify options that affect how ReportPro operates.
Print	Sends the report to the printer.
Print Preview	Sends the report to the print preview.

1 - 5 (Last used reports)	Allows you to quickly access the most recently used reports.
Exit	Terminates ReportPro.

Edit

ReportPro provides a wealth of editing features to aid in the development of reports. This section focuses on the *Edit* menu option and the features available in the *Edit* pull-down menu.

You will notice that some of these options are not always available. This is because the availability of the editing features depends on the status of objects in the work window. For example, the *Copy* option is not available if there are no selected objects in the work window and the *Paste* command is not available if there are no objects in the clipboard.

The following table identifies each editing command.

Menu Option	**Description**
Cut	Deletes the selected report objects and places them in ReportPro's clipboard. The Cut objects can later be Pasted into one of ReportPro's work windows.
Copy	Copies the selected report objects (without deleting them) into ReportPro's clipboard. These Copied objects can later be Pasted into one of ReportPro's work windows
Paste	Pastes objects that were either Cut or Copied into the current work window.
Delete	Deletes the selected objects without placing them in ReportPro's clipboard.
Bring to Front	Brings the selected objects to the front of the Z-order. The Z-order determines which objects lie on top of other objects. An object "in front" will lie on top of all other objects in the same band.
Send to Back	Sends the selected objects to the back of the Z-order. A "sent to back" object will lie behind all other objects in the same band.
Center Objects	Centers the selected objects horizontally in the report. Note that when more than one object is selected the rectangle that bounds all of the selected objects is centered and the position of each object is adjusted relative to the movement of the bounding rectangle.
Select All	Selects all objects in the current work window.
Set Font	Sets the font for all selected objects. For example, if you wish to change the font of all field and text objects on a report, you would click the Select All menu option and then click the Set Font option.

Order

This menu option does not have a sub menu. When selected, it displays the Sort Order dialog for the currently displayed section. This menu option is a short cut for accessing the Section *Sort Order* pop-up menu option in the Setup Sections dialog.

Group

This menu option does not have a sub menu. When selected it displays the Grouping dialog where you specify how data should be grouped for the current section.

Filter

This menu option does not have a sub menu. When selected it displays the Expression Builder to allow you to specify a filter for the current section.

This menu option is a short cut for accessing the Section *Filter* pop-up menu option in the Setup Sections dialog.

Tools

This menu option allows you to select the tools necessary to create a report. Each of the key menu options is identified in the table below.

Menu Option	Description
Drawing Tools	Displays another menu listing ReportPro's drawing tools.
Grid	Toggles the display of the design grid. The resolution of the design grid is set using the Report Setup dialog window and works in conjunction with the Snap To Grid feature (having the design grid displayed does not necessarily mean that your drawing will snap to it).
Snap	Toggles the snap to grid feature of the design grid. When ON all drawing will snap to the resolution of the design grid.
Proof Mode	Toggles the proof mode. When a report is in the proof mode, object shadows are displayed in the work window otherwise they are hidden.
Field Wizard	Displays the Field Wizard.
Object Inspector	Displays the Object Inspector.

Variables

This menu option does not have a sub menu. When selected it displays the Variables dialog where you can define variables.

Windows

This menu option is provided to allow you to easily arrange and select open reports.

Menu Option	Description
Cascade	Positions the open reports in a cascaded arrangement.
Tile	Positions the open reports in a tiled arrangement.
1 - N (open reports)	The lower portion of the menu displays a list of the open reports. You can give focus to a report by selecting it from this list.

Help

The Help menu allows you to access ReportPro's help system. The help system provides detailed information about ReportPro.

Menu Option	Function
Index	Displays the ReportPro help file index
Using Help	Displays help on how to use the windows help system
About	Displays the ReportPro About dialog.

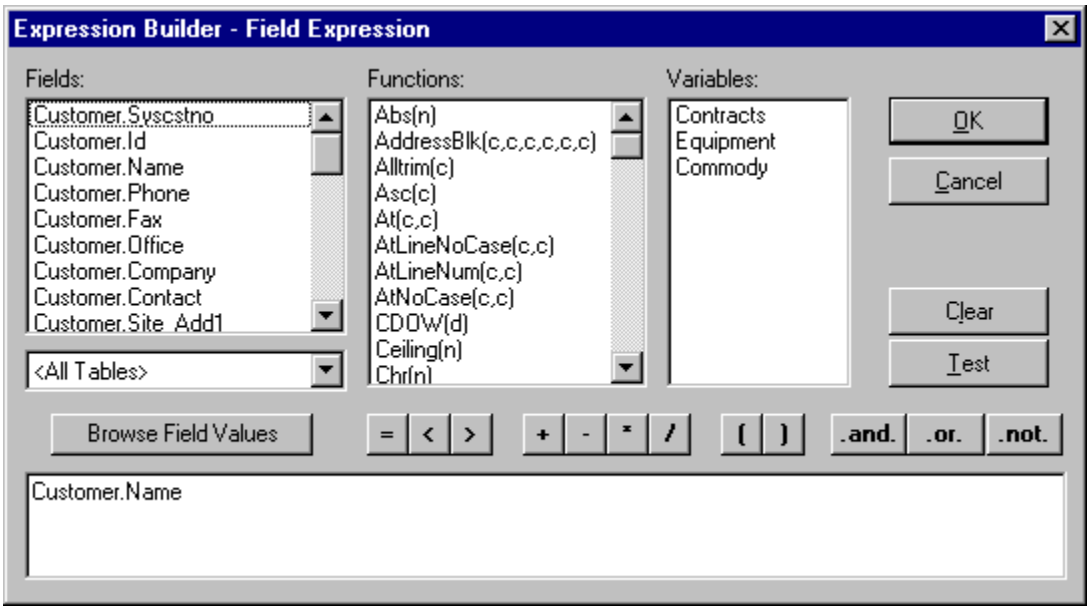
ReportPro's Pop-up Menus

ReportPro makes extensive use of pop-up menus. Most objects you see in the work window have pop-up menus. These menus are activated by clicking the right mouse button over the desired object. The menu you see below is the pop-up menu for ReportPro's Text object.

Pop-up menus are specific to the object or control they belong to and are discussed in detail in the chapter where they apply.

ReportPro's Expression Builder

The Expression Builder is an important part of ReportPro's design interface and you will encounter it frequently as you create reports. ReportPro uses expressions extensively for a variety of purposes such as retrieving information from a data source and making decisions during the printing process. Think of an expression as a series of commands that instructs ReportPro how to perform a task.



As the name implies, the Expression Builder is where you build expressions. The Expression Builder contains controls to simplify the process of building an expression. Each control is identified in the following table.

Control	Description
Fields List Box	Lists the database fields available for building expressions.
Table Select	Allows you to limit the fields which are displayed in the Field List Box. Select "<All Tables>" to show all the fields or select an individual table to see only fields from that table. This option does not make other fields unavailable for use; but simply limits the fields displayed in the Fields list box.
Functions List Box	Lists the functions available to use in building expressions. See Appendix B for a complete function listing.
Variables List Box	Lists the User Defined Variables which are available for use in the expression builder.
Expression	This is the expression being built. You can directly edit the expression you are building using this edit box.
OK Button	Accepts the expression and closes the Expression Builder.
Cancel Button	Closes the Expression Builder without accepting the expression.
Clear Button	Clears all text in the expression edit control.

Test Button	Tests the validity and syntax of the expression being built without having to exit the expression builder.
Browse Field Values Button	Presents a dialog that shows sample data for the field highlighted in the Fields List box.
Short Cut Buttons	The remaining buttons on the Expression Builder are short cut buttons that enter text in the expression edit. For example if you press the *.and.* button, the text ".and." is added to the expression edit.

Building Expressions

Building Expressions

One of the more powerful and flexible features available in ReportPro is the ability to use expressions to extract and manipulate information. If not for expressions, it would be impossible to add two numbers and display the result on a report.

Expressions are to ReportPro what sentences are to the English language. They convey a message or command to ReportPro. An expression can consist of several different components including constants, variables, functions and operators.

Data Types

Data can be thought of as the subject of an expression. The intent of most expressions is to retrieve or manipulate some type of data. To use expressions effectively, you must understand the basic types of data supported by ReportPro. The four basic data types are:

- **Character** - Defined as zero or more alpha-numeric characters. Character data types are used to express names, addresses, phone numbers, etc. In an expression, a character data type is usually stated by enclosing a sequence of characters within a pair of quotes (e.g., **"John" or "(324) 342-9999"**. To express a character data type that does not contain any characters, use double quotes with no intervening characters (e.g., **""**).

- **Numeric** - Defined as a data item on which you can perform mathematical operations (e.g., addition, multiplication and division). A numeric data type can contain the digits 0-9, a period to represent a decimal position and the plus or minus sign to indicate the sign of the number. In an expression a numeric data type is stated as **1230.982** or **-9355** (quotes are not allowed).

- **Date** - Defined as a data item which expresses a calendar date. You can manipulate dates in several ways, such as finding the number of days between two dates. A date cannot be represented directly in an expression; instead you must use a function to convert a character data type into a date. This is usually done by using the CTOD function. For example, you can represent 12/12/93 as a date by adding the sequence **CTOD("12/12/1993")** into the expression.

- **Logical** - Defined as a data item which can only contain two values. A logical can only be true (.t.) or false (.f.). In an expression, a logical is represented by **.t.** or **.f.**.

Constants

A constant is data which cannot change. A constant must be one of the pre-defined data types identified above. Constants are used in conjunction with variables, functions and operators to manipulate data. The sequence **"John"** is a character constant. **134.88** is a numeric constant.

Variables

Variables are temporary storage areas which hold data. The data being stored by the variable must be one of the pre-defined data types discussed earlier. Unlike a constant, the data held by a variable can be modified. Variables are referenced by the name which is assigned to them in the Variables dialog.

For example, suppose we define a variable named Total. We can assign data to the variable with the expression **Total := 0**. The variable Total now holds a numeric data type with a value of 0. If we change the expression to **Total := "0"**, the variable would hold a character value.

Database Fields

A special type of variable is the database field. A database field contains a value and is referenced by its name just like a variable, but the value of a database field cannot be modified.

A database field name consists of two or more parts. All field names consist of a table name separated by a period (".") followed by a field name. Fields from SQL queries can also require an owner name.

The value of a database field depends on which record in the database is current. Record positioning is controlled automatically by ReportPro.

Functions

A function can be considered the verb of the expression. It is the part that performs some action. Functions consist of several parts. The first part, the function name, describes what the function does. For example, the function LTrim() (short for Left Trim), tells us that it removes the spaces from the left side of a character data type.

The second part of a function is its arguments. An argument simply provides us with a way to send information to the function. Our LTrim() function has no way of knowing what we want to remove the spaces from unless we explicitly tell it. So we need to add a argument to our function. The proper use of LTrim is to express it as **LTrim(" John")**. Now LTrim() knows what we want to remove the spaces from.

The last part of a function is its return value. The return value is how the function reports the result of its work. The return value of a function is automatically substituted into an expression after the function has completed its work. In our LTrim(" John") example, the result of the function's work is "John". The value "John" is automatically returned to our expression. We can assign the result of the function to a variable using the expression **Name := LTrim("John ")**. The variable, Name, now contains "John".

Operators

Along with functions, constants, and variables, operators are also a basic building block of expressions. Operators are similar to functions in that they perform some operation and return a value. Operators are

expressed in a different format than functions. For example the operator **:=**, used in the examples above, is the assignment operator. It is used to assign a value to a variable. Another popular operator is the addition operator **(+)**. The addition operator can be used to add two numbers together **(100 + 50)**.

Examples

The following are examples of expressions which you can use in the Expression Builder dialog window.

- **Customer.Name** - This is the simplest of all expressions. It is a reference to a database field.
- **Upper(Customer.Name)** - This expression is slightly more complex. It accesses the field Name from the Customer table and converts all characters to upper case.

Creating a Report

There are two ways to create a report. You can create reports manually or you can use one of ReportPro's powerful Report Wizards. Using the Report Wizards is the simplest and quickest way to great looking reports, and Report Wizards are covered later in this guide. This chapter explains how to create a report manually since it provides a good way to learn ReportPro.

There are two basic steps you must perform to create a report:

- First you must define a *data source*. This process tells ReportPro where to get the information that will be included on the report. It also allows ReportPro to retrieve information about the data that is available from the data source. This information is used by our Field Wizard and to perform validation as you create the report.
- The second step is to place *report objects* in the bands of the work window. Report objects are what actually generate the printed output.

Of course, this is a very simplistic view of how to create a report. In practice you will want to specify filters, sort orders, and customize the report objects perhaps to show only a subset of the data presented in a useful sequence. For more complex reports you'll likely specify selections or complex relations between tables in your data source.

You create a report by choosing *File, New* from ReportPro's main menu. Another menu will appear that allows you to select the type of report you wish to create or which Report Wizard you wish to use. The first three menu options represent the basic types of reports supported by ReportPro.

The ***Standard*** report is the most common and covers reports that are not columnar in nature and do not generate a Cross-Tab output. Though this may sound complicated it really isn't. If you are creating a listing, or form based report, you are creating a ***Standard*** report.

Label reports are columnar in nature. Mailing address labels are a good example of this type of report. You can also use Label reports to create a listing style report that has multiple columns.

Cross-Tab reports are those that perform a summary operation on a database and report the result set in a grid type fashion.

It is important to note that you can easily change between two different types of reports if you choose to do so later. Your selection at this point is only required to allow ReportPro to prompt you for enough information to initialize an empty report template.

Creating a Standard Report

Creating a Label Report

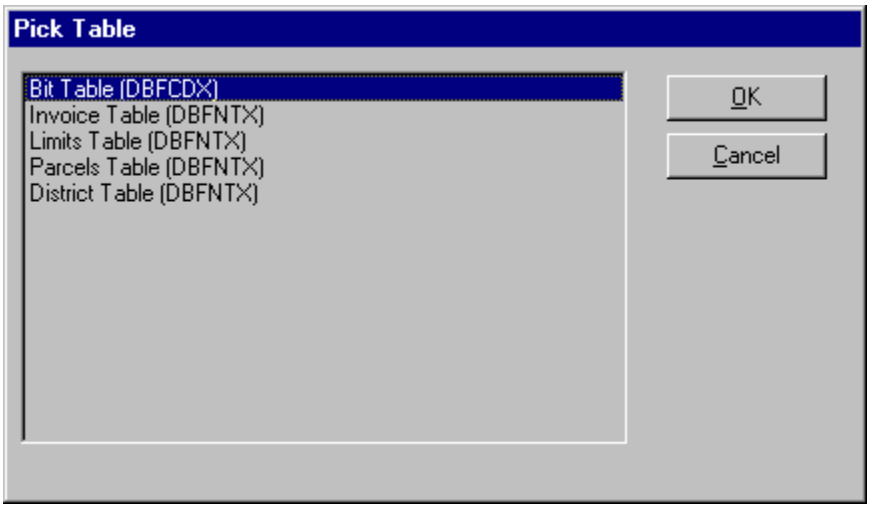
Creating a Cross-Tab Report

Creating a Standard Report

This section covers the steps required to create a standard report. The other report types are covered in subsequent sections.

Defining the Data Source

After you choose the *File, New, Standard Report* menu option, a ReportPro work window will appear along with a dialog prompting you for a primary table. Select the desired table and click the *OK* button.



After you have successfully opened the primary table, ReportPro will enter an idle state where it is waiting for you build the report.

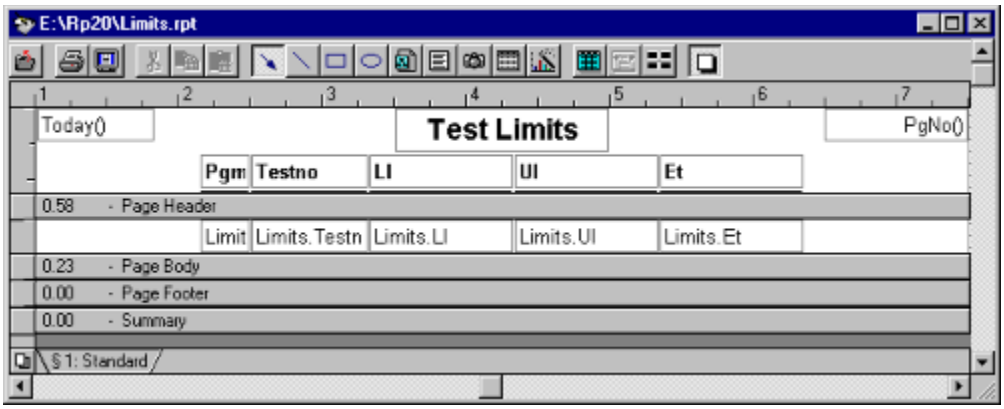
Adding Report Objects

The next step is to place report objects on the work window. Normally report objects are *drawn* on the work window. You start the process by selecting the desired drawing tool from the *Tools, Drawing Tools* menu or the toolbar. After you have selected the appropriate drawing tool, left click the mouse in the work window, then move the mouse while continuing to hold the left mouse button down. You'll notice that the object is drawn as you move the mouse.

When you release the left mouse button, the object will be created. In some cases, you will be prompted for additional information. For example, if you are drawing a Field object, you will need to specify the Field object's expression once you have finished drawing the object. For a Text object, you will need to type in the text.

The objects you draw in the work window depend on the type and format of the report you are creating. If you are creating a simple listing report, you generally draw Text objects in the Page Header band to represent the report title, and the column headings. In the Page Body band you place Field objects that extract information from the data source.

The figure below shows what a simple listing type report might look like.



For a form based report, you would might reduce the size of the Page Header and Page Footer bands to zero and increase the Page Body band size to equal the size of the paper you are printing on. You would then proceed to mix Text and Field objects in the Page Body band to generate the proper form design.

As you draw the report objects, you may find ReportPro's Grid and Snap-to-Grid handy. Both of these features are available from the *Tools* menu.

Customizing Your Report

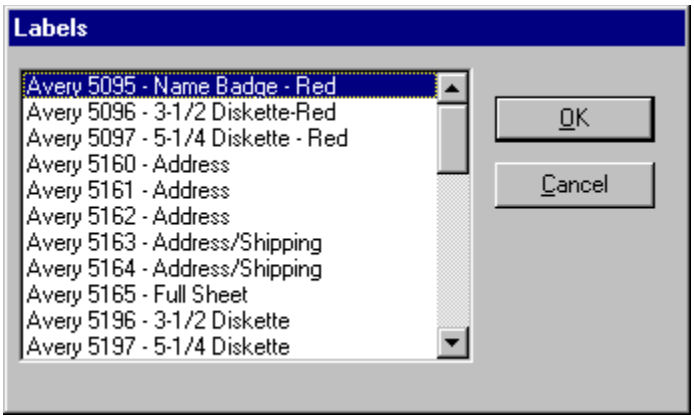
After you have added the report objects you'll probably want to customize the report. ReportPro supports many options to allow you to tailor the report to meet your needs. One thing you'll probably want to do is modify the data source by defining a sort order or filter. These options are available on ReportPro's menu bar.

Next you may wish to customize the report objects by adding shadows or borders. The object attributes are available on pop-up menus. You access an object's pop-up menu by right clicking the mouse over the object.

The report creation process is iterative, and you don't have to create the report in the order discussed here. ReportPro's design interface is flexible enough to allow you to create the report in almost any sequence.

Creating a Label Report

Creating a *Label* report is very similar to creating a Standard report except you are initially prompted to choose a label type. The dialog that appears lists many of the more popular labels types. Select the desired label type from the list, or if you are using a custom label size, choose *User Defined Format* which is the last selection in the list.



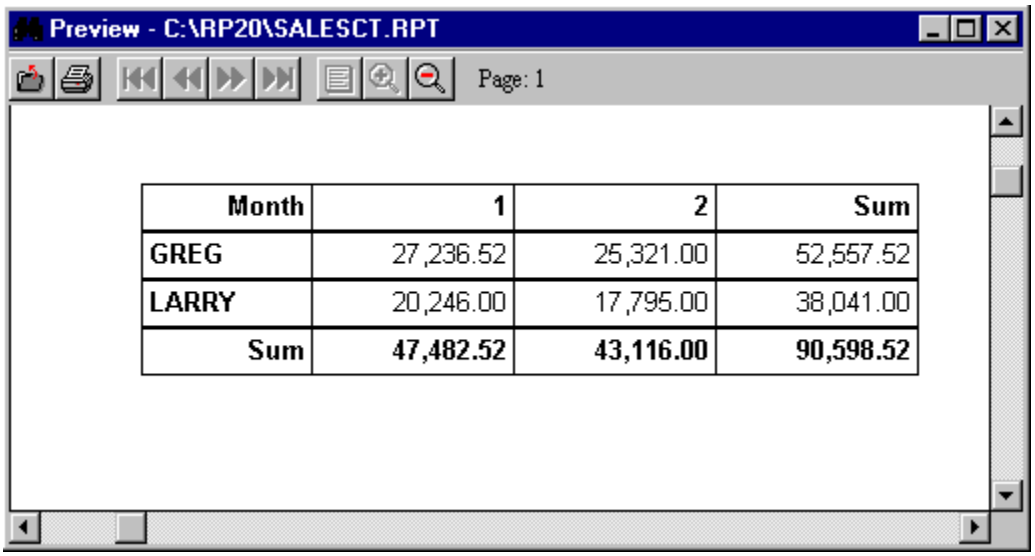
Based on your selection, ReportPro initializes the work window to represent a single label. If you choose *User Defined Format*, ReportPro will initialize the work window to 1" x 6.5" inches. You must specify the custom size in the Column/Label Setup dialog which is available by right clicking on the Section Tab at to bottom of the work window.

After you have selected a label, the report initialization process is the same as encountered when defining a standard report.

Creating a Cross-Tab Report

A Cross-Tab report is one that summarizes information from a database and displays the result in a spreadsheet like structure. Reports of this type are useful for summarizing large amounts of information and identifying trends in data.

A classic example of a Cross-Tab report is to sum monthly sales figures by salesman. Assume that you have a table that contains the following fields: salesman, sale date and sale amount. Using Cross-Tabs, you could easily generate the following report which shows each salesman's performance.

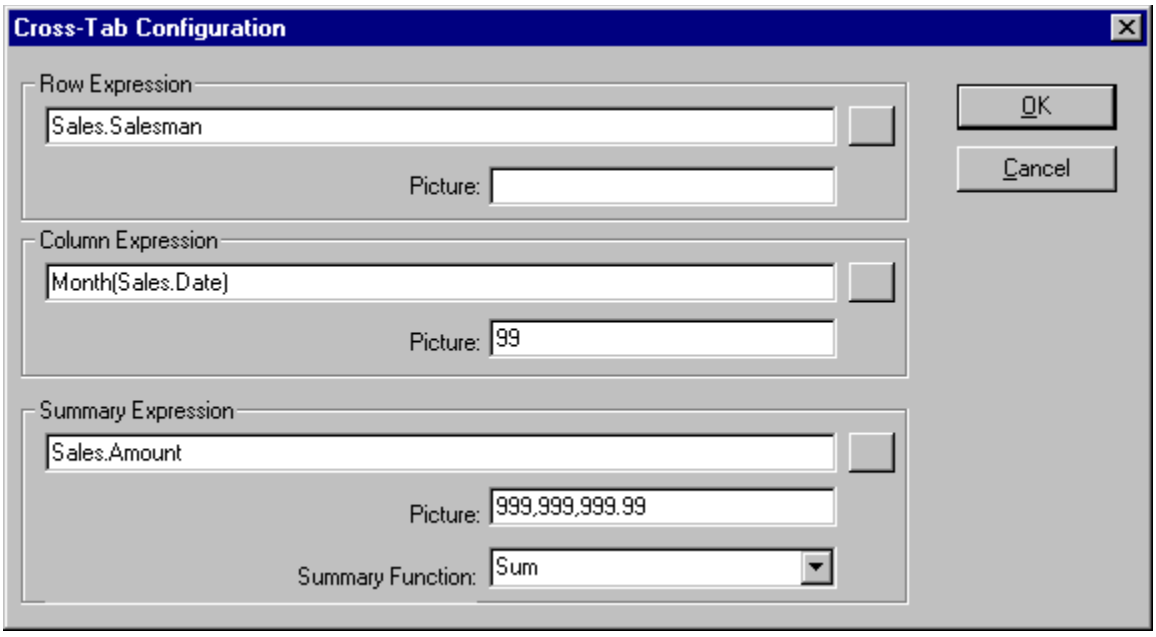


Month	1	2	Sum
GREG	27,236.52	25,321.00	52,557.52
LARRY	20,246.00	17,795.00	38,041.00
Sum	**47,482.52**	**43,116.00**	**90,598.52**

To create a Cross-Tab report, choose the *File, New, Cross-Tab Report* option from ReportPro's main menu. You can also create a Cross-Tab report using the Cross-Tab Report Wizard discussed in the next chapter.

The initial process of creating a Cross-Tab report is similar to creating a Standard report in that you must specify a data source. This process is covered in detail in the previous section.

After you define a data source, the Cross-Tab Configuration dialog will be displayed.



To define a Cross-Tab, you must specify a row, a column and a summary expression. The expressions can be entered directly or via the Expression Builder. To access the Expression Builder, click the square button to the right of the desired edit.

The *Picture* edit, which is available for each expression, allows you to specify an optional picture clause which is used to format the expression's value. See Appendix A for more information on picture clauses.

The *Row Expression* governs the values that are displayed in the Row Label (left most) column of the Cross-Tab. The row expression is evaluated for each record in the data source. Each unique value returned by the expression will get a row in the Cross-Tab.

The *Column Expression* determines the values that are displayed in the Column Label (top) row. The column expression is also evaluated for each record in the data source. Each unique value returned by the expression will get a column in the Cross-Tab.

Finally, the Row and Column values are cross-referenced and summarized using the *Summary Expression*. The Summary Function list box allows you to specify which mathematical operation is performed on the summary expression. Five summary functions are available, but Average and Sum are only available for summary expressions that return a number.

Function	**Description**
Average	Determines the average of all cross-referenced values
Count	Counts all cross-referenced values
Maximum	Determines the largest cross-referenced value
Minimum	Determines the smallest cross-referenced value

Sum	Sums all cross-referenced values

After you have entered all three expressions and selected a summary function, click the *OK* button. ReportPro will create the report and automatically layout the Cross-Tab Field objects. You can change the size and location of the Cross-Tab Field objects but they cannot be moved out of their original band.

Report Wizards

Report Wizards allow you to quickly generate reports. This is the recommended approach for creating a report since it allows you to create the basic elements of a report quickly.

The Report Wizards guide you through a series of dialogs in which you specify reporting options. After you specify the desired options, the wizard generates the report for you in a work window. Once the report is created you can customize it as desired.

You start a Report Wizard by selecting the *File, New* menu option and choosing the type of wizard you wish to use. ReportPro has a separate Report Wizard for Standard, Label and Cross-Tab reports. Each wizard is discussed below.

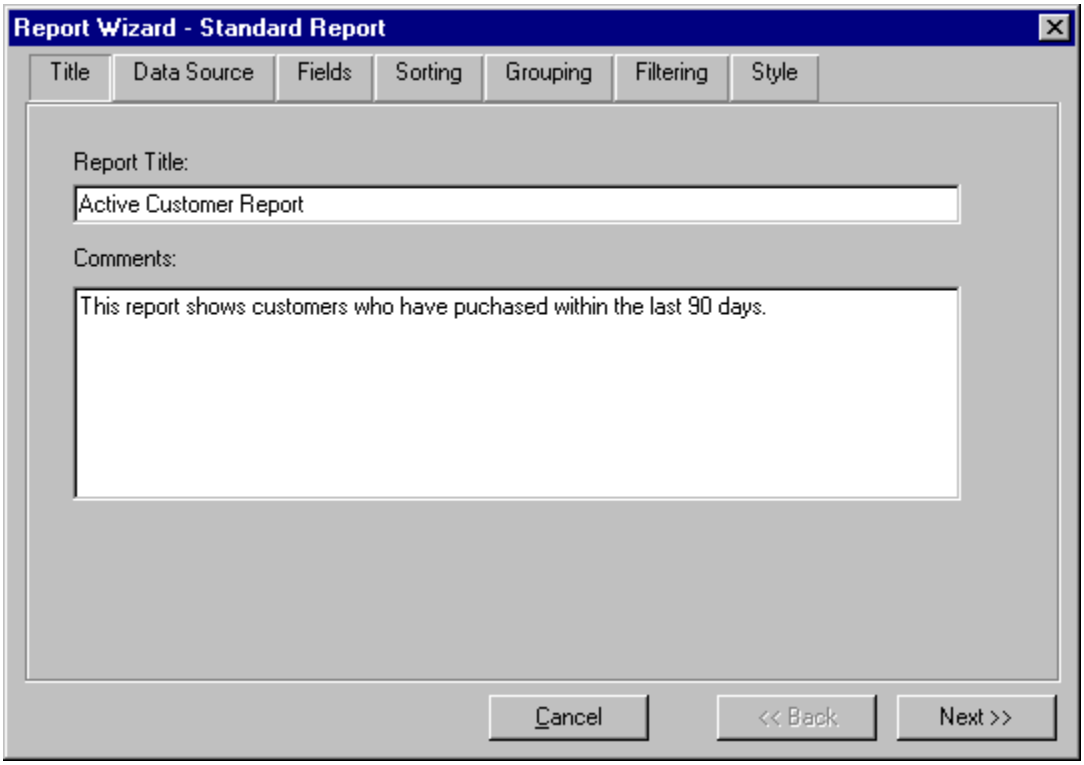
Standard Report Wizard

Label Report Wizard

Cross-Tab Report Wizard

Standard Report Wizard

The *Standard Report Wizard* guides you through the creation of a standard report. As discussed earlier, a standard report is one that does not print labels or have a Cross-Tab on the primary section.



Report Wizards are simply tabbed dialogs that allow you to specify the different attributes of the report. The tabs across the top of the dialog represent the different attributes of the report that you can specify. You select a tab by clicking the tab with the left mouse button or by pressing the *Next>>* and *<<Back* buttons until the desired tab is selected.

