



FormPort Designer[®]

User Guide

Table of Contents

Preface: FormPort Designer©	15
<i>FormPort Designer User Guide Overview</i>	15
<i>Software License Agreement</i>	17
Chapter 1: Introduction	19
<i>What is FormPort Designer?</i>	19
<i>Features and Benefits</i>	19
<i>Spool and CSV files</i>	20
<i>Spool file</i>	20
<i>.CSV file</i>	20
Chapter 2: User Interface	21
<i>Overview</i>	21
<i>File Menu</i>	22
<i>New</i>	22
<i>Open</i>	22
<i>Open Dialog</i>	22
<i>Close</i>	22
<i>Save</i>	23
<i>Save As Dialog</i>	23
<i>Save As</i>	23
<i>Compile Document</i>	24
<i>Compile Dialog</i>	24
<i>Compile Document As</i>	24
<i>Import Document</i>	25
<i>Import Dialog</i>	25
<i>Document Information</i>	25
<i>Document Setup</i>	26
<i>Print Preview</i>	27
<i>Print</i>	27
<i>Send Mail</i>	27
<i>Exit</i>	28
<i>Edit Menu</i>	29
<i>Undo</i>	29
<i>Redo</i>	29
<i>Cut</i>	29
<i>Copy</i>	30

Paste..... 30

Delete..... 30

Edit Text..... 30

Select All..... 31

Filter Selection..... 31

Library / Import..... 31

Library / Export..... 31

Properties..... 32

Default Properties... 32

View Menu 33

 Toolbar→Standard Bar 33

 Toolbar→Align Bar..... 34

 Toolbar→Canvas Bar 34

 Toolbar→Drawing Bar 35

 Toolbar→Layout Bar..... 35

 Toolbar→Nudge Bar..... 36

 Toolbar→Property Bar..... 37

 Toolbar→Rotate Bar..... 37

 Toolbar→Structure Bar..... 38

 Toolbar→Zoom Bar 38

 Toolbar→Font Bar 39

 Status Bar 41

 Ruler..... 41

 Grid 41

 Properties Window..... 41

 Zoom→Zoom 42

 Zoom→Zoom 1:1 42

 Zoom→Zoom Custom..... 42

 Zoom→Zoom To Selection... 43

 Zoom→Fit to Width..... 43

 Zoom→Fit to Height..... 43

 Zoom→Fit to Page..... 43

Page Menu..... 44

 Spool Mapping→Spool File→Load Spool File... 44

 Spool Mapping→Spool File→Reload Spool File 44

 Spool Mapping→Spool File→Unload Spool File 44

 Spool Mapping→Spool File→Display 'Copied' Area 44

 Spool Mapping→Spool File→Display 'Moved' Area 44

 Spool Mapping→Spool File→Display 'Unused' Area 45

<i>Spool Mapping</i> → <i>Spool File</i> → <i>Spool File Options</i>	45
<i>Spool Mapping</i> → <i>CSV File</i> → <i>Load CSV File</i>	45
<i>Spool Mapping</i> → <i>CSV File</i> → <i>Unload CSV File</i>	46
<i>Page Control</i> → <i>Insert Page</i>	46
<i>Page Control</i> → <i>Remove Page</i>	46
<i>Page Control</i> → <i>Go to Page</i>	47
<i>Page Setup</i>	47
<i>Insert Menu</i>	48
<i>Properties</i>	48
<i>General</i>	48
<i>Line</i>	49
<i>Line Spacing</i>	50
<i>Fill</i>	50
<i>Round Corners</i>	51
<i>Font</i>	53
<i>Image Properties</i>	54
<i>Barcode</i>	55
<i>Spool Info</i>	56
<i>Formula</i>	57
<i>CSV Field</i>	58
<i>Date Format</i>	59
<i>Time Format</i>	59
<i>Page Number</i>	60
<i>Numeric</i>	60
<i>Spool Text</i>	61
<i>Spool Image</i>	62
<i>Spool Barcode</i>	62
<i>CSV Spool Text</i>	62
<i>CSV Spool Image</i>	62
<i>CSV Spool Barcode</i>	63
<i>Field</i>	63
<i>Date Field</i>	63
<i>Time Field</i>	63
<i>Page Number Field</i>	64
<i>Numeric Field</i>	64
<i>Rich Text</i>	64
<i>Image</i>	64
<i>Barcode</i>	65
<i>Line</i>	65

Rectangle 65
Round Rectangle 65
Ellipse..... 65
Format Menu..... 67
 Font..... 67
 Justification →Left..... 67
 Justification →Center Horizontally..... 67
 Justification →Right 67
 Justification →Justify 68
 Justification →Top 68
 Justification →Center Vertically 68
 Justification →Bottom 68
 Style..... 68
Tools Menu 69
 Field Filling Order..... 69
 Group Resize... 69
 Group Align 70
 Align Top 70
 Align Middle 70
 Align Bottom 70
 Align Left..... 70
 Align Center..... 70
 Align Right 70
 Group Layout 71
 Space Across 71
 Space Down 71
 Same Width 71
 Same Height..... 71
 Same Size 71
 Group..... 71
 Group Command..... 71
 Ungroup..... 71
 Bring to Front..... 72
 Send to Back 72
 Bring Forward 72
 Send Backward 72
 Nudge..... 72
 Nudge Up 72
 Nudge Down..... 72

Nudge Left	72
Nudge Right.....	73
Rotate.....	73
Free Rotation	73
Left 90 Degrees	73
Right 90 Degrees.....	73
Snap to Grid Command	73
Angle Snap Command.....	73
Auto Select Tools.....	74
Options.....	74
Window Menu	75
Cascade.....	75
Tile	75
Arrange Icons.....	75
Close All.....	75
Open Files List.....	75
Help Menu	76
Contents.....	76
Register FormPort Designer... ..	76
Technical Support.....	76
About.....	76
Chapter 3: Tutorials.....	77
FormPort Designer Tutorial.....	78
Prerequisites	79
Getting Started.....	79
Choose Printer and Paper size	80
Set Grid Size and View.....	81
Creating a Black Box.....	81
Create a Transparent Box	82
Creating Horizontal and Vertical Lines	82
Formatting Graphic Objects	83
Completing the Graphical Layout	84
Creating Text Inside the Black Box	84
Inserting Logos and Signatures.....	87
Mapping Data	88
Add Micr Line.....	89
Create a Copy and Select Tray 2	90
Print the Form.....	90
Multiple Accounts Tutorial.....	91

<i>Determining which Check to Print by Check Number.....</i>	91
<i>Customize the PDF File Name Tutorial</i>	94
<i>Name the file using a FormPort Designer form field.....</i>	94
<i>Customize the PDF file name in FormPort Server</i>	95
<i>E-mail Tutorial.....</i>	96
<i>Using the FormPort Server Email Agent to Send the Entire Job.....</i>	96
<i>Emailing Multiple Jobs to Multiple Recipients.....</i>	97
<i>FAX Tutorial</i>	100
<i>Prerequisites.....</i>	100
<i>Configure the Fax Field in FormPort Designer</i>	100
<i>Configure the Form in FormPort Server</i>	101
<i>Fax a whole job to a single Fax number in FormPort Server</i>	103
<i>Adding Index Fields from FormPort Archive Tutorial.....</i>	104
<i>Exporting the XML File (Document Type) From FP Archive</i>	104
<i>Importing the XML File (Document Type) to FP Designer Form</i>	105
Chapter 4: Procedures and Additional Features.....	109
<i>General Procedures.....</i>	109
<i>Adding Fonts.....</i>	109
<i>Configuring.....</i>	110
<i>Inserting Objects.....</i>	110
<i>Modifying Objects</i>	113
<i>Import Forms Features.....</i>	116
<i>Importing Forms.....</i>	116
<i>Importing an old form.....</i>	116
<i>Importing a new form.....</i>	117
<i>Import Limitations.....</i>	117
<i>Importing from FormPort Designer Version 5</i>	119
<i>Importing from FormPort Designer Version 6</i>	120
<i>Importing Text objects</i>	121
<i>Importing Image objects</i>	121
<i>Importing Spooltext objects</i>	122
Chapter 5: Scripting Commands	125
<i>List of functions</i>	125
<i>ADDFILE.....</i>	125
<i>Syntax:.....</i>	125
<i>Example.....</i>	125
<i>ALLSUBSTITUTE</i>	126
<i>Syntax:.....</i>	126
<i>Example.....</i>	126

<i>AND</i>	127
Syntax:.....	127
Example.....	127
<i>CHAR</i>	128
Syntax:.....	128
Example.....	128
<i>CHOOSE</i>	129
Syntax:.....	129
Example.....	129
<i>CHOOSELIST</i>	129
Syntax:.....	130
Example.....	130
<i>CLEAN</i>	130
Syntax:.....	131
Example.....	131
<i>CODE</i>	131
Syntax:.....	131
Example.....	131
<i>CSVTEXT</i>	132
Syntax:.....	132
Example.....	132
<i>DIR</i>	132
Syntax:.....	132
Example 1.....	133
Example 2.....	133
<i>EURO</i>	133
Syntax:.....	133
Example.....	134
<i>EXACT</i>	134
Syntax:.....	134
Example.....	134
<i>EXIT</i>	135
Syntax:.....	135
Example.....	135
<i>FIND</i>	135
Syntax:.....	136
Example.....	136
<i>FORMAT</i>	136
Syntax:.....	136

Example.....	137
<i>FULLDATE</i>	137
Syntax:.....	137
Example.....	138
<i>FULLTIME</i>	138
Syntax:.....	138
Example.....	139
<i>IF</i>	139
Syntax:.....	139
Example.....	140
<i>INT</i>	140
Syntax:.....	140
Example.....	140
<i>ISBLANK</i>	141
Syntax:.....	141
Example.....	141
<i>ISNUMBER – ISNONTEXT – ISTEEXT</i>	141
Syntax:.....	142
Example.....	142
<i>LEFT</i>	142
Syntax:.....	142
Example.....	143
<i>LEN</i>	143
Syntax:.....	143
Example.....	143
<i>LOWER</i>	143
Syntax:.....	144
Example.....	144
<i>MID</i>	144
Syntax:.....	144
Example.....	144
<i>MODULO</i>	145
Syntax:.....	145
Example.....	145
<i>NOT</i>	146
Syntax:.....	146
Example.....	146
<i>OR</i>	146
Syntax:.....	146

Example.....	147
<i>RAND</i>	147
Syntax:.....	147
Example.....	147
<i>READFILE</i>	148
Syntax:.....	148
Example.....	148
<i>REPLACE</i>	149
Syntax:.....	149
Example.....	149
<i>REPT</i>	150
Syntax:.....	150
Example.....	150
<i>RIGHT</i>	151
Syntax:.....	151
Example.....	151
<i>RLMC</i>	151
Syntax:.....	151
Example.....	152
<i>ROUND</i>	152
Syntax:.....	152
Example.....	152
More Examples.....	152
<i>SEARCH</i>	153
Syntax:.....	153
Example.....	153
<i>SPOOLTEXT</i>	154
Syntax:.....	154
Example.....	154
<i>SPOOLTEXTL</i>	155
Syntax:.....	156
Example.....	156
<i>SPOOLTEXTS</i>	157
Syntax:.....	157
Example.....	157
<i>STOREN</i>	158
Syntax:.....	158
Example.....	158
<i>STORES</i>	159

Syntax:.....	159
Example.....	159
<i>SUBSTITUTE</i>	159
Syntax:.....	159
Example.....	160
<i>SUMCOL</i>	160
Syntax:.....	160
Example.....	161
<i>TEXT</i>	161
Syntax:.....	161
Example.....	162
<i>TRIM</i>	162
Syntax:.....	162
Example.....	162
<i>TRIMC</i>	163
Syntax:.....	163
Example.....	163
<i>UPPER</i>	163
Syntax:.....	163
Example.....	164
<i>VALUE</i>	164
Syntax:.....	164
Example.....	164
<i>WRITEFILE</i>	165
Syntax:.....	165
Example.....	165
<i>Dynamic Text Formatting</i>	167
<i>Commands</i>	167
<i>Text Formatting Example</i>	168
<i>Special Fieldnames</i>	169
<i>Special Names for Fields</i>	171
Chapter 6: Troubleshooting	173
<i>General Problems</i>	173
<i>Poor Quality of Imported Bitmaps</i>	173
<i>Manually Printing a Form</i>	173
<i>Validating the Input File Format</i>	175
<i>Special characters</i>	175
CR-LF.....	175
FF.....	176

Character ASCII '00'	176
Appendix A: FormPort Designer Shortcut Commands	177
Appendix B: FormPort Print Utility	179
Syntax:	179
<i>Option Parameters</i>	180
<i>Return Value</i>	184
<i>Environment Variables</i>	185
Glossary	187
Index	190

Preface: FormPort Designer©

This guide provides extensive explanations and procedures for utilizing every feature of the FormPort Designer software.

FormPort Designer User Guide Overview

This user guide is organized into the following sections:

Chapter 1: Introduction – Provides basic information on what FormPort Designer is and what it can do.

Chapter 2: User Interface – Describes every menu bar and icon of the FormPort Designer user interface.

Chapter 3: Tutorials - Basic – A few tutorials demonstrating some of the basic features of FormPort Designer.

Chapter 4: Procedures and Additional Features – Quick reference for how to perform some of the most-common FormPort Designer functions and operations.

Chapter 5: Scripting Commands – Lists and describes all of the commands and syntax for FormPort Designer’s scripting language.

Chapter 6: Troubleshooting – A list of common problems and their solutions.

The FormPort Designer User Guide also includes an index to enable fast topic navigation, and a glossary to further explain the terms and concepts mentioned throughout this manual.

Throughout this guide, the following icons are used to denote important sections of text:



Highlights procedures that, if not heeded, could damage the product or adversely affect the functionality.



Provides useful information relevant to the described feature or procedure.



Offers a tip, shortcut, or useful information.

Software License Agreement

IMPORTANT: Please read this License carefully before using “FormPort Designer,” hereinafter referred to as “SOFTWARE.” The right to use this SOFTWARE is granted only if you agree to the terms of the license. USE OF THIS SOFTWARE INDICATES YOUR ACCEPTANCE OF THE TERMS AND CONDITIONS OF THE LICENSE AGREEMENT.

In return for the payment of a one time license fee which was included in the purchase price of the Capella Technologies FormPort Designer product, you are granted a non-exclusive right to use the SOFTWARE subject to the following terms and conditions. No title or ownership of the SOFTWARE is conferred with the license.

1. The SOFTWARE may be used without time limit to enable other software products to access the Capella Technologies FormPort Designer features. The SOFTWARE must be used only with Capella Technologies FormPort Designer products.
2. The SOFTWARE may not be disassembled, decompiled, decrypted, or reverse engineered unless prior written consent is either obtained or not required by law. Upon request, the user will provide reasonably detailed information regarding any disassembly, decompilation, decryption, or reverse engineering.
3. This license will automatically terminate upon any transfer of the SOFTWARE or the Capella Technologies product. Upon transfer, you must deliver both the DIMM product, if applicable, and the SOFTWARE, including any copies and all related documentation, to the transferee. The transferee must accept these license terms as a condition to the transfer.
4. We reserve the right to terminate this license upon breach. In the event of termination, all copies of the SOFTWARE must be returned or, with prior written consent, a certificate of destruction of all copies may be provided.

Capella Technologies, Inc.
2099 S. State College Blvd. Suite 500
Anaheim, CA 92806 USA
Phone: 714-385-4900
Fax: 714-385-7936
<http://www.capellatech.com>
support@capellatech.com

Chapter 1: Introduction

What is FormPort Designer?

FormPort Designer 6.0 is a software application intended to create, design, and modify a wide range of forms. The resulting forms are then intelligently merged with raw print data and ultimately printed on standard HP printers.

The FormPort system consists of the following two parts:

- **FormPort Designer**—a WYSIWYG graphical design tool that resides on a PC platform and allows users to create line-box drawings, pictures, signatures, graphics, and text pieces to build a customized form.
- **FormPort Software**—a runtime component that resides on a network server and is a utility that takes an input file, merges and maps it to the graphical form, and outputs a “printable” file on a standard laser printer.

Features and Benefits

FormPort Designer offers the following features and benefits:

- Allows companies to print customized business documents on the newest LaserJet technology without a costly change to their legacy application.
- Graphical business documents can be created on blank laser paper instead of using costly pre-printed forms.
- It is highly customizable, and companies can easily upgrade the look of the business forms at any time with FormPort Designer.
- Provides a quick, programming-free method of replacing old impact printers and pre-printed forms with fast, high quality laser-printed forms.

Spool and CSV files

FormPort Designer can use raw data in the form of either spool or .CSV files. These file types are described in the following sections.

Spool file

Since computers and printers operate at different speeds, raw printer instructions are temporarily stored as spool data. This frees the computer to perform other tasks instead of waiting for the printer to finish. When this printer data is saved to disk, it results in a spool file.

- These spool files can be merged with and mapped to the forms created with FormPort Designer

.CSV file

CSV stands for *Comma Separated Values*. When the values of a spreadsheet are exported as a text file, each line of the spreadsheet is converted to a line, and the data in each spreadsheet cell is separated by a comma. This type of file can then be read by a wide-range of other spreadsheet and database applications.

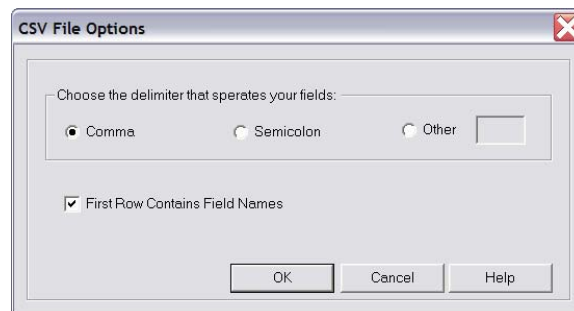
The following illustration shows a sample of a CSV file with a comma separator:

```

1 Number,Name,Address 1,Address 2,Zip Code,City,Country,Date,Currency,Amount,TAX,Discount,Total1,Total2

```

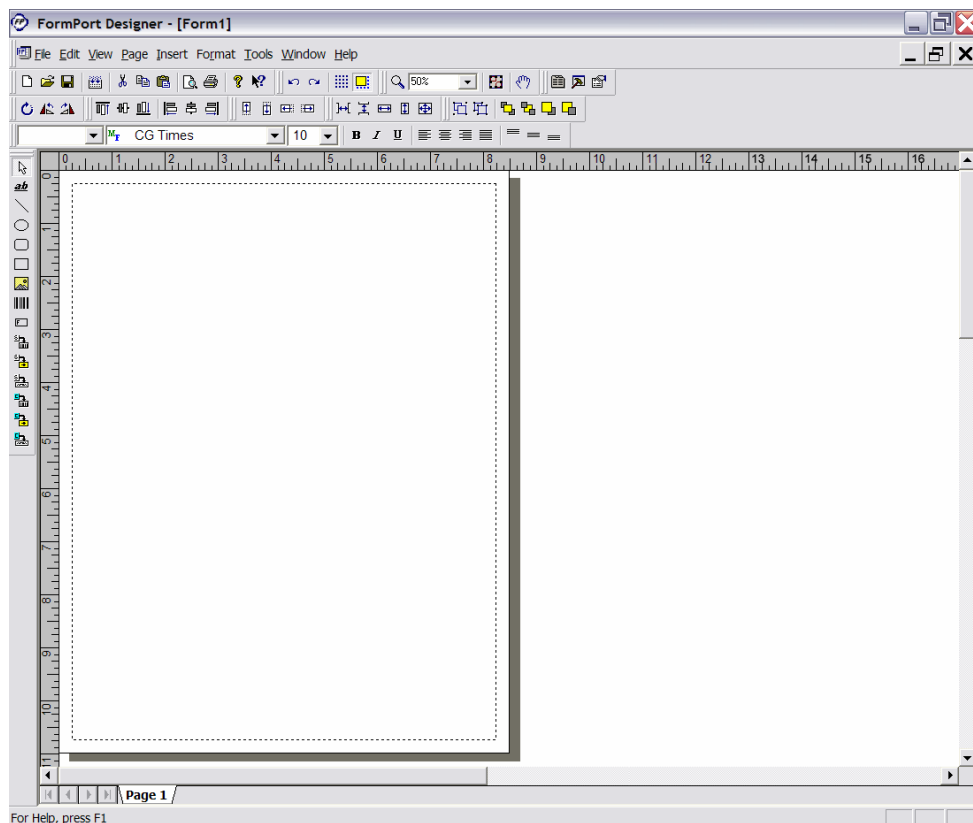
Contrary to the name, cell data in a CSV file might be separated by characters other than commas. In some instances, semicolons are used, or any number of other characters. When importing CSV files into FormPort Designer, you must indicate the manner in which data in the file is separated, as shown in the following dialog box:



Chapter 2: User Interface

Overview

Creating and customizing the look of your forms can be accomplished through FormPort Designer's graphical WYSIWYG interface. The tools, concepts, and functionality of the FormPort Designer interface should be very familiar to people who have used traditional object-oriented drawing programs such as Visio, CoreIDRAW!, and Adobe Illustrator.



In addition to the standard Windows drop-down menus, some of the more common functions also have graphical icons and keyboard shortcuts—all of which will be discussed throughout this chapter.

A complete list of all icons and shortcut keys can be found in *Appendix A: Keyboard Shortcuts*.

File Menu

The **File** menu contains the following commands:

New

This command instructs FormPort Designer to create a new, empty document. **New** can be selected on the **File** menu, or use the following icon or keyboard shortcut:



Open...

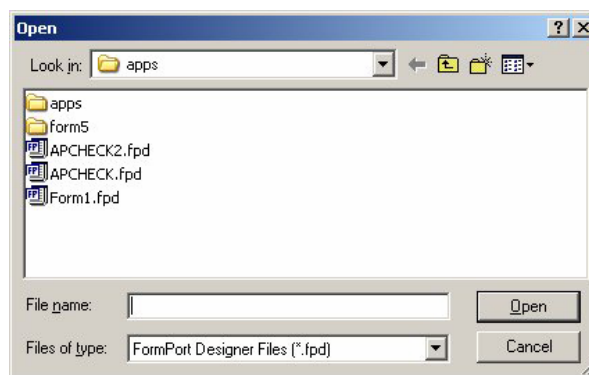
This command opens a document that has previously been saved. **Open** can be selected on the **File** menu, or use the following icon or keyboard shortcut:



Once selected, an **Open** dialog window appears and displays a list of files in the current directory. If you wish to load another file from elsewhere, indicate the correct path.