In general, you should specify report options from the left most tab working toward the right most tab.

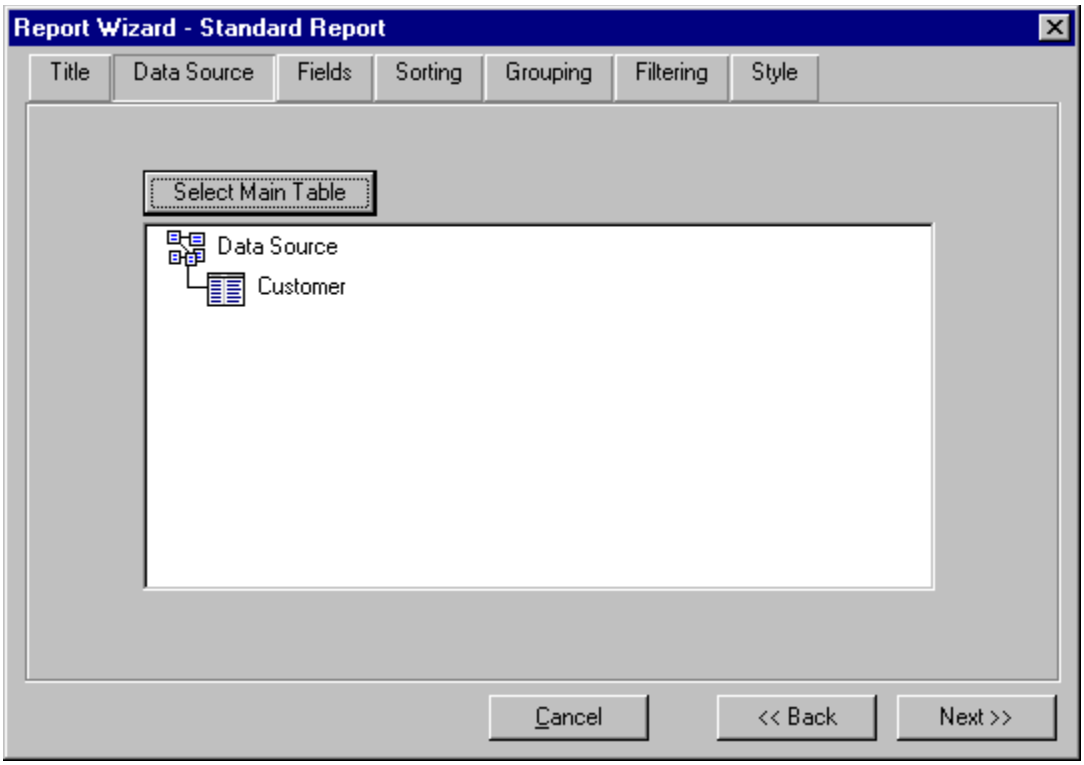
Title Tab

This tab allows you to specify a report title and store comments about the report you are creating. The title appears in the caption of the report work window making it easier to identify the report you are working on. The title is for readability only and is not used by ReportPro.

The *Comments* edit is where you can enter a brief description of the report. This is useful for recording details about how the report was implemented or the intent of the report. Again, this information is not used by ReportPro and is provided for your convenience.

Data Source Tab

This tab is where you specify what data will be available to the report.

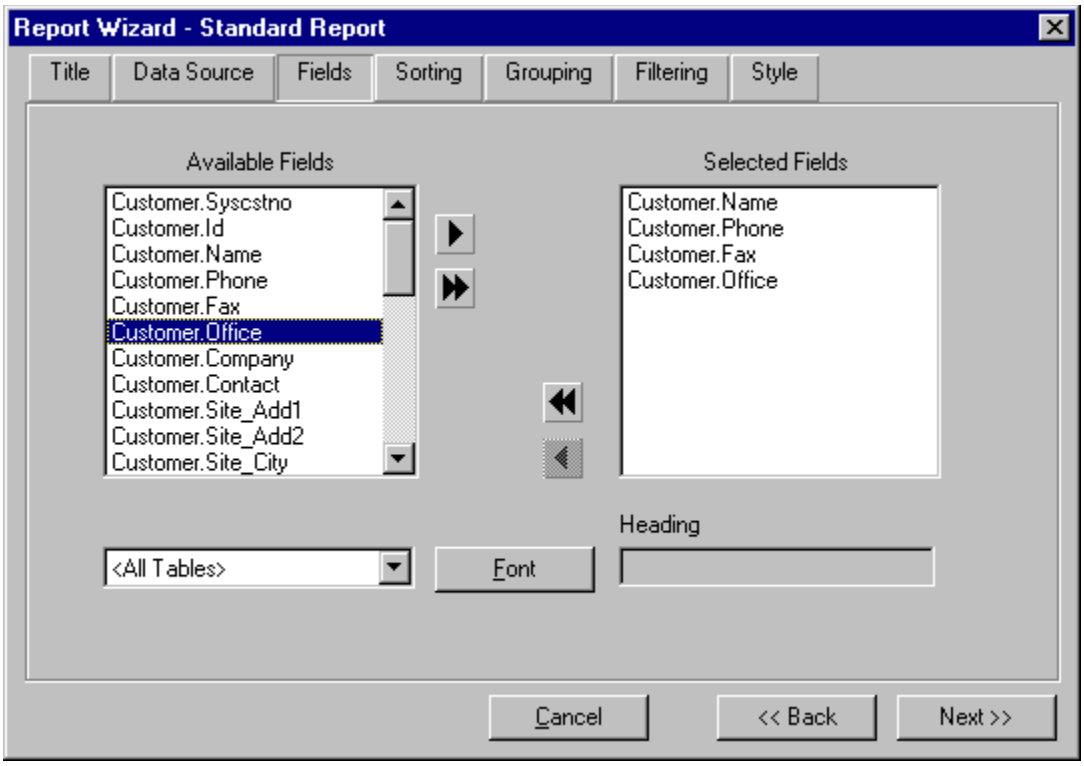


When you first select the Data Source tab, the list box in the middle of the dialog is empty. To specify the primary table or query, click the *Select Main Table* button. ReportPro will present a series of dialogs prompting you to specify a database driver and primary table. This process is the same as you encounter when manually creating a report and is covered in detail in the chapter entitled "Creating a Report".

After you have identified a primary table or query, the list box will be updated to reflect your selection. You can add tables and queries to the data source via pop-up menus that are activated by clicking the right mouse button on objects in the list box. Refer to the chapters entitled "Configuring Local Tables" and "Configuring SQL Queries" for a detailed description of each menu option.

Fields Tab

The next step is to specify the fields you wish to include on the report.



On the *Fields* tab, the *Available Fields* list box displays the fields that are available from the data source you specified on the *Data Source* tab. You specify which fields to include on the report, by adding them into the *Selected Fields* list box.

There are several ways to add fields into the *Selected Fields* list box. The single right arrow **>** moves the highlighted field from the *Available Fields* list box into the *Selected Fields* list box. The double right arrow **>>** copies all fields from the *Available Fields* list box into the *Selected Fields* list box.

If you mistakenly add a field into the *Selected Fields* list box, you can remove it by highlighting it and pressing the single left arrow **<**. You can remove all fields by clicking the double left arrows **<<**. You can also move fields between list boxes by double-clicking them with the left mouse button.

The order of the fields in the *Selected Fields* list box can be changed by clicking the left mouse button over the desired field and while holding the mouse button down moving the field to the preferred position in the list box.

The *Tables* list box allows you to limit the fields displayed in the *Available Fields* list box to those of a single table. This feature is very useful when you are creating a report that uses more than one table. To specify a table filter, click on the down arrow on the list box and choose the desired table. If you wish to see the fields for all tables select the *<All Tables>* entry.

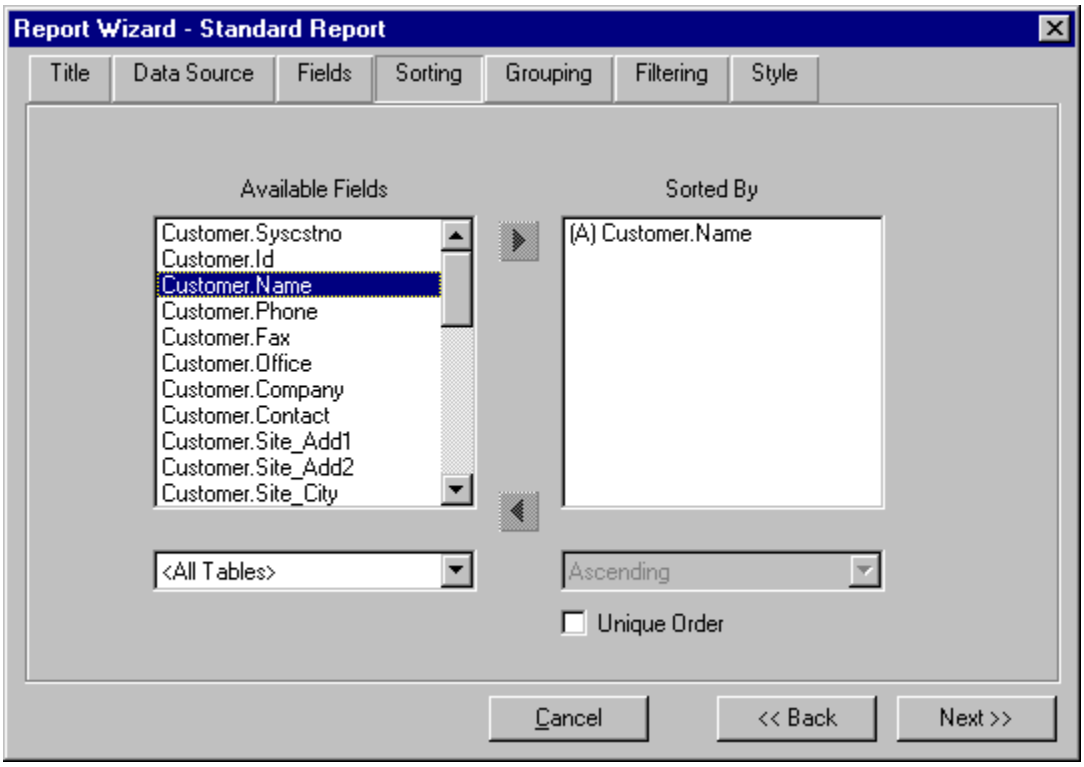
Pressing the *Font* button displays a dialog to allow you to select a font for all field and text objects that will be created for the report. If you wish to specify a font for a single field, you'll have to do that later in the work window.

The *Heading* edit allows you to specify a heading or title for each field. Headings are used in listing and form based reports to identify the field. If you choose to create a listing style report, the heading will be placed in the Page Header band directly above the field. If you choose a form style report the heading will appear to the left of the field.

Headings default to the field name but you may wish to specify a more readable name. To specify a heading, click the desired field in the *Selected Fields* list box with the left mouse button and type in the new heading in the *Heading* edit.

Sorting Tab

The Sorting tab allows you to specify how data from the data source will be sorted. The Sorting tab works very much like the Fields tab. You select fields from the *Available Fields* list box and copy them to the *Sorted By* list box.



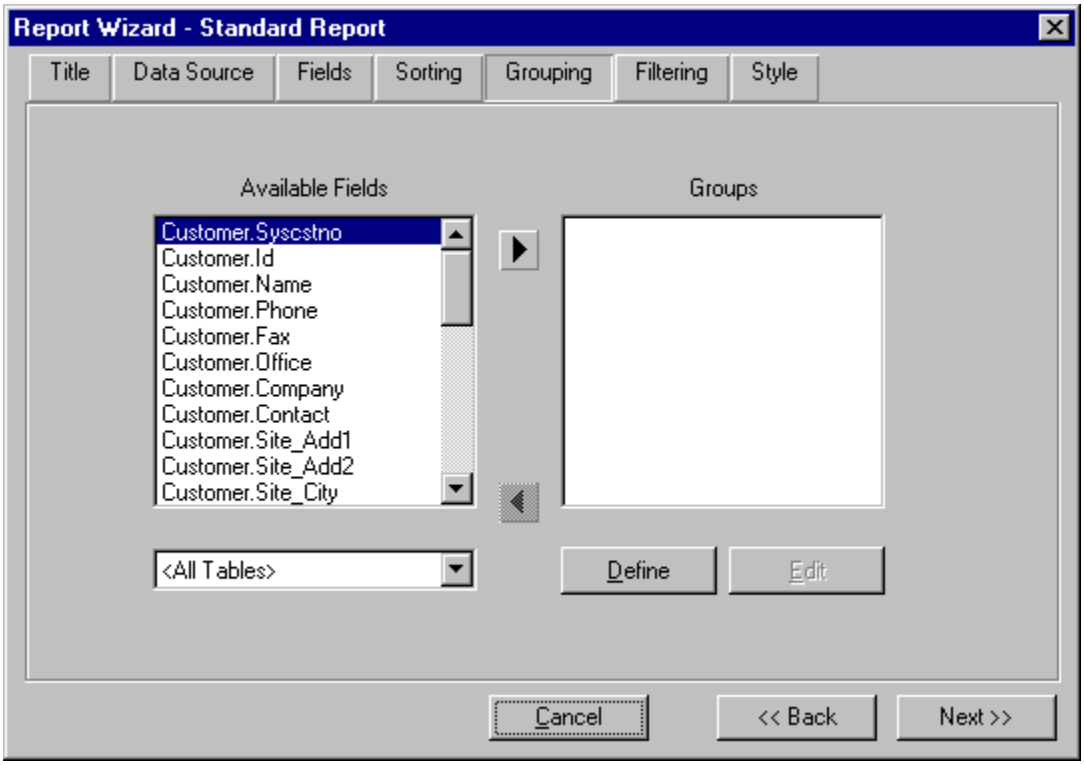
ReportPro supports sorting on multiple fields. The first field you select identifies the primary sort order. Each additional field you specify defines a sub-sort of the previous field(s). For example, if your data source consisted of a sales table and you selected the fields SALESMAN and DATE, ReportPro would sort the invoices first by salesman and then sub-sort the invoices in date order for each of the salesman.

You can specify a Ascending or Descending sort for each of the fields in the *Sorted By* list box. To do so, highlight the desired field by clicking it once with the left mouse button and then select the desired sorting method from the *Sort Method* list box. You will notice that doing so will display either the letter **A** or the letter **D** to the left of the field identifying the chosen sort method.

You can also specify that the sorting process produce only unique values by checking the *Unique Order* check box. This process eliminates duplicate rows from the data source. For example, if your sales table contained only one salesman and you created unique sort on salesman your report would only print one record.

Grouping Tab

The Grouping tab allows you to specify how data in the report will be grouped. The concept of grouping is covered in detail later in this manual, but in general, when you specify grouping, ReportPro creates additional report bands that print before and after a series of records.



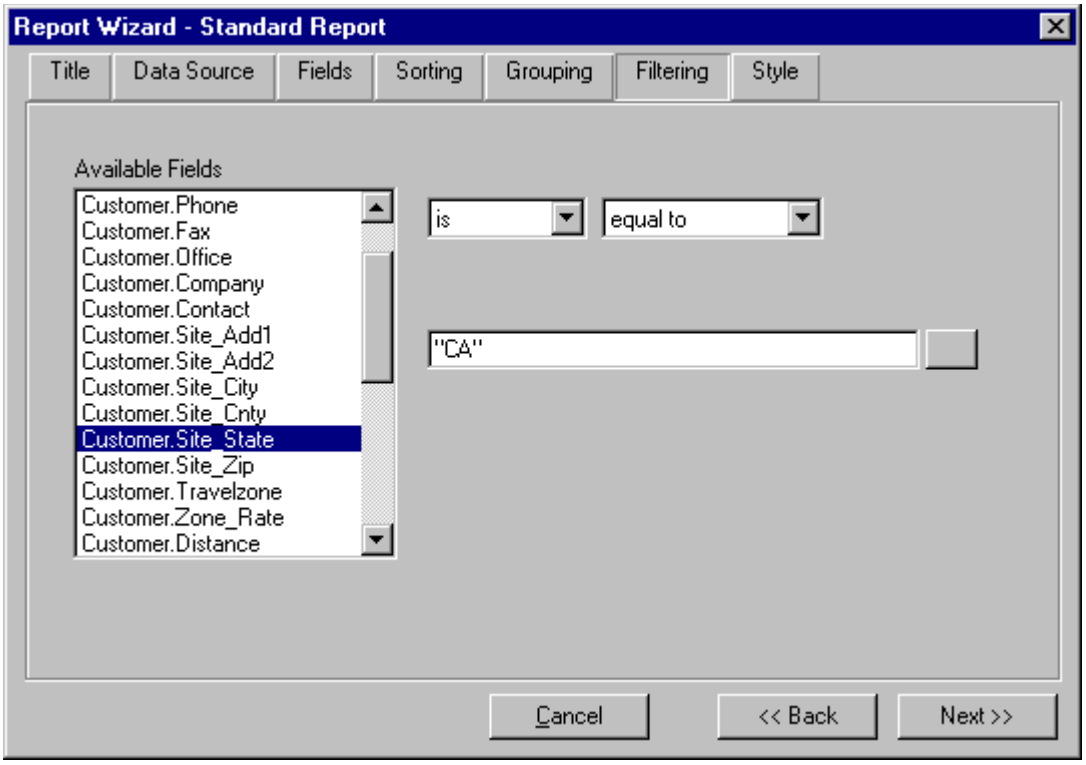
The Grouping tab works similar to the Fields and Sorting tab. You specify groups by copying fields from the *Available Fields* list box into the *Groups* list box.

Pressing the *Define* button will display another dialog where you can define an expression based group by specifying a group name and a grouping expression. You can also edit an existing group by selecting the desired group in the *Groups* list box and pressing the *Edit* button.

For more information on how Grouping works please see the chapter on Grouping later in this manual.

Filtering Tab

The Filtering tab allows you to define a filter for the data source. A filter is an expression which restricts the amount of data that is retrieved from the data source. Each record in the data source is tested against the filter expression. If the filter expression returns true that record is added to the report, otherwise it is skipped.



A filter is created by defining filter clauses for one or more fields in the *Available Fields* list box. If a filter clause is defined for two or more fields, the clauses are combined into a compound expression using a logical *AND* operator.

To define a filter clause, highlight the desired field in the *Available Fields* list box, select an operator from the *Operator* list box and enter the comparison expression in the *Expression* edit.

The *Operator* list box contains many different operators to allow you to create sophisticated filtering conditions. Each of the operators is discussed in the table below.

Statement	Description
Any Value	Any value will match. This is considered a state where no filter clause is defined for the selected field.
Equal To	Only include records where the field's value exactly matches the value

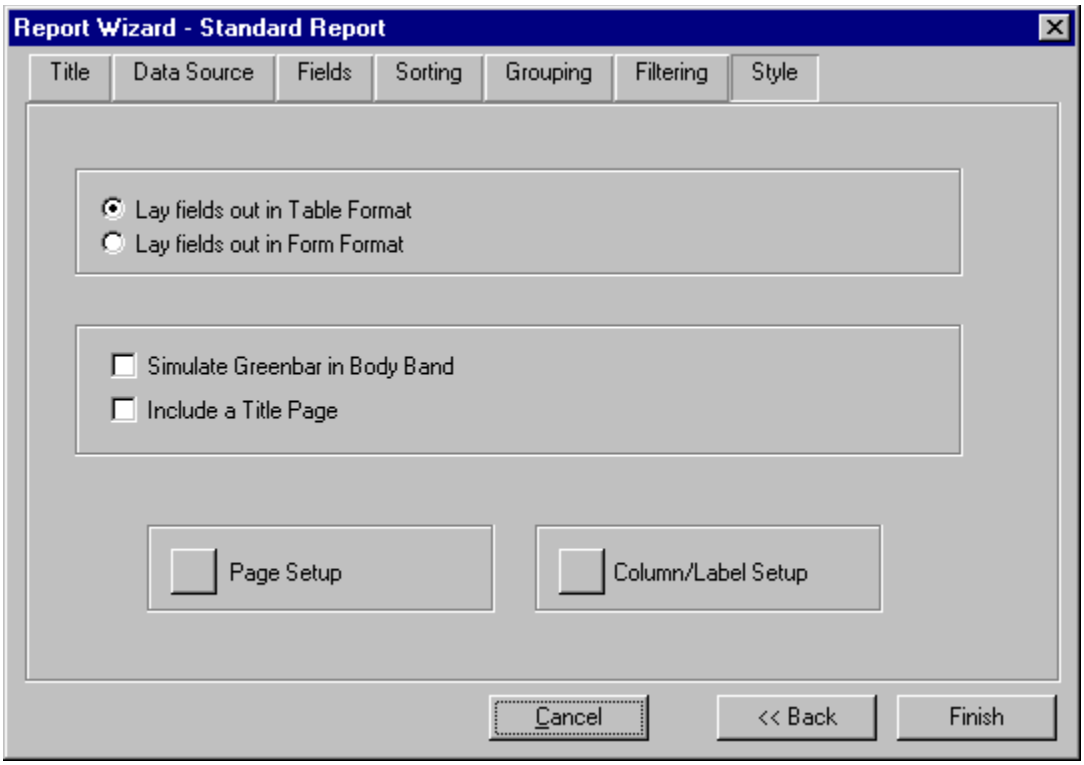
	of the expression entered in the *Expression* edit. For example, "AA" is exactly equal to "AA" but not to "AAA."
Similar To	Only include records where the field's value is similar to the value of the expression entered in the *Expression* edit. For example, "AA" is similar to "AAA" but not exactly equal.
Less Than	Only include records where the field's value is less than the value of the expression entered in the *Expression* edit. Values equal to the expression can also be included by checking the "OR EQUAL TO" check box.
Greater Than	Only include records where the field's value is greater than the value of the expression entered in the *Expression* edit. Values equal to the expression can also be included by checking the "OR EQUAL TO" check box.

The *Negate* list box contains an *IS NOT* option that allows you to negate the result of the comparison operator you specified in the *Operators* list box. Records will be included in the report if the negated result is true.

If desired, the Expression Builder can be used to build the comparison expression. To display the Expression Builder, click on the button to the right of the *Expression* edit.

Style Tab

The Style Tab allows you to specify miscellaneous style settings for the report. Here you can specify the field layout style, greenbar simulation, title page, page layout and column/label settings.



The field layout style can be set to either Table or Form format. If you choose Table format, the database fields will be laid out in a horizontal manner along the top of the Body band. If you select the Form format, the fields will be laid out vertically on the left edge of the Body band.

Greenbar paper has pre-printed horizontal bars evenly spaced down each page. This feature of the paper makes it easy to follow information that is laid out horizontally on a page especially if the paper is greater than 14 inches wide. ReportPro can simulate Greenbar by printing gray bands across the page at regular intervals. This effect looks best on laser printers but is generally unreadable when faxed.

A Title Section can be automatically included in the report by checking the *Title Page* check box. The Title section prints at the beginning of a report and is useful for creating report cover pages.

Both Page Layout and Column attributes can be specified by clicking the appropriate button at the bottom of the Style tab. Both these options are covered in detail later in this manual.

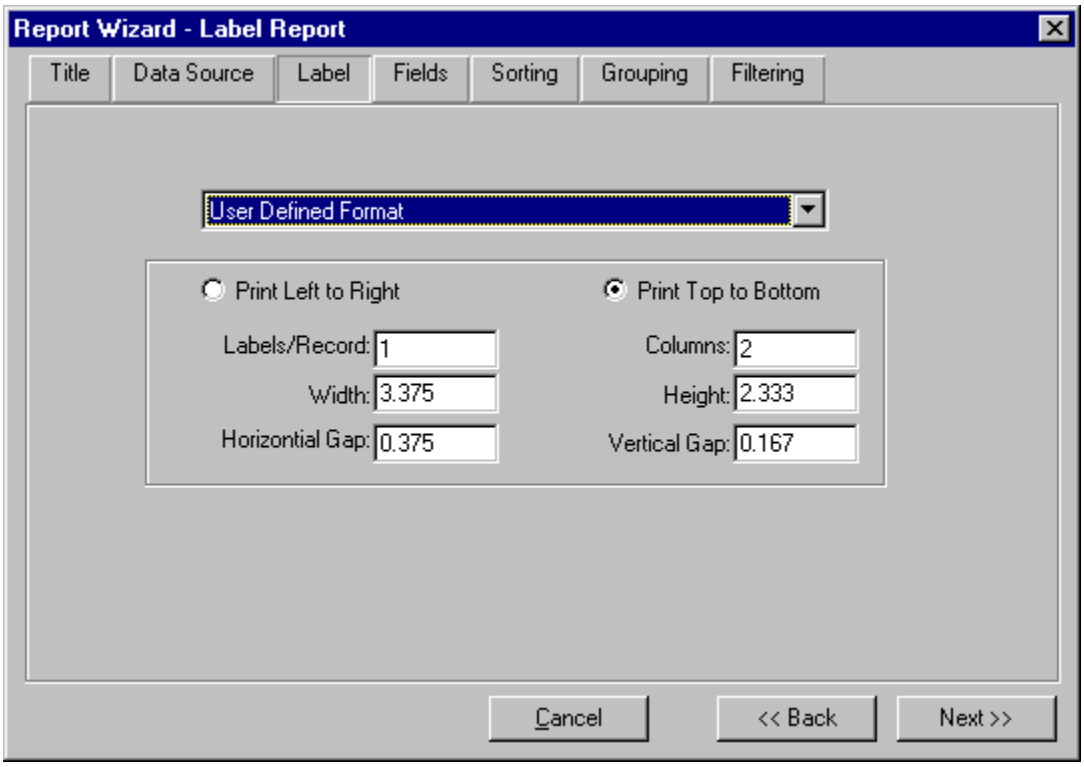
After you have defined your reporting options press the *Finish* button and the Report Wizard will open a work window and build a new report based on the options you specified. If the *Finish* button is disabled, it is because you haven't specified the required information to the Report Wizard. At a minimum you must specify a data source and select at least one field on the Fields tab.

Label Report Wizard

The Label Report Wizard guides you through the creation of a report that prints labels or contains columns. The Label Report Wizard is very similar to the Standard Report Wizard and shares the following common tabs: Title, Data Source, Sort Order, Group, and Filtering. For more information on these common tabs, please see the discussion of the Standard Report Wizard.

Label Tab

In addition to the common tabs, the Label Report Wizard contains a Label tab to allow you to define labels or column attributes.



The drop-down list box at the top of the tab contains a listing of standard Avery labels. You can select one of the pre-defined label types to have the label attributes automatically initialized or you can select *User Defined Format* to define your own attributes.

When you select the *User Defined Format* the controls in the lower section of the tab become enabled. The purpose of each control is defined in the following table:

Control	Description
Print Left To Right and Print Top To Bottom	These radio buttons control the direction in which the labels print. If the *Print Left to Right* button is selected then the labels print all columns from left to right and then move to the next row. If the *Print Top To Bottom* button is selected, ReportPro prints all labels in the first column and then moves to the next column.

Labels/Record	This option allows you to specify the number of labels that get printed for each record in the data source. For example if you were printing a customer address listing and you wanted to print two labels for each customer, you would specify two in the Labels/Record edit control.
Columns	Here you specify the number of columns you wish to print on each page of the report.
Width	Here you specify the width of each label.
Height	Here you specify the height of each label.
Horizontal Gap	In this edit control, you specify the horizontal distance between labels or columns.
Vertical Gap	This edit allows you to specify the vertical distance between labels.

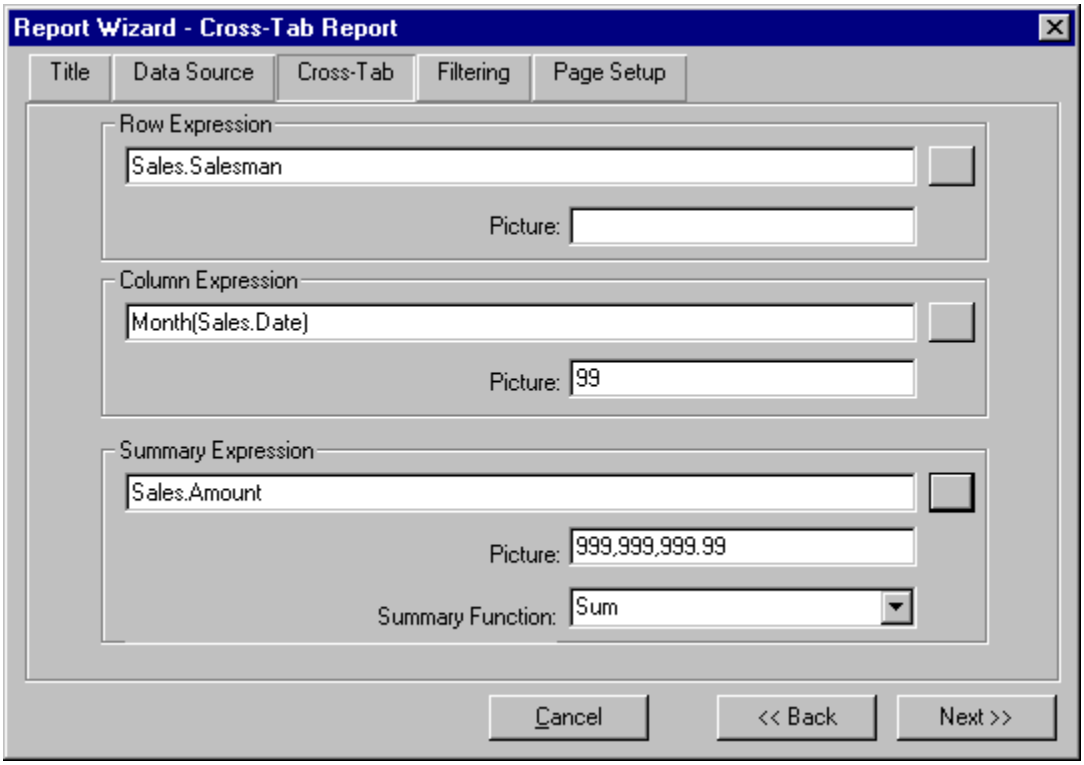
Cross-Tab Report Wizard

The Cross-Tab Report Wizard guides you through the creation of a Cross-Tab style report. The Cross-Tab Report Wizard contains the following tabs in common with the Standard Report Wizard: Title, Data Source, and Filtering. Please see the discussion of the Standard Report Wizard above for more information on the common tabs.

In addition to the common tabs, the Cross-Tab Report Wizard also contains a Cross-Tab Tab and a Page Setup Tab.

Cross-Tab Tab

The Cross-Tab tab allows you to define the expressions which are used to create the Cross-Tab.



To define a Cross-Tab, you must specify a row, a column and a summary expression. The expressions can be entered directly or via the Expression Builder. To access the Expression Builder, click the square button to right of the desired edit.

The *Picture* edit, which is available for each expression, allows you to specify an optional picture clause which is used to format the expression's value. See Appendix A for more information on picture clauses.

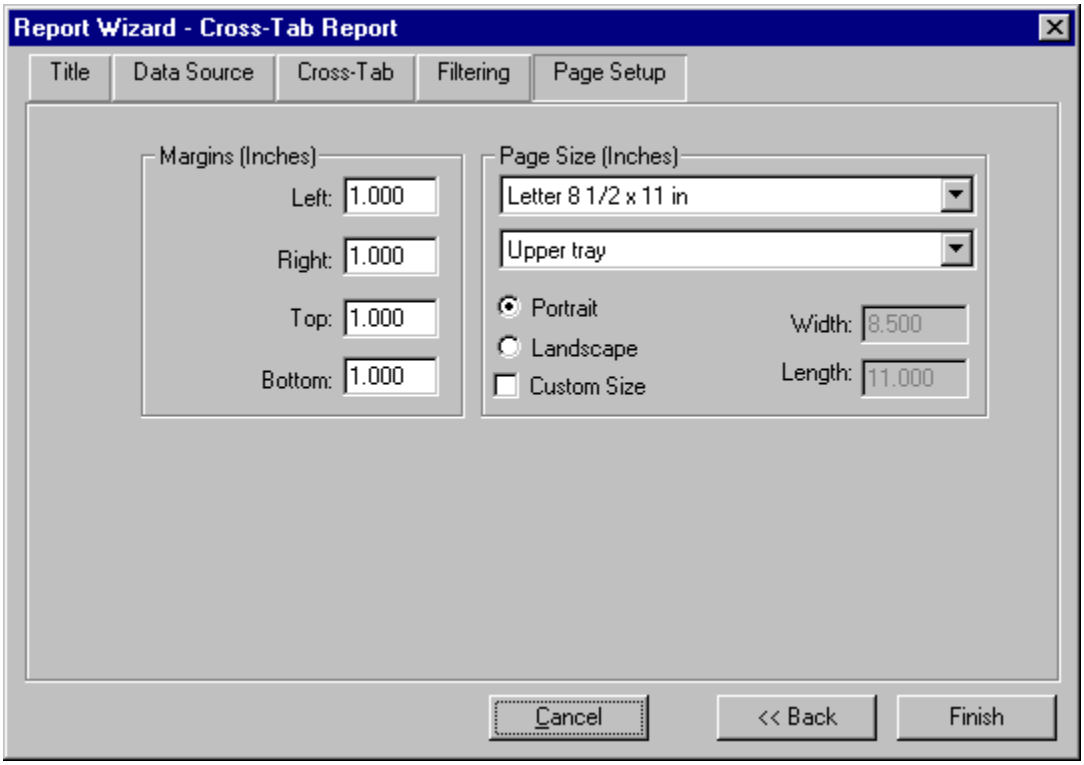
The *Row Expression* governs the values that are displayed in the Row Label (left most) column of the Cross-Tab. The row expression is evaluated for each record in the data source. Each unique value returned by the expression will get a row in the Cross-Tab.

The *Column Expression* determines the values that are displayed in the Column Label (top) row. The column expression is also evaluated for each record in the data source. Each unique value returned by the expression will get a column in the Cross-Tab.

Finally, the Row and Column values are cross-referenced and summarized using the *Summary Expression*. The Summary Function list box allows you to specify which mathematical operation is performed on the summary expression. Five summary functions are available, but Average and Sum are only available for summary expressions that return a number.

Page Setup Tab

The Page Setup tab allows you to specify paper size, orientation, and margins.



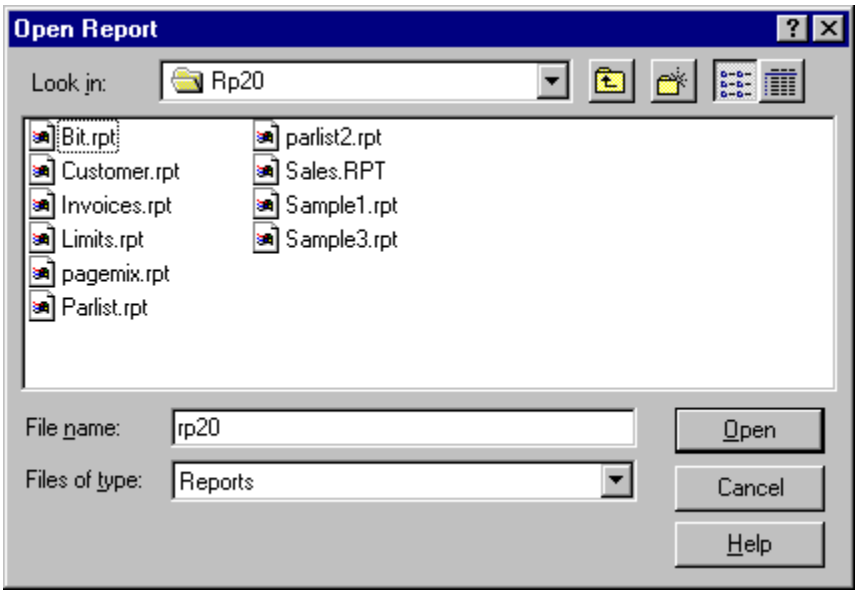
The Margin edits allow you to specify the margin sizes. The dimension of the margins will be inches or centimeters depending on how ReportPro is configured.

The page size settings allow you to set the paper size, paper source and orientation of the current section. The selections available in the lists reflect those of the default windows printer.

Selecting the *Custom Size* check box will enable the *Width* and *Height* edits where you can specify custom paper sizes. Custom paper sizes are not supported on most printers and are primarily used to specify custom sizes for continuous feed labels on dot matrix printers.

Opening a Report

To open an existing ReportPro report, choose *Open* from the *File* menu. Doing so displays the Open Report dialog where you specify the report you wish to open.



The Open Report dialog is a standard Windows dialog so its operation should be familiar to you. The drop-down list box at the top of the dialog allows you to navigate through the directory structure of your hard drive(s). When you select the desired directory, the available report files will be displayed in the large list box in the middle on the dialog.

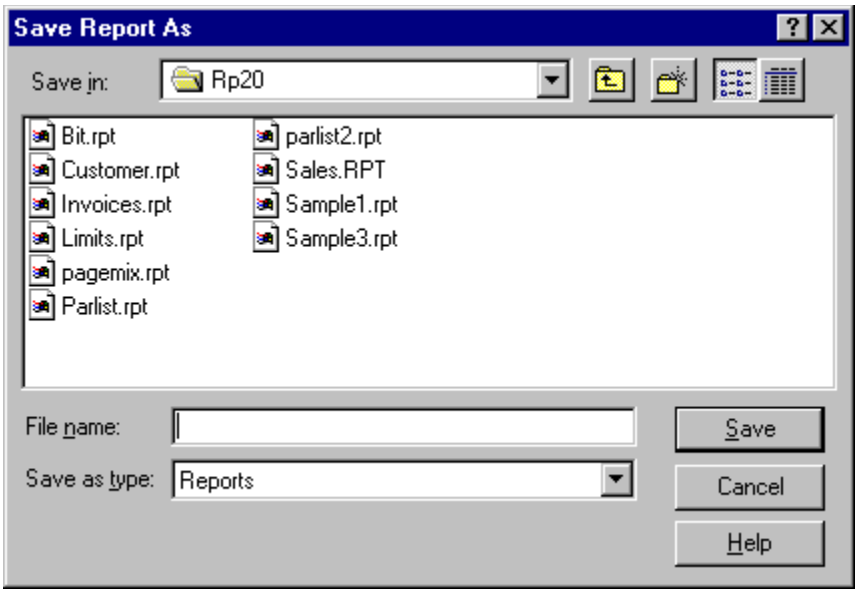
You open a report by double-clicking on the desired file or highlighting the file and pressing the Open button. Alternately you can enter the report name, including path, in the File Name edit toward the bottom of the dialog.

When ReportPro opens a report it also opens local tables, connects to SQL data sources, retrieves sample data and validates expressions used throughout the report. If ReportPro encounters an error, it presents a dialog identifying the problem.

One common error that occurs is when local tables are moved from the directory where they were when the report was created. In this case, ReportPro will look for them in the directory where the report resides. If the tables can not be located, ReportPro will prompt you for the location of the missing tables.

Saving a Report

To save a report, choose *Save* from the *File* menu. If the report has previously been saved, ReportPro will automatically save the report to the same file name. If the report is a new report or you choose *Save As* from the *File* menu, the Save Report As dialog appears allowing you to specify a file name.



The Save Report As dialog is a standard Windows dialog much like the Open Report dialog. The drop-down list box at the top of the dialog allows you to navigate through the directory structure of your hard drive(s). When you select the desired directory, the available report files will be displayed in the large list box in the middle on the dialog.

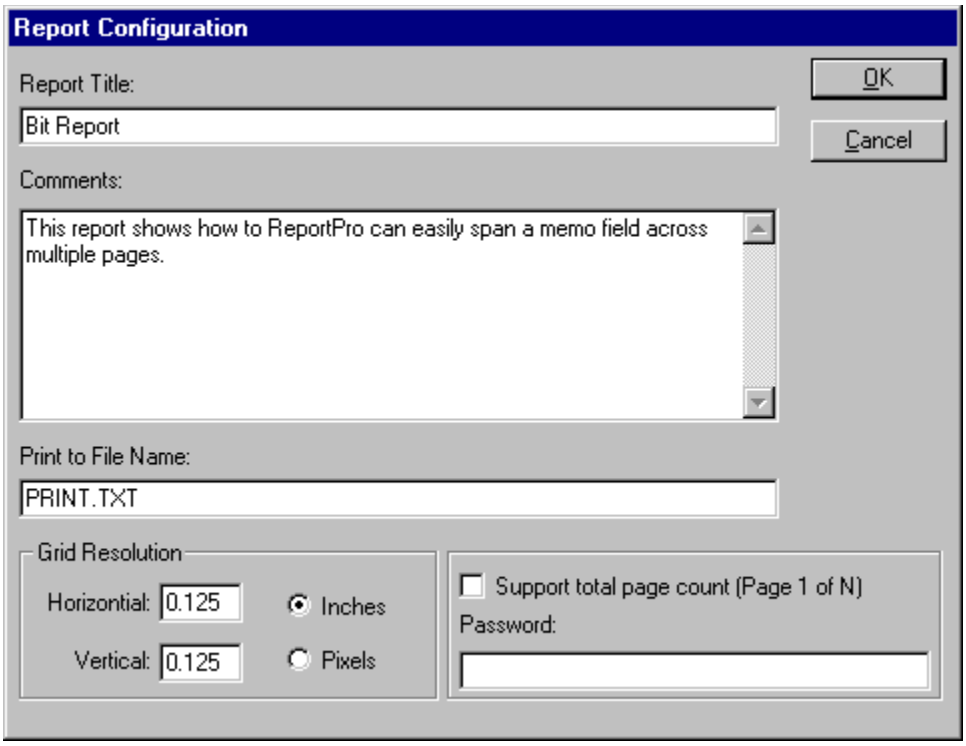
After you have identified a directory, type the report name in the File Name edit and click the Save button.

If the file you specify already exists, ReportPro will ask you if you wish to overwrite the file. If you choose Yes, ReportPro will overwrite the file, otherwise you will be returned to the Save Report As dialog.

Although ReportPro does not enforce a specific file extension, if you do not specify a file extension, ReportPro will default to "RPT".

Configuring a Report

The Report Configuration dialog allows you to specify miscellaneous properties that relate to the entire report. The Report Configuration dialog is displayed when you select *File, Setup, Report* from the work window menu.



The Report Title attribute is provided to allow you to assign an abstract name to the report. If specified, it is displayed along with the report file name in the caption of the report work window. This field is optional and is not used by ReportPro.

The Comments edit is a convenient location to keep notes specific to a report. You may find it helpful to keep notes describing the purpose of the report or technical details on how it was implemented. This field is also optional and is not used by ReportPro.

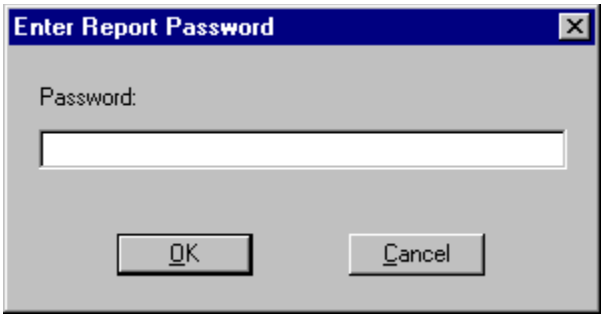
The Print to File edit allows you to identify the file name that ReportPro prints to should you select the *Print To File* option on the Print dialog. The file name specified here is only the default file name. Prior to printing to the file, ReportPro displays another dialog allowing you to change the file name.

The Grid Resolution controls allow you to set the resolution of the grid displayed in the work window. You can specify the dimension in user dimensions (Inches or Centimeters) or Pixels.

The Support Total Page Count option instructs ReportPro to calculate the number of pages in the report

prior to printing the report. This feature allows you to print the page count on each page in the report. Selecting this option causes ReportPro to generate the report twice, once in the background to determine the number of pages and the other when the report is actually printed. Do not select this feature unless you need to print the page count since it doubles reporting time.

The Password edit allows you to password protect a report, preventing others from opening the report in ReportPro's designer. When you open a report that is password protected, a dialog is displayed prompting you for the password.

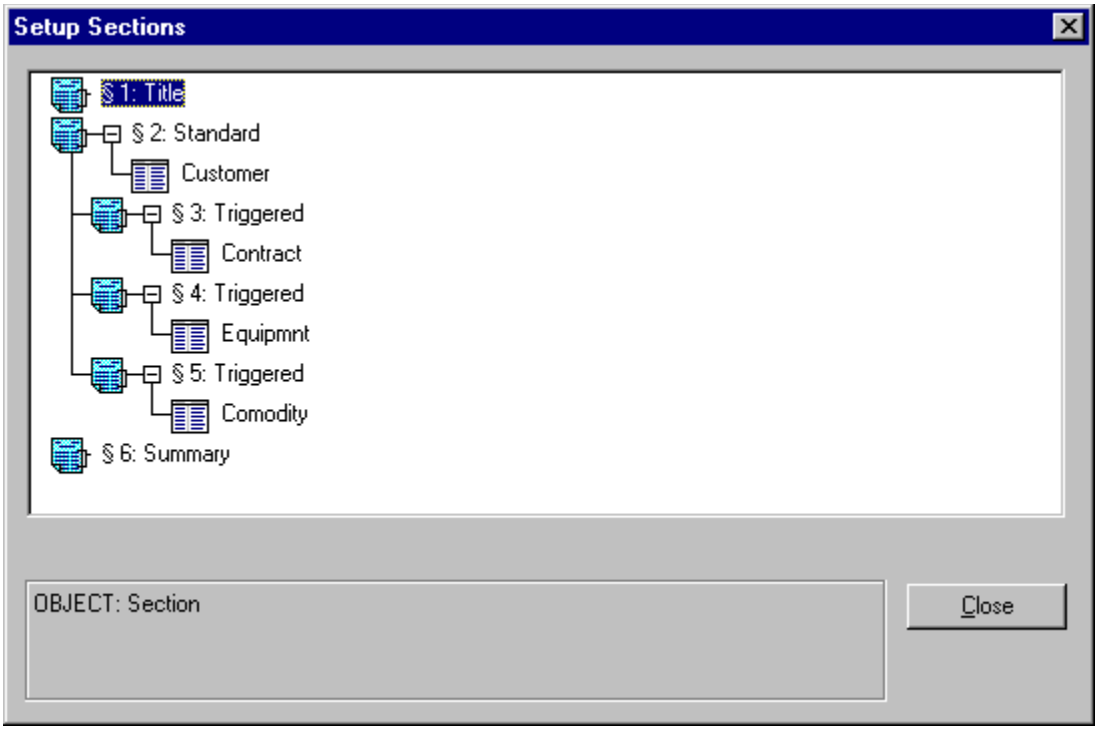


If the password is not correctly entered, ReportPro will not open the report. Make sure you do not forget the password. If you forget it, there will be no way to recover the report.

Configuring Sections

The Setup Sections dialog allows you to add, delete and configure sections in your report. This is also where you configure the data source associated with each section.

Because of the large number of options available for sections and data sources, the discussion of the Setup Sections dialog is broken-up into three chapters. Topics related to configuring sections are covered here while data source topics are covered in the next two chapters.



The Setup Sections dialog is accessed by choosing *Sections* from the *File, Setup* menu. The dialog consists of a hierarchical list box which shows all the sections of the report. It also displays the tables and queries that provide data to each of the sections. There are three icons that you'll encounter in the list box:


Represents a Section.


Represents a SQL Query.


Represents a Table. It is used for both local tables and tables that belong to a SQL query.

Each object in the list box can have children and attributes. Children are objects that are owned by another object. They are always connected to lines coming straight down from the parent object. For example, figure 30 shows that Section 2 has three child sections.

Attributes are represented by lines going right and then downward. The lines are connected by boxes that contain either + or -. For example, data sources are attributes of sections. Figure 30 shows that Section 2 uses the Customer table.

If you wish to hide the children of an object, you can double-click on the object's icon and all its children will be hidden. You can also hide the attributes of an object by double-clicking the +/- box on the attributes line.

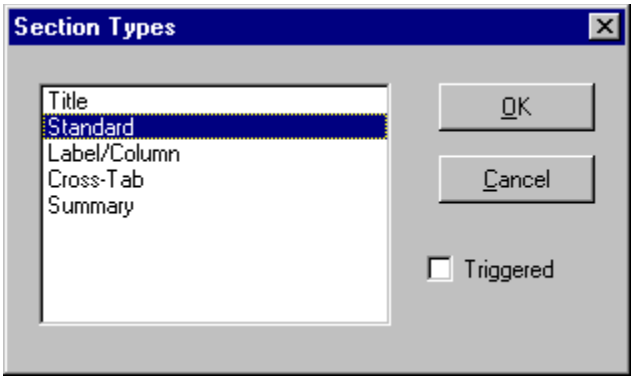
When you select an object by clicking the left mouse button on it, detailed information about the object is displayed on the status bar at the bottom of the Setup Sections dialog.

When you click the right mouse button on an object, a pop-up menu will appear with a list of features available for the selected object. You can also add objects to the list box via the pop-up menus. The options presented in the pop-up menu depend on which object you've selected.

The following options are available in the section pop-up menu:

Add Section

This menu allows you to add a section to the report. A section is like a sub-report that is embedded within the primary report. When you choose this option, you will be presented with a dialog to allow you to choose the type of section you wish to add.



The section types displayed in the dialog depend on which sections are already defined in your report. For example, since there can only be one Title and one Summary section, these section types will not be available if they have already been included in the report.

The following table provides a brief description of each section type.

Section Type	Description
Title	This section consists of a single page that prints at the beginning of

	the report. It is useful for creating cover pages for a report.
Standard	This is a standard section. It does not support columns or perform Cross-Tab calculations.
Label/Column	This section is columnar in nature. Mailing address labels are a good example. You can also use this type of section to create a report that has multiple columns.
Cross-Tab	Cross-Tab sections are those that perform summary or arithmetical processing on a database and report the result set in a grid type fashion.
Summary	This type of section prints at the end of a report after all other sections have printed. It is primarily used to print summary information.

The *Triggered* option creates a section that is a child of the highlighted section. Triggered sections are printed when an expression changes value. For example, you would use a triggered section to print a sub-report for each customer in the main report. In such a case you would specify the customer name or ID as the trigger expression. The *Triggered* option is only available for Standard, Label/Columnar and Cross-Tab sections.

The events that occur after you select a section type depends on which type of section you chose. If you selected a Title or Summary section, the sections are automatically added to the report with no user interaction. If you choose a Standard, Label/Column or Cross-Tab section you will be presented with a series of dialogs prompting you for data source information. When you choose a Cross-Tab section you are also prompted for Cross-Tab configuration information. See the chapter entitled *Creating Cross-Tab Reports* for more information on how to configure a Cross-Tab section.

If you choose the *Triggered* option, you will be prompted to specify the expression that triggers the section. A triggered section prints when the data returned by the trigger expression changes value. For example, if you are printing a customer listing report and you have a section that triggers on customer name, the section will print for each customer.

If, however, you sorted the report by the state where the customer resides, and you specify that the section triggers on state, then the section would print once for each state but not for each customer.

Delete Section

This option allows you to delete the currently highlighted section. This option is very straight forward but with one exception: you must always have at least one section (not including Title and Summary sections) in your report. When you delete a section, all its children sections are also deleted.

Header and Footer Bands

This option allows you to specify if a section has Header and Footer bands. When a section has Header and Footer bands, it always starts printing on a new page. Otherwise it prints continuously on the same page as either its parent section or the proceeding sequential section.

Without Header and Footer bands, a section has no page attributes or Background band of its own. So during the printing process it inherits these attributes from the previous section.

This option is not available for Title and Summary sections as they do not support Header or Footer bands.

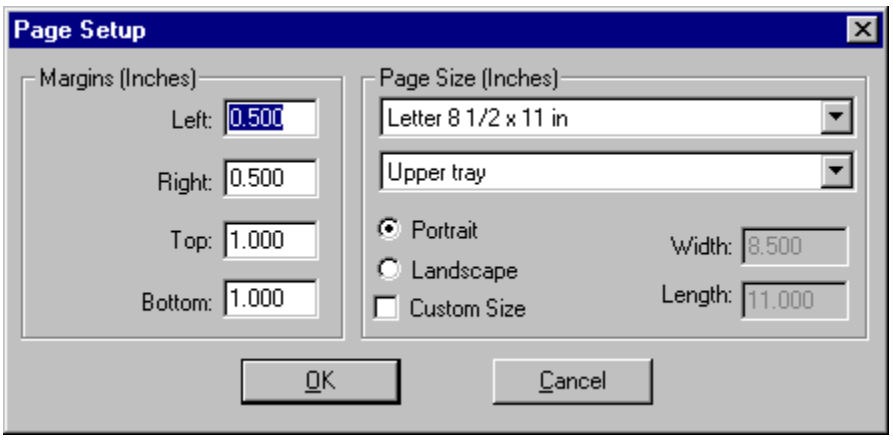
Print When

The *Print When* option allows you to specify *when* a section prints. Selecting this option displays the Expression builder where you specify the *Print When* expression. This expression is evaluated during reporting when it is time for the section to print. If the expression evaluates to true the section is printed; otherwise it is skipped.

For example, assume that you have a customer listing report that contains a triggered section that prints for each customer. If you wanted the triggered section to print for customers with purchases that amounted to more than $10,000, you would specify a *Print When* of: **"Customer.Purchases>10000"**.

Page Setup

The Page Setup option allows you to specify page related information like paper size, page orientation and the margin sizes. This option is only available for the Title section and any sections that have Header and Footer bands.



In general, the various settings in the Page Setup dialog are self explanatory. The Margin edits allow you to specify the margin sizes. The dimension of the margins will be inches or centimeters depending on which mode the report is in.

The page size settings allow you to set the paper size, paper source and orientation of the current section. The selections available in the lists reflect those of the current printer.

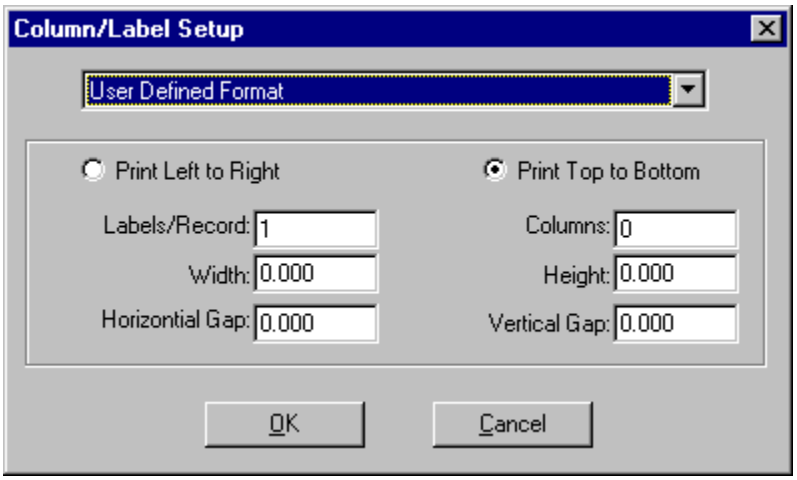
Selecting the *Custom Size* check box will enable the *Width* and *Height* edits where you can specify custom paper sizes. Custom paper sizes are not supported on most printers and are primarily used to specify custom sizes for continuous feed labels on dot matrix printers.

Trigger

This option allows you to edit the trigger expression that was specified when a triggered section was created. This option is only available for triggered sections. For more information on Triggered sections see the discussion under "Add Section".

Column/Label Setup

When you select this option, you are presented with the Column/Label Setup dialog. This dialog allows you to specify the label or column attributes of a section. This feature is available for all sections except the Title and Summary sections.



The drop-down list box at the top of the dialog contains a listing of standard Avery labels. You can select one of the pre-defined label types to have the label attributes automatically initialized or *User Defined Format* to define your own attributes.

When you select the *User Defined Format* the controls in the lower section of the dialog become enabled. The purpose of each control is defined in the following table:

Control	Description
Print Left To Right and Print Top To Bottom	These radio buttons control the direction in which the labels print. If the *Print Left to Right* button is selected then the labels print all columns from left to right and then move to the next row. If the *Print Top To Bottom* button is selected, ReportPro prints all labels in the first column and then moves to the next column.
Labels/Record	This option allows you to specify the number of labels that get printed for each record in the data source. For example if you were printing a customer address listing and you wanted to print two labels for each customer, you would specify two in the

	Labels/Record edit control.
Columns	Here you specify the number of columns you wish to print on each page of the report.
Width	Here you specify the width of each label.
Height	Here you specify the height of each label.
Horizontal Gap	In this edit control, you specify the horizontal distance between labels or columns.
Vertical Gap	This edit allows you to specify the vertical distance between labels.

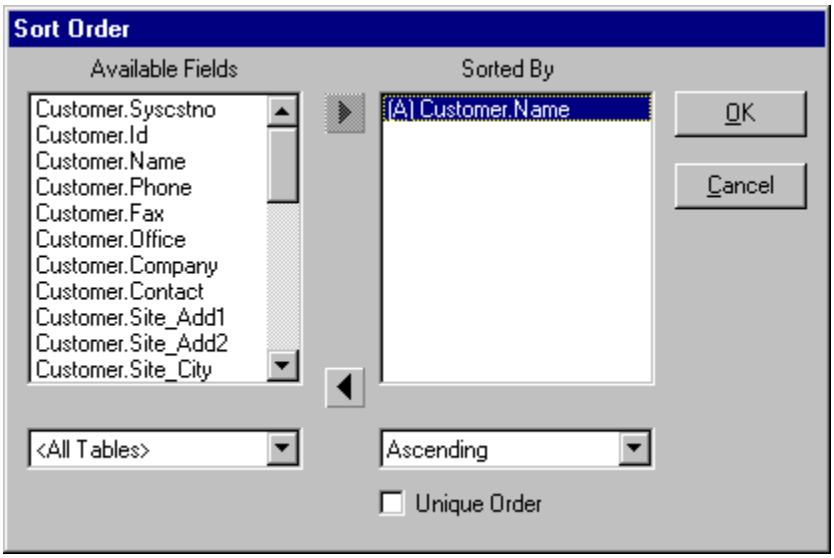
Sort Order and Optimize Sort Order

The *Sort Order* and *Optimize Sort Order* options work together to allow you to specify how data from a section's data source is sorted. ReportPro allows you to sort your data on multiple fields in either ascending or descending orders.

An ascending sort means the resulting sort will look like "A", "B", "C" or 1, 2, 3. A descending sort will produce results like "C", "B", "A" and 3, 2, 1. When ReportPro sorts data from local tables, the sort is case independent meaning that "c" and "C" will be sorted to the same location. Whether or not data sorted by a SQL server is case independent depends on how the server was configured.

Selecting the *Sort Order* option here is the same as selecting the *Order* menu option from ReportPro's main menu except the *Order* menu option sets the sort order for the displayed section while the *Sort Order* pop-up menu option allows you to set the sort order for any section in the report.

When you select the *Sort Order* option, you are presented with a dialog that allows you to specify the sort criteria.



To specify a sort order, highlight the field in the *Available Fields* list box that you wish to sort on and click on the > button . The field will be added to the *Sorted By* list box.

Fields in the *Sorted By* list box have an indicator next to them that identifies if the field will be sorted in ascending or descending order. "(A)" indicates an ascending sort order, while "(D)" indicates descending. To change the sort method, highlight the *Sorted By* field you wish to change, and select the desired method from the combo box locate at the bottom of the dialog.

More than one field can be specified in the *Sorted By* list box. Any subsequent field creates a sub-sort of the previously listed fields.

The *Unique* Sort option forces ReportPro include only unique values in the sort order. For SQL data sources this causes the "Distinct" clause to appear in the SELECT statement.