Open Dialog

To load a document, select it from the list or enter the name of the file in the **File name** field and then click **Open**.



To cancel this operation, click **Cancel**.

Close

This command closes the active document.

If the current document is new, or has changed since it was first opened, you will receive a prompt asking if you'd like to save the document before closing.



If you close a document without saving, all changes made since it was opened or created will be lost.

Save

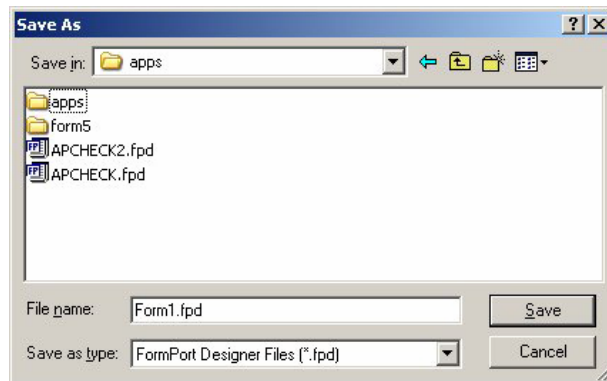
Saves the active document. Use this command to save the latest changes of the active document onto the hard disk.

Selecting this command will open up the **Save As...** dialog, and ask for a document file name if one has not been entered before. **Save** can be selected on the **File** menu, or use the following icon or keyboard shortcut:



Save As Dialog

If the document hasn't been saved yet, this dialog will automatically open and prompt for a filename and path to save the file. Click **Save** to accept the name and path you have chosen.




To cancel this operation, click **Cancel** in this dialog.

Save As...

Saves the active document with a new name. Use this command to save the active document with a new name on the hard disk. Selecting this command will open the **Save As...** dialog.

Compile Document...

After a form has been created, this command compiles the document and saves it in the *.fpf file format. Selecting this option will open the **Compile** dialog. If a name for the compile file hasn't been previously entered, you must now provide one.

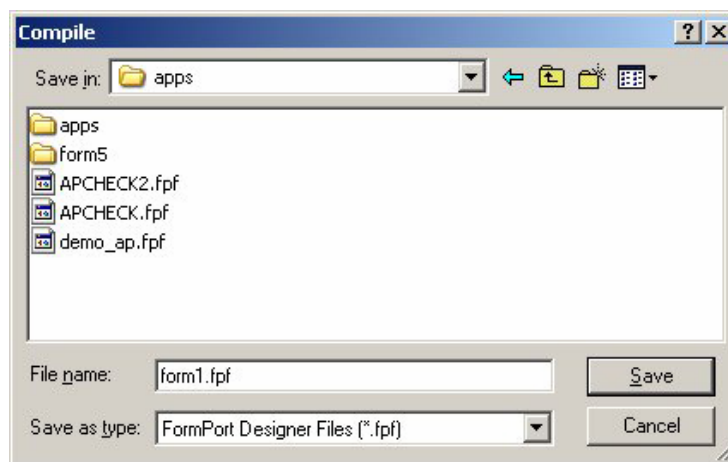
The **Compile** command can also be initiated by clicking on the compile icon () located on the Standard toolbar. Any changes to the document are saved at this point. If the document has not yet been saved, the **Save As...** dialog appears and requests a file name.

Compile Dialog

This dialog opens automatically the first time you compile a document. Enter the name of the file in the **File name** field, and then select **Save** to place the compiled file in the desired location.



Your document must be saved before it can be compiled. If it hasn't been saved, FormPort Designer will prompt you before compiling.



To halt the compile operation, click **Cancel**.

Compile Document As...

This command compiles the active document to a *.fpf file—but with a new name.

Selecting this command will automatically save the last unsaved changes of the active document, and will cause the **Save As...** Dialog to open (if the document has not been saved before).

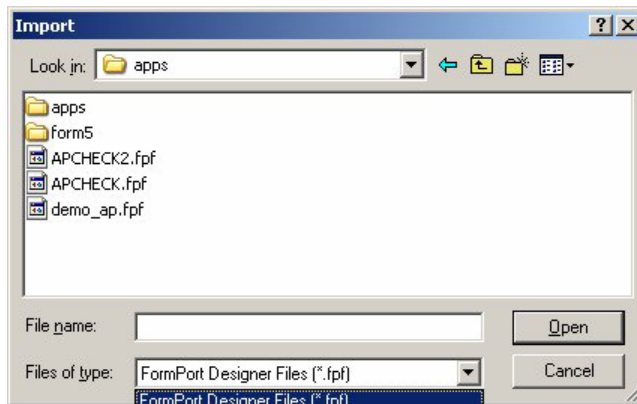
Import Document...

Imports a compiled document into the program. Use this command if you wish to insert a previously-compiled *.mff (MegaForm Designer) or *.fpf (FormPort Designer) file into the active document.

Selecting this command will open up the **Import** dialog and ask for the name and path of the file you wish to import.

Import Dialog

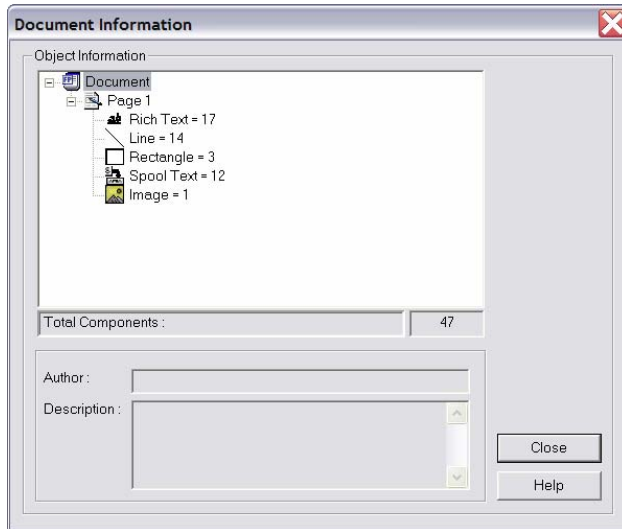
This dialog opens when the **Import Document...** command is selected from the **File** menu. Choose the desired *.mff or *.fpf file (indicating another path, if necessary) and click **Open** to load the file into the active document.



Click **Cancel** if you wish to halt this operation.

Document Information...

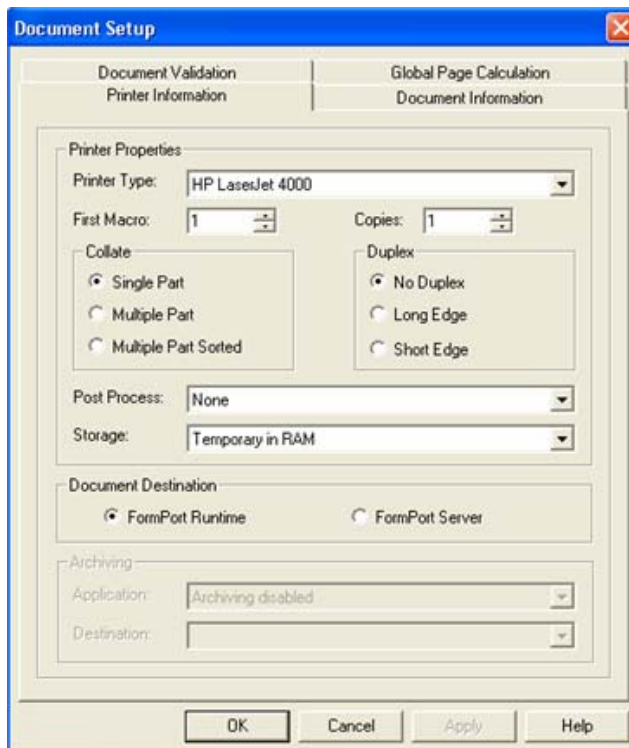
Choosing this command displays a screen that lists all pages in the active document as well as the components contained therein.



This screen is for information purposes only, and cannot be altered directly. Click **Close** when finished.


Document Setup...

Choosing this command displays the active document setup screen, as shown in the figure below:



This property sheet contains the following tabs: **Printer Information**, **Document Information**, **Document Validation**, and **Global Page Calculation**. Information in this section is used during form creation and printing.

Print Preview

Selecting this command or icon  displays an approximation of how your form will look once it is printed. Since the output of different models of printers can vary, you need to define a Printer Type in the **Printer Information Property Page** (located in **Document Setup**).

Also, documents must be saved and compiled before they can be previewed. If neither of these has been done recently, and there are unsaved changes, a dialog box will appear and prompt you to do both.



You can disable this feature by selecting **Options** on the **Tools** menu.

Print...

This command sends the active document to the printer, using the settings specified by selecting **Page Setup** on the **Page** menu.

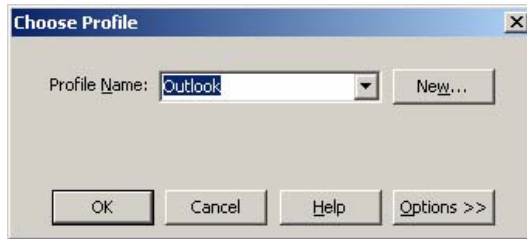
Print can be selected on the **File** menu, or use the following icon and keyboard shortcut:



Send Mail

This command sends the active document through electronic mail as an attachment.

Selecting this command will open up the **Choose Profile** dialog. Select the correct **Profile Name** from the drop-down list, or create a new profile for sending your mail.



The **Choose Profile** dialog will only appear if an email client is not setup on the computer. If you have a working email client set up, a new message will open with the .fpd as an attachment, ready to be emailed.

Exit

This command closes the FormPort Designer application. If there are any unsaved changes in any of the open documents, a **Save As...** dialog will appear and prompt you to save.

To save, enter a file name and click **Save**. Otherwise, click **Cancel** and the application will close without saving any open documents.

Edit Menu

From the **Edit** menu, you can perform common Windows functions such as editing text objects, importing library files, and defining object properties. Every feature of the **Edit** menu is described in the following section.

Undo

Use this command to undo the last action you performed. Continuing to select this command will keep undoing previous actions. By default, FormPort Designer has 15 undo levels, but can have as many as 50. However, increasing the number of undo levels could impact system performance, and should be used with caution. Undo levels can be adjusted by selecting **Options** on the **Tools** menu.

The **Undo** command can be selected on the **Edit** menu, or use the following icon and keyboard shortcut:



Redo

In the event you have chosen **Undo** too many times, **Redo** allows you to step forward through all the undone steps.

Redo can be selected on the **Edit** menu, or use the following icon and keyboard shortcut:



Cut

This command places a copy of the selected item into memory and then deletes the original. Performing this action will replace previously-cut or copied data that is currently stored in memory.

Cut can be selected on the **Edit** menu, or use the following icon and keyboard shortcut:



Copy

This command puts a copy of the selected object into memory. This differs from the **Cut** command because the original object remains intact. Copying an object will replace previously-cut or copied data stored in memory.

Copy can be selected on the **Edit** menu, or use the following icon and keyboard shortcut:



Paste

This command places a copy of the item stored in the clipboard (placed there by using either the **Cut** or **Copy** command) into the current page. If there is no object stored in memory, this command will be grayed-out.

Paste can be selected on the **Edit** menu, or use the following icon and keyboard shortcut:



Delete

This command deletes the current selection. Use this command to delete the selected objects from the active page. Once deleted, an object can only be restored by choosing **Undo**.

Delete can be selected on the **Edit** menu or by pressing the **Delete** key.



Edit Text

This command places text objects into edit mode. When adding a Rich Text object, for instance, you would enter this mode to define the text and apply basic formatting such as font type, size, attributes and color.



You must select a text object in order for this command to be available.

Select All

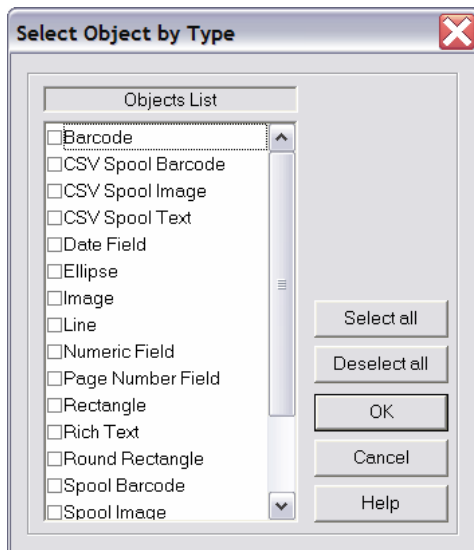
This command selects all the drawing objects available in the active page.

Select All can be selected on the **Edit** menu or use the following keyboard shortcut:



Filter Selection...

This command allows you to select specific types of objects within your document. Select **Filter Selection...** from the **Edit** menu to display the following dialog:



Check the types of object you wish to select and click **OK**.

Library / Import

This command loads previously-saved library objects from the disk.

Choosing this command will open up the **Import Components From Library** dialog asking to enter the needed information for the components.

Library / Export

This command saves the selected components to the hard disk.

Choosing this command will open up the **Export Components To Library** dialog asking to enter the needed information for the components.

Properties...

This command displays the properties related to the currently-selected object. When chosen, it opens the associated property sheet of the component.

The property sheet can be launched by double-clicking on the drawing object or through the following icon, keyboard shortcut and action:



Default Properties...

This command displays the default properties of the active document. These settings are used whenever a new object is created.

Selecting this command opens the **Default Properties** property sheet containing the following pages:

- Line Property Page
- Fill Property Page
- Font Property Page
- Round Corners Property Page

View Menu












The **View** menu allows you to display or hide system toolbars, which provide convenient, single-click access to common functions.

Toolbar→Standard Bar

This command either displays or hides the **Standard** toolbar, as shown in the following figure:



Below is a complete list of icons available in the **Standard** toolbar:

Click	To
	New. Creates an empty new document.
	Open. Opens an existing document.
	Save. Saves the active document to the hard disk.
	Compile. Compiles the active document to the *.fpf file format.
	Cut. Removes selected component(s) from the active document and copies them to the Clipboard. This icon is only enabled if there is at least one selected component.
	Copy. Copies the selected component(s) of the active document to the Clipboard. This icon is only enabled if there is at least one component selected.
	Paste. Inserts a copy of the components(s) stored in the clipboard into the active document.
	Print Preview. Displays an approximation (based on printer selection and settings) of how the document will appear once it is printed.
	Print. Sends the active document to the printer.
	About. Displays the About dialog, containing information about the application and vendor.
	Help. Activates context-sensitive help.







Toolbar→Align Bar

Shows or hides the **Alignment** toolbar, as displayed below:



The icons on the **Alignment** toolbar will only be enabled if there is more than one component selected on the active document.

Below is a complete list of icons available in the **Alignment** toolbar:





Click	To
	Align Top. Moves the selected objects so that the topmost edges are aligned.
	Align Middle. Moves the selected objects so their center areas are aligned vertically.
	Align Bottom. Moves the selected objects so that the bottommost edges are aligned.
	Align Left. Moves the selected objects so that their leftmost edges are aligned.
	Align Center. Moves the selected objects so their center areas are aligned horizontally.
	Align Right. Moves the selected objects so that their rightmost edges are aligned.

Toolbar→Canvas Bar

Shows or hides the **Canvas** toolbar, as displayed below:



Below is a complete list of icons available in the **Canvas** toolbar:
















Click	To
	Undo. Undoes the last change or action performed on a component.
	Redo. Steps forward through undone actions.
	Toggle Grid. Shows or hides the user-defined grid.
	Snap to Grid. Enables or disables grid snap.

Toolbar→Drawing Bar

Shows or hides the **Drawing** toolbar, as displayed below:

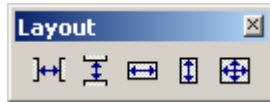


Below is a complete list of icons available in the **Drawing** toolbar:

Click	To
	Select. Selects components for editing or manipulation.
	Rich Text. Inserts a Rich Text object.
	Line. Allows the user to draw a line on the document.
	Ellipse. Adds a user-definable oval object.
	Round Rectangle. Inserts a rectangle with rounded corners into the document.
	Rectangle. Adds a rectangle object.
	Image. Imports an external bitmap image and places it in the document.
	Barcode. Places a barcode in the document.
	Field. Inserts field text object.
	Spool Barcode. Places spool barcode data. This icon will only be available if a spool file is currently loaded.
	Spool Image. Places spool image data. This icon will only be available if a spool file is currently loaded.
	Spool Text. Places spool text data. This icon will only be available if a spool file is currently loaded.
	CSV Spool Barcode. Places CSV barcode data. This icon will only be available if a spool CSV file is currently loaded.
	CSV Spool Image. Places CSV image data. This icon will only be available if a spool CSV file is currently loaded.
	CSV Spool Text. Places CSV text data. This icon will only be available if a spool CSV file is currently loaded.

Toolbar→Layout Bar






Shows or hides the **Layout** toolbar, as displayed below:



NOTE

The icons on this toolbar will only be available if there is more than one component selected on the active document.

Below is a complete list of icons available in the **Layout** toolbar:

Click	To
	Space Across. Moves three or more selected components so that the horizontal space between them is the same.
	Space Down. Moves three or more selected components so that the vertical space between them is the same.
	Same Width. Compares two or more selected objects and changes their width to match that of the last one selected.
	Same Height. Compares two or more selected objects and changes their height to match that of the last one selected.
	Same Size. Compares both the height and width of two or more selected objects and changes their dimensions to match that of the last one selected.

Toolbar→Nudge Bar

Shows or hides the **Nudge** toolbar, as displayed below:







The **Nudge** command enables you to move selected objects in single-pixel increments in any of four different directions.



NOTE

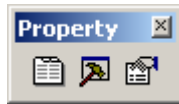
The icons on this toolbar will only be available if there is at least one component selected on the active document.

Below is a complete list of icons available in the **Nudge** toolbar:

Click	To
	Nudge Up. Moves the selected component(s) up one pixel.
	Nudge Down. Moves the selected component(s) down one pixel.
	Nudge Left. Moves the selected component(s) left one pixel.
	Nudge Right. Moves the selected component(s) right one pixel.




Toolbar→Property Bar

Shows or hides the **Property** toolbar, as displayed in the figure below:



The **Property** toolbar allows for quick access to the properties of the currently-selected object.

Below is a complete list of icons available in the **Property** toolbar:

Click	To
	Properties. Displays the associated properties of the selected component(s). This icon is only available if there is at least one component selected.
	Output. Shows or hides the output window
	Properties Window. Shows or hides the property window, displaying the related properties of the active selected component(s).




Toolbar→Rotate Bar

Shows or hides the **Rotate** toolbar, as displayed in the figure below:



The icons on the **Rotate** toolbar will only be available if there is at least one selected component on the active document that supports rotation.

Below is a complete list of icons available in the **Rotate** toolbar:

Click	To
	Rotate. Rotates the selected component(s) to any angle. Use Tools→Angle Snap to restrict rotation to 15 degree increments.
	Rotate Left. Rotates the selected component(s) 90 degrees to the left.
	Rotate Right. Rotates the selected component(s) 90 degrees to the right.







Toolbar→Structure Bar

Shows or hides the **Structure** toolbar, as shown in the figure below:



From the **Structure** toolbar, you can group and ungroup objects, bring objects to the top, send them to back, or move them up or down one layer.

Below is a complete list of icons available in the **Structure** toolbar:





Click	To
	Group. Links the selected components together into a single collection. This icon will only be available if more than one component is selected in the active document.
	Ungroup. Breaks apart a previously-grouped object. This icon is only available when a grouped object is selected.
	Front. Brings the selected component(s) to the top of all the other components.
	Back. Sends the selected component(s) back below all the other objects in the active document.
	Forward. Brings the selected component(s) up one level within the stack of objects on the active document.
	Backward. Sends the selected component(s) down one level within the stack of objects on the active document.

Toolbar→Zoom Bar

Shows or hides the **Zoom** toolbar, as displayed in the figure below:

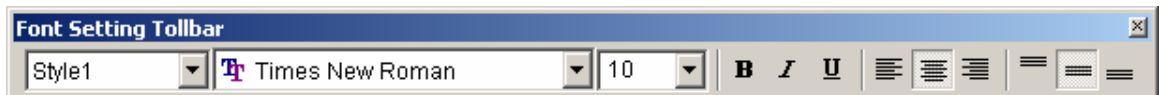


Below is a complete list of icons available in the **Zoom** toolbar:

Click	To
	Zoom. Activates the zoom tool. Clicking on an area of the current document will increase the magnification of that area.
	Current zoom percentage. This value can be set manually to get a specific, desired zoom percentage.
	Zoom to Selection. Displays a view that shows all the components in the active document.
	Pan. Allows the user to move the active document in any direction.


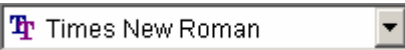











Toolbar→Font Bar

Use this command to show or hide the **Font** toolbar, as displayed below:



The controls on the **Font** toolbar will only be available if all of the selected components support font and alignment features.

Below is a complete list of icons available on the **Font** toolbar:

Click	To
	Style. Indicates the style of the selected component. Can also be used to apply a particular style to a component.
	Font. Indicates the current font of the selected component. Can also be used to change the font.
	Size. Indicates the font size of a component, and allows for the font size to be changed.
	Bold. Indicates if the selected font has a bold attribute assigned, or can be used to add or remove a bold attribute.
	Italic. Indicates if the selected font has an italic attribute assigned, or can be used to add or remove an italic attribute.
	Underline. Indicates if the selected font has an underline attribute assigned, or can be used to add or remove an underline attribute.
	Left Align. Indicates if the selected text object is left-justified, or can be used to change the current formatting to left-justified.
	Center Horizontally. Indicates if the selected text object is horizontally centered, or can be used to change the current formatting to centered.
	Right Align. Indicates if the selected text object is right-justified, or can be used to change the current formatting to right-justified.
	Justify. Indicates if the selected text object is fully-justified, or can be used to change the current formatting to fully-justified.
	Top Align. Indicates if the selected text object is aligned with the top, or to can be used change the current text positioning to top-aligned.
	Center Vertically Align. Indicates if the selected text object is aligned in the center, or can be used to change the current text positioning to center-aligned.
	Bottom Align. Indicates if the selected text object is aligned with the bottom, or can be used to change the current text positioning to bottom-aligned.