ReportPro is capable of sorting information from dissimilar data sources. When you sort information from different sources, ReportPro must retrieve all the information from the data sources and store the information on a local hard drive. Though this is an extremely power feature, it can require a large amount of disk space and processing time depending on how much information is being gathered.

If the *Optimize Sort Order* option is selected, ReportPro will query the database driver to determine if the driver can sort the data. In most cases, the driver can sort the data and this dramatically speeds up the sorting process. If the driver cannot handle the sorting process, ReportPro will automatically create the sort.

Occasionally ReportPro may not be able to properly determine if the driver can sort the data. ReportPro will incorrectly assume the driver can perform the sort and an error will result. In this instance you should deselect the *Optimize Sort Order* option and have ReportPro sort the data for you. Otherwise, you should always leave the *Optimize Sort Order* option selected.

Filter and Optimize Filter

The *Filter* and *Optimize Filter* options work together to limit the data that is included in the report. When you select the *Filter* option the expression builder is displayed to allow you to enter a filter expression. The filter expression must return a logical value (i.e. the expression must be either true or false).

The filter expression is evaluated prior to printing each record. If the expression evaluates to true, the record is printed; if not, the record is skipped.

Selecting the *Filter* option here is the same as selecting *Filter* from ReportPro's main menu.

If the *Optimize Filter* option is selected, ReportPro passes the filter expression to the database driver for processing. In many cases this greatly improves reporting speeds. For local CDX tables, the driver will try to use bit mapped query optimization to speed up the query. Many SQL data sources also support

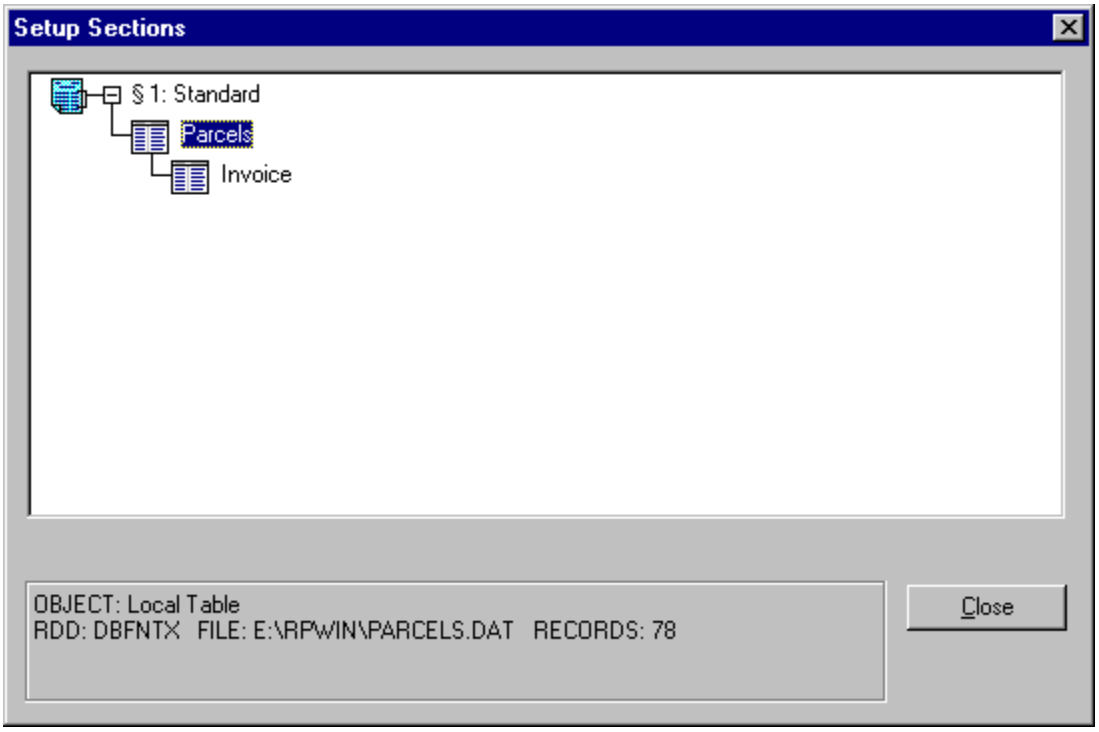
query optimization.

If you want to filter on an expression that is not supported by the database driver, you can deselect the *Optimize Filter* option and have ReportPro perform the filtering.

Configuring Local Tables

Data source configuration and management is handled via the Section Setup dialog. This is where you add, delete, configure and specify the relationships between local tables. For information on the basic operation and layout of the Setup Sections dialog, please refer to the previous chapter.

When local tables are used to supply data to a section the Setup Sections dialog will look similar to figure 28. The figure shows two dBase tables related together. In this example, the Invoice table is a child of the Parcel table.



Tables are added, deleted and configured via pop-up menu options that are available when you click the right mouse button over a table object. The options available for local tables are:

Add Child Table/SQL Query

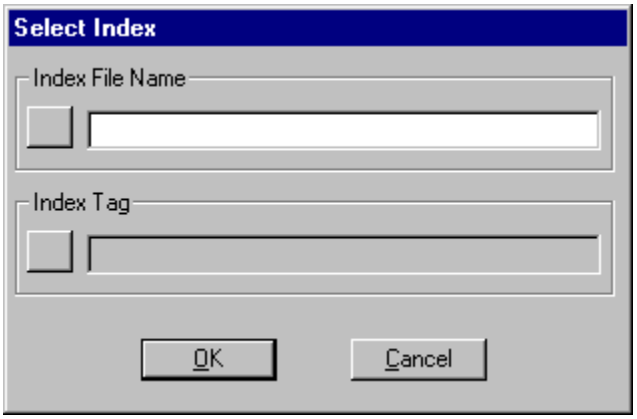
This option allows you to add a child table or SQL query to the selected table. If you choose this option, you will be prompted to select a data source. See "Creating a Standard Report" for more information on selecting a data source.

When you add a child table or query, ReportPro automatically defines the relationship between the parent and the child. To modify the relationship, select the *Relationship* option for the child table or query.

Index

The Index option allows you to specify an existing index to use for the sort order. This option is only available for the primary table in the data source. Specifying this option can reduce the time required to generate a report since the sort order doesn't have to be created by ReportPro.

When you select the *Index* option you are presented with a dialog prompting you for the name of the index file.



You can enter the index file name directly or click on the button to the left of the Index File Name edit to be presented with a standard File Open dialog.

If the database driver you are using supports compound indexes, you must also specify the Index Tag name. To select the index tag, click the button to the left of the Index Tag edit and you will be presented with a list of tags in the selected index file. If you do not specify an Index Tag name, then the table will be accessed in record order.

Seek Expression

The Seek expression allows you to specify a starting point for the report. If you specify a Seek expression, ReportPro will evaluate the expression before reporting starts and then seek to that location in the primary table. Reporting will start at the point of the seek.

Seek expressions are only valid if an index for the primary table or sort order is specified. The seek that is performed is a soft seek; that is, it locates the point were the value returned by the expression would reside in the index even if the value is not present in the index.

If ReportPro creates the sort order, it is important to note that ReportPro converts all field values to character strings. For example, if you create a sort order on a numeric field, ReportPro converts the field to a string value and then builds the index. Generally this has no affect on reporting, but it does affect how you must specify a Seek expression since the Seek expression must also be specified as a character string.

Here's how ReportPro converts fields when creating an index:

Field Type	Generated Expression	Sample Seek Expression
Numeric	str(field name, width, decimals)	" 100"
Character	upper(field name)	"SMITH"
Date	dtos(field name)	"19961205"
Logical	iif(field name,"1","2")	"1"
Memo	upper(left(field name,40))	"MEMO SEEK"

While Expression

The While expression allows you to specify an expression which can conditionally terminate the report. ReportPro starts at the top of the primary table (or at the position left by the Seek expression) and prints records that satisfy the filter condition. If you specify a While condition the report prints until the While expression returns false at which point reporting stops.

Relationship

ReportPro allows you to control the relationship between tables. This feature gives you precise control over how your data is retrieved from the database. When you select the *Relationship* option you are presented with the Table Relations dialog. Here you can specify the relationship between a parent and a child table.

Local tables are linked by an expression in the Parent Table to an expression in the Child Table via an index. The link between the two tables governs how the records in the Parent Table are related to the

records in the Child Table.

Parent & Child Expressions

You can change the Parent and Child expression by choosing the appropriate *Change* button. The Expression Builder is displayed to allow you to make the necessary changes. The following algorithm is used by ReportPro to relate records between two tables:

PARENT EXPRESSION = CHILD EXPRESSION

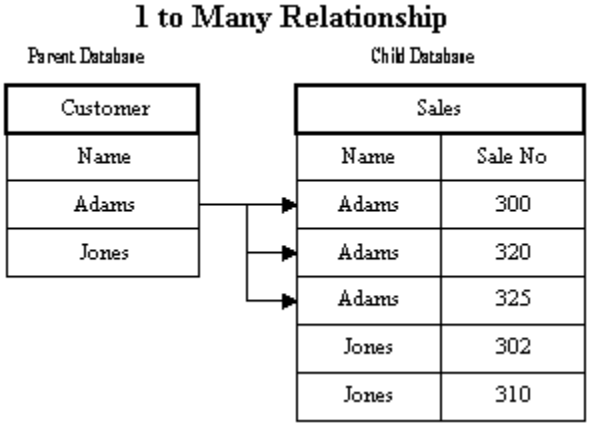
This expression must evaluate to a logical TRUE before records are considered related.

Relationship Type

ReportPro allows you to specify how records in a child table are processed when a report is created. There are two types of relationships supported by ReportPro.

The first type of relationship, the *One to One* relationship, occurs if one and only one record exists in a child table for each record in the parent table. An example of this is when a Sales table is related to a Customer table. Only one Customer can exist for each sale.

The second type of relationship, the *One to Many* relationship, occurs when more than one record exists in a child table for a record in the parent table. This case exists when a Customer table is related into a Sales table. Many Sales Orders can exist for each customer.



Refer to the figure above for the following discussion. When the *Relationship Type* is set to *One to One*, records are processed in the following manor. First, the record pointer in the parent table is moved. Let's assume the current record after the move has a Customer Name of "Adams". As you can see, there are three Sales Orders which belong in that Customer. Next, the record pointer in the child table is moved to the first record that has a Customer Name of "Adams". Then the Customer and Sales table

records are printed on the report. After printing, the Customer record pointer is moved to Customer "Jones". The Sales record pointer is moved to the first Sales Order for "Jones". As you can see, we skipped over two Sales records. The records were not printed on the report.

If the *Relationship Type* had been set to *One to Many*, the following would have occurred. After the Customer record pointer was moved to Customer "Adams" and the Sales record pointer was moved to the first record for "Adams", the records would have been printed on the report. Next, instead of moving the Customer record pointer to the next Customer, the Sales record pointer would be moved to the next Sales record for Customer "Adams". The Customer record pointer does not move. The new Sales record is printed along with the Customer record. The Sales record pointer is moved again. This continues until there are no more Sales records for Customer "Adams". Only at that time is the Customer record pointer moved to "Jones" and the sequence of events repeats.

The record movement method you choose depends on the type of report you are trying to produce. The One to Many method can take significantly longer to traverse a large data set since it is accessing many more records.

Child Index for Relationship

When a local table is linked to another table, an index must be used on the child table to support the relationship. The report engine creates this index automatically if an existing index is not specified.

The drawback to automatic index creation is that it increases the time it takes to run a report. If an index already exists that supports the relationship, it can be specified by choosing the *Select* button. If an index is specified, the report engine will assume that it correctly supports the relationship and will skip the automatic index creation; this greatly speeds up the process of producing the report.

Index Tag

RDD Drivers such as the CDX driver, support index files that contain multiple indexes. A tag is required to identify which index should be used in the index file. You can select the proper index tag by choosing the *Change* button next to the Index Tag information box.

Change RDD

This option allows you to change the database driver you are using for a table. When you select this option you will be presented with a list of available drivers to select from. The driver you choose depends on the type of index and memo files used by the table.

The NTX driver should be used with Clipper indexes and .DBT memo files. The CDX driver should be used with FoxPro indexes and .FPT memo files. The MDX driver is used to access dBase 4 indexes and memo files.

Delete Item

This option deletes the selected table. It is important to note that when you delete a table, you also automatically delete all the children of that table.

Configuring SQL Sources

If you create reports that use ODBC, you'll eventually need to customize and configure ReportPro's SQL Query object. The SQL Query object is used by ReportPro to retrieve information from ODBC sources.

As with sections and local tables, the SQL Query object is configured in the Setup Sections dialog. Here you can add and delete tables from a query. You can also control the relationship between tables and even customize the SQL Select statement.

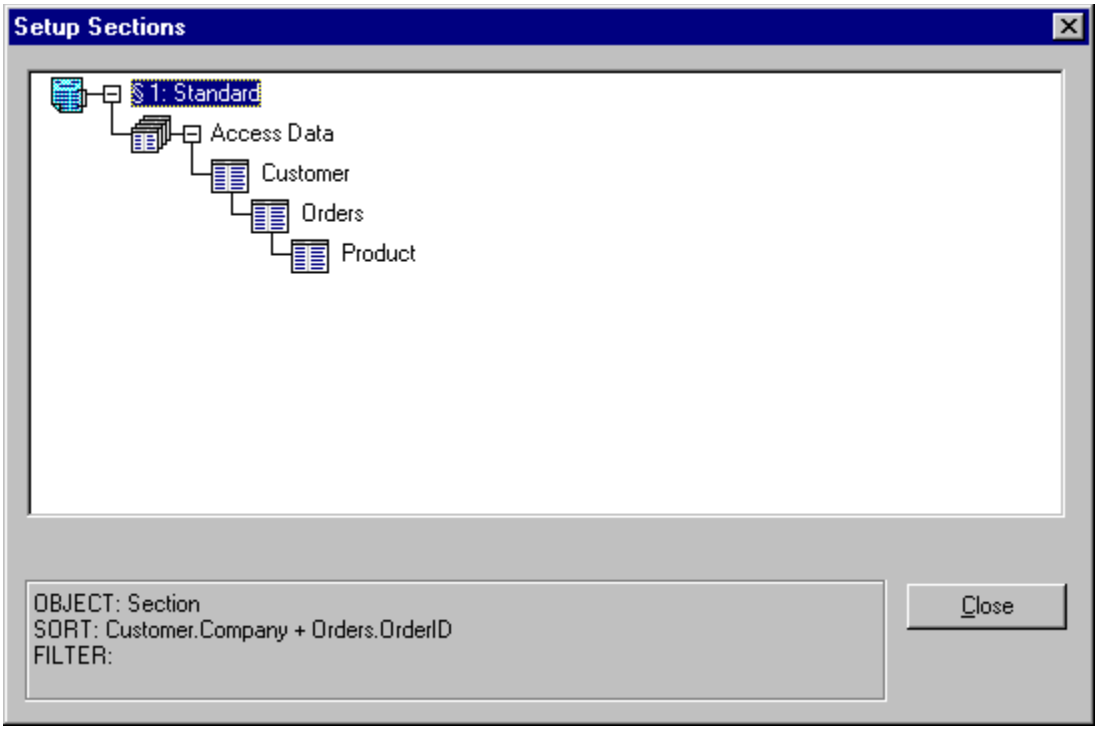


Figure 33 illustrates a typical SQL Query object that uses three SQL Tables. SQL Tables are considered attributes (rather than children) of the SQL Query object since a query is simply a collection of related tables.

Child objects are normally separate entities that are joined to a parent object by a relationship as is the case when SQL Tables are related to each other. In the figure above, note that the *Orders* table is a child of the *Customers* table.

SQL Query Options

Configuration options are available via a pop-up menu which is activated by clicking the right mouse button over the SQL Query object. Each of the options available in the pop-up menu is covered below.

Add Child Table/SQL Query

This option allows you to add a *child* local table or SQL Query to the SQL Query. If you choose this option, you will be prompted to select a data source. See "Creating a Standard Report" for more information on specifying a data source.

When you add a child table or query, ReportPro automatically defines the relationship between the parent and the child. To modify the relationship, select the *Relationship* option for the child table or query.

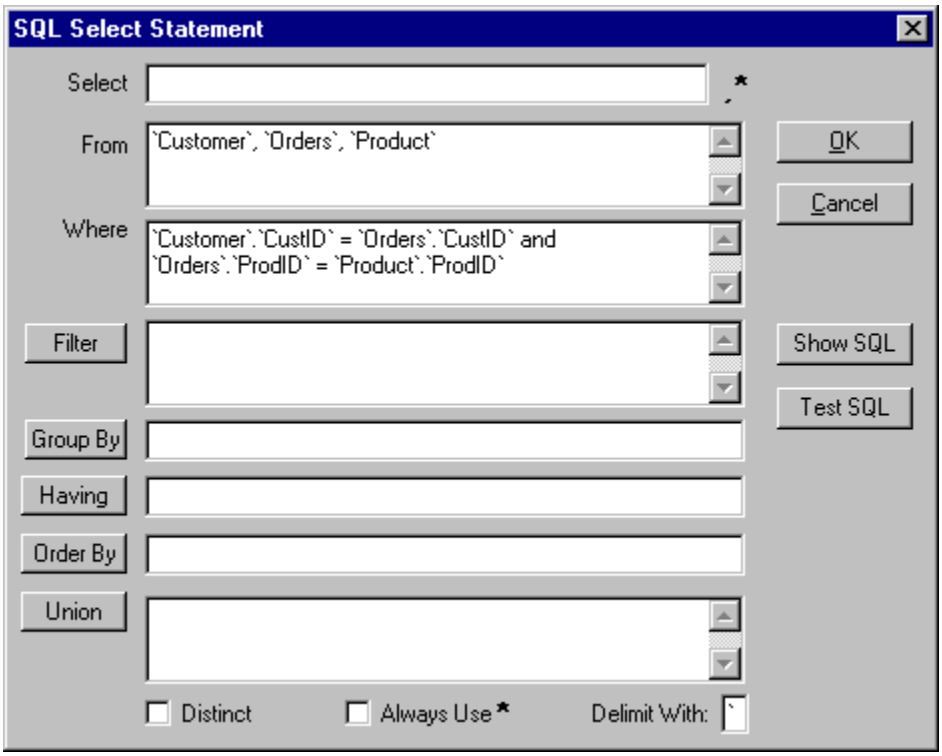
Adding a *child* table or SQL Query does not modify the SQL Query. The procedure to add SQL Tables to a query is discussed under the SQL Table options below.

Login

This option allows you to specify a login to use to connect to the SQL server. When you select this option you are presented with a dialog where you can specify a User ID and Password. The specified User ID and Password will be used to connect to the server the next time the report is opened. The current connection remains unchanged.

Setup SQL

To provide complete control over the data retrieval process, ReportPro allows you to manipulate the SQL Select statement that is sent to the ODBC driver. A discussion of SQL is beyond the scope of this manual. Furthermore, it is assumed that the reader has a basic understanding of SQL.



The SQL Select Statement dialog is broken up into sections. Each section represents a particular clause in the SQL Select statement. Each section and option is discussed below.

Select Clause	Description
Select	By default, ReportPro retrieves only columns from an SQL data source. Some data sources, however, support calculated columns which are not accessible through ReportPro's design environment. This edit allows you to retrieve calculated columns and system variables from a SQL data source. The columns you define here are accessible in a report expression by calling the RpSQLCol() function. For example, if you specify "Count(*)" here, you could access that column in a report with "RpSQLCol(1)".
From	This edit allows you to modify the SQL From clause. This information is automatically maintained by ReportPro and normally should not be modified.
Where	ReportPro splits the Where clause in two sections. This edit allows you to modify the table join specification portion of the Where clause. This section is automatically maintained by ReportPro and normally should not be changed.
Filter	This edit allows you to specify the selection specification portion of the Where clause. If you are using a single SQL Query, you should specify the filter criteria at the Section level via the *Filter* pop-up menu option. If the Section *Optimize Filter* option is selected, the Section filter is automatically passed to the SQL source. Specifying a filter at the this level is useful if you are using multiple SQL Queries in a report.
Group By	This edit allows you to modify the SQL Group By clause. This information is passed directly to the SQL source and is not used by ReportPro.
Having	This edit allows you to modify the SQL Having clause. This information is also passed directly to the SQL source and is not used by ReportPro.
Order By	This edit allows you to modify the SQL Order By clause. This information is automatically maintained by ReportPro via the *Order* main menu option or the Section's *Sort Order* pop-up menu option. If the Section *Optimize Sort Order* option is selected, the Section sort order is automatically passed to the SQL source. This section should only be modified if you can not achieve the proper results at the section level.
Union	This edit allows you to modify the Union clause. This information is passed directly to the SQL source and is not used by ReportPro.
Distinct Check box	This option forces ReportPro to issue "SELECT DISTINCT ...."
Always Use * Check box	This options forces ReportPro to issue "SELECT *" instead of explicitly stating the column names.
Delimit With	This edit allows you to change the character delimiter used for the column names. This information is retrieved from the ODBC driver and normally should not be modified.
Show SQL	This button allows you to view the SQL Select statement that is sent to the SQL data source. Note that the Section's sort order and filter information is not included in the Select statement.
Test SQL	This button sends the SQL Select statement to the SQL source and displays a dialog window that identifies if the statement executed

successfully. If execution fails, an error message is displayed. Note that the Section's sort order and filter information is not included in the Select statement/

Delete Item

This option deletes the selected SQL Query. It is important to note that when you delete the query, the children of the query are also deleted.

SQL Table Options

Configuration options for the SQL Tables are accessed via pop-up menu like all objects in the Setup Sections dialog. Each of the options are covered below.

Add Child SQL Table

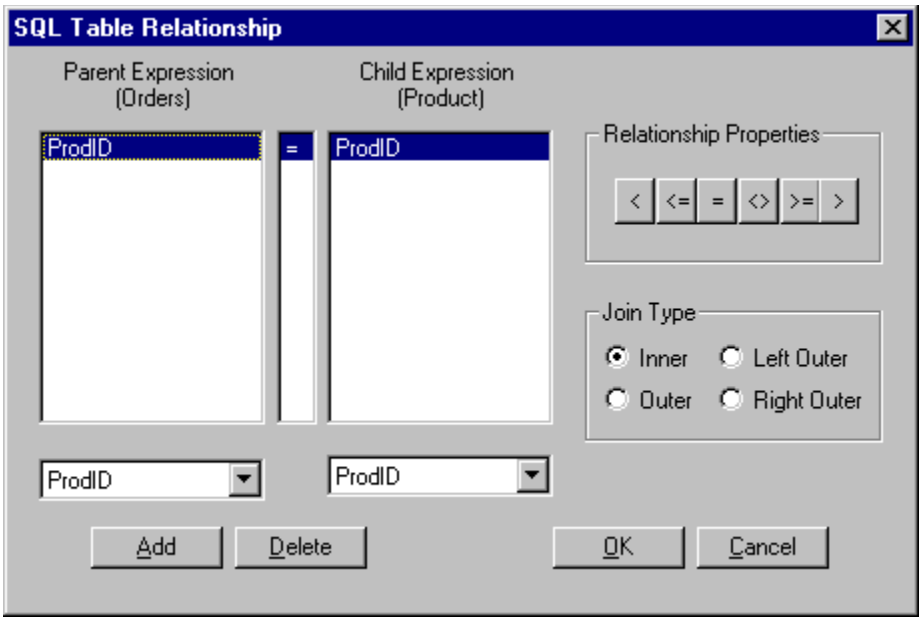
This option adds a new table into the SQL Query as a child of the selected SQL Table. ReportPro automatically creates the relationship between the parent and child table.

Delete Item

This option deletes the selected SQL Table. It is important to note that when you delete a table you also automatically delete all the children of that table.

Relationship

The SQL Table relationship implementation is different than that used for local tables since SQL hides the user from the implementation aspects of the table relationship and leaves that to the server.



The SQL Table Relationship dialog focuses on specifying conditions that relate tables rather than how they are related. The dialog contains three list boxes. The left list box holds columns from the parent table. The right list box holds columns from the child table. The center list box holds the operator that defines the relationship between the parent and child columns.

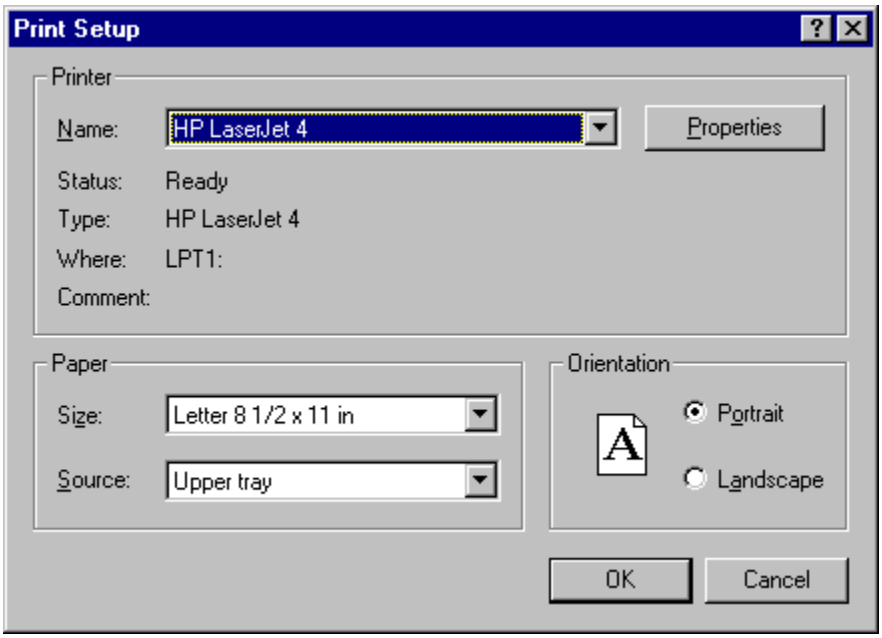
To add a new condition, click the *Add* button and a new row will be added to the list boxes. To change either the parent or child column, select the desired column from the combo box located below the appropriate list box. To change the relationship operator, click the desired button in the *Relationship Operators* group.

To delete a relationship condition, highlight the desired row and click the *Delete* button.

The *Join Type* group allows you to specify the SQL join type. This feature is server dependent and may not be supported by all servers. This feature also affects how the SQL Select statement is generated. If the *Inner Join* option is selected, ReportPro generates a SQL 1.0 compatible Select statement. Any other option causes ReportPro to generate a SQL 2.0 compatible statement.

Configuring the Printer

The Print Setup dialog is available via the *File, Setup, Printer* menu option and allows the configuration of various printer related options such as printer name, page orientation and paper size.



Each of the options available in the Print Setup dialog and how they affect your report is discussed below.

Printer

The *Printer Name* list box allows you to select which printer you wish to send the report to. The printers listed here are those that are installed on your computer.

A *Properties* button will be displayed if additional printer options are available. This feature allows access to more advanced printer options. If the *Properties* button is selected, an additional dialog will be displayed. The options available in this dialog depend on which printer you are using. Some common options include Print Quality, Color and Default Paper Source.

Paper

The Paper controls allow you to specify the paper size and source. The sizes and sources available are printer dependent.

Orientation

This option designates the page orientation. Choose from Portrait (normal) and Landscape (sideways)

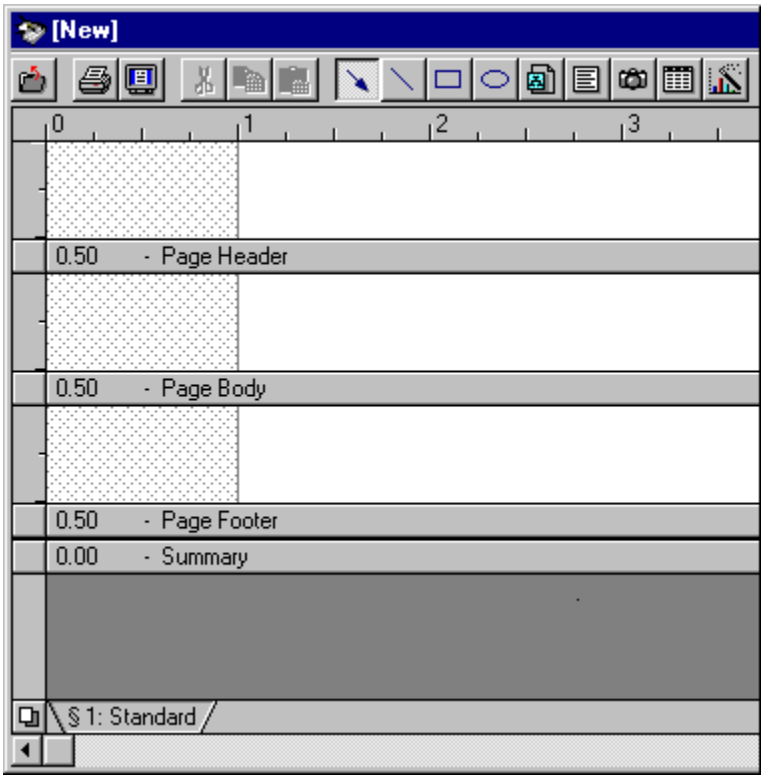
page orientations.

As discussed earlier, sections that have Header and Footer bands support their own paper attributes. This feature allows one section to print portrait while the next prints landscape.

When you modify the paper or orientation information in the Printer Setup dialog and you have multiple sections defined in your report, ReportPro will ask if you wish to apply these changes to all sections in the report. If you wish the changes to apply only to the displayed section, select *No*.

Bands

Bands are how you tell ReportPro where to print report objects. ReportPro supports seven types of bands: Background, Page Header, Group Header, Page Body, Group Footer, Page Footer and Summary. Each section that you define in your report will have its own set of bands. Some sections support all band types while others, like the Title Section, only support the Page Body and Background bands.



The Background, Page Header, Page Body, Page Footer and Summary bands are created automatically each time you create a new section. The Group Header and Footer bands are created when you define a grouping.

Background Band

The Background band prints on each page of a report and is always the same size as the printable paper area (paper height - top margin - bottom margin). Objects on the Background band print behind objects in other band types.