Status Bar

Shows or hides the FormPort Designer **Status** bar, as displayed in the following figure:



The left area of the **Status** bar describes the actions of menu items when you navigate through the menus using the arrow keys. Similarly, this area shows messages that describe the actions of toolbar icons as you click them, and before releasing them.

If, after viewing the description of the toolbar icon command, you decide not to execute the command, then release the mouse button while the pointer is off the toolbar icon.

The right areas of the status bar indicate which of the following keys are latched down:

Indicator	Description
CAP	The Caps Lock key is currently latched down.
NUM	The Num Lock key is currently latched down.
SCROLL	The Scroll Lock key is currently latched down.

Ruler

Shows or hides the interface ruler.

The **Ruler** is used to show the unit of measurement in the active document, and to facilitate the creation of components with specific sizes.

Grid




Shows or hides a grid  in the work area.

The grid is useful for placing components at precise locations within a document.

Properties Window

Shows or hides the **Properties** Window  as shown below.

The **Properties** Window is a dockable window that shows the properties of the selected component(s) in the active document. The properties shown in this window can either be changed directly, or through the traditional object-editing method.

Name	Field1
Field Type	Text Field
Border Style	Solid
Line Transparent	Yes
Border Width	0.000000
Border Color	 0, 0, 0
Font	CG Times
Fill Transparent	Yes
Fill Pattern	Solid
Fill Color	 0, 0, 0
Back Color	 255, 255, 255
Line Spacing	0.125000
Validation Formula	
Calculation Formula	
Rotation	None
Left	0.687500
Top	0.791667
Width	6.833333
Height	2.145833

Zoom→Zoom

Activates the **Zoom** tool .

After clicking on this icon, the cursor will change to a magnifying glass. Left-clicking anywhere in the document will increase the magnification in 25% increments (to a maximum of 650%). Right-clicking will zoom out in 25% increments (to a minimum of 25%). You can zoom a specific region by clicking and dragging to define the area and then releasing the mouse button.

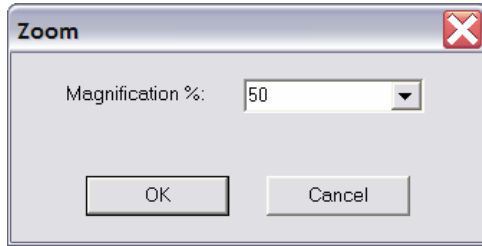
Zoom→Zoom 1:1

Zooms the document to the normal scale of 100%.

Zoom→Zoom Custom...


This command is used to zoom in 25% increments.

When selected, the custom zoom dialog appears, as shown below:



Select a value from the drop-down menu and then click **OK** to initiate. Clicking on **Cancel** will abort the custom zoom procedure.

Zoom→Zoom To Selection...

This tool  determines the perfect zoom level necessary to fill the screen with the currently-selected object.



This feature is only available if at least one object is selected.

Zoom→Fit to Width

Use this command to set the zoom of the document so that all of the components are visible in the screen horizontally.


Zoom→Fit to Height

Use this command to set the zoom of the document so that all of the components are visible in the screen vertically.

Zoom→Fit to Page

Use this command to change the view of the active document so that all of the components are visible on the screen both horizontally and vertically.



To exit the **Zoom** feature, click the select arrow  on the **Drawing** toolbar.

Page Menu

From the **Page** menu, you can map both spool and CSV files, define page properties, add or remove pages, and jump to a specific page in a multi-page document. For detailed descriptions of these features, proceed to the following section.

Spool Mapping→Spool File→Load Spool File...

This command loads a spool file into the active document. Before you can insert a spool object, a spool files must first be loaded. This menu option is only available when no spool or CSV file is currently loaded in the active document.

Once selected, a **File Open** dialog appears and asks for the name and location of the spool file to be loaded.

Spool Mapping→Spool File→Reload Spool File

This command *reloads* a previously-loaded spool file, and is only available if a spool file has already been loaded.

Spool Mapping→Spool File→Unload Spool File

This command *unloads* a previously-loaded spool file (and all associated spool objects) from the active document, and is only available if a spool file has already been loaded.

Spool Mapping→Spool File→Display 'Copied' Area

This command shows or hides the original spool data after it has been copied and manipulated within FormPort Designer. This old data is normally displayed in blue text.



This command can also be invoked by pressing the F3 key.

Spool Mapping→Spool File→Display 'Moved' Area

This command shows or hides the original spool data after it has been moved and manipulated within FormPort Designer. This old data is normally displayed in red text.



This command can also be invoked by pressing the F2 key.

Spool Mapping→Spool File→Display ‘Unused’ Area

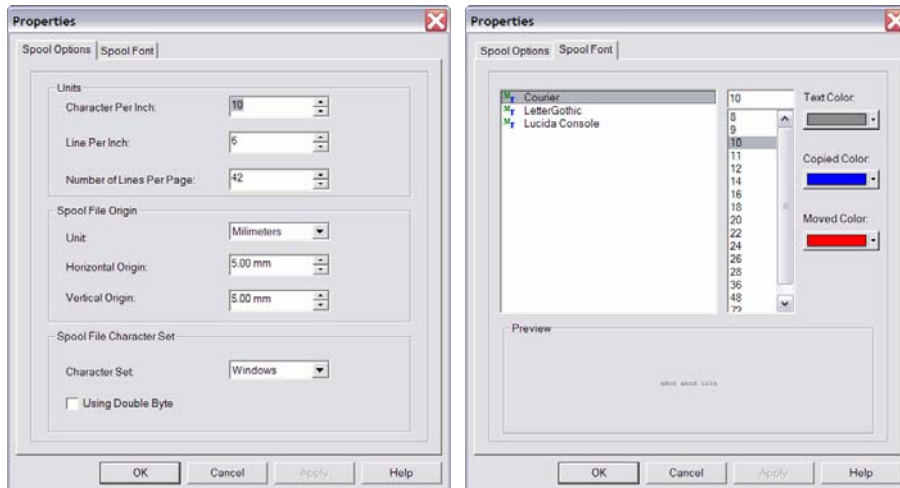
Use this command to show or hide the portions of the spool file that will not be used in the final document. This data is normally displayed in gray text.



This command can also be invoked by pressing the F4 key.

Spool Mapping→Spool File→Spool File Options

Use this command to change the spool file options, as shown in the following images:

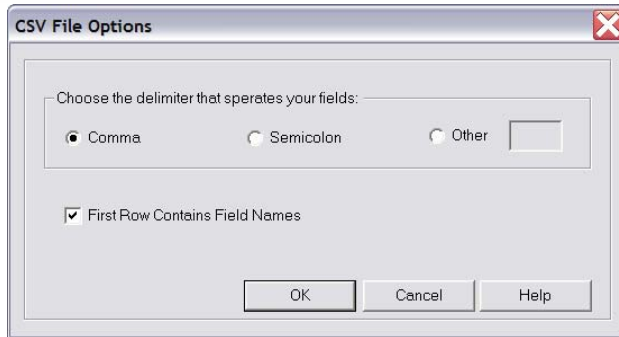


From the **Properties** dialog, you can define the appearance, font, and color of spool data within your document.

Spool Mapping→CSV File→Load CSV File

Use this command to load a CSV file into the active document. Loading a CSV file is required before a CSV object can be inserted.

This menu command will only be available if there is no spool file or CSV file already loaded into the document.



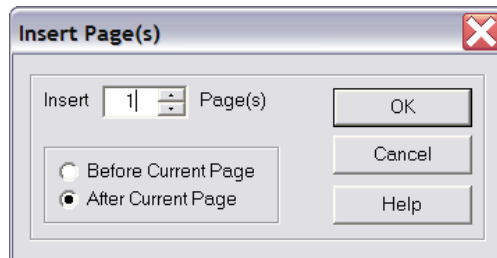
When loading a new CSV file, you must tell FormPort Designer which method was used to separate the data in the file (comma, semicolon, etc.).

Pool Mapping→CSV File→Unload CSV File

Use this command to unload a previously loaded CSV file (and all associated CSV objects) from the active document. This command is only available when a CSV file has already been loaded.

Page Control→Insert Page

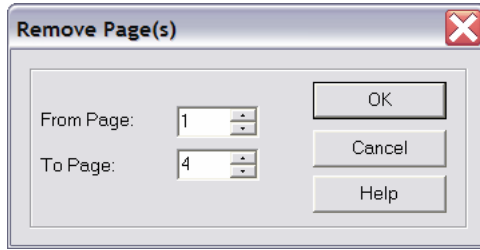
Use this command to insert new empty pages into the active document.



Multiple pages can be added either before or after the current page.

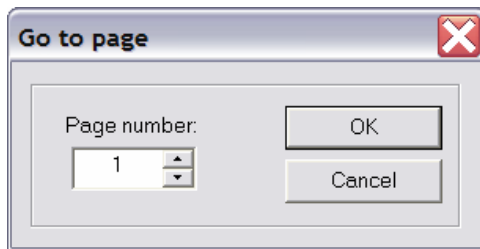
Page Control→Remove Page

Use this command whenever you need to remove a page or a specific number of pages (and all the components currently on those pages) from the active document.



Page Control → Go to Page

Use this command to go directly to a specific page number within the active document.

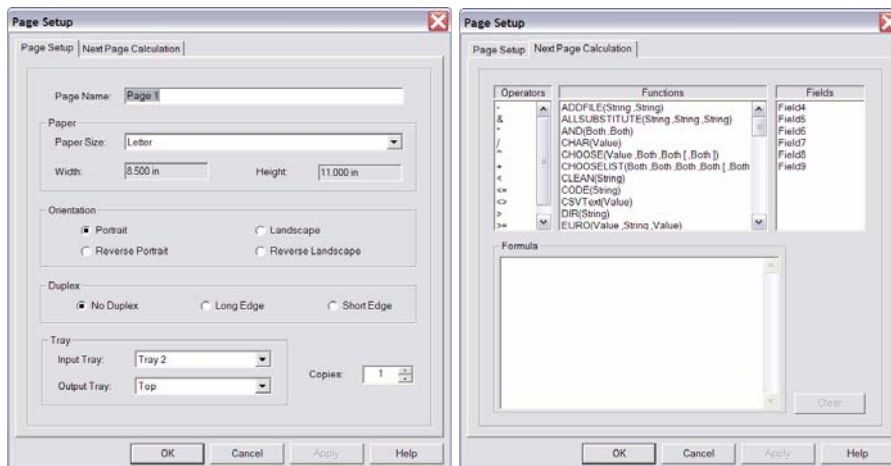


Enter the page number in the **Page number** field and then press **OK** to go to that page.

Page Setup

Use this command to modify the properties associated with the current page of the active document. These settings can be the same for every page in a multi-page document, or completely unique for each page.

The image below illustrates both tabs of the **Page Setup** dialog:



Insert Menu

From this menu you can insert external data files into your FormPort Designer form.

Properties

Once an object has been inserted into a document, various attributes and properties can be assigned to further define its appearance and function. This section describes each of these properties.

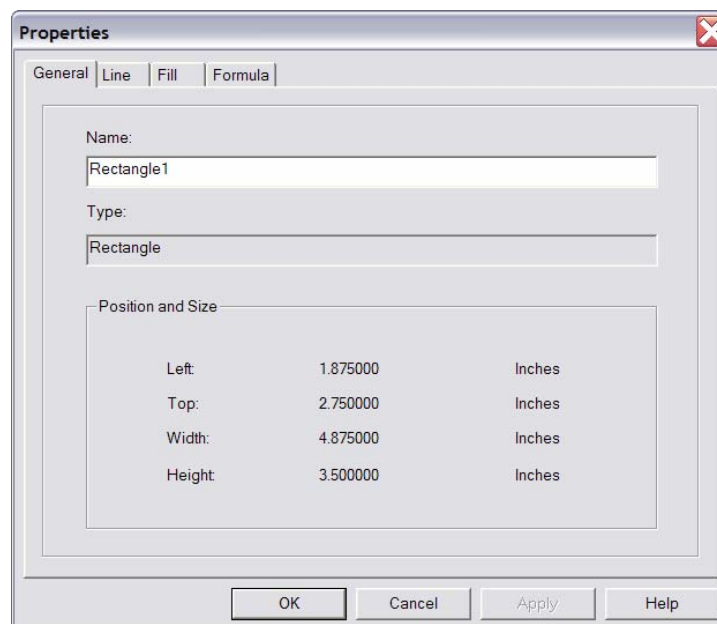
To access the **Properties** dialog, right-click on the object.



The properties available depend on the type of object selected.

General

General properties are displayed on this page.

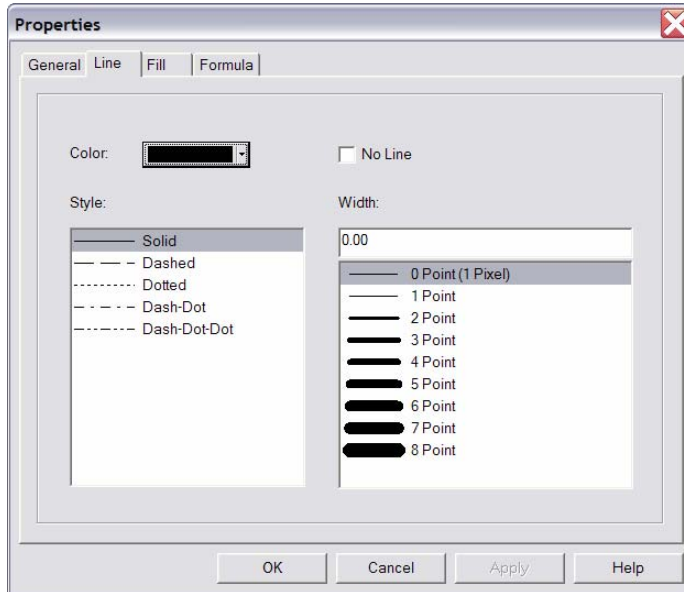


This page contains the information about the component name, component type, and component position and size.

This page is for informational purposes only, and other than the **Name** field, these values cannot be edited directly.

Line

This page contains the information related to the line settings of components with borders.



Color

This icon determines the border color of the component. Click on it to see a popup window and other color choices.

No Line

Check this option if you want the component border to be invisible.

Style

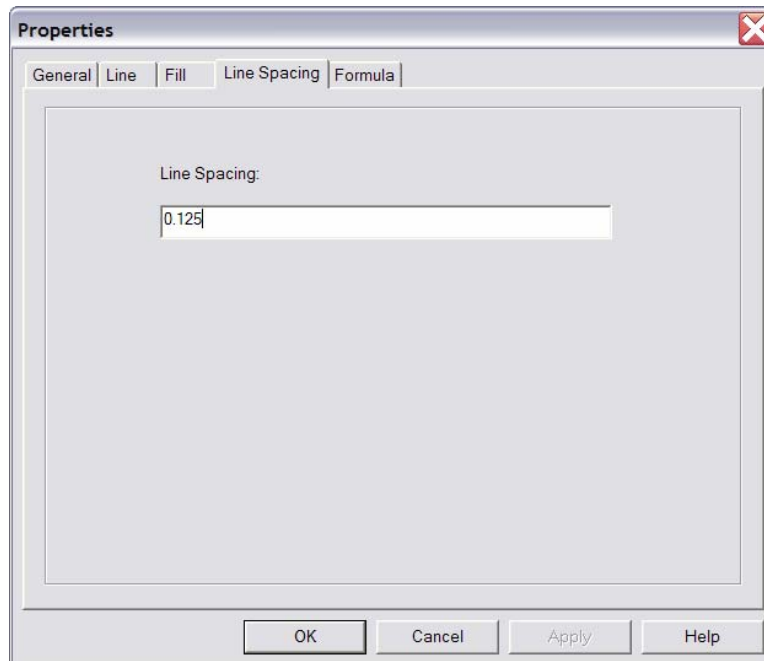
The style of single-pixel borders can be selected from this list. If the current border is thicker than one pixel, this list will be grayed-out.

Width

The width of solid borders can be determined through this list. If the border is using a style other than solid, this list will be grayed-out and not available.

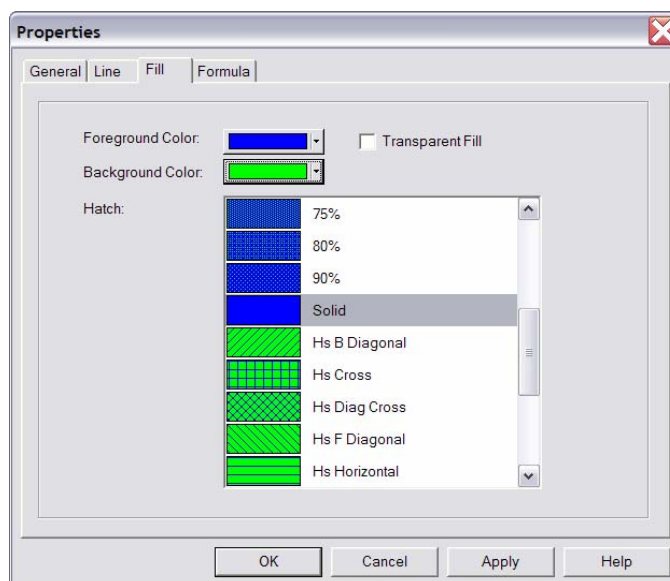
Line Spacing

This page contains the information related to the field spacing. Enter the correct line spacing for the field object in the related edit box.



Fill

This page contains the information related to the manner in which the component can be filled.



Foreground Color

This color icon indicates the foreground color of both solid and non-solid hatch patterns used to fill components. Click the icon and select the desired color from the popup window. This option is not available when **Transparent Fill** is checked.

Background Color

This color icon indicates the background color in hatch patterns. If the current fill is solid, this color is not used. Like foreground color, this option is not available when **Transparent Fill** is checked.

Transparent Fill

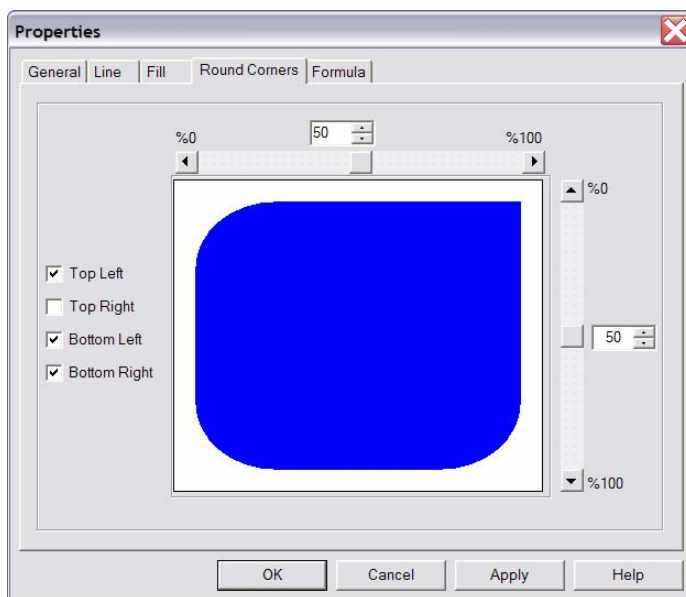
This option turns the non-border region of components transparent. When this box is checked, the color and hatch options are not available.

Hatch

Displays the patterns available for components that support the fill attribute. Fills can be either solid or a number of two-color patterns. This option is not available when **Transparent Fill** is checked.

Round Corners

This page contains the information related to the rounded corners of the round rectangle object.



Top Left

If checked, indicates the **Top Left** corner will be rounded.

Top Right

If checked, indicates the **Top Right** corner will be rounded.

Bottom Left

If checked, indicates the **Bottom Left** corner will be rounded.

Bottom Right

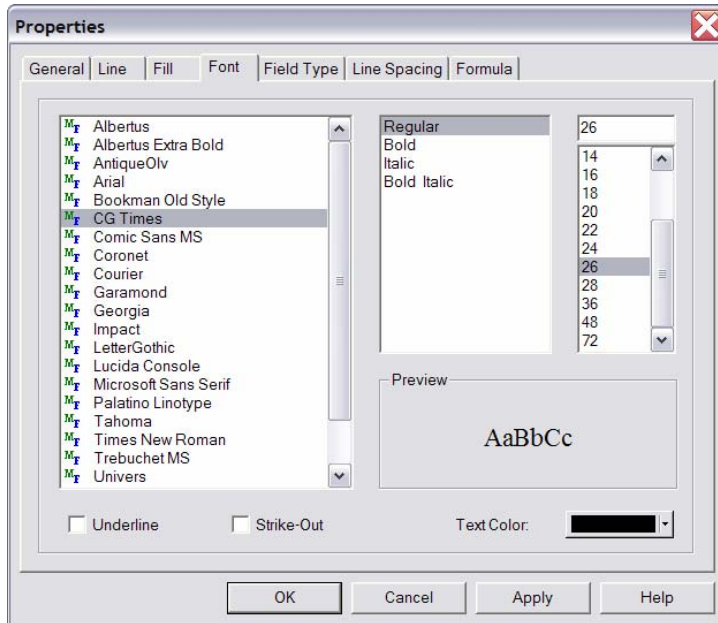
If checked, indicates the **Bottom Right** corner will be rounded.

Percentage

Determines the amount of curve to apply to the edges of the object. Both of these values are linked; changing one automatically changes the other.

Font

This page contains the information related to the font properties of supported components.



Font Name

This list indicates the currently-assigned font as well as other fonts available.

Font Style

Indicates the current style as well as the other available styles for assignment.

Font Size

Shows the current font size and a list of all the sizes available.

Underline

Selecting this checkbox will add an underline to the current text object.

Strike-Out

Selecting this checkbox will add a line through the current text object.