Only sections that have Page Header and Page Footer bands support Background bands. During printing, sections that do not have Page Header and Page Footer bands inherit the Background band from the most recently printed section that has a Background band.

To access the Background band, press the Background mode button on the lower left corner of the work

window. This button toggles the work window display between the foreground and background modes.

Page Header Band

The Page Header band is used to print objects at the top of each page of the report. The Page Header band prints just below the top margin of the report. This is a convenient location to place the report title as well as the page number and date.

Group Header Band

This band is printed when a report grouping starts. The grouping must contain data or the header will not print. This feature prevents a Group Header / Group Footer combination from printing unless a Page Body band prints.

Group Header bands useful for placing sub-title information. For example if you were creating an invoice listing report that was sorted and grouped by customer, you would probably want to place the customer name in the Group Header band so you could easily identify who the invoices belonged to.

If you create a report that has multiple levels of grouping you'll get a Group Header / Group Footer band combination for each grouping you defined.

Page Body Band

This band is printed for each record in the data source that satisfies the data source's filter criteria. The Page Body band is where you place the report objects that print the body of the report. For example if you were creating a customer listing report, you might place field objects to print the customer name and mailing address in this band.

Group Footer Band

This band is printed after a grouping ends. It is useful for printing totals based on information in the grouping.

Page Footer Band

The Page Footer band is used to print objects that appear at the bottom of each page of the report. The Page Footer band prints the height of the band above the bottom margin. For example, if the bottom margin is one inch and the band height is ½ inch, the band would start printing 1½ inches from the bottom of the page. The Page Footer band is a convenient location to place your company name and report page number.

Summary Band

The Summary band prints after all data associated with its section has been printed. It prints after the Page Body and Group Footer bands but before the last Page Footer band. The Summary band is designed to allow you to print totals and summary information for the data in the data source.

Configuring Bands

Each band type has certain options that affect how and when they are printed. The options available depend on the band type. The band options are accessible by clicking the right mouse button over the band or the band's button on the band bar. Each band option is discussed below.

Force Page Eject

This option is a toggle, that when selected, causes ReportPro to eject a page after the band prints. This feature is useful if you wish the next band to print at the top of a page. This option is not available for the Page Header and Page Footer bands.

Force Eject When...

This option allows you to specify a conditional page eject. When you select this option, you are presented with the Expression Builder to allow you to enter the expression that will be used to determine if the page should be ejected.

After all objects in the band are printed, the expression is evaluated. If the expression returns true the page is ejected; otherwise printing continues normally. This option is not available for Page Header and Page Footer bands.

Print When...

This option allows you to set a condition for which the band will be printed. When you select this option the Expression Builder is displayed. The expression you define is evaluated before the band is printed. If the expression evaluates to true the objects in the band are printed; otherwise the band is skipped.

Fixed Size

This option forces the band to occupy the same size on the report as it does in the work window. By default, ReportPro dynamically sizes bands during printing based on the objects in the band. Normally the printed band height is the height of the largest object plus the distance between the band bottom and the bottom of the lowest object in the band.

When you set the Fixed Size option the band is always the same size regardless the size of the objects in the band. Be careful when using this feature with field objects that have the *Vertical Size* option selected. It is possible to have an object print outside its band and into another band's print area.

Skip if Empty

This option causes the band to be skipped altogether if no information is printed in the band. A band is

considered empty when none of the objects in the band print. A band is not considered empty if a field or text object prints an empty string of characters even though it may look as if nothing was printed.

Print On Every Page

This option is only available for Group Header bands. It forces the Group Header band to be printed on every page of its section. This feature is useful when a group contains a large amount of information and spans several pages. By printing the Group Header on every page you can easily determine the group you are looking at without having to flip through several pages of information to find the Group Header information.

Select All In Band

This option is an editing feature. When you select this option, ReportPro will automatically selected each object in the band. This feature is useful when you wish to set an attribute for all objects in a band.

Set Band Size

This option allows you to manually set a band's height. This feature is useful when you want to size a band to an exact height.

You can also set a band's height by pressing the left mouse button over the button directly adjacent to the band and while holding down the mouse button, sliding the button to the desired height.

Drawing Tools

With the exception of the Pointer Tool, ReportPro's Drawing Tools are used to create the report objects which generate the printed output. ReportPro provides several different types of drawing tools, one for each object type. To select a drawing tool, use the *Tools, Drawing Tools* menu or select the tool directly from ReportPro's toolbar.

A report object is created by selecting the proper drawing tool and using the tool to draw the object on the report surface. The object is drawn by pressing the left mouse button at the desired origin of the object, then dragging the mouse until the desired object size is reached while still holding down the left mouse button. When the left mouse button is released, the object is created at the size and location you specified.

In some cases, ReportPro will prompt you for additional information before the object is created. For example, when you create a field object, ReportPro prompts for the expression used by the field object to extract data from the data source.

Each of the drawing tools supported by ReportPro is discussed below.

Pointer Tool

The Pointer Tool allows you to manipulate other report objects. The following table lists the tasks that can be performed with the pointer tool. It also provides a brief description on how to perform the task.

Function	Description
Selecting an Object	To select an object, point at the object with the mouse and click the left mouse button.
Selecting Multiple Objects	To select multiple objects, first select an individual object. Next select the other objects while holding down the Shift key. Multiple objects can also be selected by drawing a rectangle area with the pointer. All objects touched by the rectangle will be selected.
Moving	To move an object, point the pointer tool at the object and drag the object while holding down the left mouse button. It is also possible to move multiple objects.
Sizing	To size an object, select the object you wish to size. Then point the pointer tool at the sizing boxes located on the object's bounding box. Stretch the object to the new size while holding down the left mouse button.

Line Tool

The Line tool is used to add straight lines to the report.

Rectangle Tool

Use the Rectangle tool to add squares and rectangles to your reports.

Ellipse Tool

Use the Ellipse too to add circles and ellipses to your reports.

Field Tool

Use the Field tool to create objects that retrieve information from your databases. You can also use the field objects to access user defined variables.

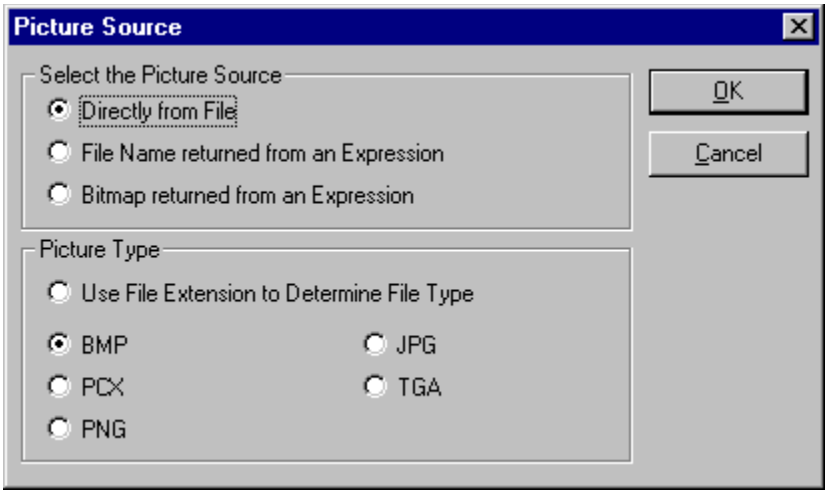
After you draw the field object on your report, ReportPro's Expression Builder appears. For more information on expressions, see the chapter entitled "ReportPro's Expression Builder".

Text Tool

To add fixed text to a report, use the Text Tool. After you draw the text box, the text object will be displayed in the *Edit* mode. Type the desired text into the text object. When you select a menu option or another report object, the text object will exit the edit mode. To return to the edit mode, select the Text Tool, point the mouse pointer at the text object and click the left mouse button.

Picture Tool

To add images to your report, use the Picture Tool. After you draw the picture object, you are presented with a series of dialogs that allow you to specify the source of the image.



ReportPro can retrieve the image from several different sources:

Image Source	Description
Directly From File	This option allows you to specify a file name to retrieve the image from.
File Name returned from an Expression	With this option, you specify an expression that returns file names. This feature is useful if the file names of the images are stored in a database. If you place the image in the Page Body band, this would result in a different image being printed for each record in the database.
Bitmap returned from an Expression	The feature allows you to display *Bitmap* images that are stored in a database.

ReportPro supports a variety of image formats including BMP, PCX, PNG, JPG and TGA. If you choose to load the image from a file, you can have ReportPro automatically detect the image format based on the file extension. If you know the image type, or if you don't use standard file extensions, you should specify the proper image type.

The only image format that is currently supported when an image is returned from an expression is BMP.

Depending on the Picture Source you choose in the Picture Source dialog, you will either be prompted to specify a file name or an expression which will be used to retrieve the image.

Embedded Section Tool

This tool allows you to create an Embedded section. An Embedded section is a section that prints within a rectangular area inside another section. Embedded sections are useful for placing Cross-Tab or look-up type information along side data from another section.

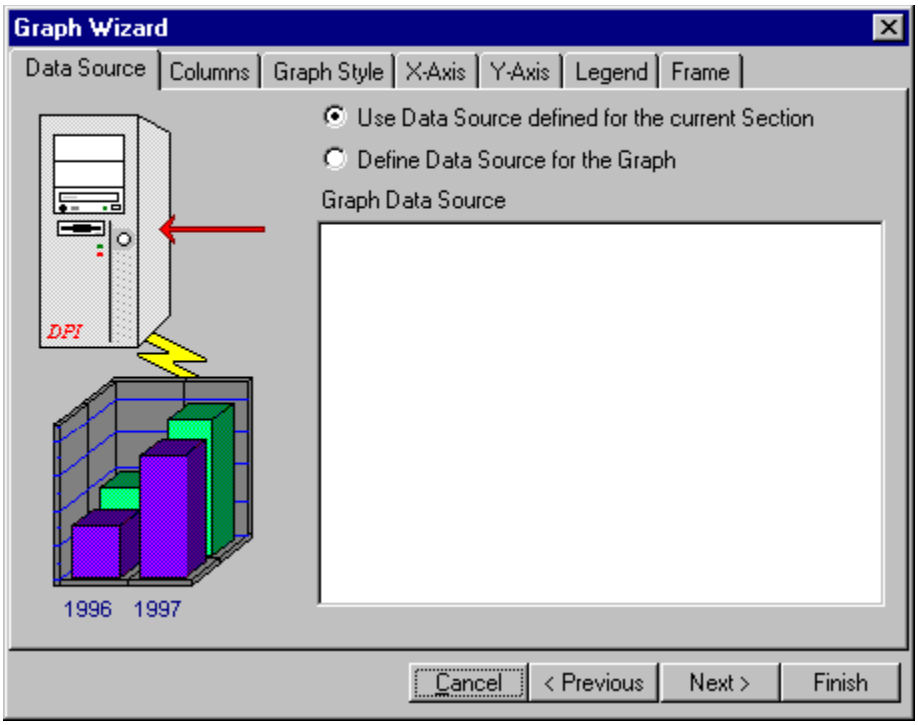
After you draw the bounding box of the Embedded Section, you will be prompted to specify information about the section. This information includes the section type and data source.

After you specify the section information, the section is created on the report surface. A new section tab is also added to the work window's section tab bar. You cannot edit the Embedded section object from the parent section. You must first select the Embedded section by clicking on it's tab located on the Section Tab Bar. Once the section is selected, you can manipulate the section like any other report section.

Graph (32-bit version only)

The Graph tool allows you to create stunning graphs on your report. After you create a graph object, the Graph Wizard is displayed. The Graph Wizard is a tabbed dialog that allows you to specify all the attributes of a graph. You navigate through the Graph Wizard dialog by clicking on the desired tab at the

top of dialog or by pressing the *Next* > and < *Previous* buttons.



Data Source

The first step in creating a graph is to define where the graph will get its information. Graphs can retrieve information from the data source of the section where they reside or they can have their own data source.

It is important to correctly define the data source. Otherwise, you will not be able to create the proper column expressions. To use the data source of the current section, select the *Use Data Source defined for the current Section* radio button.

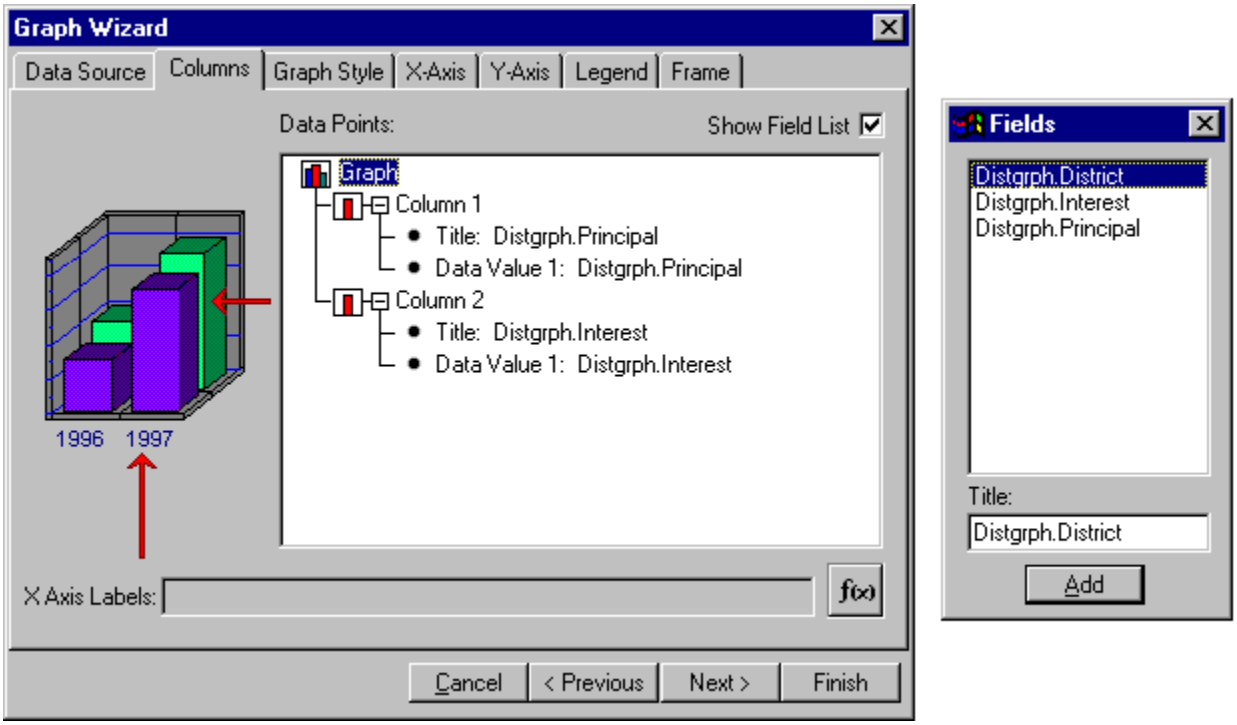
If you wish to define a data source for the graph, select the *Define Data Source for the Graph* radio button. When you do so, you will be prompted with a series of dialogs that allow you to specify data source information. This procedure is similar to that of adding a new section to your report.

After you have identified the data source, the list box in the middle of the dialog will be updated to reflect your selection. This list box operates in the same manor as the one used in the Setup Sections dialog. You add tables and queries to the data source via pop-up menus that are activated by clicking the right mouse button. Refer to the chapters entitled "Configuring Local Tables" and "Configuring SQL Queries" for a detailed description of each menu option.

Columns

The *Columns* Tab allows you to define the expressions that are used to retrieve information from the data

source. You define the graph columns by adding column objects into the hierarchical list box shown in the figure 39.



There are two ways to add columns to the graph. If you click the *Show Fields List* check box, ReportPro will display a floating dialog that contains a list of all the fields available from the data source. You add columns by selecting the desired field in the field list and pressing the *Add* button. The *Title* edit at the bottom of the floating dialog allows you to specify a title for the selected field.

You can also add columns by clicking the right mouse button over the Graph icon in the *Data Points* list box and selected *Add Column* from the pop-up menu. When you select this method you will be prompted with the Expression Builder where you can specify more complicated expressions.

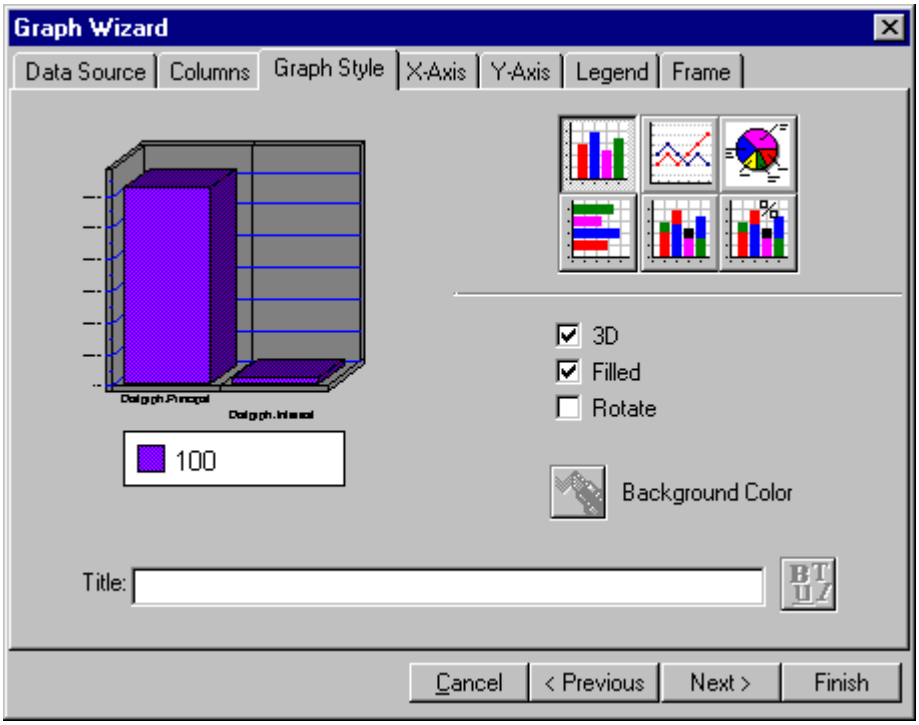
Once you have added a column to the graph, you can access options related to it by clicking the right mouse button over the desired column icon. Available options include *Insert Column*, *Delete Column* and *Add Data Point*. You can also click the right mouse button over the column title and data value expressions to edit them.

To define an expression that retrieves the x-axis labels from the data source, click on the **f(x)** icon and the Expression Builder will be displayed to allow you to specify the desired expression. The information returned by this expression will be printed on the x-axis to identify each of the columns.

Graph Style

The Graph Style tab allows you to define the attributes that relate to the entire graph. You'll notice that the remaining tabs display a sample graph in the upper left side of the dialog. This sample grid updates

every time you make a change to the graph to give you an indication of what your graph is going to look like.



The buttons in the top section of the Graph Style dialog allow you to set the graph type. ReportPro supports Bar, Line, Pie, Horizontal Bar, Stacked Bar and Stacked Percentage bar graphs.

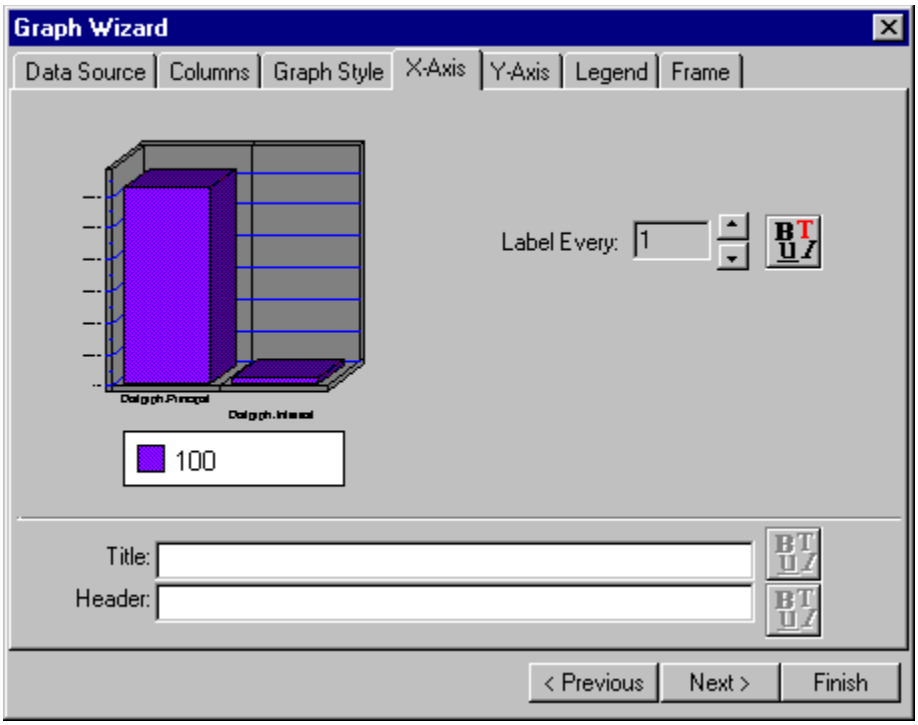
The *3D* check box toggles the graph between 2D or 3D modes. The *Filled* check box fills the bars of the graph when checked otherwise the bars are transparent. The *Rotate* check box swaps the information on the x-axis with the information in the legend.

The *Background Color* button sets the background color of pie charts. This feature is not available for the other graph types.

The Title edit allows you to specify the report title. The report title displays at the very top of the graph. The button to the right of the title edit allows you to specify a font for the graph title.

X-Axis

The X-Axis tab allows you to specify attributes related to the x-axis.

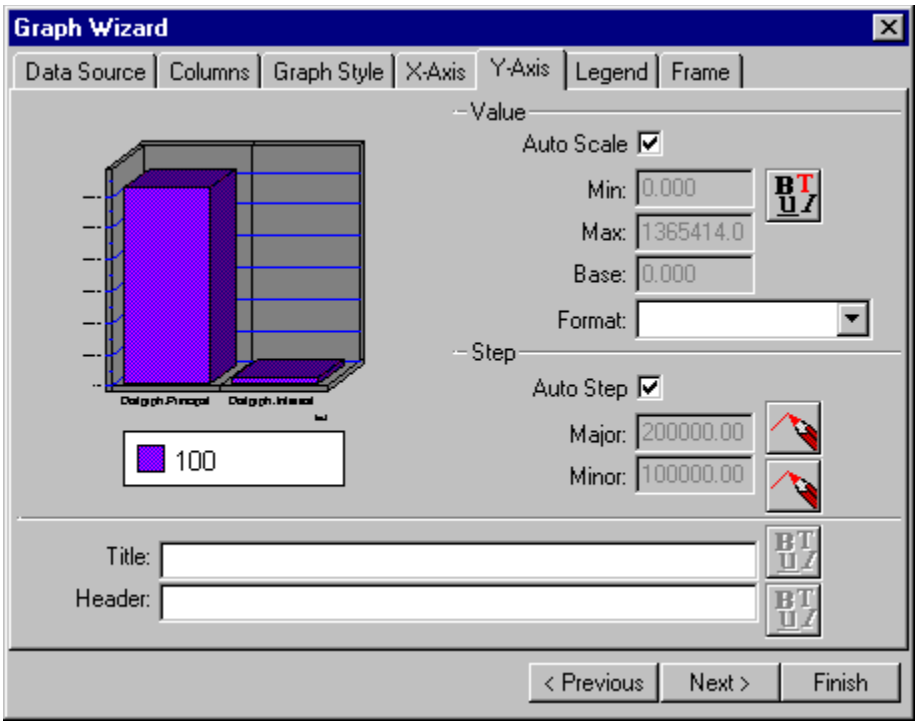


The *Label Every* spinner allows you to specify the interval at which the labels are displayed. The default interval is one which means a label is displayed for each column in the graph. If you set the interval to two a label will be displayed for every other column. The *Label Font* button allows you specify a font for the label text.

The *Title* and *Header* edits allow you to display descriptive information on the x-axis. The title displays directly below the x-axis while the header displays on the right side of the x-axis. Each edit has an associated font button to allow you to identify the font to use for the title and header.

Y-Axis

The Y-Axis tab allows you to specify attributes related to the y-axis of the graph.



The Value section allows you to control the range of values displayed on the y-axis. The Graph object has an auto scale feature that automatically determines the y-axis scale values. Normally you would leave this option checked. If you wish to manually configure the y-axis range information, uncheck the *Auto Scale* check box and the remaining edits in the value section will become enabled.

The *Min* value edit allows you to specify the minimum value that is displayed on the y-axis. The *Max* value sets the maximum value displayed. The *Base* edit sets the value where the columns are drawn from.

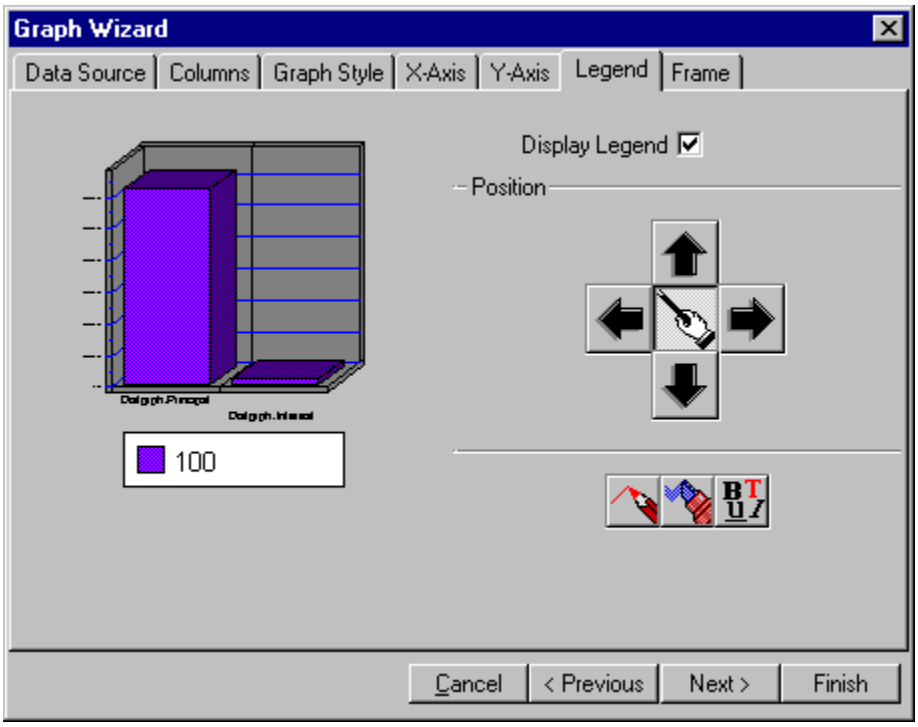
The *Format* combo box allows you to specify a picture clause to format the y-axis values with. For more information on picture clauses refer to Appendix A. The *Value* font button allows you to specify a font for the y-axis values.

The *Step* section allows you to control the resolution of the graph grid. The graph displays both major and minor grid lines. These edits allow you specify the interval at which the grids lines are drawn. These edits are normally disabled and the Graph object automatically calculates the best grid resolution. If wish to manually set the grid resolution, you must uncheck the *Auto Step* check box. The color buttons next to each edit allows you to define the color of the grid lines.

The *Title* and *Header* edits allow you to attach descriptive information on the y-axis. The title displays vertically beside the y-axis values while the header displays at the top of the y-axis. Each edit has an associated font button to allow you to identify the font to use for the title and header.

Legend

The Legend tab allows you to control the attributes of the graph legend. From here you can specify the position, color and font of the legend.



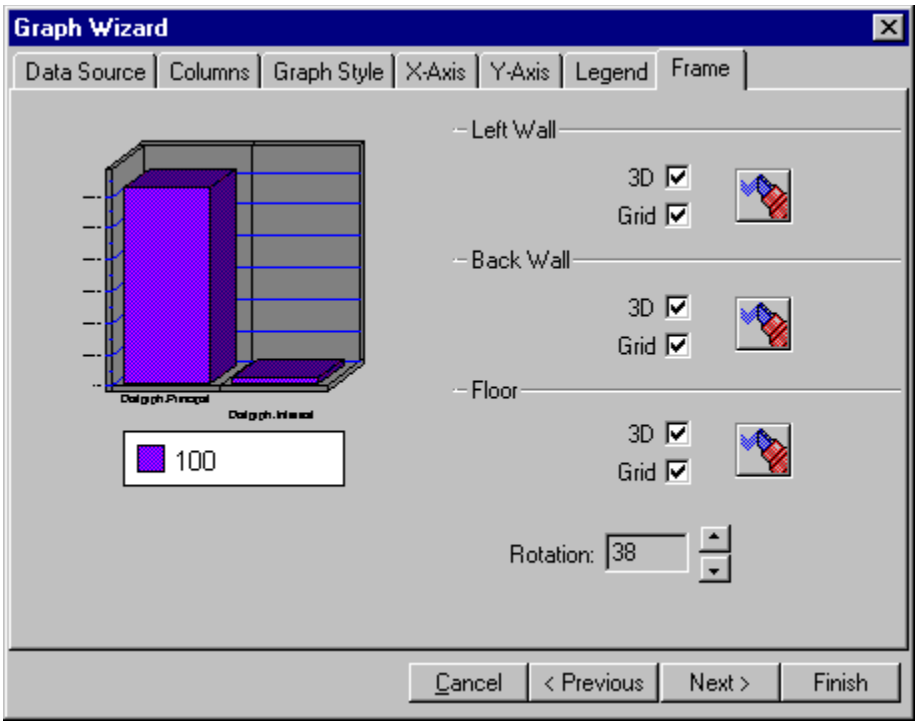
The Display Legend check box allows you to toggle the display of the legend. If this option is unchecked, the legend will not display.

The position buttons allow you to control where the legend is displayed. To move the legend, click the desired arrow. The button in the middle allows the Graph object to automatically calculate the best position for the legend.

The three buttons at the bottom of the dialog allow you to specify the border color, fill color and font respectively.

Frame

The Frame tab allows you to set the attributes related to the frame of the graph. The controls are broken into three sections, one for each of the three walls.



The *3D* check box causes the Graph object to draw a three dimensional wall. The *Grid* check box toggles the display of a grid on the wall. The color button allows you specify the color of the wall.

The *Rotation* spinner allows you to rotate the graph to modify the user's perspective of the graph.

Configuring Report Objects

Each report object supports various attributes to allow you to customize the appearance of the report output. Several attributes are supported by all objects while some are peculiar to a specific object type. For example, all objects support alignment and size but only Text and Field objects have font attributes.

You configure an object by clicking the right mouse button over the desired object. A pop-up menu will be presented to allow you to select the attribute you wish to configure.

You'll notice that the pop-up menu also contains the editing commands *Cut*, *Copy* and *Paste*. These commands are provided here for convenience and have the same affect as the *Edit* menu options on ReportPro's main menu.

This chapter starts by discussing the attributes that are common to all objects and then discusses those attributes that apply to specific objects.

Align

Set Size/Position

Background Style

Border Style

Print When

Shadowing

Line Style

Auto Size

Font

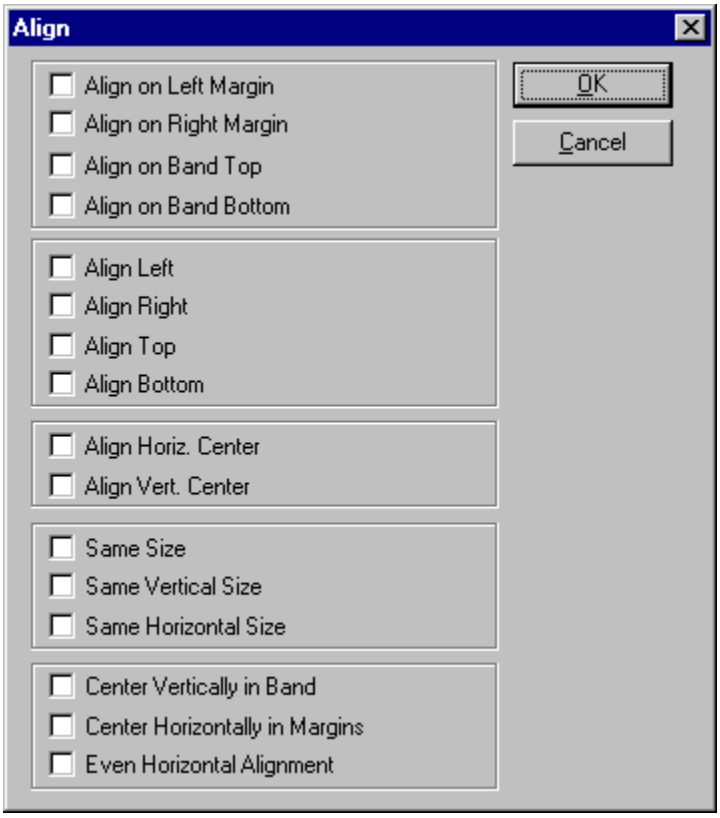
Text Align

Field Style

Picture Style

Align

When you select the *Align* menu option, you are presented with the Align dialog where you can select the type of alignment you wish to perform.



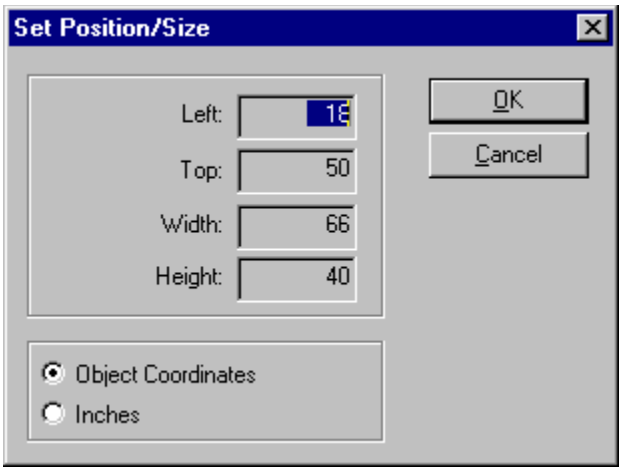
Using the Align dialog, you can align objects relative to the report margins and band positions. You can also center an object in its band.

If multiple objects are selected when you choose *Align*, you can align and size the objects relative to each other. This is a very nice feature when you want to left align several objects or when you want multiple objects to have the same dimension.

To perform the alignment procedure, check the desired option(s) using the mouse and press the *OK* button.

Set Size/Position

This option allows you to explicitly specify the size and location of an object. Normally you use the Pointer tool to stretch and drag an object to its desired size and position, but in some cases you may wish to specify an object's size and position with greater precision. The Set Position/Size dialog allows you to do so.

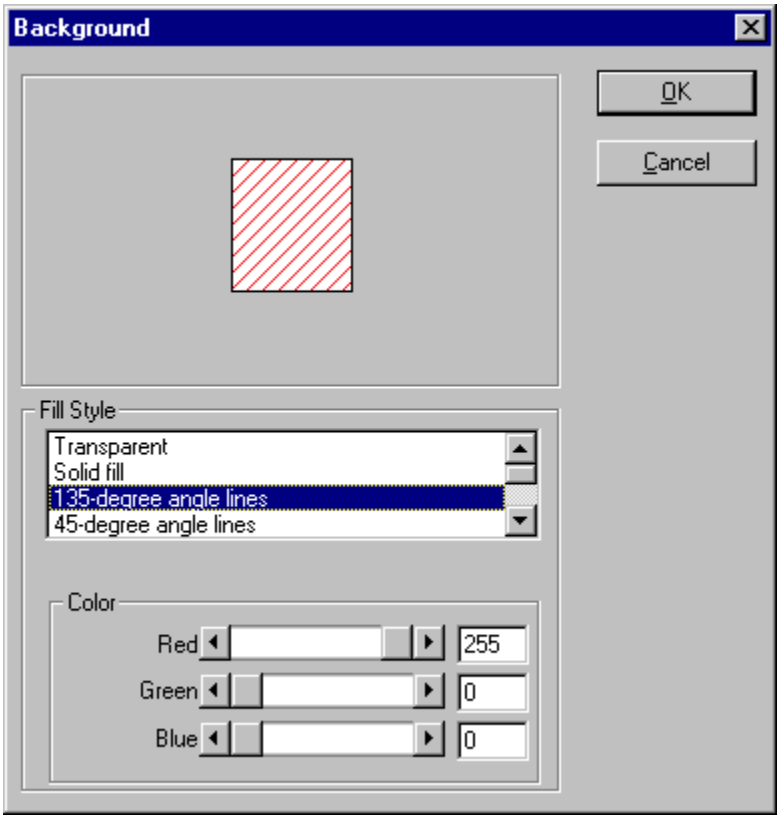


The location and size of the object can be specified in either object coordinates or user coordinates. Object coordinates are what ReportPro uses internally when printing and displaying objects. Object coordinates are always integer values so that there are no rounding problems. There are roughly 96 object units per inch but this can vary depending on your video mode.

User coordinates are either in inches or centimeters depending on how your report is configured. If you specify an object's position or location in user coordinates, ReportPro automatically converts these values to object units so some minor rounding errors may occur. Also it is possible for you to specify a value in user coordinates and later determine that the object position isn't exactly what you specified. This is a also due to rounding errors.

Background Style

All report objects except the Line object support background attributes. When you select the Background pop-up menu option, you are presented with the Background dialog.

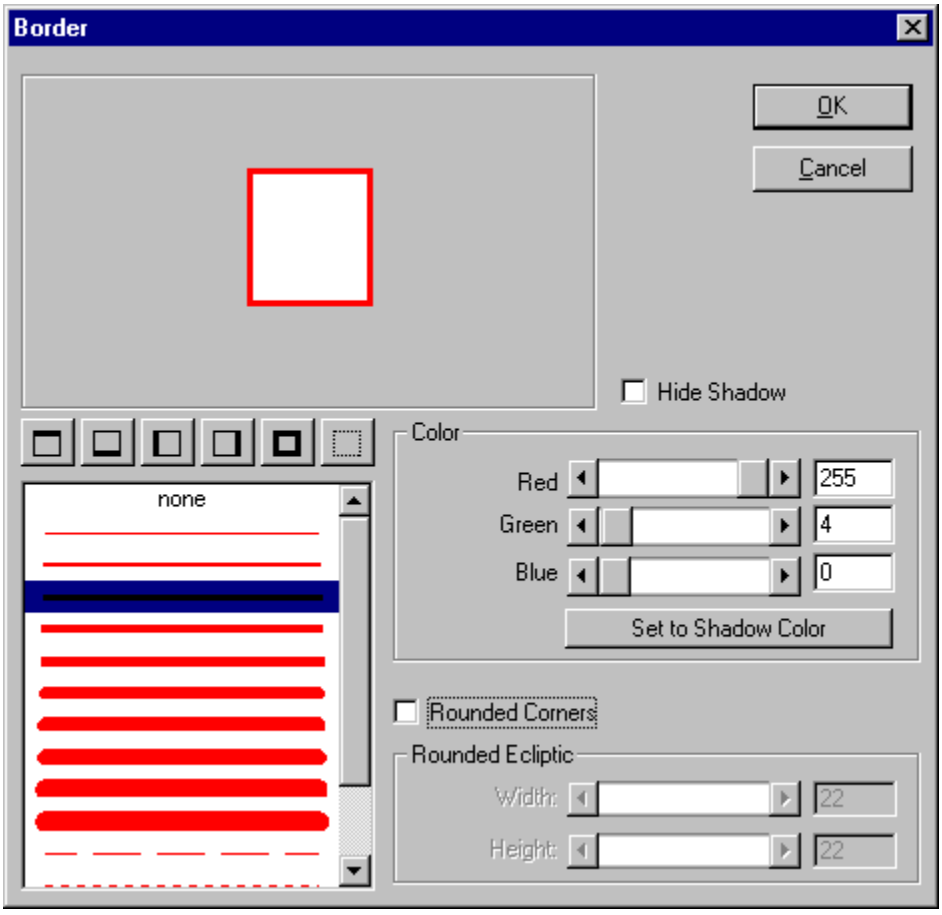


The Background dialog allows you to specify how the background of an object is drawn on the report. The dialog presents an example of the object you are configuring at the top of the dialog.

In the Background dialog you can specify the pattern used to paint the background and you can also specify the background color. To select a *Fill Style,* select the desired pattern from the *Fill Style* list box. To set the color, use the *Red*, *Green* and *Blue* scroll bars to select the desired color. Alternately you can type the color value directly into the edit boxes next to the color scroll bars.

Border Style

You can customize the border around a report object with the Border dialog.



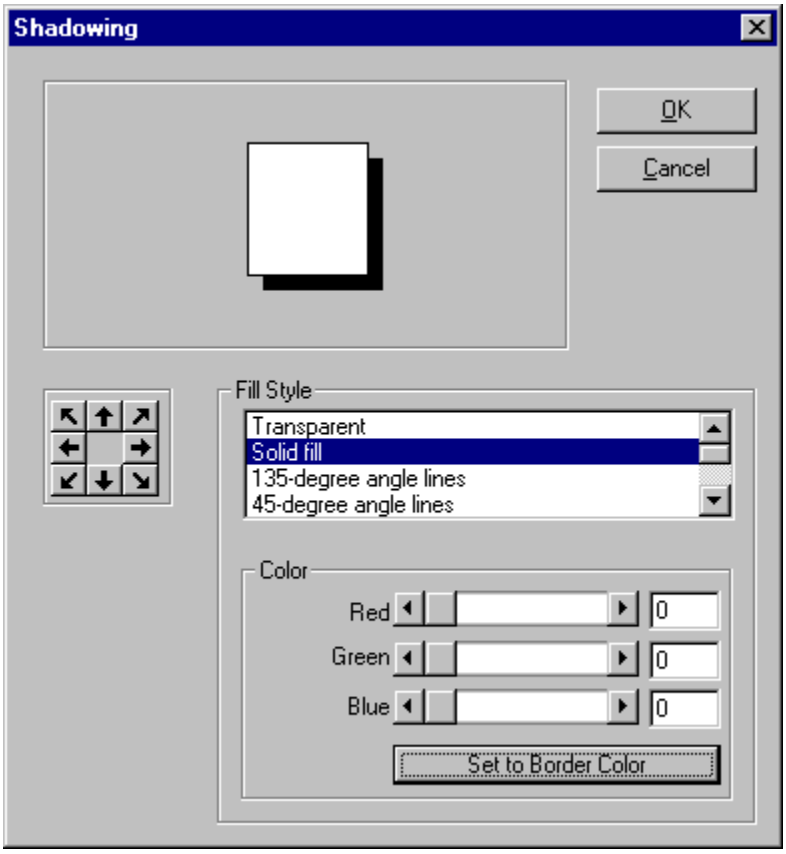
From here you can set the line style, thickness and color of each side of an object's border. To change a border setting, select the desired line style in the list box, set the desired color using the color scroll bars and then press one of the border buttons below the sample object. You can optionally specify that the object uses rounded corners for the border by selecting the *Rounded Corners* check box.

Print When

The Print When attribute allows you to specify an expression that is evaluated before the object is printed on the report. If the expression returns true, the object is printed, otherwise, the object is not printed. When you select this option, the Expression Builder is displayed prompting you for the expression.

Shadowing

The Shadowing dialog allows you to define a shadow for the report object. Shadowing allows you to give your reports a 3D effect.

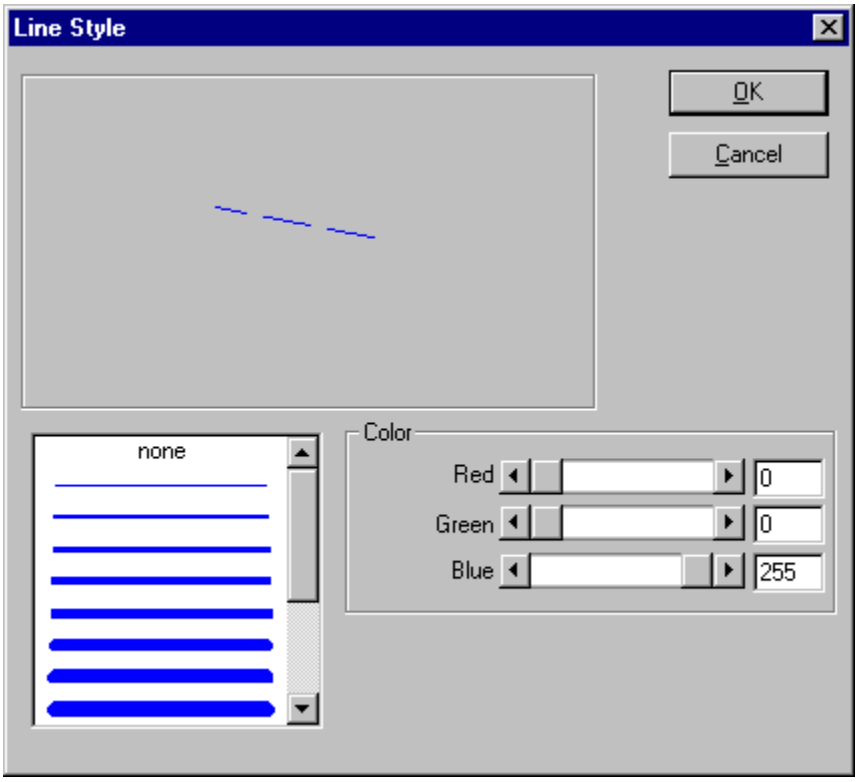


The shadow can be placed on all sides of the object and its width can also be adjusted. You can also specify the fill type and the color of the shadow. To specify a shadow, select the fill type and then use the arrow buttons to move the shadow to the desired position.

Note: If the shadow fill style is transparent, you will not able to see it on the sample object. It will appear as if the positioning buttons have no effect, but in reality they do, you just can't see the shadow.

Line Style

This option is available only for Line object. It allows you to customize how the Line object is drawn.



You'll probably notice that the Line Style dialog has many of the same controls as the Border Style dialog. The functionality of both dialogs is basically the same. You select the desired line thickness and style from the Line list box and adjust the line color using the color scroll bars or edit controls.

As with the border styles, the solid style is the only style that supports a thickness greater than one.

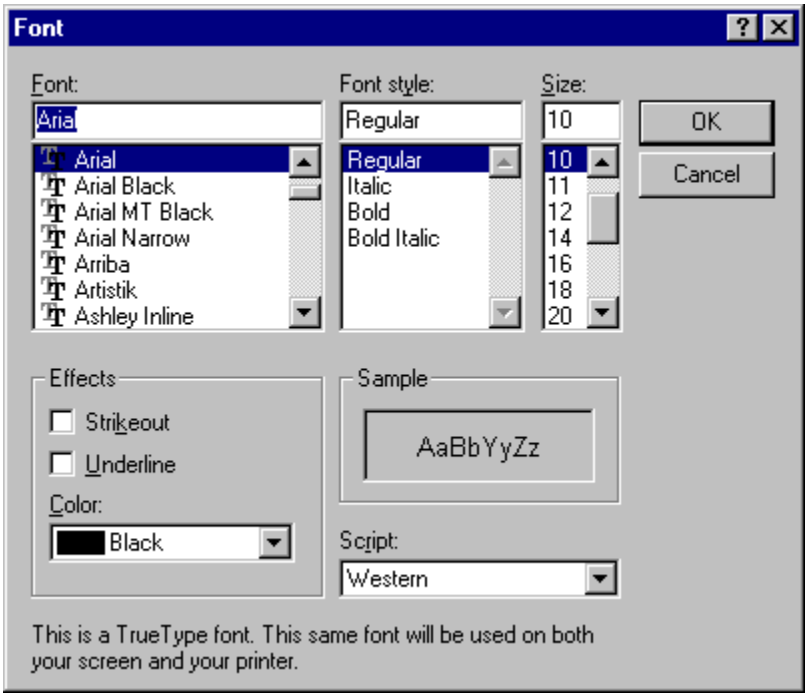
Auto Size

Auto size is a command that is available for Field and Text objects. When you select this option, ReportPro automatically sizes the object so it can properly display the text associated with the object.

For Text objects this procedure is very straight forward, but for field objects you may get unwanted results with Auto Size. The reason for this is that ReportPro only has one row of data available to use as a sample. It is possible that the one row of sample data doesn't correctly represent the data that will be encountered during reporting. This problem is most apparent when working with memo fields.

Font

Both Field and Text objects support font attributes. When you select this option, the Font dialog is displayed.



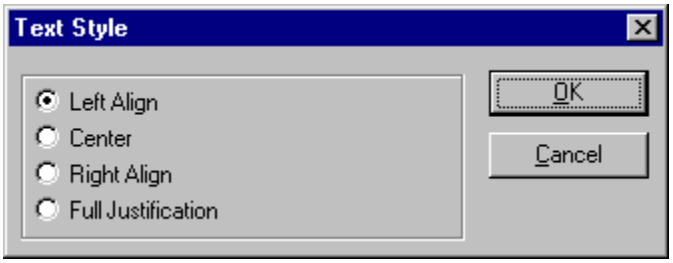
In the Font dialog you can specify the font, font style and size. You can also specify strikeout and underline effects here. The Font dialog is also where you specify the color of the text.

ReportPro supports all True Type and printer fonts. True Type fonts provide the best mapping between ReportPro's print preview and the printer and therefore are recommended in most cases.

Printer fonts are provided because they can be faster and have less overhead than True Type fonts during printing. Printer fonts however do not scale well in the print preview mode and do not provide a very high level of WYSIWYG.

Text Align

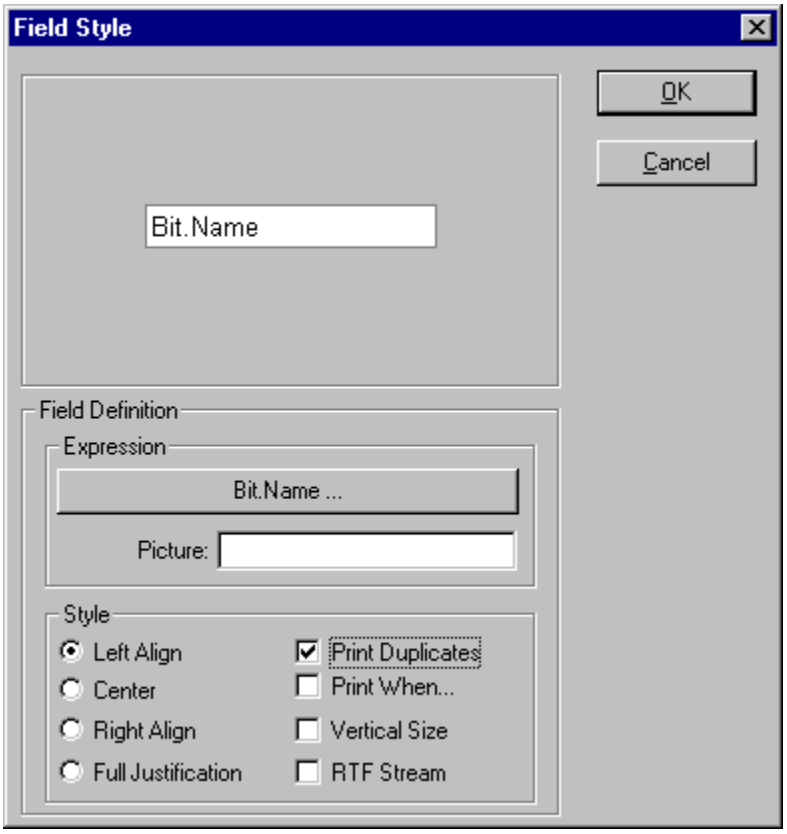
This option allows you to specify how the text of a Text or Field object is aligned within the bounding box of the object.



ReportPro can align text on the left, center and right side of the object's bounding box. ReportPro can also fully justify the text.

Field Style

This menu option is only available on the Field object pop-up menu since it allows you to specify attributes that are specific to field objects.



From the Field Style dialog, you can modify the expression used to retrieve information from the data source, specify a picture clause, set the text alignment and specify other miscellaneous options that affect how the field object prints.

To modify the Field object's expression, click the large button in the Expression group box. Doing so will display the Expression Builder where you can modify the expression.

The *Picture* clause is useful for formatting the value returned by the field object's expression. For example, if the expression returns a numeric value that you wish to format as currency, you would specify a picture clause of **"$ 999,999.99"**. There are many different types of picture clauses which you can use. Refer to Appendix A for a complete listing of the picture clauses supported by ReportPro.

In the *Style* section, you can set the alignment of the Field object's output. This is the same setting that is available in the Text Align menu option. It is included here for convenience.

Printing sorted records can produce rows of text with the same information (duplicates) on them. The *Print Duplicates* check box allows you to control how duplicate values are printed. For example,

consider the following output:

Salesman	Closed Date
GAK	12/10/96
GAK	12/15/96
LLA	12/10/96
LLA	12/15/96
LLA	12/17/96

The sorted Salesman field contains duplicate records of information. "GAK" is duplicated once and "LLA" is duplicated twice. The following output is the same except that the *Print Duplicates* option is not selected for the Salesman field:

Salesman	Closed Date
GAK	12/10/96
	12/15/96
LLA	12/10/96
	12/15/96
	12/17/96

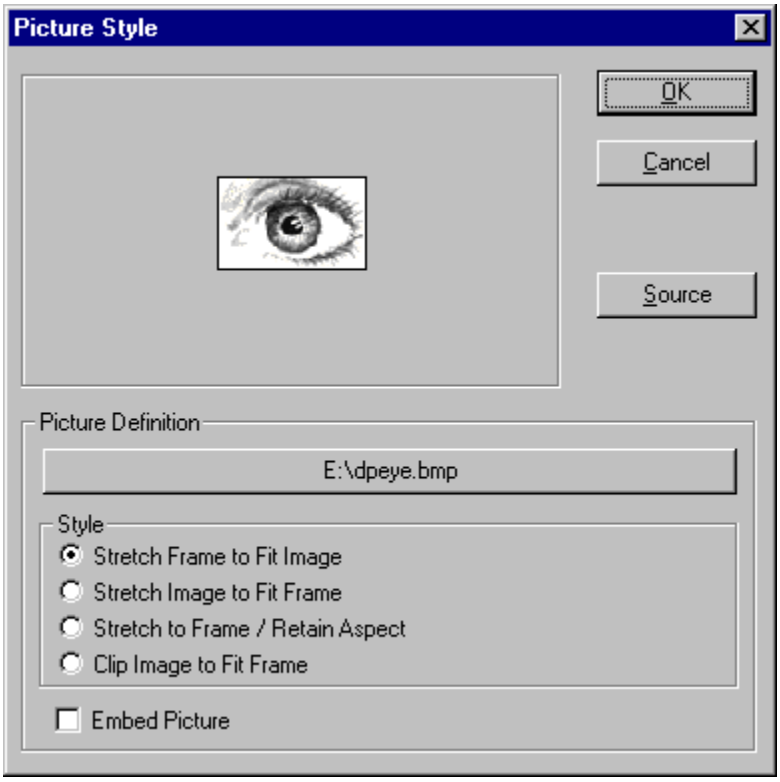
The *Print When* check box allows the object to be conditionally rendered. This is the same option that is available via the *Print When* menu option discussed earlier. It is included here for convenience.

The *Vertical Size* option causes the Field object to re-calculate its height each time it is printed. The Field object height can decrease to zero or increase in size to span multiple pages. This is option is useful for printing variable length memo fields. If you select this option, it is recommended that you do not select the *Fixed Size* band option for the band where the Field object resides.

The *RTF Stream* option informs the field object that the string returned by the Field object's expression is a RTF stream. You can use this option to embed RTF formatted text into your report. The RTF stream must be a valid RTF 1.0 stream or it will not print. Since the stream is retrieved from an expression, the source of the stream can be a memo field or the return value of a function. Currently, only a subset of the RTF standard is supported. Supported features include: mixed fonts, text colors and paragraph formatting.

Picture Style

The Picture Style dialog allows you to configure attributes associated with the Picture object.



The large button in the *Picture Definition* section has a dual purpose depending on the source of the picture. If the picture was retrieved from a file, pressing the button displays the File Open dialog where you can select another image file. If the picture was retrieved as the result of an expression, pressing the button displays the Expression Builder where you can modify the expression.

The image *Clipping Style* can also be specified as follows:

- *Stretch Frame to Fit Image* - This option causes the frame of the Picture object to be resized to match the size of the image.
- *Stretch Image to Frame* - This option will resize the image to match the Picture object dimensions you specified when you created the Picture object.
- *Stretch to Frame / Retain Aspect* - This option will resize the image to match the Picture object dimensions but maintains the image aspect ratio so the image isn't distorted.
- *Clip Picture to Fit Frame* - Choosing this option will cause the image to be clipped to the picture object dimensions.

You can also choose to *Embed the Picture* in the report file so that the report is not dependent on the image file being available when the report is run. Use this option when you cannot guarantee that the image file will exist on the system where the report will be printed.

The *Source* button allows you to redefine the source of the image. The dialog displayed when you press this button is the same one encountered when creating a Picture object. See the chapter entitled "Drawing Tools" for more details.

Field Wizard

The Field Wizard allows you to quickly add Field objects to a report. The Wizard is accessible via the *Tools* menu option on ReportPro's main menu. You can also access the Field Wizard by using the hot key combination *Ctrl-F5*.

Field objects are added to a report by clicking the left mouse button over the desired field in the Field Wizard, then dragging the field to the desired position on the report while still holding down the left mouse button. When you release the left mouse button, the object is added to the report.

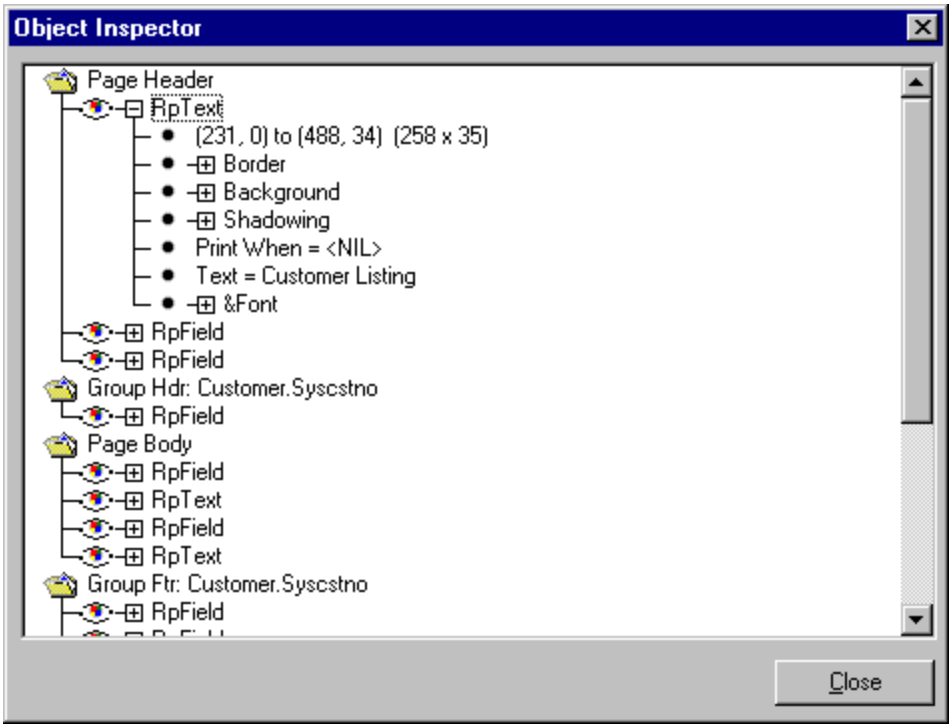
You can restrict the fields displayed in the list box to a specific table by selecting the desired table in the *Table* combo box. To display fields for all the tables, select the *<All Tables>* option.

You can also browse sample values of a field by highlighting the desired field and pressing the *Browse Field Values* button.