Text Color

This icon defines the color of the current text object. Click it to display a popup window with other color selections. Most text objects are black by default.

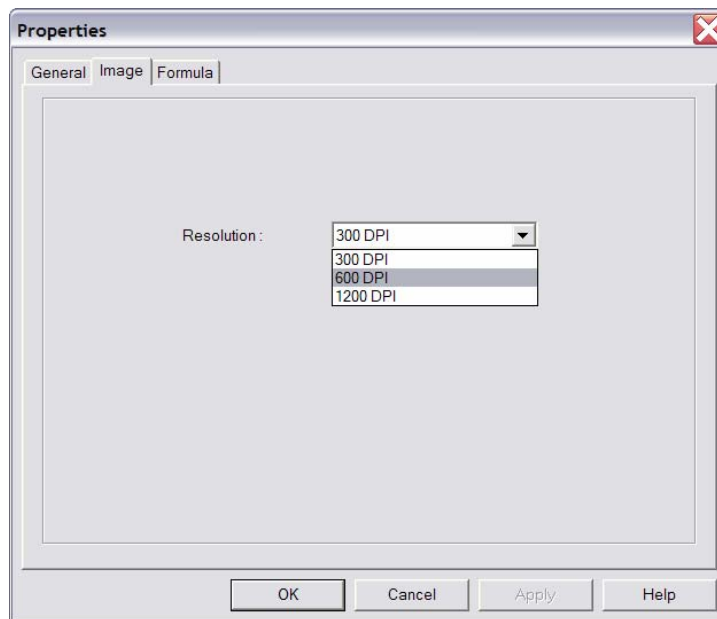
Preview

This area provides a sample of what the resulting text object will look like.

After making any changes, click either **OK** or **Apply** to accept them, or **Cancel** to reset the text object to its original configuration.

Image Properties

This page allows you to define the resolution of an imported image.



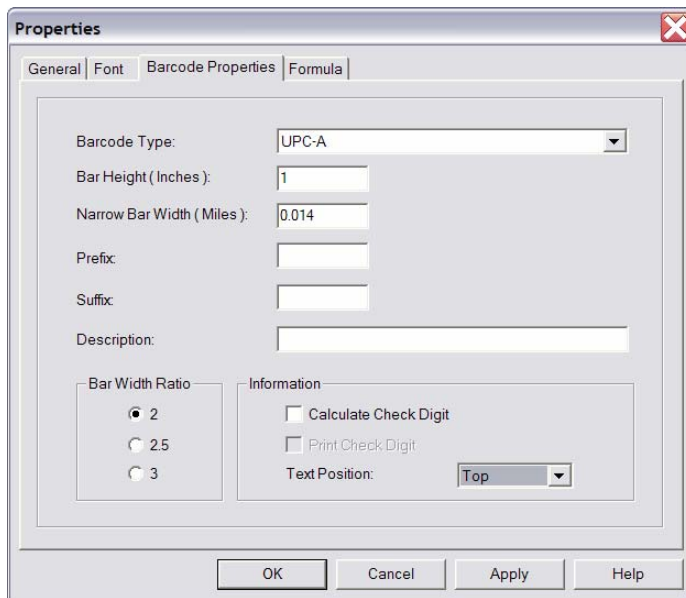
Select the desired resolution from the drop-down list and choose **OK**.



Increasing the resolution of an image will make it appear to shrink.

Barcode

This page contains the information related to the barcode settings.



Barcode Type

Indicates the type of the current barcode object. This value can be changed through the drop-down menu.

Bar Height

Specifies the height of the barcode.

Narrow Bar Width

Determines the width of the barcode.

Prefix

Barcode prefixes can be entered in this field.

Suffix

Barcode suffixes can be entered here..

Description

Enter a user-defined description of the current barcode here.

Bar Width Ratio

Sets the barcode width ratio to either 2, 2.5, or 3.

Calculate Check Digit

Should be checked to enable the **calculate check digit** feature.

Print Check Digit

Check to enable the **print check digit** option.



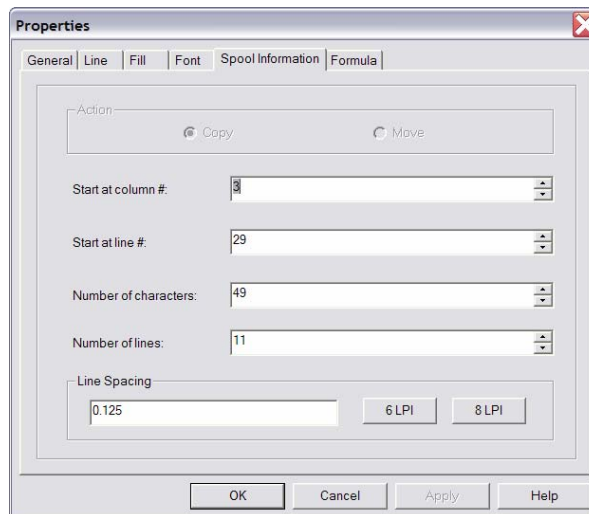
Print Check Digit will be grayed-out unless the Calculate Check Digit option is checked.

Text Position

Choose between aligning text along the top or bottom of the barcode.

Spool Info

This page contains the information related to the spool objects settings.



Action

This section indicates how the spool data was manipulated after being inserted. This option is only available at insertion time and is only shown for informational purposes.

Start at column

Displays the starting column number of the selected text area.

Start at line

Displays the starting line number of the selected text area.

Number of characters

Displays the number of characters in the widest available line in the selected text area.

Number of lines

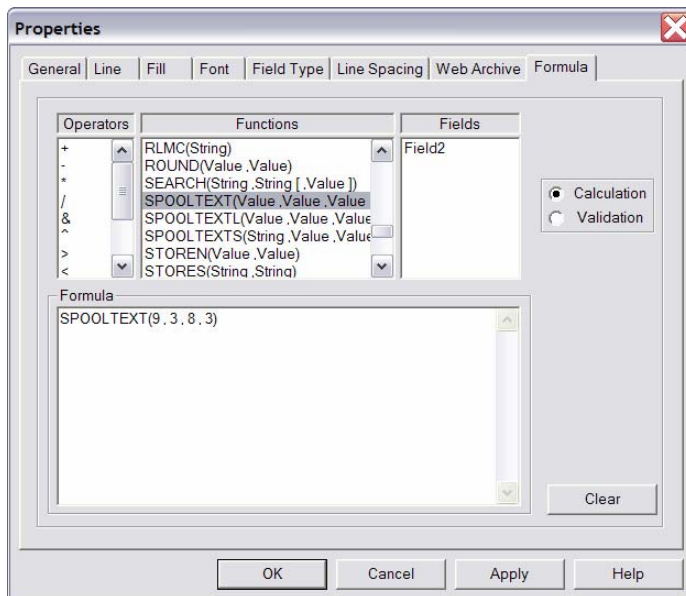
Displays the maximum number of lines that will appear in the spool object.

Line Spacing

Displays the space between each line of the spool object in LPI (Line-Per-Inch) units. The two icons (labeled 6 LPI and 8 LPI) are simply shortcut icons used to enter the specified values automatically.

Formula

This page contains information about any formulas assigned to the selected object.



Operators

Displays the list of operators. Double-click any of these to append them to the object's formula.

Functions

Displays the list of functions. Double-click any of these to append them to the object's formula.

Fields

Displays the list of fields in the active document. Double-click any of these to append them to the object's formula.

Formula

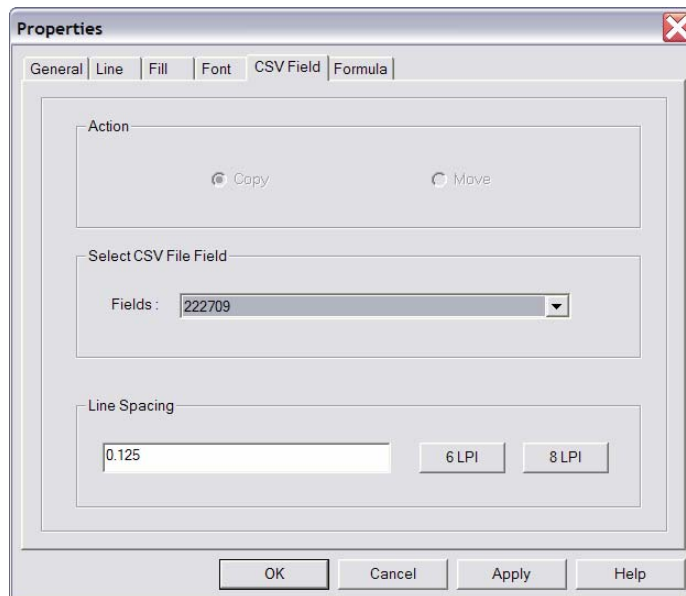
Displays the object's final formula. This formula can also be entered or edited manually by typing in the Formula edit box.

Calculation & Validation

This part of the page will only be visible if the object supports both the Calculation Formula and Validation Formula. If present, ensure the correct value is set.

CSV Field

This page contains the information related to the CSV files that have been mapped and inserted into the document.

**Action**

This section indicates how the CSV data was manipulated after being inserted. This option is only available at insertion time and is only for informational purposes.

Fields

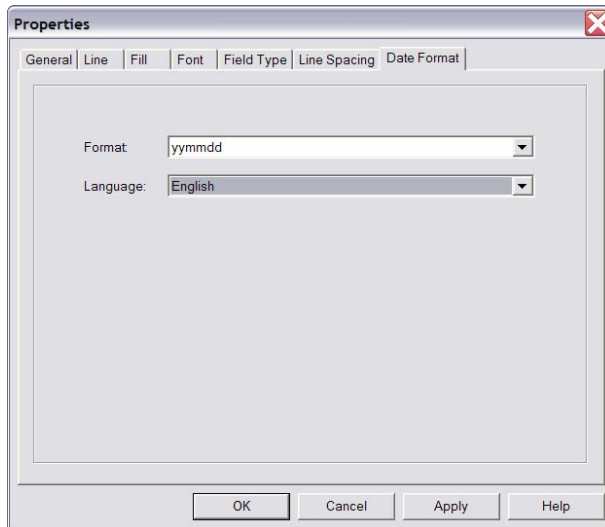
This drop-down menu lists the separated objects from the first line of the CSV file. Choose the one you wish to display.

Line Spacing

Displays the space between each line of the spool object in LPI (Line-Per-Inch) units. The two icons (labeled 6 LPI and 8 LPI) are simply shortcut icons used to enter the specified values automatically.

Date Format

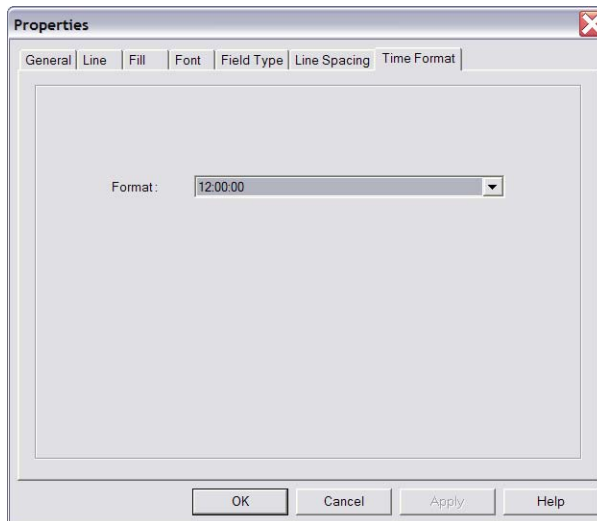
This page allows you to choose a specific date format.



Select a **Format** and **Language** from the drop-down menus and then click **OK** to accept.

Time Format

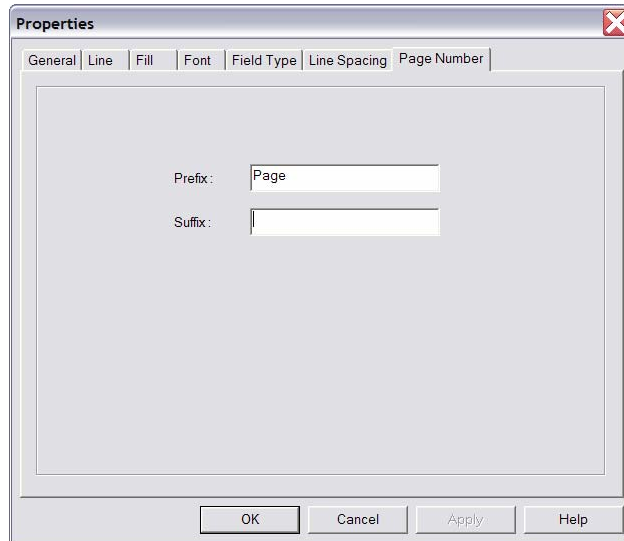
This page allows you to pick a specific time format.



Select the desired time **Format** from the drop-down menu and then click **OK** to accept.

Page Number

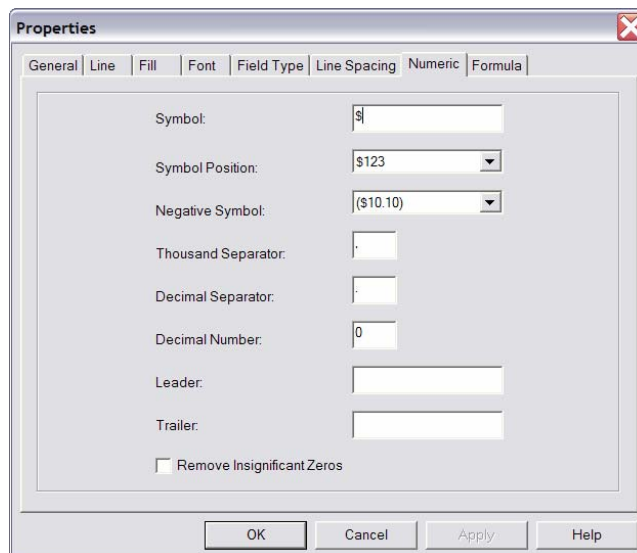
This allows you to define the look of the page number object.



Enter a **Prefix** and **Suffix** into the appropriate edit field and then click **OK** to accept.

Numeric

This page contains the information related to the numeric field.



The following settings are available through the numeric page:

Symbol

Defines the currency symbol to display with monetary values.

Symbol Position

Indicates where the monetary symbol should be placed in relation to the amount.

Negative Symbol

Allows you to specify where a negative symbol (if used) will be placed in relation to the amount.

Thousand Separator

Enter the character you would like to serve as a separator for values in excess of a thousand.

Decimal Separator

Enter the character you would like to separate whole and decimal numbers.

Decimal Number

Specify the number of digits to follow the decimal separator symbol.

Leader

Enter a Leader string of up to 12 characters.


Trailer

Enter a Trailer string of up to 12 characters.

Remove Insignificant Zeros

Check this checkbox if you wish the insignificant zeros in the numbers to be removed from the monetary amounts.


Spool Text

Selecting this command or icon  inserts a spool text component into the active page. This command is only active if there is a spool file currently loaded into the active document.

The **Spool Text** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Spool Info Property Page, Formula Property Page


Spool Image

Selecting this command or icon  inserts a spool image component into the active page. This command is only active if there is a spool file currently loaded into the active document.

The **Spool Image** component supports the following properties:

General Property Page, Spool Info Property Page, Image Property Page, Formula Property Page


Spool Barcode

Selecting this command or icon  inserts a spool barcode component into the active page. This command is only active if there is a spool file currently loaded into the active document.

The **Spool Barcode** component supports the following properties:

General Property Page, Font Property Page, Spool Info Property Page, Barcode Property Page, Formula Property Page


CSV Spool Text

Selecting this command or icon  inserts a CSV spool text component into the active page. This command is only active if there is a CSV file currently loaded into the active document.

The **CSV Spool Text** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, CSV Field Property Page, Formula Property Page


CSV Spool Image

Selecting this command or icon  inserts a CSV spool image component into the active page. This command is only active if there is a CSV file currently loaded into the active document.

The **CSV Spool Image** component supports the following properties:

General Property Page, CSV Field Property Page, Image Property Page, Formula Property Page


CSV Spool Barcode

Selecting this command or icon  inserts a CSV spool barcode component into the active page. This command is only active if there is a CSV file currently loaded into the active document.

The **CSV Spool Barcode** component supports the following properties:

General Property Page, Font Property Page, CSV Field Property Page, Barcode Property Page, Formula Property Page

Field

Selecting this command or icon  inserts a field component into the active page. Although the default border for this component is transparent, a virtual red border appears around the field object to indicate the bounding rectangle and assist placement.

The **Field** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Field Type Property Page, Line Spacing Property Page, Formula Property Page

Date Field

Selecting this command inserts a date field component into the active page. Although the default border for this component is transparent, a virtual red border appears around the field object to indicate the bounding rectangle and assist placement.

The **Date Field** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Field Type Property Page, Line Spacing Property Page, Date Format Property Page

Time Field

Selecting this command inserts a time field component into the active page. Although the default border for this component is transparent, a virtual red border appears around the field object to indicate the bounding rectangle and assist placement.

The **Time Field** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Field Type Property Page, Line Spacing Property Page, Time Format Property Page

Page Number Field

This command inserts a page number field component into the active page. Although the default border for this component is transparent, a virtual red border appears around the field object to indicate the bounding rectangle and assist placement.

The **Page Number Field** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Field Type Property Page, Line Spacing Property Page, Page Number Property Page


Numeric Field

This command inserts a numeric field component into the active page. Although the default border for this component is transparent, a virtual red border appears around the field object to indicate the bounding rectangle and assist placement.

The **Numeric Field** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Font Property Page, Field Type Property Page, Line Spacing Property Page, Numeric Property Page, Formula Property Page


Rich Text

Choosing this command or icon  inserts a text component field into the active page. After insertion, a Rich Text component automatically goes into edit mode.

The **Rich Text** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Line Spacing Property Page, Formula Property Page


Image

Choosing this command or icon  inserts an external image component into the active page.

The **Image** component supports the following properties:

General Property Page, Image Property Page, Formula Property Page


Barcode

Use this command or icon  to insert a barcode component into the active page.

The **Barcode** component supports the following properties:

General Property Page, Font Property Page, Barcode Property Page, Formula Property Page


Line

Use this command or icon  to insert a line component into the active page.

The **Line** component supports the following properties:

General Property Page, Line Property Page, Formula Property Page


Rectangle

Selecting this command or icon  inserts a rectangle component into the active page.

The **Rectangle** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Formula Property Page


Round Rectangle

Use this command or icon  to insert a round rectangle component into the active page.

The **Round Rectangle** component supports the following properties:

General Property Page, Line Property Page, Fill Property Page, Round Corners Property Page, Formula Property Page

Ellipse

Selecting this command or icon  inserts an ellipse component into the active page.

The **Ellipse** component supports the following properties:

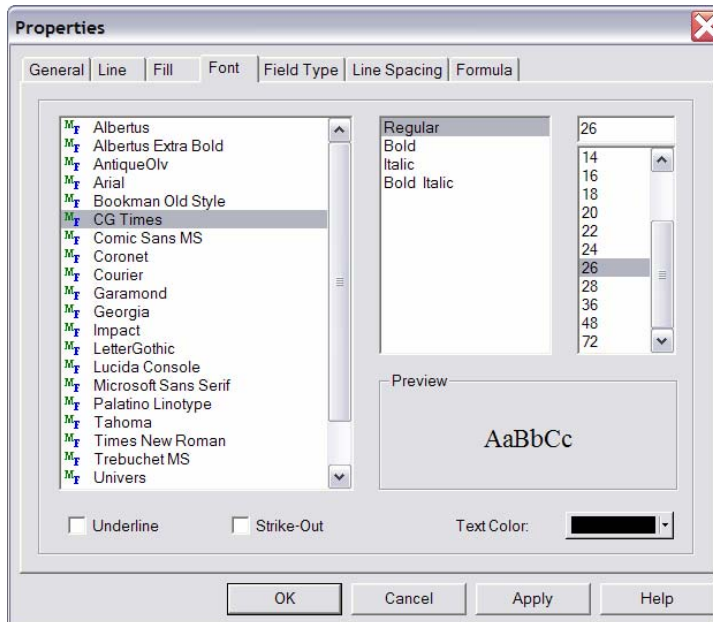
General Property Page, Line Property Page, Fill Property Page, Formula Property Page

Format Menu

The **Format** menu enables you to modify the look of text objects.

Font

Use this command to display and set the font attributes of the currently-selected text object(s). This command will open the component property sheet, with the **Font** property page tab selected, as shown below:



Justification→Left

Use this command to align the *left* edge of the currently-selected text component(s).

Justification→Center Horizontally

Use this command to *center* the currently-selected text component(s).

Justification→Right

Use this command to align the *right* edge of the currently-selected text component(s).

Justification→Justify

Use this command to align *both* the *left* and *right* edges of the currently-selected text component(s).

Justification→Top

Use this command to align the text in the currently-selected text component against the *top* border of the object.

Justification→Center Vertically

Use this command to vertically *center* the text in the currently-selected text component within the borders of the object.

Justification→Bottom

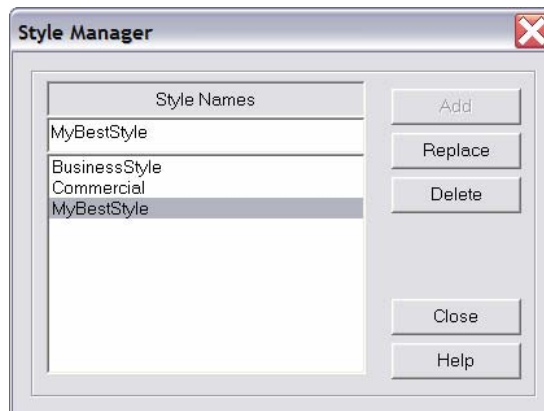
Use this command to align the text in the currently-selected text component against the *bottom* border of the object.

Style...

Use this command to open the **Style Manager** dialog window.



You must have a text object selected for the **Style...** command to be enabled.