Object Inspector

The Object Inspector gives you a hierarchical view of the report objects contained in the current report section. From the Object Inspector, you can see what band each object belongs to and where they reside in the Z-order. You can also see each of the object's attributes and the value of each attribute.

To display the Object Inspector select the *Tools, Object Inspector* option from ReportPro's main menu.



There are three icons that you will encounter in the Object Inspector:

	Represents a Band.
	Represent a Report Object.
	Represents an attribute of a Report Object.

When you click the right mouse button on a report object, a pop-up menu will appear with a list of features available for the object. The options presented in the pop-up menu depend on which object you've selected. For a detailed description of each menu option, please refer to the chapter entitled "Configuring Report Objects".

Grouping

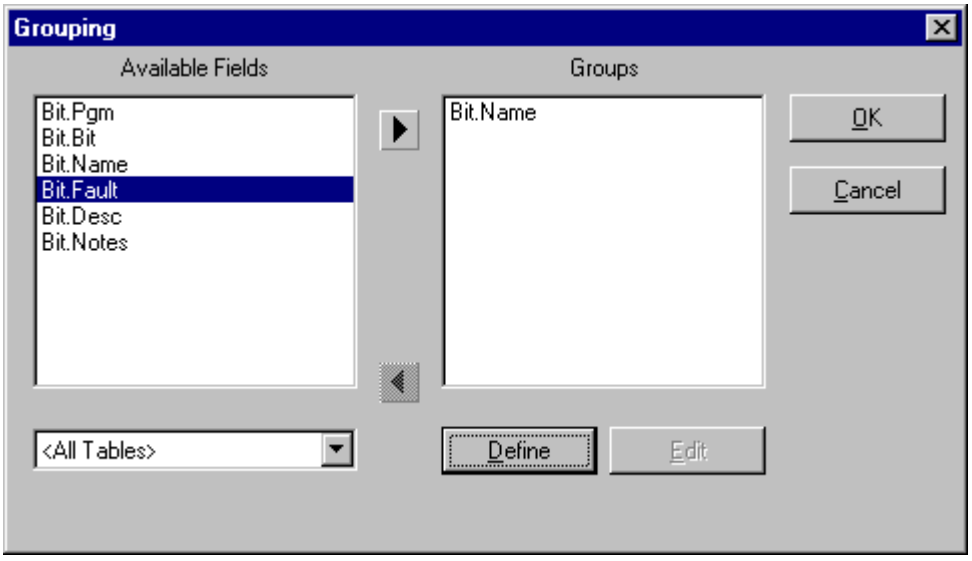
When you sort information, the resultant data sometimes contains groups of similar data. An example of this is when you sort a sales table on the salesman name. After the sort, records are grouped by salesman and there are typically multiple sales orders for each salesman. ReportPro allows you to identify the grouped data and print information before and after each of the groups.

Specifically, groups allow you to:

- Print header and footer bands to identify the details about each group
- Print each group on a new page
- Perform calculations on each of the groups

When you create a grouping in ReportPro, you identify the expression that reflects how the data is grouped. In our salesman example, the grouping expression would be the salesman field since it defines how the data is grouped. ReportPro uses this expression to trigger the printing of the Group Header and Group Footer bands that are created with you define a grouping.

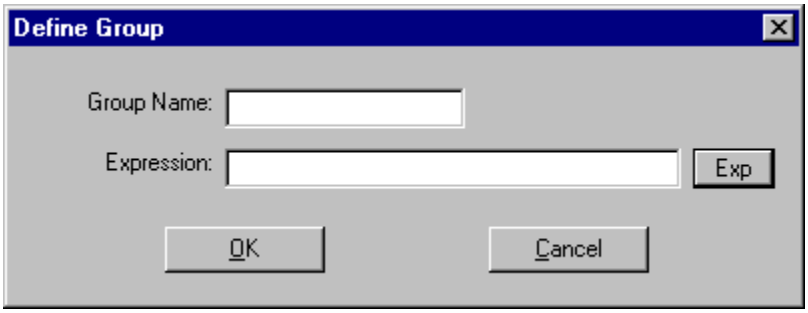
To specify a grouping, select the *Group* menu option from ReportPro's main menu. The Grouping dialog that is displayed is very similar to the Sort Order dialog.



To specify a grouping simply highlight the field you wish to group on in the *Available Fields* list box and click the left arrow button. The field will be added into the *Groups* list box. When a second field is added to the *Groups* list box, it defines a sub-grouping of the first grouping.

ReportPro allows you to create groups based on more complex expressions than just field names. For

example, you may wish to create a grouping on the first few characters of a field rather than the entire field. To define a grouping, click the *Define* button at the bottom of the Grouping dialog.



The Define Group dialog allows you to manually create a new grouping. The *Group Name* edit is required and must be a unique group name. The *Expression* edit is where you enter the expression you wish to use for the grouping. If you want to work in the Expression Builder, press the *Exp* button to the right of the *Expression* edit. Once you define the grouping, you are returned to the Grouping dialog.

The ability to group on expressions has a few not so obvious benefits. Not only is it an extremely powerful feature, but it also lets you create multiple Group Header and Group Footer bands that trigger on the same expression. This is extremely useful when printing multiple variable length fields on a report. Also, if you correctly define a variable and use it as your grouping expression, you can simulate multiple Body bands.

When you close the Grouping dialog, ReportPro will automatically create the required Group Header and Group Footer bands in the report work window. The Group Header band prints before each group and the Group Footer band prints after each group.

Grouping: How it works

For the sake of discussion, assume that we are creating a Sales listing report and we wish to list all the Sales for each salesman in Closed Date order. Let's assume that our Sales table looks like:

Salesman	Closed Date
LLA	12/10/96
GAK	12/15/96
GAK	12/15/96
LLA	12/15/96
LLA	12/17/96

First we specify a that the sort order for the report is Salesman first and Closed Date second. Next we specify that the grouping is on Salesman.

When the report is printed the following events will occur:

- The table is sorted based on the information specified in the Sort Order dialog.

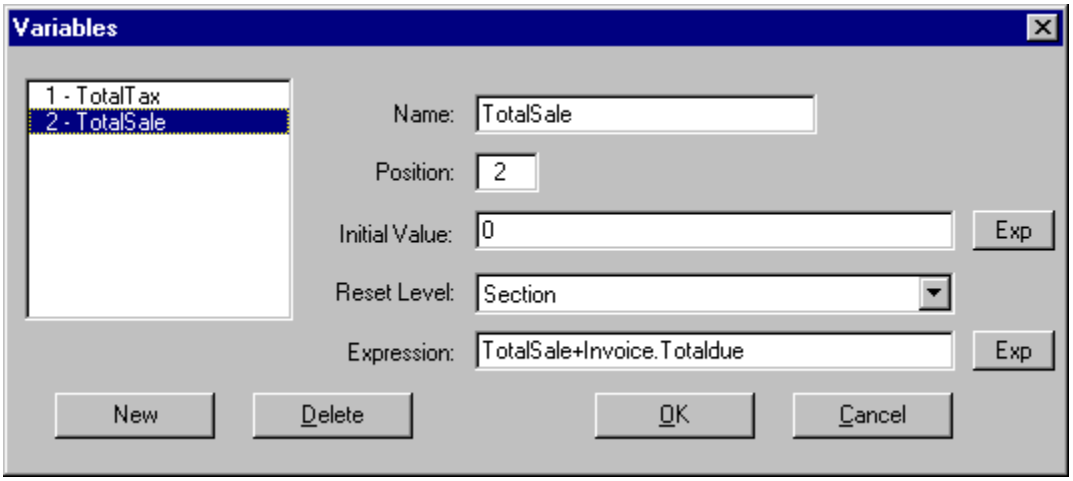
Salesman	Closed Date
GAK	12/10/96
GAK	12/15/96
LLA	12/10/96
LLA	12/15/96
LLA	12/17/96

- A group will start and remain in effect until Salesman is no longer equal to "GAK".
- The Group Header band will print.
- The Body band will print for each record for which Salesman equals "GAK" resulting in Dates "12/10/93" and "12/15/93" being printed.
- When the Salesman no longer equals "GAK" the Group Footer band is printed.
- Another group will start and remain in-effect until Salesman is no longer equal to "LLA".
- The Group Header band will print.
- The Body band will print for each record for which Salesman equals "LLA" resulting in Dates "12/10/93", "12/15/93" and "12/17/93" being printed.
- When the Salesman no longer equals "LLA" the Group Footer band is printed.
- Since there are no more records in the database the report terminates.

Variables

Variables are temporary storage locations where you can save information that can be accessed at a later time. Variables are useful for totaling field values across many records or within a group of records.

Variables actually consist of several components. Each variable must have a unique name. The variable name can be any combination of letters and numbers, but it is recommended that the variable be named something that describes the value it will hold. For example, if we are creating a variable that will hold the total of all sales, we might name the variable "TotalSale".



Each variable must also have a position indicator. The position indicator determines the order in which the variable is updated relative to the other variables in the variable list. This position can be very important when variables reference other variables. If our variable "TotalSale" references the variable "TotalTax" in its update expression, we must make sure that "TotalTax" has a position less than "TotalSale" so "TotalTax" will contain a meaningful value when it is accessed.

Each variable must also have an initial value. The variable is set to its initial value immediately prior to running the report and again during the report depending on the setting of the reset level.

The reset level determines when the variable is reset to its initial value. The variable can be reset at the report, section, page or group level. When the variable is reset at the report level, the variable is initialized immediately prior to running the report and is not reset until after the report has terminated. Use the report reset level to create variables that sum values for the entire report.

Variables that reset at the section level are initialized prior to the start of the section they are defined in. These variables are handy for creating totals for the section. If a report has only one section, these variables behave the same as variables that reset at the report level.

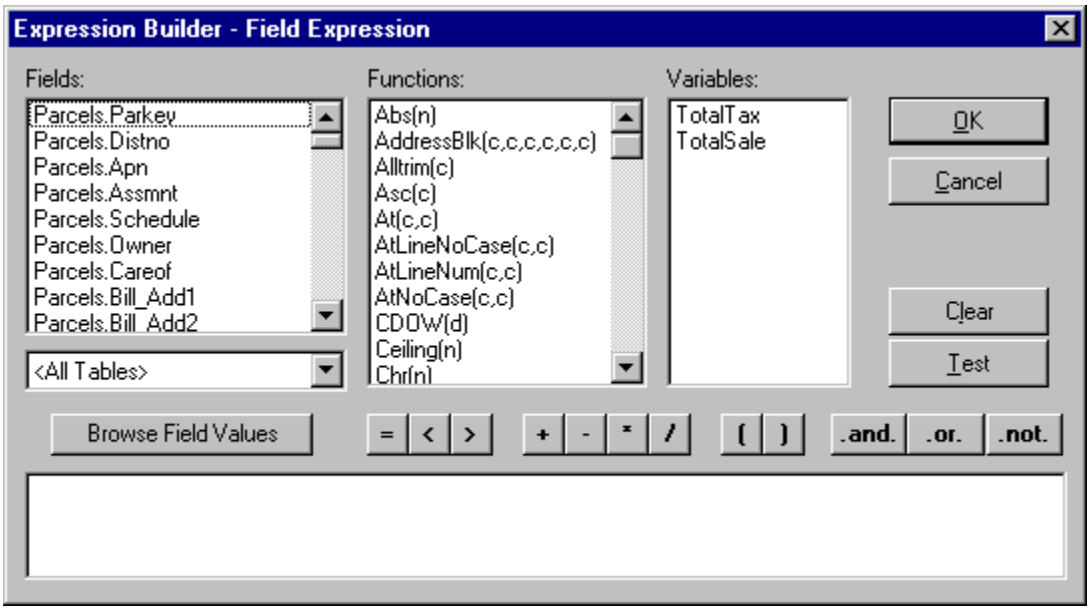
When a variable is reset at the page level, the variable is initialized prior to the start of each page. These

variables are handy for creating page totals.

When a variable is reset at the group level, it is initialized prior to the start of each new group. Use the group reset levels to create variables which total on groupings.

The variable expression determines the value that is saved into the variable during the execution of the report. The variable expression is similar to the expression used in the Field object. Remember, the value returned by the expression you enter here is stored in the variable. If you define a variable which does not seem to contain the proper value, the expression is the first place to look.

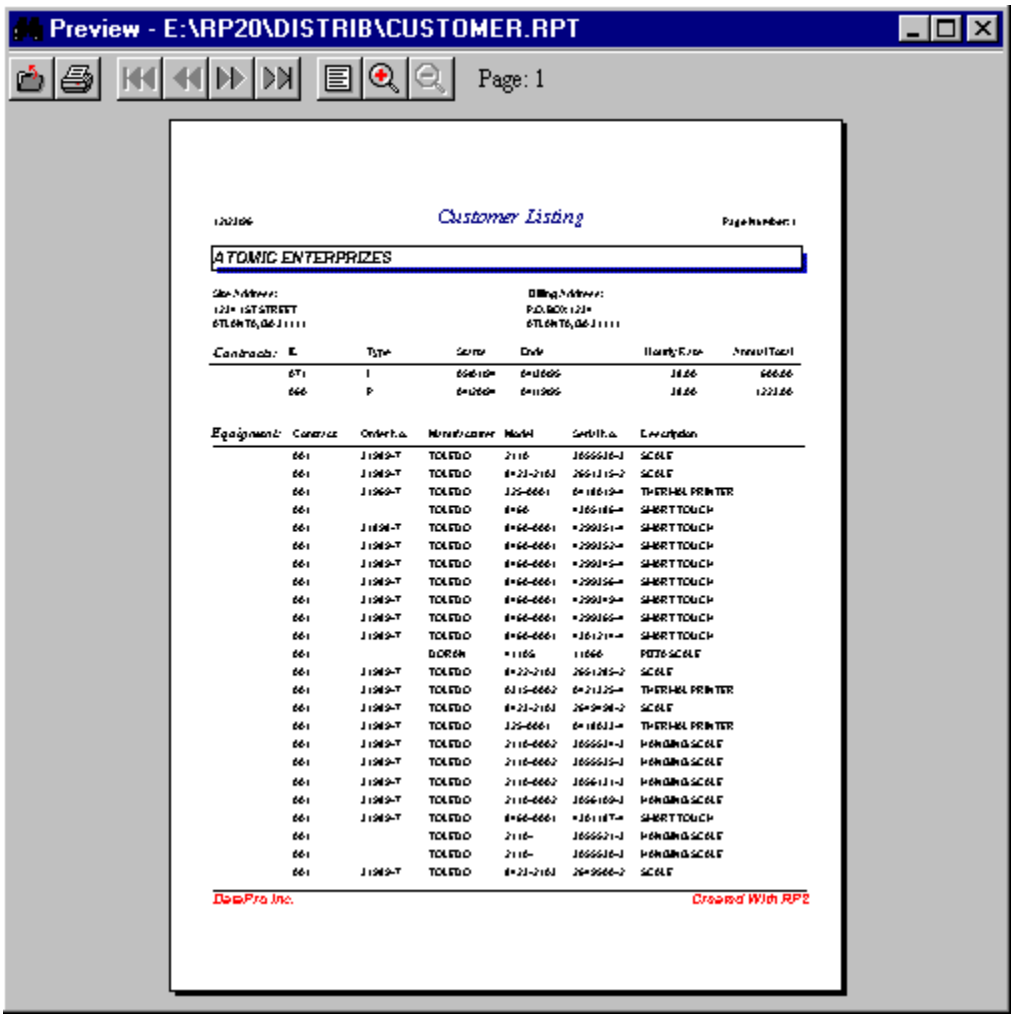
Now that you know how to define a variable, how do you use it in a report? This is the easy part. When a variable name is referenced in an expression, the variable returns to the expression the value that it contains. Simply put, once you have defined a variable, you can include it in an expression.



Notice that variables you define are available in the *Variables* list box of the Expression Builder. The variables can now be selected and placed on the report in the same manor as a database field.

Print Preview

ReportPro's Print Preview is a great way to get a sneak preview of how your report will look when it is printed. You can use the Print Preview throughout the report creation process to see how changes made to the report will look when they are printed. To display the Preview window, select the *File, Print Preview* menu option from ReportPro's main menu. You can also access the Print Preview via the toolbar on the report work window.



The white area in the middle of the preview window represents a page of the report. The page shown in Figure 69 is displayed in the *Full Page* zoom mode where you can see the contents of the entire page. If you position the mouse over the page and click the left mouse button, the paged is zoomed to the *Page Width* mode.

In the *Page Width* mode, ReportPro scales the page so that it is the same width as the Preview window. Depending on the height of the paper, vertical scrollbars may appear to allow you to scroll the page vertically.

If you click the left mouse button again, the page will zoom to the *Full Scale* mode. In this mode, the page is shown without any scaling. Here you get the best representation of what the printed output will look like. Unless you are printing on very small paper, you may not be able to see the entire page. If necessary, vertical and horizontal scrollbars will appear allowing you to scroll across the page.

You can also click the right mouse button over the preview page to zoom out through the three different zoom modes. The size of the preview also has an affect on page scaling and how much of the page can be displayed.

The Preview window has it's own menu and toolbar. The options available on the menu enable you to move through the report and zoom the page display. Each of the menu options is outlined in the table below.

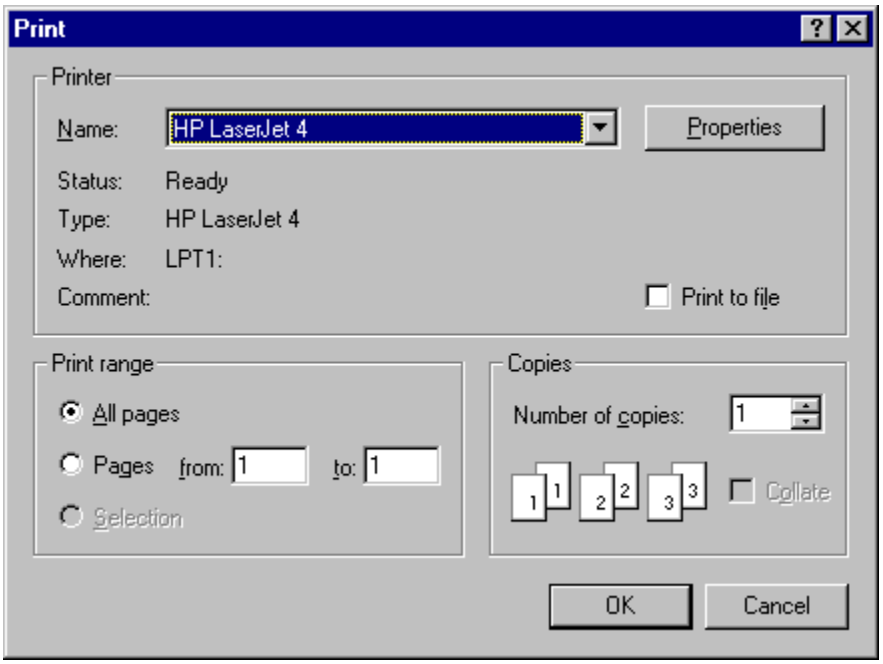
Menu Option	**Description**
Close	Closes the Preview window and returns to the work window.
Print	Sends the report to the printer.
Page, First	Displays the first page of the report.
Page, Previous	Displays the previous page of the report.
Page, Next	Displays the next page of the report.
Page, Last	Displays the last page of the report.
Page, Go to	Displays a dialog where you can enter a page number. If the page is available in the report the preview will display that page otherwise it will display the last page of the report
Zoom, In	Increases the size of the displayed page.
Zoom, Out	Decreases the size of the displayed page.
Window, Cascade	Positions the open reports in a cascaded arrangement.
Window, Tile	Positions the open reports in a tiled arrangement.
Window, 1-n	The lower portion of the menu displays a list of the open reports. You can give focus to a report by selecting it from this list.

Printing

After you have created a report and customized it to meet you needs, you will want to send the output to the printer. To print a report, choose the *File, Print* menu option from ReportPro's main menu. Printing can also be initiated from the work window toolbar.

Print Dialog

Before printing starts the Print dialog is displayed to allow you to specify printing parameters.



Each of the options available in the Print dialog is discussed in detail below.

Printer

The *Printer Name* combo box allows you to select which printer you wish to send the report to. The printers listed here are those that are installed on your system.

A *Properties* button will be displayed if additional printer options are available. This feature allows access to more advanced printer options. If the *Properties* button is selected, an additional dialog will be displayed. The options available in this dialog depend on which printer you are using. Some common options include Print Quality, Color and Default Paper Source.

Print to File

This check box instructs ReportPro to send the report output to file. If this option is selected, a dialog will be displayed before printing starts to allow you to specify a file name. The report output will be saved in the file you specify. The format of the file is dependent on the printer driver and normally cannot be opened or edited using a word processor.

Print Range

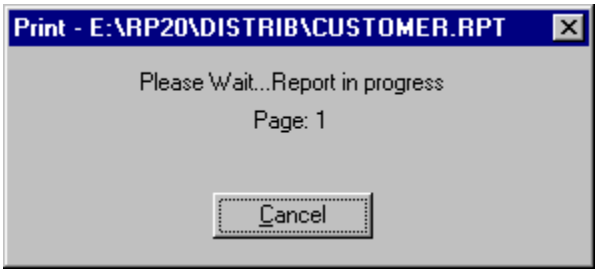
The *Print Range* controls allow you to specify which portion of the report is printed. The entire report is printed if the *All Pages* radio button is selected. If you wish to specify a range of pages to print, select the *Page* radio button and enter the desired range in the *From* and *To* edit controls.

Copies

These controls allow you to specify the number of copies you wish to print. To change the number of copies, click the arrow buttons on the spinner control until the desired value is displayed in the adjacent edit control. The *Collate* check box allows you to specify if the multiple copies should be collated. If the printer doesn't support collation, ReportPro can simulate collation by printing multiple copies of the report.

Printing Progress Dialog

After you specify the desired options in the Print dialog and press the *OK* button, ReportPro will start printing the report. While the report is printing, a progress dialog will be displayed showing you the page number that is currently printing. To cancel the printing operation press the *Cancel* button on the dialog.



Appendix A

Picture Codes

Format pictures give you the capability to format the way data is printed. The format picture consists of two distinct parts, a function string and template string, either or both of which may be present.

Function Strings

The function string specifies formatting that applies to the entire piece of information, rather than to particular character positions within it. The function string consists of the @ character, followed by one or more additional characters, each of which has a particular meaning (see table below). The function string must be the first element of a format picture and cannot contain spaces. A function string may be specified alone or with a template string. If both are present, the function string must precede the template string, and the two must be separated by a single space.

Function	Type	Action
B	N	Displays numbers left-justified
R	C	Non-template characters are inserted in the display but not saved in the field.
Z	N	Displays zero as blank.
!	C	Converts alphabetic character to uppercase

Template String

A template string specifies formatting on a character by character basis. The template string consists of a series of characters, some of which have special meanings (see table below). Each position in the template string corresponds to a position in the displayed value. Characters in the template string which do not have assigned meanings are copied verbatim into the displayed value. If you use the @R picture function, these characters are inserted between characters of the displayed value; otherwise they overwrite the corresponding characters of the displayed value.

Template	Action
N	Displays only alphabetic and numeric characters
9	Displays only digits (and sign character for numeric fields)
L	Displays logicals as T or F
Y	Displays logicals as Y or N

!	Converts an alphabetic character to upper case.
$	Displays a dollar sign in place of a leading space in a numeric.
.	Displays a decimal point.
,	Displays a comma.

Examples

The following are examples of values and the affect that format pictures have on the way they are displayed.

Value:	123654987
Format Picture:	999,999,999
Displayed:	123,654,987

Value:	John Smith
Format Picture:	@!
Displayed:	JOHN SMITH

Value:	009564311
Format Picture:	@R 999-99-9999
Displayed:	009-56-4311

Function Reference Appendix B

Abs()
AddressBlk()
Alltrim()
Asc()
At()
AtLineNoCase()
AtLineNum()
AtNoCase()
CDOW()
Ceiling()
Chr()
CMonth()
Cosine()
Cotangent()
CTOD()
Day()
Descend()
DOW()
DTOC()
DTOS()
ElapsedTime()
Empty()
GroupMembers()
HardCR()
HoursBetween()
iif()
Integer()
IsAlpha()
IsBetween()
IsDigit()
IsLower()
IsUpper()
Left()
Len()
Log()
Lower()
LTrim()
Max()
MakeDate()
MakeTime()
MemoLine()
MemoTran()
Min()
MinutesBetween()
MLCount()
Modulus()
Month()
Num2CDOW()
Num2CMonth()
NumTrim()
OccursIn()
Pad()
Power()
PgCount()
PgNo()
Properize()
Rand()
RAt()
Replicate()
Right()
Round()
RpEOF()
RpSQLCol()
Sec2Days()
Sine()
Space()
SqRt()
Str()
StrTran()
Stuff()
Tangent()
Time()
Today()
Trim()
Upper()
Val()
Year()

Abs()

Purpose

To calculate the absolute value of a number

Syntax

Abs(<*nNumber*>) *nAbsValue*

Argument

<*nNumber*> is the number to determine the absolute value of

Returns

Abs() returns the absolute value of <*nNumber*>.

AddressBlk()

Purpose

To format an address for printing

Syntax

AddressBlk(*<cName>, <cAddress1>, <cAddress2>, <cCity>, <cState>, <cZip>*) *cAddressString*

Arguments

<cName>	The recipient's name
<cAddress1>	Address line 1
<cAddress2>	Address line 2
<cCity>	The city name
<cState>	The state
<cZip>	The zip code

Returns

AddressBlk() creates an address block string based on the arguments passed. Blank arguments are ignored.

Alltrim()

Purpose

To remove leading and trailing spaces from a character string

Syntax

Alltrim(<*cString*>) *cTrimString*

Argument

<*cString*> is the character string to trim spaces from

Returns

Alltrim() returns a character string with leading and trailing spaces removed.

Asc()

Purpose

To determine the ASCII value of a character

Syntax

Asc(<*cCharacter*>) *nNumber*

Argument

<*cCharacter*> is the character to determine the ASCII value of

Returns

Asc() returns the ASCII value of <*cCharacter*>.

At()

Purpose

To determine the position of a string within another string

Syntax

At(<*cSearchFor*>, <*cSearchIn*>) *nLocation*

Argument

<*cSearchFor*> is the character string to search for

<*cSearchIn*> is the character string to search

Returns

At() returns a numeric indicating the location of <*cSearchFor*> within <*cSearchIn*>. If <*cSearchFor*> is not found, At() returns 0.

AtNoCase()

Purpose

To determine the position of a string within another string without regard for case

Syntax

AtNoCase(<*cSearchFor*>, <*cSearchIn*>) *nLocation*

Argument

<*cSearchFor*> is the character string to search for

<*cSearchIn*> is the character string to search

Returns

AtNoCase() returns a numeric indicating the location of <*cSearchFor*> within <*cSearchIn* >. If <*cSearchFor* > is not found, AtNoCase() returns 0.

AtLineNum()

Purpose

To determine the line number of a string within a multi-line string

Syntax

AtLineNum(<*cSearchFor*>, <*cSearchIn*>) *nNumber*

Argument

<*cSearchFor*> is the character string to search for

<*cSearchIn*> is the character string to search

Returns

AtLineNum() returns a numeric indicating the line number of <*cSearchFor*> within <*cSearchIn*>. If <*cSearchFor*> is not found, AtLineNum() returns 0.

AtLineNoCase()

Purpose

To determine the line number of a string within a multi-line string without regard for case

Syntax

AtLineNoCase(<*cSearchFor*>, <*cSearchIn*>) *nNumber*

Argument

<*cSearchFor*> is the character string to search for

<*cSearchIn*> is the character string to search

Returns

AtLineNoCase() returns a numeric indicating the line number of <*cSearchFor*> within <*cSearchIn*>. If <*cSearchFor*> is not found, AtLineNoCase() returns 0.

CDOW()

Purpose

To convert a date value to a character day of the week

Syntax

CDOW(<*dDate*>) *cDayName*

Argument

<*dDate*> is the date to convert

Returns

CDOW() returns the name of the day of the week as a character string.

Ceiling()

Purpose

To round a number up to the highest integer

Syntax

Ceiling(<*nNumber*>) *nRoundedNumber*

Argument

<*nNumber*> is the number to round

Returns

Ceiling() returns <*nNumber*> rounded up to the next highest integer.

Chr()

Purpose

To convert a numeric to a character

Syntax

Chr(<*nNumber*>) *cCharacter*

Argument

<*nNumber*> is the ASCII number to convert to a character

Returns

Chr() returns the character representation of <*nNumber*>.

CMonth()

Purpose

Convert a date to a character month name

Syntax

CMonth(<*dDate*>) *cMonth*

Argument

<*dDate*> is the date to convert.

Returns

CMonth() returns the name of the month as a character string.

Cosine()

Purpose

To calculate the cosine of a number

Syntax

Cosine(<*nAngle*>) *nCosine*

Argument

<*nAngle*> is an angle in radians

Returns

Cosine() returns the cosine of <*nAngle*>.

Cotangent()

Purpose

To calculate the cotangent of a number

Syntax

Cotangent(<*nAngle*>) *nCotangent*

Argument

<*nAngle*> is an angle in radians

Returns

Cotangent() returns the cotangent of <*nAngle*>.

CTOD()

Purpose

To convert a character value to a date

Syntax

CTOD(<*cDate*>) *dDate*

Argument

<*cDate*> is a character string consisting of numbers representing month, day, and year digits separated by the "/" character. <*cDate*> must be expressed as "mm/dd/yy" or "mm/dd/yyyy".

Returns

CTOD() returns the date equivalent of <*cDate*>.

Day()

Purpose

To calculate the day of the month as a numeric value.

Syntax

Day(<*dDate*>) *nDay*

Argument

<*dDate*> is the date to convert.

Returns

Day() returns a number in the range of 1 to 31 as a numeric value.

Descend()

Purpose

To create a descending order key value

Syntax

Descend(<*cString*>) *cInvertedValue*

Argument

<*cString*> is the string to calculate the descending value for

Returns

Descend() returns an inverted <*cString*>.