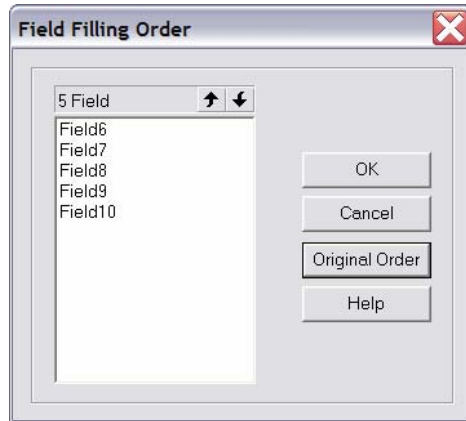
After you have modified the appearance of a text object, you can give a name to that look and save it in the style manager. This style can then be applied to other text objects to save time and ensure a consistent look.

Tools Menu

Under the **Tools** menu, you can further refine and optimize the components that make up your form.

Field Filling Order...

In this dialog window, you can specify the order in which the text fields will be filled.

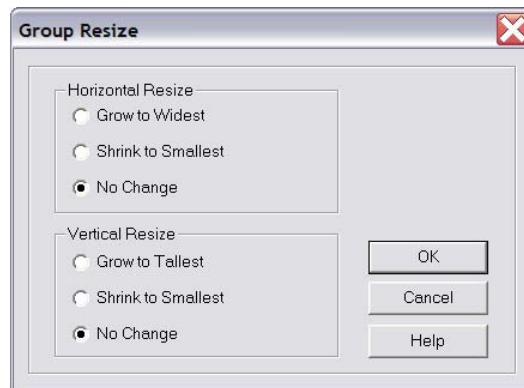


There must be at least two fields in the document for this option to be available.

To change the order, select a field and then use either the up or down arrow keys to move it around the list. Click **OK** to accept the new order, or **Cancel** to revert back to the original order.

Group Resize...

This feature allows you to resize multiple objects in a single step. Resizing can be performed horizontally, vertically, or both dimensions at once.







At least two objects must be selected for this command to be enabled.

Group Align


Align Top

Use this command or toolbar icon  to move the selected object(s) into alignment with the topmost portion of the last selected object. At least two objects must be selected for this command to be available.


Align Middle

Use this command or toolbar icon  to move the selected object(s) into horizontal alignment with the last selected object. At least two objects must be selected for this command to be available.


Align Bottom

Use this command or toolbar icon  to move the selected object(s) into alignment with the bottommost portion of the last selected object. At least two objects must be selected for this command to be available.


Align Left

Use this command or toolbar icon  to move the selected object(s) into alignment with the leftmost portion of the last selected object. At least two objects must be selected for this command to be available.

Align Center


Use this command or toolbar icon  to move the selected object(s) into vertical alignment with the last selected object. At least two objects must be selected for this command to be available.

Align Right


Use this command or toolbar icon  to move the selected object(s) into alignment with the rightmost portion of the last selected object. At least two objects must be selected for this command to be available.

Group Layout


Space Across

Use this command or toolbar icon  to equally space three or more selected objects *horizontally* on the current page.


Space Down

Use this command or toolbar icon  to equally space three or more selected objects *vertically* on the current page.


Same Width

Use this command or toolbar icon  to change the *width* of the selected components so that they match the width of the last selected object. At least two objects must be selected for this command to be available.

Same Height


Use this command or toolbar icon  to change the *height* of the selected components so that they match the width of the last selected object. At least two objects must be selected for this command to be available.

Same Size


Use this command or toolbar icon  to change the both the *height and width* of the selected components so that they match the width of the last selected object. At least two objects must be selected for this command to be available.

Group


Group Command

Use this command or toolbar icon  to group selected components into a single object. Grouped objects can easily be ungrouped later if needed. This icon is only available if there is more than one selected component in the active document.


Ungroup

Use this command or toolbar icon  to ungroup the selected group component. This icon is only available after you select a group in the active document.


Bring to Front

Use this command or toolbar icon  to bring the selected component(s) to the topmost layer of the object stack. You must select at least one object for this command to be available.


Send to Back

Use this command or toolbar icon  to send the selected component(s) to the bottommost layer of the object stack. You must select at least one object for this command to be available.

Bring Forward

Use this command or toolbar icon  to elevate the selected component(s) up one level in the object stack. You must select at least one object for this command to work, and it ceases to have an effect once the object has reached the top of the stack.


Send Backward

Use this command or toolbar icon  to send the selected component(s) down one level in the object stack. You must select at least one object for this command to work, and it ceases to have an effect once the object has reached the bottom of the stack.


Nudge

The **Nudge** command moves the selected object(s) a single pixel in the direction you specify.


Nudge Up

Use this command or toolbar icon  to move the selected component(s) *up* one pixel.


Nudge Down

Use this command or toolbar icon  to move the selected component(s) *down* one pixel.

Nudge Left


Use this command or toolbar icon  to move the selected component(s) *left* one pixel.

Nudge Right


Use this command or toolbar icon  to move the selected component(s) *right* one pixel.

Rotate


Free Rotation

Select this icon  to freely rotate the current object. Using angle snap constrains free rotation to increments of 15°.


Left 90 Degrees

Click this icon  to rotate the selected object 90° to the *left*.

Right 90 Degrees

Click this icon  to rotate the selected object 90° to the *right*.

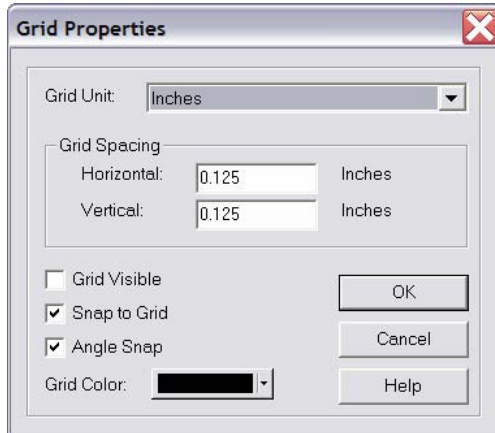
Snap to Grid Command

Use this command or toolbar icon  to enable or disable the **Snap to Grid** feature for components on the active document page. Grid settings can be customized by selecting **Tools** on the menu, and then selecting **Tools**, and **Grid Properties**.

Angle Snap Command

Use this command to activate or deactivate the **Angle Snap** feature used when rotating objects. When enabled, objects rotate in 15° increments. Grid settings can be customized by selecting **Tools** on the menu, and then selecting **Grid Properties**.

Use this command to open the **Grid Properties** dialog. This dialog will give the ability to set the grid properties.

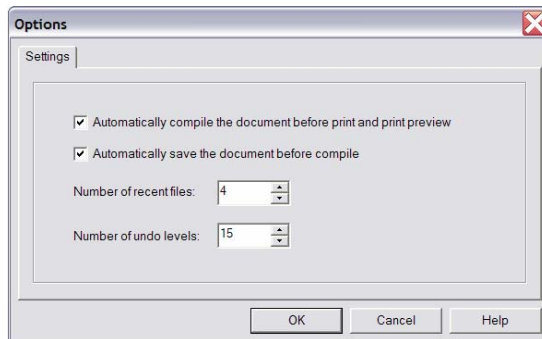


Auto Select Tools

Enable this command to automatically select all inserted or drawn objects. When this feature is disabled, you must manually select new objects before modifying their properties.

Options

Use the **Options** dialog to customize how some functions of FormPort Designer behave.



In addition, you can specify whether or not FormPort Designer automatically compiles documents prior to displaying a print preview or before printing the document. You can also set how many recent files appear in the **File** menu history list (up to a maximum of 16) and set the number of undo levels (50 maximum).



Setting excessive undo levels could impact system performance, and should always be used with caution.

Window Menu

The **Window** menu contains the items listed below:

Cascade

This command arranges multiple opened windows in an overlapped fashion.

Tile

This command arranges multiple opened windows in a non-overlapped fashion.

Arrange Icons

This command arranges the icons for minimized windows at the bottom of the main window. If there is an open document window at the bottom of the main window, then some or all of the icons may not be visible since they will be underneath this document window.

Close All

Use this command to close all of the currently-open documents. If there are any unsaved changes in any of the documents, the **Save** dialog will open and prompt you to save.

Open Files List

The application displays a list of currently-open document windows at the bottom of the **Window** menu. A check mark appears in front of the document name of the active window. Select any document from this list to make it active.

Help Menu

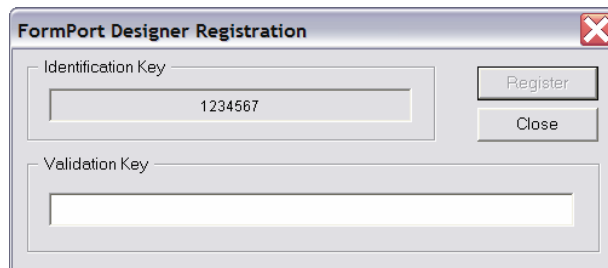
The **Help** menu contains links to various assistance resources.

Contents

Use this command to open the online help documentation's table of contents.

Register FormPort Designer...

This menu item launches a dialog box allowing you to register FormPort Designer.



To register the product, perform the following procedure:

1. Copy the **Identification Key** and Email it to register@capellatech.com with a valid return address.
2. When you receive the **Validation Key**, enter it into the **FormPort Designer Registration** dialog and click the **Register** icon.

FormPort Designer is now registered and ready to use without restrictions.

Technical Support

Contact information for technical support is displayed by selecting this menu item.

About

Use this command to display a window with version and copyright information.

Chapter 3: Tutorials

These tutorials are intended to give you examples of FormPort Designer functions and get you started creating your first forms.

This section consists of the following basic tutorials, which should be performed in order:

- FormPort Designer Tutorial
- Multiple Accounts Tutorial
- Customize the PDF Filename Tutorial
- Email Tutorial
- FAX Tutorial

FormPort Designer Tutorial

The purpose of this tutorial is to help you become familiar with the basic features and functions of Formport Design. The form you will be creating is an AP Check that will be printed on a standard HP printer.

This tutorial explores and describes the following features and functions: lines, boxes, fixed text, logo, signature, micr line, mapped data original copy from tray 1, and internal copy from tray 2

Once complete, the form should look like the sample below.

DESCRIPTION		CHECK NO.:	0000123
		INVOICE NO.	AMOUNT
12/12/99	XYZ Widgets	80413	74.50
12/12/99	Maintenance on p/n 249-OPC	80414	25.50
12/14/99	Circuit boards for model T-090	80510	50.00
12/14/99	Memory modules for model S-090	80512	75.25
12/15/99	Model S-090 with accessory kit.	80513	50.00
12/16/99	Custom design fee	80601	27.75
12/16/99	Case p/n 024-570	80622	01.50
12/16/99	ABC alpha kits	80642	97.50
12/18/99	Consumable p/n 024-572, 1 pallet	80800	40.00
		TOTAL \$	442.00

DETACH BEFORE DEPOSITING

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

FIRST UNION NATIONAL BANK
IRVINE, CA

DATE
12/30/99

AMOUNT
842.00

PAY THE
SUM OF Four Hundred forty two dollars & zero cents

PAY TO THE ORDER OF
FormPort Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

0000123

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

FormPort Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

John Doe
AUTHORIZED SIGNATURE

Prerequisites

To complete this tutorial, you must have the following files:

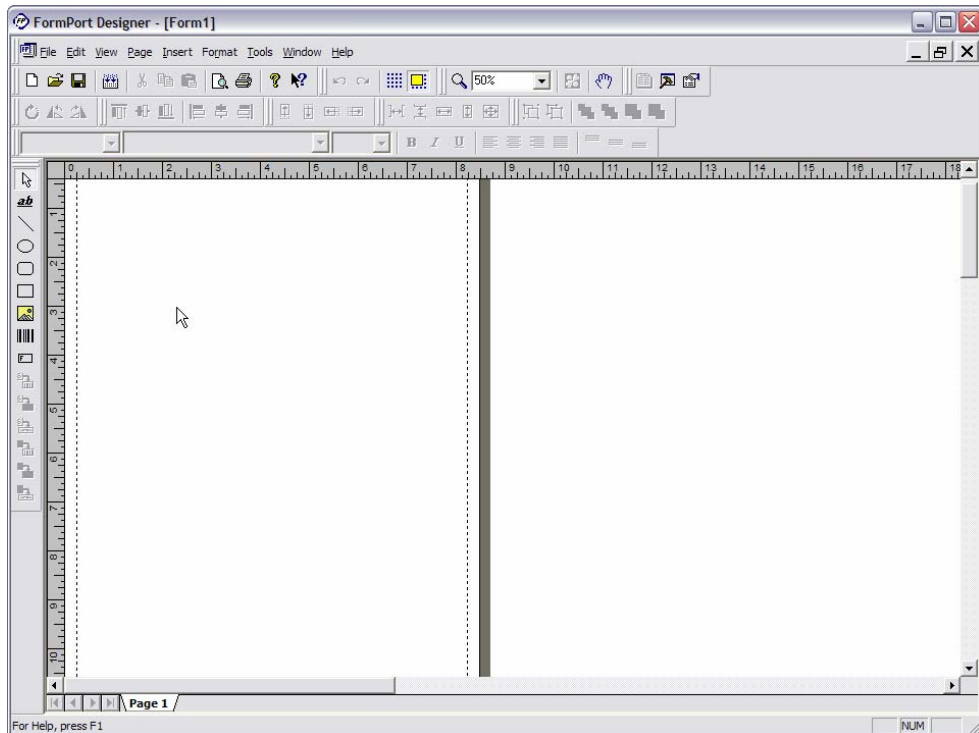
- jdoe.jpg
- Demodata.txt
- Demospool.txt

These files can be found at the following location:

C:\Program Files\CapTech\FormPort Designer\Tutorials and Test Files\FormPort Designer Tutorial

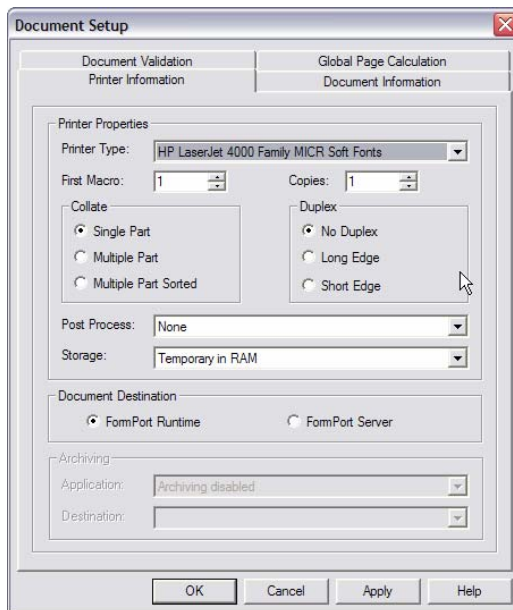
Getting Started

To launch this program, click **Start**, select **Programs**, **Formport**, and then **Formport Designer**. The following screen appears:

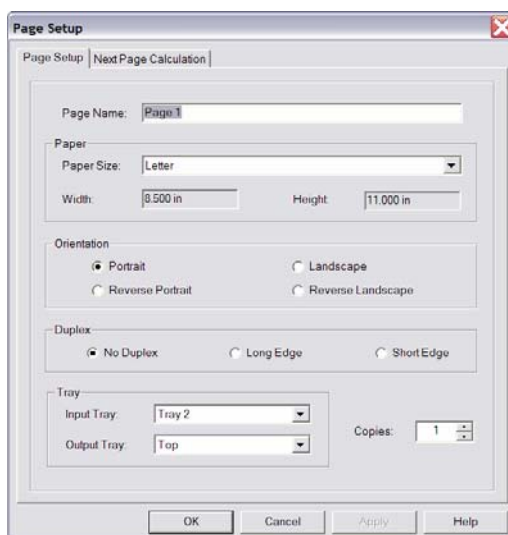


Choose Printer and Paper size

1. Click **File** and then select **Document Setup**. The Document Setup page appears.



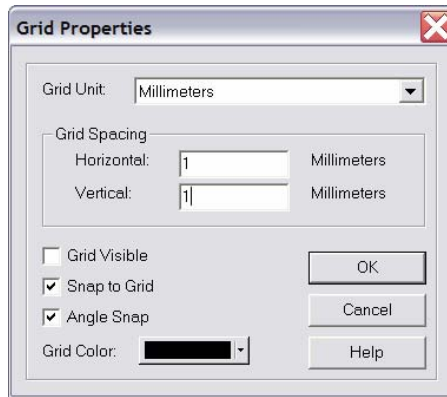
2. For Printer Type, select **HP LaserJet 4000 Family MICR Soft Fonts** from the drop-down menu list and then click **OK**.
3. On the FormPort Designer menu, select **Page**, and then select **Page Setup**. The following **Page Setup** screen appears:



4. Verify Paper Size is set to **Letter** and Orientation is set to **Portrait**.

Set Grid Size and View

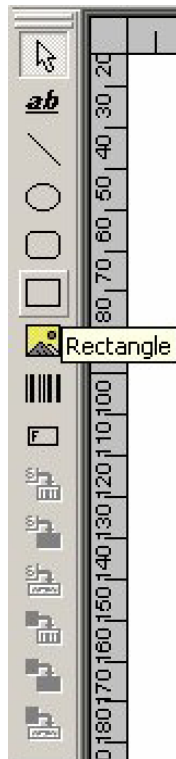
From the FormPort Designer menu, click **Tools**, and then select **Grid Properties** to bring up the **Grid Properties** screen, as shown below:



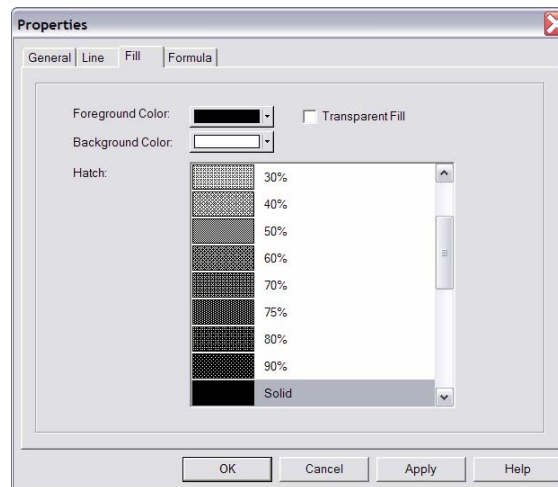
1. For **Grid Units**, select **Millimeters**.
2. For **Grid Spacing**, enter **1** for both Horizontal and Vertical.
3. Make sure the **Grid Visible** box is unchecked.

Creating a Black Box

Begin by clicking the **Rectangle** icon on the **Drawing** toolbar on the left side of the screen.

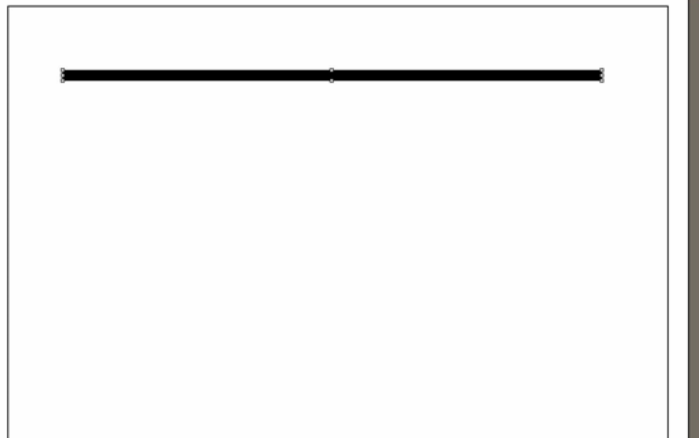


1. Click and drag the cursor across the screen.
2. Right-click inside the rectangle and choose **Properties** from the menu shown.
3. Click the **Fill** tab and uncheck the **Transparent Fill** box.



4. Choose **Solid** from the **Hatch** menu and click **OK**

The document should now resemble the following illustration:



Create a Transparent Box

Repeat the previous procedure to create another box directly below the first one. On this one, make sure the **Transparent Fill** option is checked.

Creating Horizontal and Vertical Lines

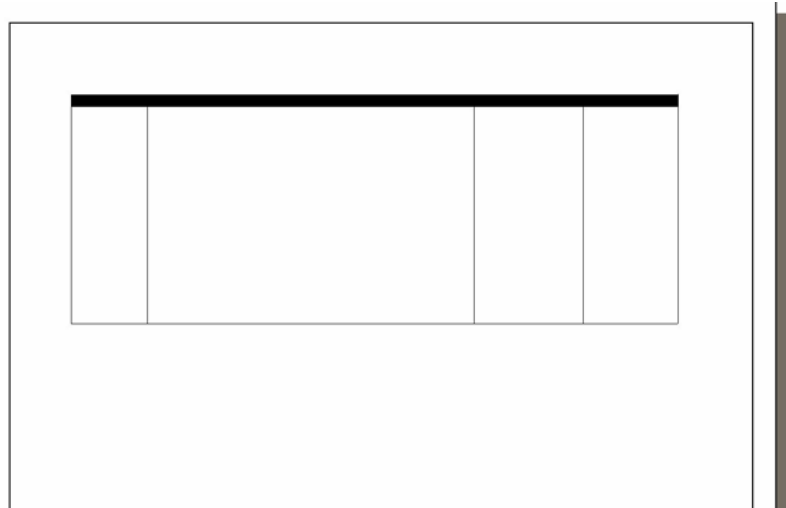
Select the Line icon  on the **Drawing** toolbar.



NOTE

You can constrain the line tool to horizontal or vertical lines (as opposed to diagonal) by holding down the **SHIFT** key.

Click and drag the cursor to the desired length on either side of the black rectangle, as shown in the following image:

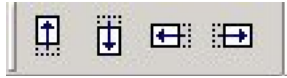


Formatting Graphic Objects

The following tools are available to assist you as you build your page:

Nudge Tool

The Nudge tool allows you to move a selected object a pixel at a time in any of four directions.



Select an object, click any of the **Nudge** tools and watch the result.

Aligning Objects

The alignment tools allow you to precisely align objects along their edge or through their midpoints. Alignment tools only become active after you have selected one or more objects.



When performing an alignment, the last object selected is used as the reference object to which all other objects are positioned. After selecting an object, you can choose additional objects by holding down either the **CTRL** or **SHIFT** keys.