DOW()

Purpose

Convert a date value to a numeric day of the week

Syntax

DOW(<*dDate*>) *nDay*

Argument

<*dDate*> is the date to convert.

Returns

DOW() returns the day of the week as a number between 1 and 7.

DTOC()

Purpose

To convert a date value to a character string

Syntax

DTOC(<*dDate*>) *cDate*

Argument

<*dDate*> is the date value to convert

Returns

DTOC returns a character string in the format "mm/dd/yy" that represents <*dDate*>.

DTOS()

Purpose

To convert a date value to a character string

Syntax

DTOS(<*dDate*>) *cDate*

Argument

<*dDate*> is the date value to convert

Returns

DTOS returns a character string in the format "yyyymmdd" that represents <*dDate*>.

ElapsedTime()

Purpose

To calculate the interval between two times

Syntax

ElapsedTime(<*cTime1*>, <*cTime2*>) *cTimeDifference*

Argument

<*cTime1*> is the starting time

<*cTime2*> is the ending time

Returns

ElapsedTime() returns the amount of time that has elapsed from <*cTime1*> to <*cTime2*> as a time string in the format hh:mm:ss.

Empty()

Purpose

To determine if a field or variable contains a empty value

Syntax

Empty(<*xValue*>) *lResult*

Argument

<*xValue*> is the value to check and can be any data type.

Returns

Empty() returns true if the value is empty; otherwise, it returns false.

Data Type	Empty Values
Character	Spaces, tabs or "" (empty string)
Numeric	0
Date	NULL
Logical	.f.

GroupMembers()

Purpose

To determine the number of members in a grouping.

Syntax

GroupMembers(<*nGroup*>) *nNumber*

Argument

<*nGroup*> is the group to query.

Returns

GroupMembers() returns the number of members in <*nGroup*> group as a numeric value.

HardCR()

Purpose

To replace all soft carriage returns (chr(141)) with hard carriage returns (chr(13))

Syntax

HardCR(<*cString*>) *cHardCRString*

Argument

<*cString*> is the string containing the soft carriage returns

Returns

HardCR() returns a string where all soft carriage returns are replaced with hard carriage returns.

HoursBetween()

Purpose

To calculate the hours between two date/time values

Syntax

HoursBetween(*<dDate1>, <cTime1>, <dDate2>, <cTime2>, <lWeekEnds>*) *nNumber*

Argument

<dDate1> The starting date

<cTime1> The starting time as character string formatted as "hh:mm:ss" (24 hour format)

<dDate2> The ending date

<cTime2> The ending time as character string formatted as "hh:mm:ss" (24 hour format)

<lWeekEnds> A logical indicating whether or not to count weekends

Returns

HoursBetween() returns the number of hours between *<cTime1>* on *<dDate1>* and *<cTime2>* on *<dDate2>*. The first date/time value must be smaller of the two.

iif()

Purpose

To return the result of an expression based on a condition

Syntax

iif(<*lCondition*> , <*expTrue*> , <*expFalse*>) *xValue*

Arguments

<*lCondition*> is a logical expression to be evaluated.

<*expTrue*> is the value, of any data type, returned if <*lCondition*> is true.

<*expFalse*> is the value, of any data type, returned if <*lCondition*> is false.

Returns

iif() returns the evaluation of <*expTrue*> if <*lCondition*> evaluates to true and <*expFalse*> if it evaluates to false.

Integer()

Purpose

To truncate a number with decimal digits to a whole number

Syntax

Integer(<*nNumber*>) *nInteger*

Arguments

<*nNumber*> is the decimal number to convert to an integer

Returns

Integer() returns <*nNumber*> truncated to an integer.

IsAlpha()

Purpose

To determine if the first character of a string is alphabetic

Syntax

IsAlpha(<*cString*>) *lLogical*

Arguments

<*cString*> is the string to test

Returns

IsAlpha() returns TRUE if the first character in <*cString*> is an alphabetic (A-Z, a-z) character.

IsBetween()

Purpose

To determine if a number is between two other numbers

Syntax

IsBetween(<*nTestNumber*>, <*nLowerLimit*>, <*nUpperLimit*>) *lLogical*

Arguments

<*nTestNumber*> is the number to test

<*nLowerLimit*> is the lower limit to test against

<*nUpperLimit*> is the upper limit to test against

Returns

IsBetween() returns TRUE if <*nTestNumber*> is between <*nLowerLimit*> and <*nUpperLimit*>.

IsDigit()

Purpose

To determine if the first character of a string is a numeric digit

Syntax

IsDigit(<*cString*>) *lLogical*

Arguments

<*cString*> is the string to test

Returns

IsDigit() returns TRUE if the first character in <*cString*> is a digit (0-9).

IsLower()

Purpose

To determine if the first character of a string is lower case

Syntax

IsLower(<*cString*>) *lLogical*

Arguments

<*cString*> is the string to test

Returns

IsLower() returns TRUE if the first character in <*cString*> is lower case (a-z).

IsUpper()

Purpose

To determine if the first character of a string is upper case

Syntax

IsUpper(<*cString*>) *lLogical*

Arguments

<*cString*> is the string to test

Returns

IsUpper() returns TRUE if the first character in <*cString*> is upper case (A-Z).

Left()

Purpose

To extract a substring beginning with the first character in a string

Syntax

Left(<*cString*> , <*nCharacters*>) *cSubString*

Arguments

<*cString*> is a character string from which to extract characters.

<*nCharacters*> is number of characters to extract beginning with the leftmost character.

Returns

Left() returns a substring of <*nCharacters*> from <*cString*> starting at the leftmost character.

Len()

Purpose

To calculate the length of a character string

Syntax

Len(<*cString*>) *nLength*

Arguments

<*cString*> is the character string to determine the length of

Returns

Len() returns the length (in characters) of <*cString*>.

Log()

Purpose

To calculate the natural logarithm of a number

Syntax

Log(<*nNumber*>) *nLogarithm*

Arguments

<*nNumber*> is a number greater than 0

Returns

Log() returns the natural logarithm of <*nNumber*>.

Lower()

Purpose

To convert upper case characters to lower case

Syntax

Lower(<*cString*>) *cLowerCase*

Arguments

<*cString*> is the string to convert to lower case.

Returns

Lower() returns the lower cased equivalent of <*cString*>.

LTrim()

Purpose

To remove leading spaces from a character string

Syntax

LTrim(<*cString*>) *cTrimmed*

Arguments

<*cString*> is the character string to trim.

Returns

LTrim() returns <*cString*> without leading spaces.

Max()

Purpose

To determine the larger of two numeric or date values

Syntax

Max(<*dnValue1*> , <*dnValue2*>) *dnMaxValue*

Arguments

<*dnValue1*> and <*dnValue2*> are the date or numeric values to compare. They must be the same data type.

Returns

Max() returns the maximum value of <*dnValue1*> and <*dnValue2*>.

MakeDate()

Purpose

To create a date from three numerics representing a year, month and day.

Syntax

MakeDate(<*nYear*> , <*nMonth*>, <*nDay*>) *dDate*

Arguments

<*nYear*> is the year of the new date

<*nMonth*> is the month of the new date

<*nDay*> is the day of the new date

Returns

MakeDate() returns a date that represents <*nYear*>, <*nMonth*> and <*nDay*>.

MakeTime()

Purpose

To create a time string from three numerics representing hours, minutes and seconds.

Syntax

MakeTime(<*nHours*> , <*nMinutes*>, <*nSeconds*>) *cTime*

Arguments

<*nHours*> is the hour of the new time

<*nMinutes*> is the minutes of the new time

<*nSeconds*> is the seconds of the new time

Returns

MakeTime() returns a string in the format "hh:mm:ss".

MemoLine()

Purpose

To extract a line of text from a multi-line string

Syntax

MemoLine(<*cString*> , <*nLineLength*>, <*nLineNumber*>, <*nTabSize*>, <*lWrap*>) *cText*

Arguments

<*cString*> is the string to extract the line of text from

<*nLineLength*> is the default line length in characters

<*nLineNumber*> is the line number to extract

<*nTabSize*> is the tab size in characters

<*lWrap*> is TRUE to turn on word wrap, FALSE to turn it off

Returns

MemoLine() returns the <*nLineNumber*> line of text if it exists in <*cString*>.

MemoTran()

Purpose

To replace carriage return and linefeed characters with a character that can be displayed

Syntax

MemoTran(<*cString*> , <*cHardCR*>, <*cSoftCR*>) *cText*

Arguments

<*cString*> is the string that contains the carriage return and linefeed characters

<*cHardCR*> is the character to replace hard carriage returns with

<*cSoftCR*> is the character to replace soft carriage returns with

Returns

MemoTran() returns a string with the carriage returns and linefeeds replaced

Min()

Purpose

To determine the smaller of two numeric or date values

Syntax

Min(<*dnValue1*> , <*dnValue2*>) *dnMinValue*

Arguments

<*dnValue1*> and <*dnValue2*> are the date or numeric values to compare. They must be the same data type.

Returns

Min() returns the minimum value of <*dnValue1*> and <*dnValue2*>.

MLCount()

Purpose

To determine the number of lines in a string

Syntax

MLCount(<*cString*> , <*nLineLength*>, <*nTabSize*>, <*lWrap*>) *cText*

Arguments

<*cString*> is the string to determine the number of lines for

<*nLineLength*> is the default line length in characters

<*nTabSize*> is the tab size in characters

<*lWrap*> is TRUE to turn on word wrap, FALSE to turn it off

Returns

MLCount() returns the number of lines in <*cString*>.

Modulus()

Purpose

To determine the remainder of a division operation

Syntax

Modulus(<*nDividend*> , <*nDivisor*>) *nRemainder*

Arguments

<*nDividend*> The dividend of the division operation

<*nDivisor*> The divisor of the division operation

Returns

Modulus() returns the remainder of <*nDividend*> divided by <*nDivisor*>.

MinutesBetween()

Purpose

To determine the number of minutes between two date/time values.

Syntax

MinutesBetween(<*dDate1*>, <*cTime1*>, <*dDate2*>, <*cTime2*>, <*lWeekEnds*>) *nNumber*

Argument

<*dDate1*> The starting date

<*cTime1*> The starting time as character string formatted as "hh:mm:ss" (24 hour format)

<*dDate2*> The ending date

<*cTime2*> The ending time as character string formatted as "hh:mm:ss" (24 hour format)

<*lWeekEnds*> A logical indicating whether or not to count weekends.

Returns

MinutesBetween() returns the number of minutes between <*cTime1*> on <*dDate1*> and <*cTime2*> on <*dDate2*>. The first date/time value must be the smaller of the two.

Month()

Purpose

To convert a date value to a month number

Syntax

Month(<*dDate*>) *nMonth*

Arguments

<*dDate*> is the date to convert.

Returns

Month() returns a numeric value in the range of 1 to 12 representing the month of <*dDate*>.

Num2CDOW()

Purpose

To convert a numeric day number to a character day of the week string

Syntax

Num2CDOW(<*nDay*>) *cDayName*

Arguments

<*nDay*> is the number (1-7) to convert into the day name

Returns

Num2CDOW() returns a character day name.

Num2CMonth()

Purpose

To convert a numeric month number to a character month name

Syntax

Num2CMonth(<*nMonth*>) *cMonthName*

Arguments

<*nMonth*> is the number (1-12) to convert into the month name

Returns

Num2CMonth() returns a character month name.

NumTrim()

Purpose

Returns the number passed as a left-trimmed string

Syntax

NumTrim(<*nNumber*>) *cString*

Arguments

<*nNumber*> is the number to convert to a string

Returns

NumTrim() returns the value <nNumber> as a left-trimmed string.

OccursIn()

Purpose

To determine how many time one string occurs in another

Syntax

OccursIn(<*cSearchFor*>, <*cSearchIn*>) *nTimes*

Arguments

<*cSearchFor*> is the string to search for

<*cSearchIn*> is the string to search

Returns

OccursIn() returns the number of times <*cSearchFor*> occurs in <*cSearchIn*>.

Pad()

Purpose

To pad character, date and numeric values with spaces

Syntax

PadC(<*xValue*> , <*nLength*>, <*cPadChar*>) *cPaddedString*

PadL(<*xValue*> , <*nLength*>, <*cPadChar*>) *cPaddedString*

PadR(<*xValue*> , <*nLength*>, <*cPadChar*>) *cPaddedString*

Arguments

<*xValue*> is the character, date or numeric value to pad

<*nLength*> is the length of the string to return

<*cPadChar*> is the character to pad with

Returns

PadC() returns <*xValue*> as a character string of length <*nLength*> centered between <*cPadChar*>. PadL() returns <*xValue*> as a character string of length <*nLength*> padded with <*cPadChar*> on the left side. PadR() returns <*xValue*> as a character string of length <*nLength*> padded with <*cPadChar*> on the right side.

Power()

Purpose

To raise a number by a power

Syntax

Power(<*nNumber*>, <*nExponent*>) *nTimes*

Arguments

<*nNumber*> is the base number

<*nExponent*> is the power to raise the base number by

Returns

Power() returns <*nNumber*> raised to the power of <*nExponent*>.

PgCount()

Purpose

To retrieve the ReportPro page count.

Syntax

PgCount() *nCount*

Returns

PgCount() returns the internal ReportPro page number count. This function will only return a meaningful value if the "Support Total Page Count" option is set in the Report Configuration dialog.

PgNo()

Purpose

To retrieve the ReportPro page number

Syntax

PgNo() n*Page*

Returns

PgNo() returns the internal ReportPro page number.

Properize()

Purpose

To change the first character of each word in a string to upper case

Syntax

Properize(<*cString*>) *cProperString*

Arguments

<*cString*> is the string to modify

Returns

Properize() returns <*cString*> with the first character of each word in uppercase.

Rand()

Purpose

To retrieve a random number between 0 and 1

Syntax

Rand(<*nSeedNumber*>) *nRandomNumber*

Arguments

<*nSeedNumber*> is a number used to seed the random number generator

Returns

Rand() returns a random number.

RAt()

Purpose

To retrieve the last position of one string within another

Syntax

RAt(<*cSearchFor*>, <*cSearchIn*>) *nLocation*

Arguments

<*cSearchFor*> is the string to search for

<*cSearchIn*> is the string to search

Returns

RAt() returns the last location of <*cSearchFor*> in <*cSearchIn*>.

Replicate()

Purpose

To create a string by repeating another string

Syntax

Replicate(<*cString*>, <*nTimes*>) *cNewString*

Arguments

<*cString*> is the string to repeat

<*nTimes*> is the number of times to repeat <*cString*>

Returns

Replicate() returns <*cString*> repeated <*nTimes*>.

Right()

Purpose

To retrieve a substring beginning with the rightmost character

Syntax

Right(<*cString*> , <*nCount*>) *cSubString*

Arguments

<*cString*> is the character string from which to extract characters.

<*nCount*> is the number of characters to return.

Returns

Right() returns the rightmost <*nCount*> characters of <*cString*>.

Round()

Purpose

To return a numeric value rounded to a specified number of digits.

Syntax

Round(<*nNumber*> , <*nDecimals*>) *nRounded*

Arguments

<*nNumber*> is the numeric value to round.

<*nDecimals*> is the number of decimals places to retain.

Returns

Round() returns a numeric value rounded to <*nDecimals*> decimals.

RpEOF()

Purpose

To return a logical indicating if ReportPro has reached the end of the report.

Syntax

RpEOF() *lEOF*

Returns

RpEOF() returns TRUE if ReportPro has finished processing all records in the report.

RpSQLCol()

Purpose

To retrieve a SQL column value by column position number.

Syntax

RpSQLCol(<*nColumn*>) *xColumnValue*

Arguments

<*nColumn*> is the number of the column to retrieve

Returns

RpSQLCol() returns the value of the column that occupies the <*nColumn*> position in the SQL select statement.

Sec2Days()

Purpose

To convert seconds to days

Syntax

Sec2Days(<*nSeconds*>) *nDays*

Arguments

<*nSeconds*> is the number of seconds to convert to days

Returns

Sec2Days() returns <*nSeconds*> converted to days.

Sine()

Purpose

To calculate the sine of a number

Syntax

Sine(<*nAngle*>) *nSine*

Arguments

<*nAngle*> is the angle in radians

Returns

Sine() returns the sine of <*nAngle*>.

Space()

Purpose

To return a string of spaces

Syntax

Space(<*nCount*>) *cString*

Arguments

<*nCount*> is the number of spaces to return.

Returns

Space() returns a string of <*nCount*> spaces.

SqRt()

Purpose

To calculate the square root of a positive number

Syntax

SqRt(<*nNumber*>) *nRoot*

Arguments

<*nNumber*> is the positive number to calculate the square root of

Returns

SqRt() returns the square root of <*nNumber*>.

Str()

Purpose

To convert a numeric expression to a character string

Syntax

Str(<*nNumber*> , <*nLength*> , <*nDecimals*>) *cNumber*

Arguments

<*nNumber*> is the numeric value to convert

<*nLength*> is the length of string to return including decimals and decimal point

<*nDecimals*> number of decimals to return

Returns

Str() returns <*nNumber*> formatted as a character string.

StrTran()

Purpose

To search and replace characters in a string

Syntax

StrTran(<*cSearchFor*> , <*cSearchIn*> , <*cReplaceWith*>, <*nStart*>, <*nCount*>) *cNewString*

Arguments

<*cSearchFor*> is the character string to search for

<*cSearchIn*> is the character string to search

<*cReplaceWith*> is the character string to replace <*cSearchFor*> with

<*nStart*> is the first occurrence of <*cSearchFor*> to replace

<*nCount*> is the number of replacements to make

Returns

StrTran() returns <*cSearchIn*> where <*cSearchFor*> is replaced with <*cReplaceWith*>.

Stuff()

Purpose

To insert and delete characters from a string

Syntax

Stuff(<*cString*> , <*nStart*> , <*nDelete*>, <*cInsert*>) *cModifiedString*

Arguments

<*cString*> is the character string to modify

<*nStart*> is the character position where the delete/insert begins

<*nDelete*> is the number of characters to delete

<*cInsert*> is the character string to insert at <*nStart*>

Returns

Stuff () returns <*cString*> with characters deleted or inserted at <*nStart*>.

SubStr()

Purpose

To extract a substring from a character string

Syntax

SubStr(<*cString*> , <*nStart*> , <*nCount*>) *cSubString*

Arguments

<*cString*> is the character string in which to extract a substring

<*nStart*> is the starting position in <*cString*>

<*nCount*> is the number of characters to extract

Returns

SubStr() returns a substring of <*cString*>.

Tangent()

Purpose

To calculate the tangent of a number

Syntax

Tangent(<*nAngle*>) *nTangent*

Arguments

<*nAngle*> is the angle in radians

Returns

Tangent() returns the tangent of <*nAngle*>.

Time()

Purpose

To retrieve the system time

Syntax

Time() *cTimeString*

Returns

Time() returns the system time as a character string in the form "*hh:mm:ss*".

Today()

Purpose

To retrieve the computer clock date.

Syntax

Today() *dSystemDate*

Returns

Today() returns the system clock date as a date value.

Trim()

Purpose

To remove the trailing spaces from a string

Syntax

Trim(<*cString*>) *cTrimmedString*

Arguments

<*cString*> is the string to remove the spaces from

Returns

Trim() returns <*cString*> with the rightmost spaces removed.

Upper()

Purpose

To convert lower case characters to upper case

Syntax

Upper(<*cString*>) *cUpperString*

Arguments

<*cString*> is the character string to convert

Returns

Upper() returns a copy of <*cString*> with all alphabetic characters converted to uppercase.

Val()

Purpose

To convert a character number to numeric type

Syntax

Val(<*cNumber*>) *nNumber*

Arguments

<*cNumber*> is the character string to convert.

Returns

Val() returns <*cNumber*> converted to a numeric value including decimal digits.

Year()

Purpose

To retrieve the year from a date

Syntax

Year(<*dDate*>) *nYear*

Arguments

<*dDate*> is the date to determine the year from

Returns

Year() returns the year of the specified date.

Appendix C

RpWin.INI Reference

The RpWin.INI file allows you to control certain operating parameters within ReportPro. The INI files are divided into several sections.

The Environments section allows you to set certain internal flags which control the way data is manipulated.

[Environment]

SetAnsi=TRUE

SetCollation=WINDOWS

SetDeleted=TRUE

SetCentury=FALSE

SetEpoch=1900

UnitofMeasure=IN

RptPath=

DBFPath=

SetAnsi	Valid options are TRUE and FALSE. If TRUE, ReportPro will automatically convert between OEM and ANSI for existing database files that are in OEM format. Is FALSE, no automatic conversion is performed.
SetCollation	Valid options are CLIPPER and WINDOWS. Set this option to CLIPPER if you wish to use or create indexes that are compatible with CA-Clipper™. Set this option to WINDOWS to create indexes compatible with international language characters.
SetDeleted	Valid options are TRUE and FALSE. Set this option to TRUE to hide deleted records or set it to FALSE to include them in reports.
SetCentury	Valid options are TRUE and FALSE. Setting this option to TRUE causes dates to be displayed with a 4 digit year.
SetEpoch	Specifies the base year of a 100-year period in which all dates containing only two year digits are assumed to fall. The initial default is 1900, causing dates with no century digits to be interpreted as falling within the twentieth century.
UnitOfMeasure	Valid options are IN and CM. CM forces ReportPro to display all measurements in centimeters. IN forces ReportPro to display measurements in inches.
RptPath	This is the default report directory.
DBFPath	This is the directory ReportPro uses as a default when tables are added

to a RDD report.

The QuickFiles section is automatically maintained by ReportPro. Do not modify this section.

[QuickFiles]
QF1=
QF2=
QF3=
QF4=
QF5=

The DataDrivers section allows you to instruct ReportPro to use additional RDDs.

[DataDrivers]
RDD1=XBase Server,DBFCDXAX
RDD2=
RDD3=

The Default Font section allows you to specify a font that is assigned to Field and Text objects when they are created.

[Default Font]
FontName=ARIAL
FontHeight=13
FontBold=FALSE
FontItalic=FALSE
FontUnderline=FALSE
FontStrikeout=FALSE
FontColorRed=0
FontColorGreen=0
FontColorBlue=0

Appendices

Picture Codes
Function Reference
RpWin.INI Reference

Imaging and Graphing

Technology provided by Light Lib

The following is a list of features provided by the Standard and Professional Editions of Light Lib Images and Light Lib Business. Thanks to a special agreement between DataPro and DFL, as a ReportPro user, you get all the functionality of the Professional Editions within ReportPro reports. If you wish to use Light Lib products outside of ReportPro, please contact DFL for upgrade pricing information.

Light Lib Images

Advanced imaging

- Direct support for TWAIN scanners and devices
- High performance image processing with ULTRA fast loading, converting, printing, zooming, flipping, rotating, ...
- Extensive file format support : BMP, PCX, PNG, TGA, uncompressed and 24-bit TIF
- Compression modes: RLE, LZ77, CCITT Group 1D
- Advanced data binding and database interaction
- Save and convert file formats
- Scaling, fit to window
- Grayscale, intelligent dithering and optimized color palettes

Light Lib Business

Data aware graphs

- Dynamic navigation
- Interactive graphs
- Truly data aware control
- Drill down capabilities
- ULTRA fast graph display
- Completely configurable : colors, labels, fonts, legends, ...
- Automatic resizing, scaling, legends, dialogs
- Line, bar, pie, ribbon, stacked bar, percent, stock, 2D, 3D, filled and non-filled graphs
- Full BLOB support for easy saving of all graph settings
- ODBC connection to any data source

PRO edition

In addition to features found in the Standard Edition

- Native access to TWAIN devices and multipage scanning
- BLOB Support
- File formats: GIF, JPG, TIF (including multi-page TIF)
- Compression : LZW, CCITT G3&4, Huffman, JPEG
- Mixed and complex graphs
- Right Y-Axis support
- Rotate skipper, Gantt, Complex or Multiple-columns, Stock (high-low-close).

Contact our offices for ordering information

North America

DFL Software, Inc.
55 Eglinton Ave. E., Suite 208
Toronto, ON, Canada M4P 1G8

Tel (416) 487-2660
Fax (416) 487-3656

Internet: http://www.dfl.com

Europe and other countries

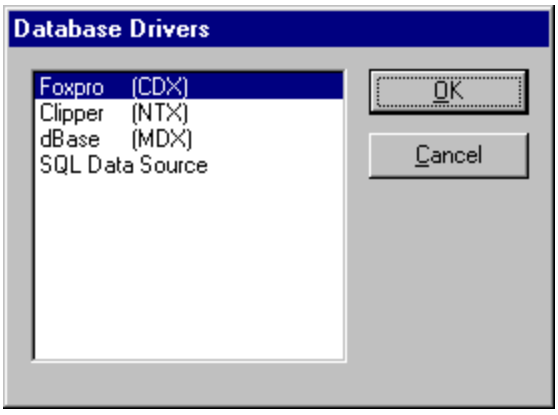
DFL Europe
39-41, rue de la Saussière
92100 Boulogne - France

Tel (33 1) 46 05 20 66
Fax (33 1) 46 04 10 39

CompuServe: GO DFLSW

Database Drivers

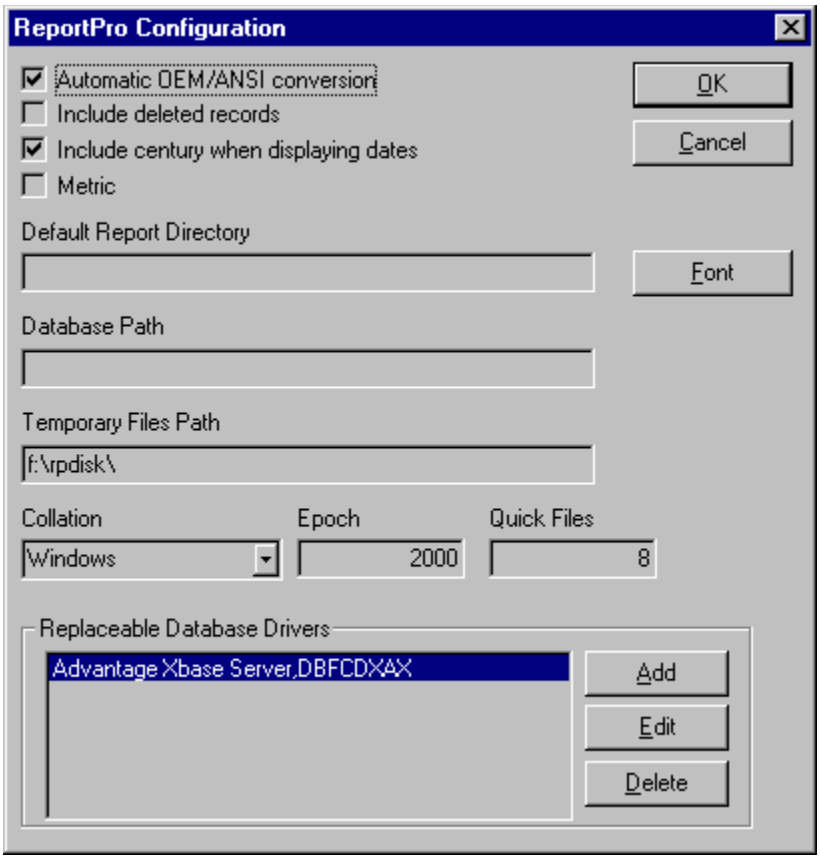
ReportPro natively supports FoxPro™, dBase™ and CA-Clipper™ file and index formats via Replaceable Data Driver (RDD) technology. ReportPro uses Open Database Connectivity (ODBC) technology to access SQL data sources. In general, ReportPro works the same regardless of which database driver you select. SQL data sources do require a few addition steps when a report is created. Please see the chapter "Using an ODBC Data Source" for more information on how to create an ODBC report.



From the Database Drivers dialog, select the desired driver by clicking the driver name with the mouse and then clicking the *OK* button. Alternately you can double-click the left mouse button on the desired database driver.

Configuring ReportPro

The ReportPro Configuration dialog allows you to specify miscellaneous properties that related to the default behavior of ReportPro. The ReportPro Configuration dialog is displayed when you select *File, Setup, ReportPro* from the work window menu.



The Automatic OEM/ANSI Conversion checkbox allows you to select whether ReportPro will automatically convert between OEM and ANSI for existing database files that are in OEM format. If this checkbox is not checked, no automatic conversion is performed.

The Include Deleted Records checkbox tells ReportPro whether to include deleted records as it processes local tables in a data source.

The Include Century When Displaying Dates checkbox tells ReportPro whether to automatically display the century when formatting dates. For example, displaying the century for 1/9/67 would result in 1/9/1967.

The Metric checkbox sets the default ReportPro unit of measure. If this box is checked, the default unit of measurement is Centimeters; otherwise, the default unit of measurement is Inches.

The Default Report Directory edit allows you to specify the default directory used when ReportPro

prompts for which report to open when Open Report is selected.

The Font pushbutton displays a dialog which allows you to select the default font ReportPro will use when creating drawing objects.

The Database Path edit tells ReportPro the default directory used when prompting for the name of a local table.

The Temporary Files Path allows you to specify the directory ReportPro will use to store its temporary files. In ReportPro/16-bit only the drive letter is used to determine the location of temporary files. In ReportPro/32-bit, the entire path specified is used.

The Collation combo box specifies the index collation ReportPro will use when creating temporary indices. Valid options are CLIPPER and WINDOWS. Set this option to CLIPPER if you wish to use or create indexes that are compatible with CA-Clipper™. Set this option to WINDOWS to create indexes compatible with international language characters.

The Epoch edit specifies the base year of a 100-year period in which all dates containing only two year digits are assumed to fall. The initial default is 1900, causing dates with no century digits to be interpreted as falling within the twentieth century.

The Quick Files edit allows you specify how many quick references to files ReportPro will display on the *File* menu. The value must be a number between 1 and 8.

The Replaceable Database Drivers group allows you to tell ReportPro about additional RDDs that can be used for accessing local tables.