From the **Alignment** toolbar, select the manner in which to align the objects. The choices are: align top, center horizontally, align bottom, align left, center vertically, and align right.

Changing the Layout of Multiple Objects

The Layout tools are only active when more than one object is selected, and are used to ensure the height, width, and spacing between objects is uniform. Like the alignment tools, the last object selected is used for reference.



Select the objects whose layout you wish to change, click the desired icon, and observe the result.

Changing Object Structure

Objects can also be grouped to facilitate easier manipulation and prevent accidental alterations. Grouped objects can always be ungrouped later if needed.

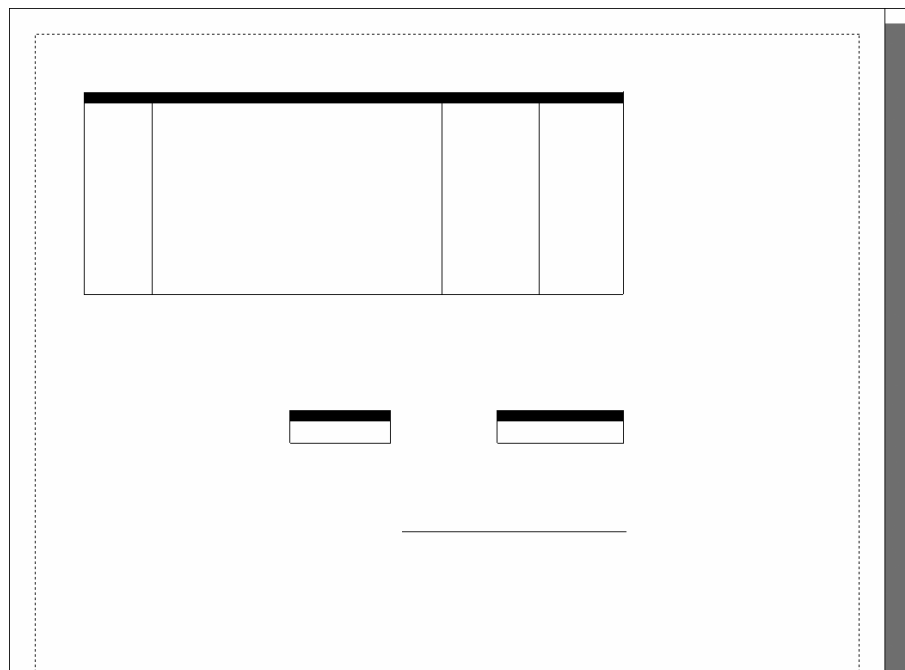


To group, select two or more objects by holding down either the **SHIFT** or **CTRL** keys. Click the **Group** icon on the **Structure** toolbar, or choose **Tools** on the menu, and then select **Group**, and **Group**.

Objects on a page can also be moved forward and backward through the stack of other objects in a document. Tools to move selected objects forward and backward can also be found on the **Structure** toolbar.


Completing the Graphical Layout

Use the tools described above to create a page that resembles the illustration below:

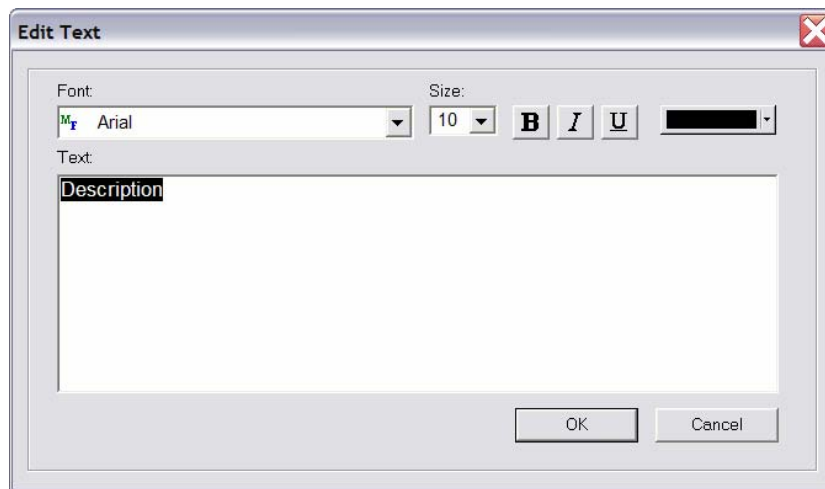


Creating Text Inside the Black Box

To add a text element to your page, follow the procedure below:

1. On the **Drawing** toolbar, click the **Rich Text** icon .
2. Click and drag the cursor to place text at the desired location.

3. Right-click inside the text space and choose **Edit Text** from the menu, to open the **Edit Text** dialog as shown below:



4. Enter the text you wish to appear on the form. To alter the appearance of the text, highlight it, select font **Arial Size 10** and change the color to white. Click **OK** to accept the new format changes.

By default, text objects have a single pixel border around them. To turn off this border, right-click on the text object and choose **Properties** from the resulting menu. Click the **Line** tab and check the **No Line** box.

Rotating Text Objects

Text objects can be rotated through the Rotate toolbar.



To rotate a text object, first select the text you wish to rotate.

- To rotate 90° to the left, select **Rotate Left** on the **Rotate** toolbar.
- To rotate 90° to the right, select **Rotate Right** on the **Rotate** toolbar.
- To manually rotate text objects to any angle, select **Rotate** on the **Rotate** toolbar and drag the cursor out from the center of the object. Text rotation can be constrained to 15° increments by using **Tools** and selecting the **Angle Snap** command.

Continue to add text objects until your document resembles the following illustration:

DESCRIPTION	INVOICE NO.	AMOUNT
	TOTALS	442.00

PAY THE SUM OF

PAY TO THE ORDER OF

DATE


CHECK NO.:

AMOUNT

AUTHORIZED SIGNATURE

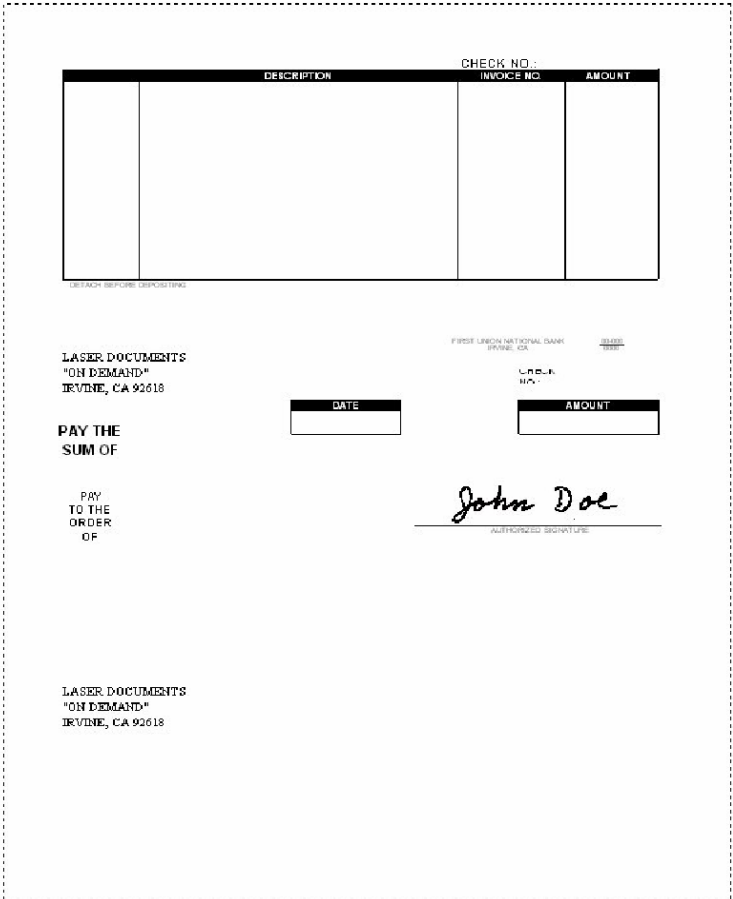
Inserting Logos and Signatures

Forms can be further enhanced by using bitmap images of company logos and signatures. To import a bitmap image, follow the procedure below:

1. Click the **Image** icon  on the **Drawing** toolbar.
2. Click and drag the cursor in the spot you'd like to place the image.

The **Open** dialog appears and you must provide a path and filename to load. Choose the image `Jdoe.jpg` from the current directory.


3. By default, images come into FormPort Designer at a resolution of 300 dpi. To reduce the size of this image, right-click on the image, select the **Image** tab and increase the resolution to **600 dpi**. Position the signature image, as shown in the following image:



The image shows a check form template enclosed in a dashed border. At the top, there is a table with three columns: 'DESCRIPTION', 'CHECK NO. / INVOICE NO.', and 'AMOUNT'. Below the table, the text 'DETACH BEFORE DEPOSITING' is visible. The check is from 'LASER DOCUMENTS "ON DEMAND" IRVINE, CA 92618' and is payable to 'FIRST UNION NATIONAL BANK IRVINE, CA'. There are input fields for 'DATE' and 'AMOUNT'. The check is payable to the order of 'LASER DOCUMENTS "ON DEMAND" IRVINE, CA 92618'. A signature 'John Doe' is written in the 'AUTHORIZED SIGNATURE' field.

Mapping Data

Now that the look of the page has been established, it is time to import and map the spool data.

1. Click **Page** on the menu, and select **Spool Mapping**, **Spool File**, and then **Load Spool File**. From the resulting dialog box, select the file Demospool.txt.
2. The data file should now be displayed in the background of the form in gray text.
3. Click the **Spool Text** icon .
4. Drag the cursor over the data that you wish to map. A **Spool Properties** dialog appears that indicates displays the line and column positions of the text you have selected. Make sure **Copy** is checked and click **OK**.

The spool data has been extracted from the spool file and can now be moved anywhere on the page.

Continue to map the spool data until your form resembles the following image:

DESCRIPTION		CHECK NO.	0000123
INVOICE NO.	AMOUNT		
12/12/99	XYZ Widgets	80413	74.50
12/12/99	Maintenance on p/n 349-OPC	80414	25.50
12/14/99	Circuit boards for model T-090	80510	50.00
12/14/99	Memory modules for model S-090	80512	75.25
12/15/99	Model S-090 with accessory kit	80513	50.00
12/16/99	Custom design fee	80601	27.75
12/16/99	Case p/n 024-570	80622	01.50
12/16/99	AEC alpha kits	80642	97.50
12/18/99	Consumable p/n 024-572, 1 pallet	80800	40.00
TOTALS			442.00

DETACH BEFORE DEPOSITING

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

DATE
12/20/99

PAY THE
SUM OF Four Hundred forty two dollars & zero cents

PAY TO THE ORDER OF
FormFit Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

0000123

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

FormFit Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

FIRST UNION NATIONAL BANK
IRVINE, CA

CHECK NO. 0000123

AMOUNT 842.00

John Doe
AUTHORIZED SIGNATURE

Add Micr Line

1. Highlight the mapped **Check No.** field. Copy and paste the Check No. field and move it to the **Micr Line** area.
2. Right-click on the field and select **Properties**, and then click the **Formula** tab. Enter the text below into the formula box:

"C"& (spooltext(64,27,9,1))&"C A123456789A C458795C"



Your actual SPOOLTEXT formula might vary slightly depending on the location of the Check No field on the page.

3. Click **Font** on the menu bar, and select the **E13B** font.

Create a Copy and Select Tray 2

1. Insert a page by clicking **Page** on the menu. Select **Page Control** and then **Insert Page**, and choose **After Current Page**. Click **OK**.
2. Return to the first page by clicking on the **Page 1** tab at the bottom of the screen.
3. Select all objects by choosing **Edit** on the menu, then **Select All**, or by holding down the **CTRL** key and pressing the **A** key.



4. Select **Edit** on the menu, and select **Copy** or hold down the **CTRL** key and press **C**.



5. Go back to the blank, new page by clicking on the **Page 2** tab at the bottom of the screen.
6. Select **Edit** on the menu, and select **Paste** or hold down the **CTRL** key and press **V** to paste a copy of the data from the first page.



7. While still on the second page, delete the **micr** field and the signature objects.
8. Select **Page** on the menu, then click **Page Setup** and select **Tray 2**.
9. Select **File** on the menu, then click **Document Setup** and select **Multiple Part** under **Collate**.

Print the Form

1. Click **File** on the menu, then click **Print** and enter a name to save and compile the form.
2. Check the box **Merge the page with sample spool data**.
3. Select the spool data file (.txt).
4. Select the printer to print to and then click **OK**.

Multiple Accounts Tutorial



This tutorial can only be performed after creating the form in the FormPort Designer Tutorial, which demonstrates the basics of FormPort Designer.

This tutorial provides an example on how to print a specific check depending on the check number range. A sample .fpd file, testCheckNum.fpd, is located in C:\Program Files\CapTech\FormPort Designer\Tutorials and Test Files\Multiple Accounts Tutorial.

Determining which Check to Print by Check Number

1. Open FormPort Designer and load your document from the previous tutorial, as shown in the figure below:

DESCRIPTION		CHECK NO.:	0000123
		INVOICE NO.	AMOUNT
12/12/99	XYZ Widgets	80413	74.50
12/12/99	Maintenance on p/n 349-0PC	80414	25.50
12/14/99	Circuit boards for model T-090	80510	50.00
12/14/99	Memory modules for model S-090	80512	75.25
12/15/99	Model S-090 with accessory kit	80513	50.00
12/16/99	Custom design fee	80601	27.95
12/16/99	Case p/n 024-570	80622	01.50
12/16/99	ABC alpha kits	80642	97.50
12/16/99	Consumable p/n 024-572, 1 pallet	80800	40.00
		TOTALS	442.00

DETACH BEFORE DEPOSITING

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

FIRST UNION NATIONAL BANK
IRVINE, CA 92618

CHECK NO. 0000123

DATE	AMOUNT
12/20/99	842.00

PAY THE SUM OF Four Hundred forty two dollars & zero cents

PAY TO THE ORDER OF FormPort Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

John Doe
AUTHORIZED SIGNATURE

0000123

LASER DOCUMENTS
"ON DEMAND"
IRVINE, CA 92618

FormPort Print Solutions
Accounting Department
1400 Designer Hwy
Anytown, CA 90099

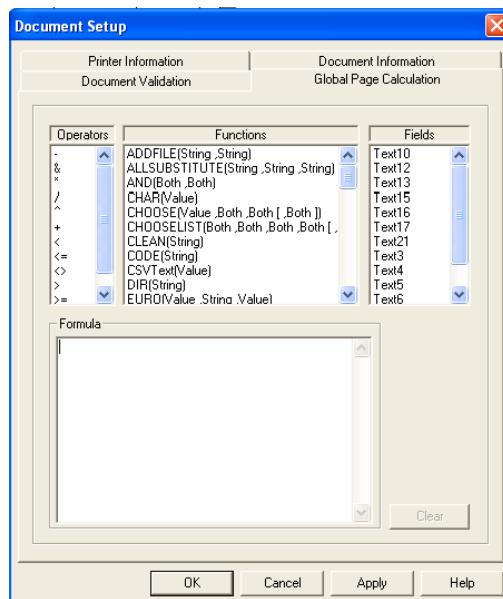
2. Insert a new page after this one and create a check for a different account.



NOTE

Load the same spool file used for the previous page (Demospool.txt).

3. When both checks are finished, select **File** on the menu, then **Document Setup** and then select the **Global Page Calculation** tab.



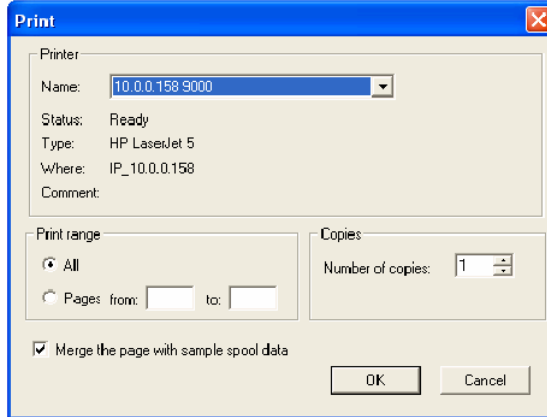
4. In the **Formula** box, type in the following:

```
IF ( VALUE ( SPOOLTEXT ( 67, 2, 7, 1 ) ) > 5000000, 2, 1 )
```

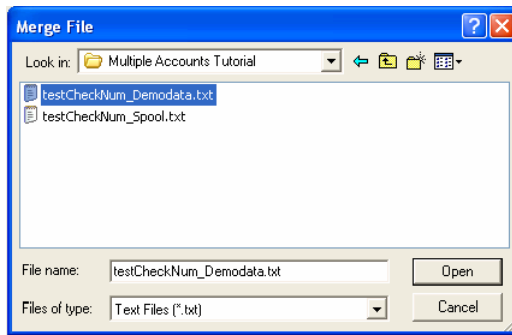
This formula checks if the value of the check number is greater than 5000000. If true, page 2 will be selected; otherwise page 1 will be selected. The “SPOOLTEXT (67,2,7,1)” formula is the formula for the check number. To make sure this is correct, you can right click on the check number, select **Properties**, and click the **Formula** tab.

5. Click **OK**, and then compile the document.

6. To test the application, go to **File** on the menu, select **Print** and check the **Merge the page with sample spool data** option.



7. When the **Merge File** dialog appears, select the testCheckNum_Demodata.txt file in the C:\Program Files\CapTech\FormPort Designer\Tutorials and Test Files\Multiple Accounts Tutorial directory.



8. Click **OK**, and then check to make sure the correct checks print.

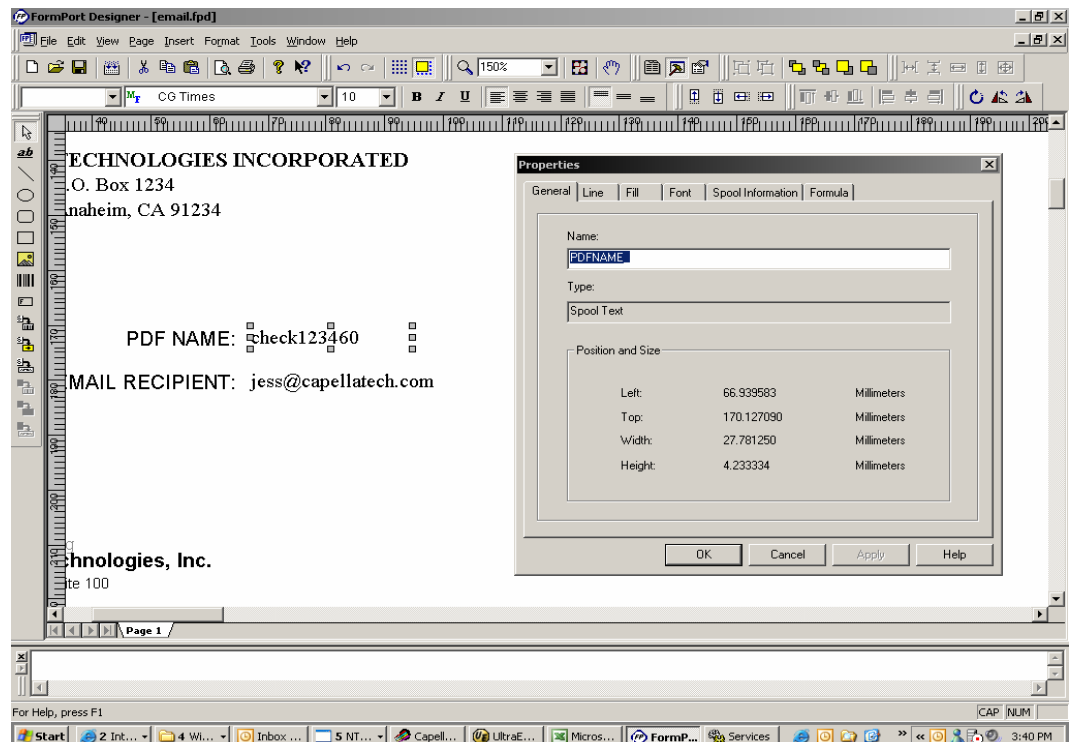
Customize the PDF File Name Tutorial

This tutorial will demonstrate how to customize the output name of the PDF file produced by the Email Agent and PDF Agent in FormPort Server.



Name the file using a FormPort Designer form field

You can use any field in your form design as the NAME of the PDF file that will be attached to an email when using the FormPort Server Email Agent, or saved to a folder by the FormPort Server PDF Agent.

1. Determine which form field to use as the name of the PDF file.
2. Right-click on the field and select **Properties**.
3. In the **General** tab, modify the name of the field, and add the suffix PDFNAME_<field name>. For example:
PDFNAME_checknumber will give a good description of the field that is mapped.

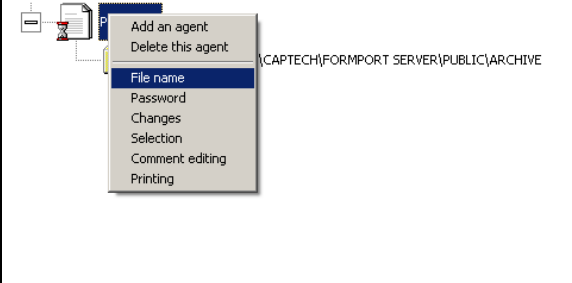
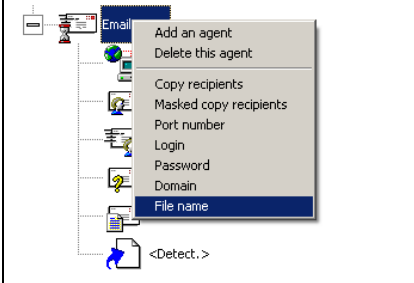


4. Save and compile the form.

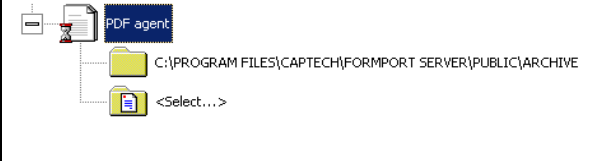

 <p>NOTE</p>	<p>In the FormPort Server Email Agent, be sure to select PDF in the attachment type parameter.</p> <div style="text-align: right;">  </div>
--	---

Customize the PDF file name in FormPort Server



1. Right-click on the **PDF Agent** or **Email Agent**.
2. Select **File name**.

	
---	--

3. Click on the new **<Select...>** parameter under the PDF Agent or Email Agent.

	
--	---

4. Type the PDF filename.

	
--	---

5. This names the new PDF output.

E-mail Tutorial



This tutorial can only be performed after creating the form in the FormPort Designer Tutorial, and will only work if run through FormPort Server.

There are two ways to email documents with FormPort Server. The first is to use the Email Agent to send the entire job to a specified email address, and the second is to extract the email address of the recipient from the data file. With the latter method, you are able to divide the job into several parts and email them individually to the specified email address.

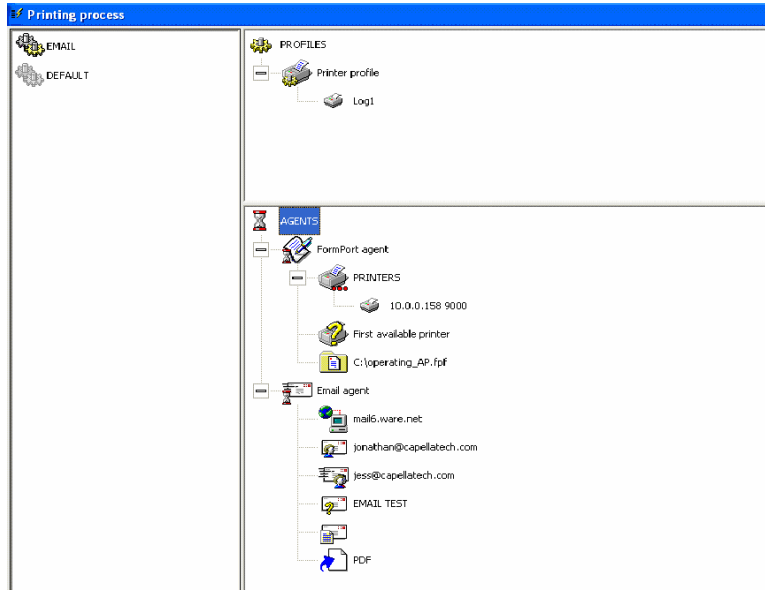
Using the FormPort Server Email Agent to Send the Entire Job

1. Create a new Printing process called “EMAIL.”
2. Add a Printer profile and select a logical printer (For this example, the printer is referred to as “Log1”)
3. Add a FormPort agent and select the appropriate parameters
4. Right-click FormPort Agent and select “Add an agent.”
5. Select the Email agent.



The Email agent must exist *below* the FormPort agent.

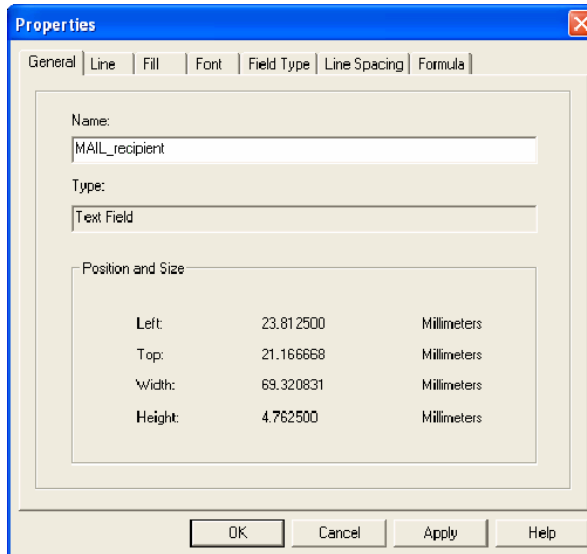
6. Fill in the basic parameters (host name, address of sender, address of recipients, email subject and message, and attachment file format)
7. When a data file is sent to the Log1 printer, the job will be sent to the printer specified in the FormPort agent. This job will then be emailed using the data provided in the Email Agent.



For more information on the Email agent, please consult the *FormPort Server Users Manual*.

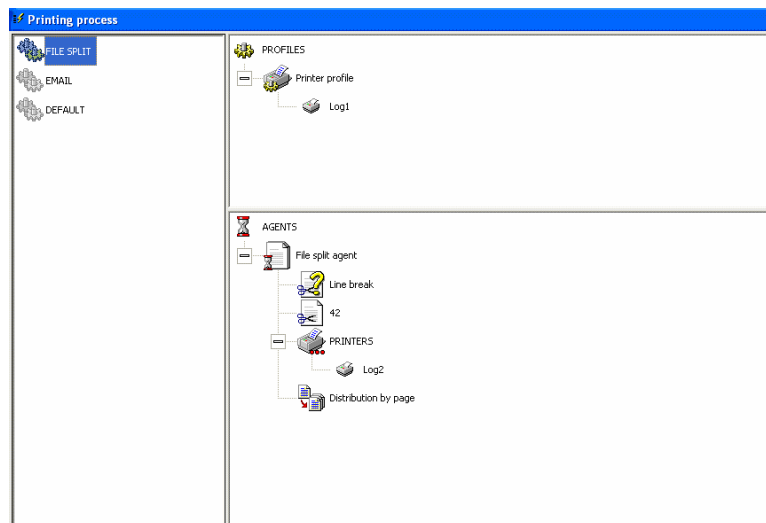
Emailing Multiple Jobs to Multiple Recipients

In order for FormPort Server to detect a recipient's email address within the data file, a field must be created within the application containing the formula for the mapped email address. The name of this field **MUST** begin with the word MAIL. Two processes and two logical printers will be used for this function.



1. Create a new Printing process called **FILE SPLIT**.

2. Add a new Printer profile and select a logical printer (For this example, this printer will be referred to as Log 1).
3. Add a **File split agent**.
4. Select the appropriate **Pages break mode**.
5. Input the value for the cut sequence. This depends on which page break mode is selected.
 - If **Line break** is selected, the cut sequence should contain the number of lines in which the job will be divided into. In other words, a new page will be created every n lines, where n is the number inputted.
 - If **String break** is selected, the cut sequence should contain a text sequence. The File split agent divides the job when it reaches this sequence. This sequence can also include the carriage return, “\r\n”.
 - If **Form Feed break** is selected, the cut sequence parameter is not used. The job will be divided when a form feed is encountered in the data.
6. Right click the printer in the **File split agent** and select a second logical printer (For this example, this printer will be referred to as Log2).
7. Select the strategy to send the spool. This is what the first process should look like.



8. Create a new process called EMAIL below the first process just made.
9. Add a new Printer profile and select the **Log2** printer.

10. Add a FormPort agent and select the appropriate parameters (for testing purposes, use the form check_email.fpf in C:\Program Files\CapTech\FormPort Designer\Tutorials and Test Files\Email Tutorial).

11. Right click **FormPort Agent** and select **Add an agent**.

12. Select the **Email** agent.

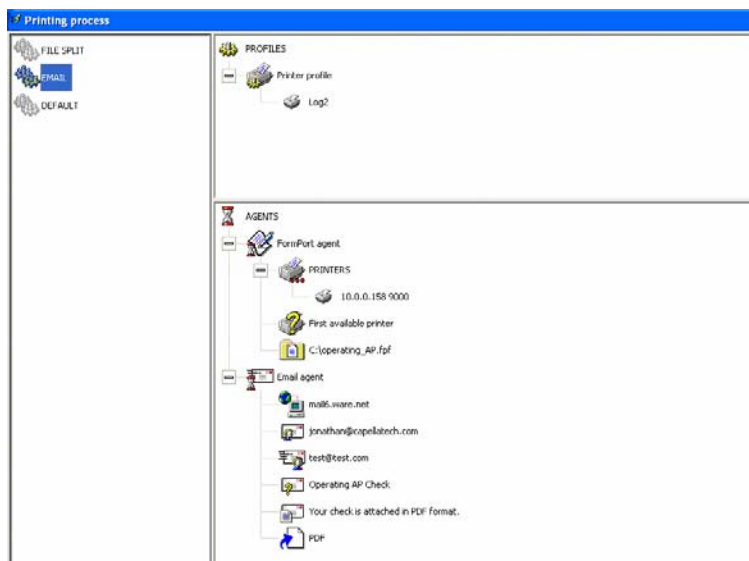


The Email agent must exist *below* the FormPort agent.

13. Fill in the basic parameters (host name, address of sender, email subject and message, and attachment file format).

The **List of recipients** parameter will be overridden by the email addresses extracted from the data file. This parameter must not be empty or an error may occur.

The following illustration is what the second process should look like:



When a data file is sent to the Log1 printer, the job will be split according to the parameters set in the File split agent. These jobs are then sent to the Log2 printer. Each job will be sent to the printer specified in the FormPort agent, then emailed to the extracted email address in the data file. To test this, open the Demodata_email.txt file and change the email address at the top of each record to your email address. Send this data file to the log1 printer. You should receive the check in your email.

FAX Tutorial

This tutorial will demonstrate how to configure the FormPort Designer form to support a dynamic fax number supplied by the data file to be used with the FormPort Server FAX Agent. Each record will use a different Fax number.

Prerequisites

- Windows Fax service must be functional
- Fax Agent must be active on the FormPort Server version purchased
- The data file used must have a valid fax number for each record/page

Configure the Fax Field in FormPort Designer

1. Open the form that you want to configure for faxing.
2. Using the Spooltext Tool, map the fax number field.
3. Right-click the fax number field to open the **Properties** dialog.
4. Rename the field with a prefix of FAX_. You can add more to the field name to make it more descriptive. For example: FAX_recipient, FAX_number, etc.

Purchase Order

PURCHASE ORDER NO. 44 REVISION 0 PAGE 1

CAPELLA TECHNOLOGIES
 1234 Main Street
 Boise, ID 88888-8888
 (555) 555-5555
 www.capellatech.com

VENDOR: Technologies Networks
 12345 Reddersford st.
 Nairobi
 Japan

CUSTOMER ACCOUNT NO. 41 VENDOR NO. 26-SEP-03

PAYMENT TERMS 30 NET

FREIGHT TERMS Prepaid McLaughlin

ITEM	PART NUMBER / DESCRIPTION	DELME
1	275232 TABLET, FL FACED BVL D, RD, EMBSD 100mg o SHIP TO: Address at top of page	20000

Properties

General Line Fill Font Spool Information Formula

Name: FAX_receipient

Type: Spool Text

Position and Size

Left:	3.052083	Inches
Top:	0.614583	Inches
Width:	1.093750	Inches
Height:	0.166667	Inches

OK Cancel Apply Help

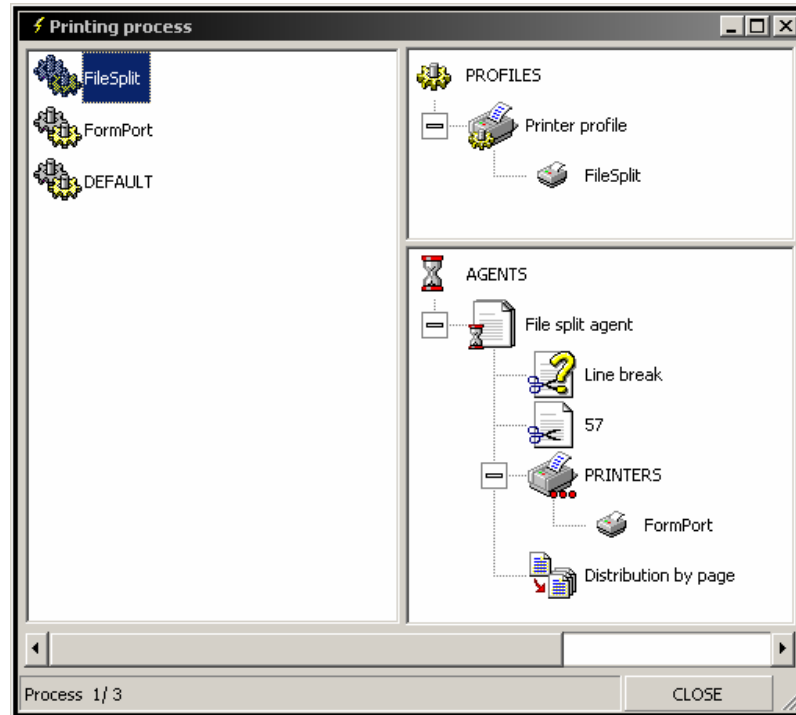


The FAX_ field should contain *only* the fax number that the Windows Fax service will use to dial out. This field must be free from any text or any other number that would make the fax number invalid when dialed.

Configure the Form in FormPort Server

To configure the form in FormPort Server, you will need to create two separate processes:

1. Split the file into separate records, so that each page will be a separate job.

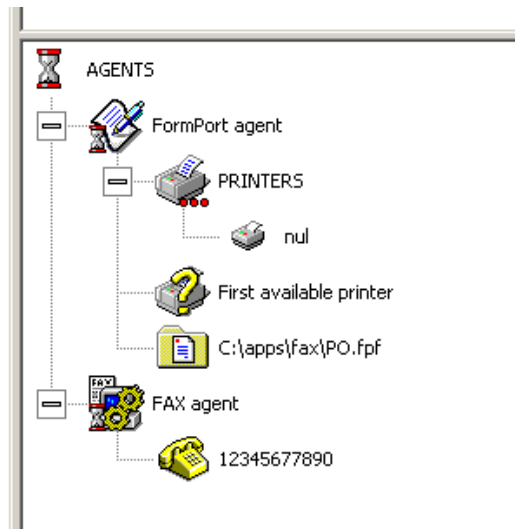


2. Provide a default fax number in the FormPort Agent - FAX Agent.
The default fax number does not need to be valid.



Fax a whole job to a single Fax number in FormPort Server

1. Add the **FAX Agent** after the FormPort Agent.
2. Provide a default **Fax number**. The default fax number does not have to be valid.



Adding Index Fields from FormPort Archive Tutorial

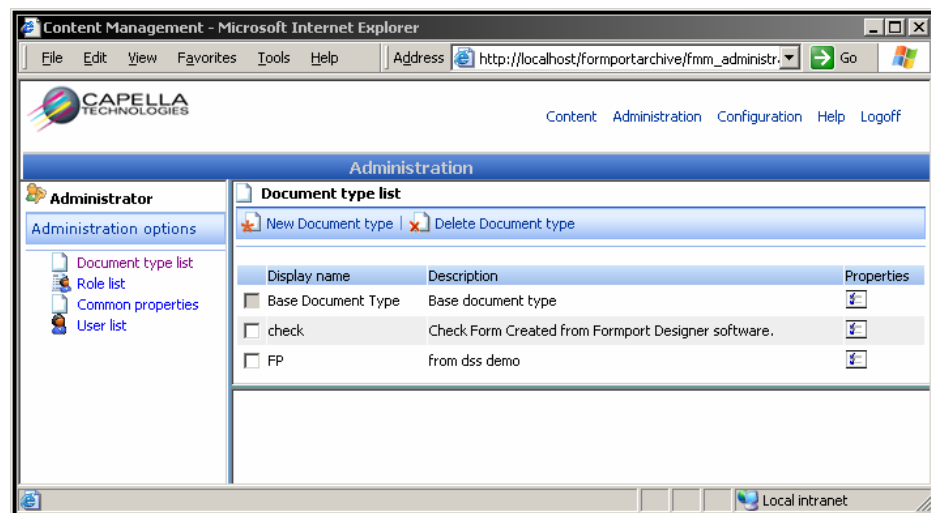


This tutorial can only be performed if FormPort Archive and FormPort Designer are installed and functioning properly.

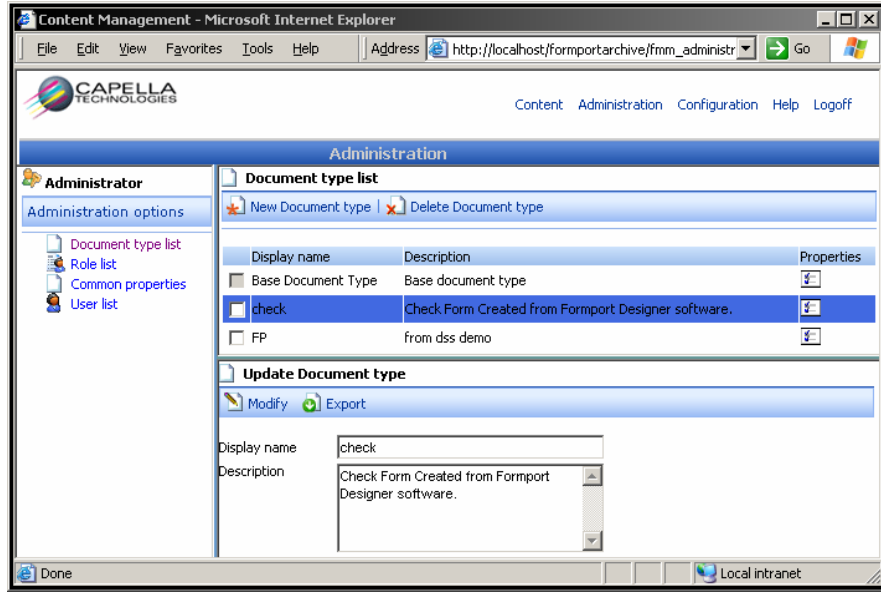
This tutorial will explain how to export the document properties that contain index field information to a FormPort Designer form, for later indexing by the FormPort Archive software.

Exporting the XML File (Document Type) From FP Archive

1. Log in to Formport Archive.
2. Click **Administration** and then select **Document Type List**. This will display all the Document Types that you have. If you do not see the document type that you want to use, please refer to the *FormPort Archive Software User Guide* for instructions on adding a new document type.



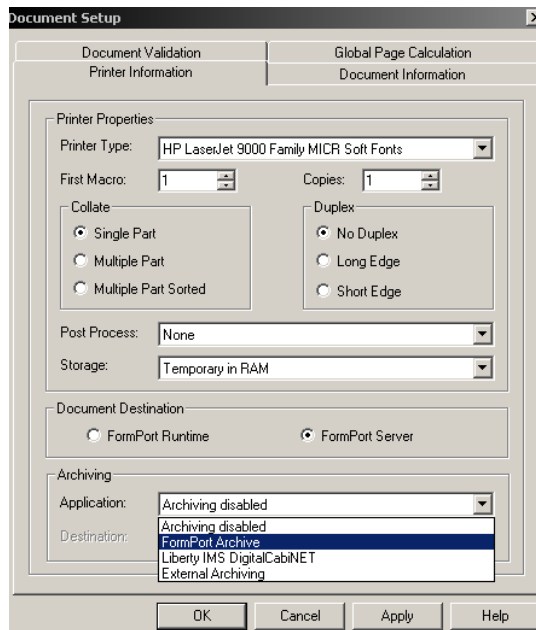
3. Click on the document type you want to use, to highlight and select it.



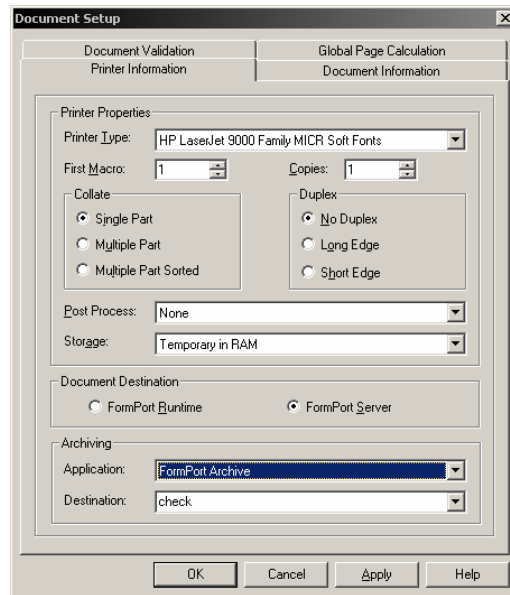
4. Click on the **Export** icon and save the XML document into the directory where FormPort Designer is installed.

Importing the XML File (Document Type) to FP Designer Form

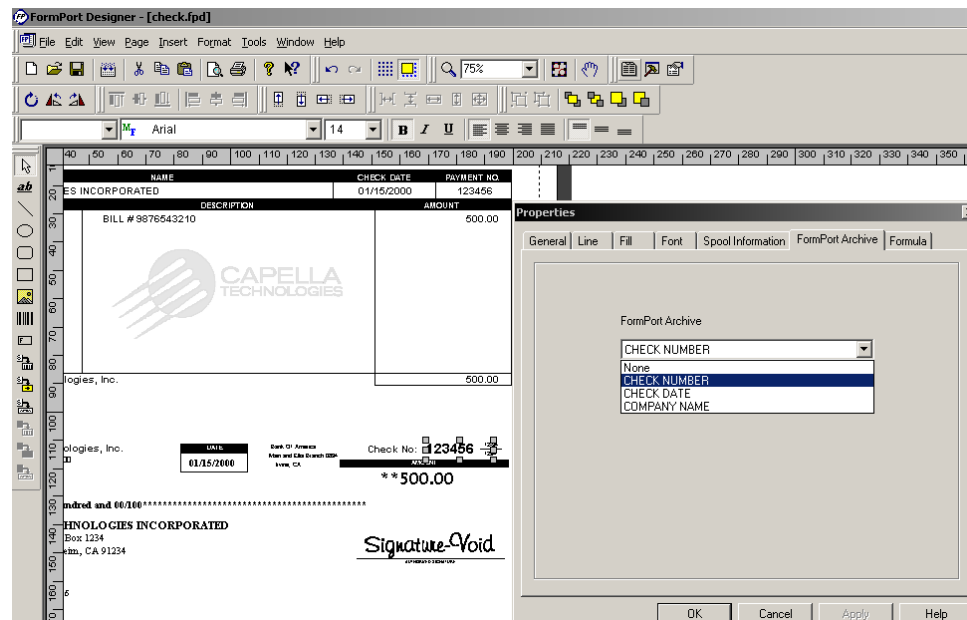
1. Open the form that you want to use.
2. Click **File** on the menu bar, and then select **Document Setup**.
3. On the **Printer Information** tab, select FormPort Server as the **Document Destination**, and in the **Archiving** section, select FormPort Archive from the drop-down **Application** list.



- When prompted to **Load Archive Definitions**, locate and select the XML file that you exported from FP Archive.



- Click **Apply** and then click **OK** to close the Document Setup dialog.
- On the FormPort Designer form, select the field that you want to index in FormPort Archive and open the **Properties** dialog.



- In the **FormPort Archive** tab, select the index field that you want to use for the selected form field. These are the Document Type

Properties. For information about adding a new document type property, please refer to the *Formport Archive Software User Guide*.

8. Click **Apply** and then click **OK** to apply the changes.

Chapter 4: Procedures and Additional Features

This chapter explains how to perform some of the most-common functions and procedures available in FormPort Designer, as well as how to use some of the additional, special features.

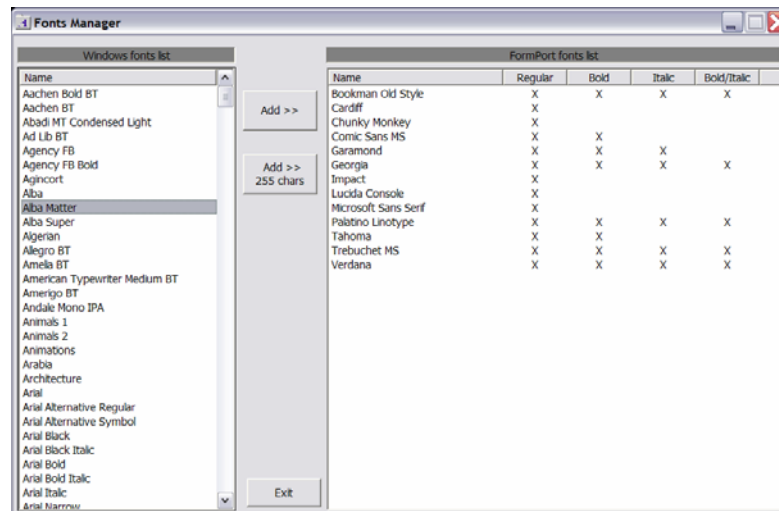
General Procedures

This section provides detailed instructions for a few common FormPort Designer operations.

Adding Fonts

To ensure document consistency, FormPort Designer uses a proprietary set of fonts that differ from what may be installed on your system. If you wish to use some of your own fonts in a document, you must use the Fonts Manager utility.

1. To launch the Fonts Manager, click **Start**, then select **Programs**, **FormPort**, and then **Fonts Manager**.



2. Select the font you wish to add, and then click the **Add>>** icon to convert it and add it to the **FormPort fonts list**.



If you will be printing your document on a different system, you must make sure any new fonts are also installed on that system.

Configuring

How to alter FormPort Designer's settings.

Question:

How can I configure the document settings?

Answer:

Settings for the active document can be specified through the Document Setup Property sheet, which can be accessed by selecting **File** on the menu, then choosing **Document Setup**

Question:

How do I change default properties?

Answer:

Default properties are a collection of settings applied to components as they are placed on the screen. Defining specific settings can greatly increase document creation workflow.

Default properties can be set through the Default Properties Property Sheet that can be accessed by selecting **Edit** on the menu, and then choosing **Default Properties**

Question:

How can I change the page settings?

Answer:

Page settings for the current document can be accessed and changed by selecting **Page** on the menu, and then choosing **Page Setup**.

Question:

How do you change the grid properties?

Answer:

The grid is useful for placing components at precise locations on a page, and can be adjusted to by selecting **Tools** on the menu, and then choosing **Grid Properties**

Inserting Objects

This section describes how to import external and graphical objects and place them in your document.

Question:

How do I associate a spool file with the active document?

Answer:

The first step to loading a spool file into the active document is selecting **Page** on the menu, and then choosing **Spool Mapping, Spool File, and Load Spool File**. The contents of the spool file should become visible on the page and ready for mapping.

Question:

Okay, after loading a spool file, how do I insert a text component into the active page?

Answer:

Selecting **Insert** on the menu, and then choosing **Spool Text** allows you to draw a marquee around the desired text object. This text object can then be moved and formatted as needed.

Question:

How do I insert a spool image component into the active page?

Answer:

Spool image components can be inserted into the active page by selecting **Insert** on the menu, and then choosing **Spool Image**.

Question:

How do I insert a spool barcode component into the active page?

Answer:

Spool barcode components can be inserted into the active page by selecting **Insert** on the menu, and then choosing **Spool Barcode**.

Question:

How can I insert a field component into the active page?

Answer:

Field components can be added by selecting **Insert** on the menu, and then choosing **Field**.

Question:

How do I associate a CSV file with the active document?

Answer:

The first step to loading a spool file into the active document is selecting **Page** on the menu, and then choosing **Spool Mapping, CSV File, and Load Spool File**.

Question:

Once a CSV file is loaded, how do I insert a CSV spool text component into the active page?

Answer:

Selecting **Insert** on the menu, and then choosing **CSV Spool Text** command allows you to insert text by drawing a marquee anywhere on the page.

Question:

How do I insert a CSV spool image component into the active page?

Answer:

CSV Spool image components can be inserted into the active page by selecting **Insert** on the menu, and then choosing **CSV Spool Image**.

Question:

How do I insert a CSV spool barcode component into the active page?

Answer:

CSV Spool barcode components can be inserted into the active page by selecting **Insert** on the menu, and then choosing **CSV Spool Barcode**.

Question:

How do I add a date field to the active page?

Answer:

A date field component can be added to the current document by selecting **Insert** on the menu, and then choosing **Date Field**.

Question:

How can I add a time field component to the active page?

Answer:

To add a time field component, select **Insert** on the menu, and then choose **Time Field**.

Question:

How do you add a page number to the current document?

Answer:

A page number field can be added by selecting **Insert** on the menu, and then choosing **Page Number Field**.

Question:

How can I add graphical objects to the current document?

Answer:

Add lines, rectangles, rectangles with rounded corners, and ellipses to your document by selecting **Insert** on the menu.

Question:

Is it possible to export an object from the current page for use in other documents?

Answer:

Yes. Objects can be exported and stored in a library for use in other documents. To export an object from the current page, select the object and then select **Edit** on the menu, and choose **Library**, and **Export**.

Question:

After I've placed an object into a library, how can I insert it into a new document?

Answer:

With the new document loaded, select **Edit** on the menu, and then choose **Library**, and **Import** to insert and place the object on your page.

Modifying Objects

Once objects have been inserted into the active document, they can be selected and modified to suit your particular needs.

Question:

I want to modify some objects on the page, but how can I select specific objects by their type?

Answer:

Select **Edit** on the menu, and choose **Filter Selection** to select specific components.

Question:

How do I add more pages to my document?

Answer:

Pages can be added by selecting **Page** on the menu, and then choosing **Page Control**, and **Insert Page**. An **Insert Page** dialog box will appear, allowing you to specify how many pages you'd like to add.

Question:

How can I change the font properties of text objects?

Answer:

Select the text object and then choose **Format** on the menu, and then choose **Font**. From the resulting dialog box, you can choose different font types, styles and colors.

Question:

I've modified my fonts just the way I like them. Is there some way I can create text styles to save time and ensure a consistent look?

Answer:

The font properties of text components can be saved as a style by choosing the desired object and then selecting **Format** on the menu, and then choosing **Style**. When the dialog box appears, give your style and name and click **Add**. Your new style will now be available from the drop-down menu of the Font toolbar.

Question:

Now that my document is complete, how can I change the field filling order?

Answer:

The order in which fields are filled can be changed at any time by selecting **Tools** on the menu, and then choosing **Field Filling Order**. A dialog box appears and allows you to select any field and move it either up or down through the list with the arrow keys.

Question:

Once I have objects precisely how I want them, is it possible to group them together?

Answer:

Yes. To group two or more objects, simply select them and then either click the group icon, or select **Tools** on the menu, and then choose **Group**, and **Group**.

Import Forms Features

FormPort Designer allows you to import compiled forms from this and previous versions. This feature is useful for opening and editing older documents or for recreating forms when the original has been accidentally deleted or lost.

Importing Forms

This section explains how to import both old and new forms—and the inherent limitations of each.

Importing an old form

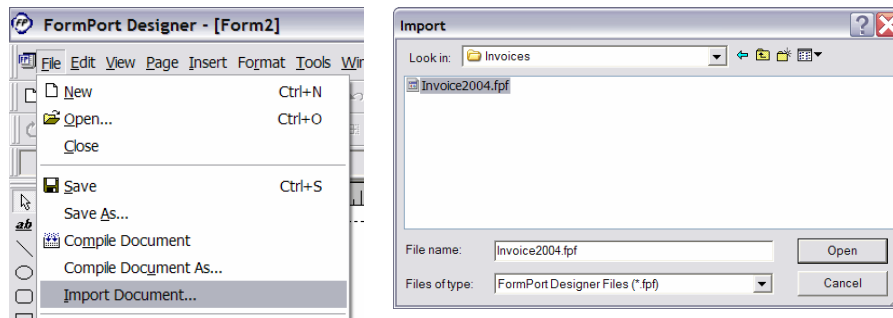
Suppose you created forms with FormPort Design 5 and now you wish to convert them with the FormPort Design 6 format so that you can use the new functionalities.



NOTE

Be aware you cannot open or import FormPort Designer WF3 files directly. However, compiled MFF/FPF files from any previous version can be imported.

To import a compiled form from a previous version, select **File** on the menu, and then choose **Import Document**.

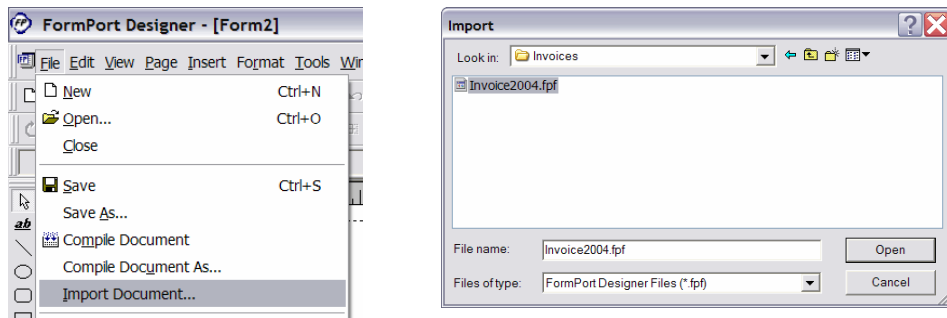


From the resulting **Import** dialog, indicate the path of the compiled document and click **Open**.

Importing a new form

Now, suppose you created a form with FormPort Designer version 6 and then saved and compiled the form. This would result in a compiled FPF file and the original FPD file. But if something happened to the original FPD file and it was accidentally deleted or lost, you can recreate it by importing the compiled FPF file.

To import the compiled document, select **File** on the menu, and then choose **Import Document**



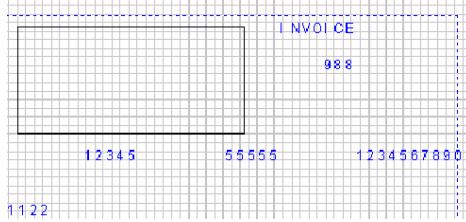
From the resulting dialog box, provide a path and then click **Open** to import it.

As you can see, the import procedure for old and new forms is the same; FormPort Designer automatically detects the version of the compiled document. However, the limitations of the import feature are the same.

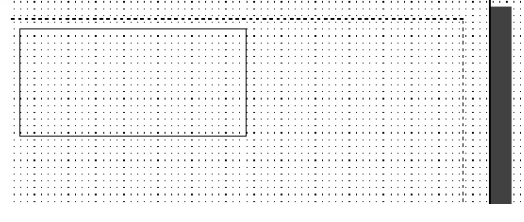
Import Limitations

It is important to understand that compiled FPF files don't contain all the information from the original form, and only have the essential data required by the print engine. Whenever you load a spool file into FormPort Designer, the data is displayed in the page background before being mapped. Although this data is stored in the FPD (or WF3) file, it serves no function in the compiled document and is left out.

This is why when you import an FPF file you will not see the spool data in the background of the page, as shown in the following screenshots:



Original Document

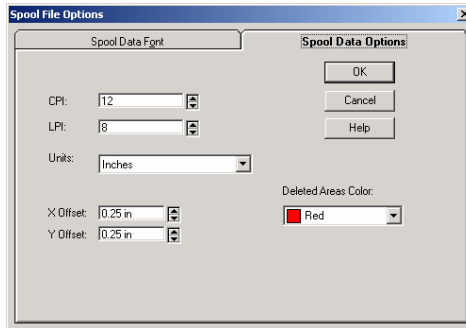


Imported Document

However, some characteristics of the page are known. For example, the number of lines per page and character set. But there are also some page characteristics that are unknown, such as number of characters per inch, etc.

Importing from FormPort Designer Version 5

Be aware of the following limitations when importing older FPF files:



CPI

Character per inch. This setting is not included in the FPF file and is therefore not imported, since it is only used to display the spool file in FormPort Designer and is not used by the print engine. Whenever importing an FPF file, a default CPI value will be used.

LPI

Lines per inch. This setting is also not stored in the compiled document and not imported. Instead, a default value of 6 LPI is used for all imported forms.

Units

Since units are used to calculate the origin of the spool file within FormPort Designer, it is not included in the compiled form, and reverts to a default value when imported.

X Offset, Y Offset

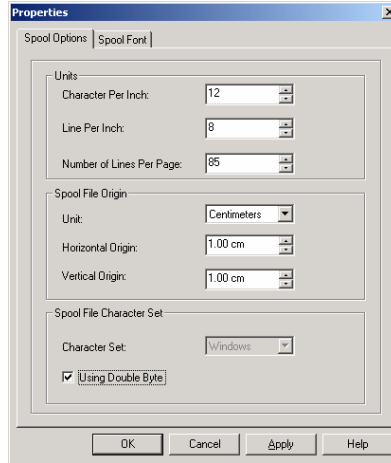
These values are only used to position the spool file and are not included in the compiled form. Instead, a default value is used.

Deleted Areas Color

This parameter is not stored in the compiled form and reverts to the default setting when a document is imported.

Importing from FormPort Designer Version 6

Forms created with version 6 of the application have similar limitations when imported.



CPI

Character per inch. This setting is not included in the FPF file and is therefore not imported, since it is only used to display the spool file in FormPort Designer and is not used by the print engine. Whenever importing an FPF file, a default CPI value will be used.

LPI

Lines per inch. This setting is also not stored in the compiled document and not imported. Instead, a default value of 6 LPI is used for all imported forms.

Number of lines

Since the print engine needs to know the number of lines, this value is included in compiled forms and is correctly imported.

Unit, Horizontal, Vertical Origin

These values are only used to position the spool file and are not included in the compiled form. Instead, a default value is used.

Character Set

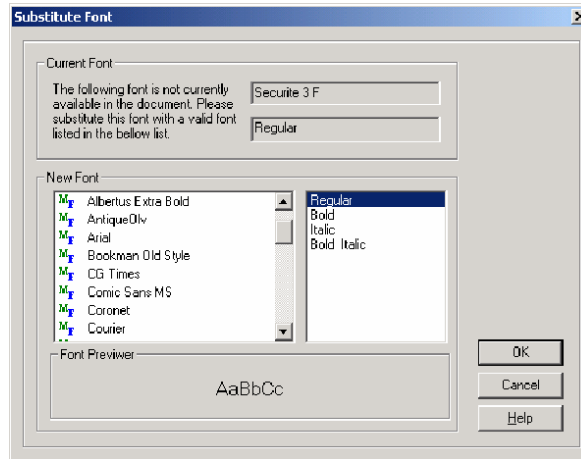
Character Set information is properly imported.

As you can see, many parameters are not imported. However, since the number of lines per page is so important to the print engine, this value is.

In previous versions of FormPort Designer this parameter was fixed, but in the current version it is variable.

Importing Text objects

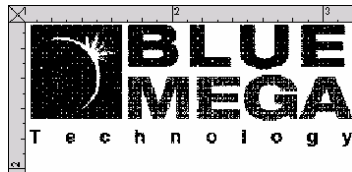
When importing a form containing a special font that is not currently installed on the system, the following dialog will appear and ask you to make a font substitution:



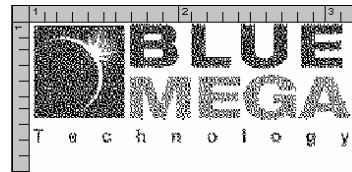
Choose an appropriate font to substitute (and applicable attributes) and click **OK**.

Importing Image objects

Form images that conform to standard resolutions (300, 600, or 1200 DPI), should import correctly. Depending on the original image resolution, the imported object might appear to differ slightly, but the printed result should be the same.



Original Image



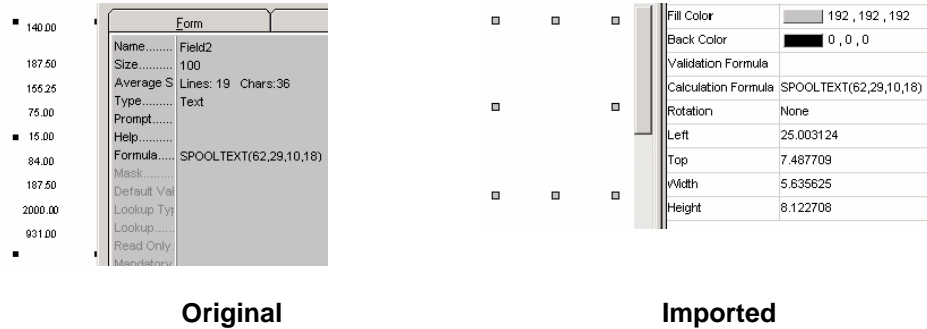
Imported Image

As you can see, the size of the image is the same, but the quality of gray is different because the graphic library used to display the image has changed.

If the resolution is not standard, the image will not import correctly. In this case, you will need to delete the image and then insert a clean image to replace it.

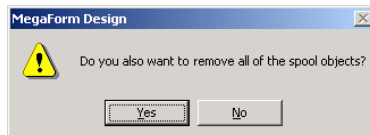
Importing Spooltext objects

Although spooltext objects are imported correctly, their contents are not displayed since they are not used by the print engine. However, all formulas assigned to the object are imported, as you can see in the figure below:



As you can see, although the spooltext object is empty after importing it, the formula remains the same and will print correctly.

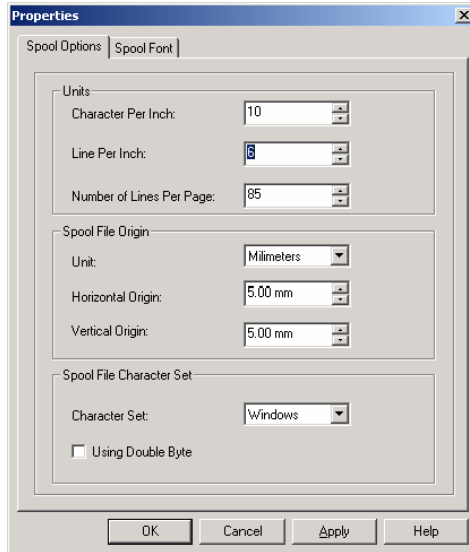
If you wish to see the spool file data in the background, you must first unload the sample spool page by selecting **Page** on the menu, and then choosing **Spool Mapping**, **Spool File**, and **Unload Spool File**. The following dialog will then appear asking if you want to remove all spool objects.



Be sure to click **No**.

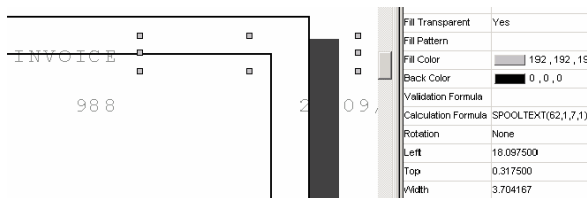
You can now reload the original spool file by selecting **Page** on the menu, and then choosing **Spool Mapping**, **Spool File**, and **Load Spool File**.

If the spool data does not appear correctly, you may have to modify the **Lines Per Inch** value, as shown below:



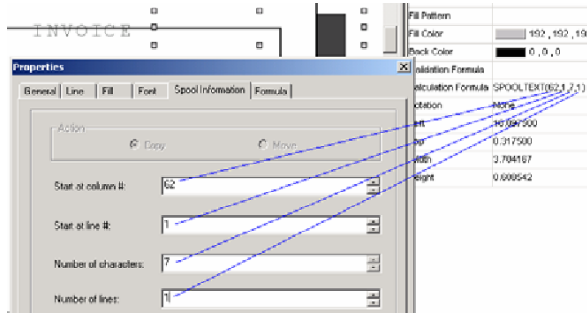
When you reload a spool file, the contents of spooltext objects are not automatically updated and will remain blank.

To update the contents of a spool file, open FormPort Designer's **Properties** window. This information window will appear along the right-hand side of the application, as shown in the following figure:



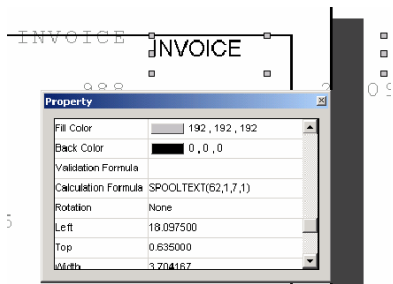
Whenever you select a spooltext object on the page, the associated information and formula will appear in the **Properties** window.

Now modify the spooltext object by opening the object's **Properties** window and selecting the **Spool Information** tab. Using the formula information displayed in the **Properties** window on the right for reference, modify the parameters on the **Spool Information** page to match.



Once you have made the necessary changes, click **OK**.

The spooltext object will now update and correctly display its contents.



Chapter 5: Scripting Commands

Although most common tasks can be performed using the WYSIWYG interface, to fully unleash the power and functionality of FormPort Designer you need to define formulas using the powerful, embedded scripting language. These simple commands enable a wide range of additional features.

Advanced users may discover they can actually perform some actions more quickly through the scripting commands, as opposed to using the graphical interface. Each of these commands is described in detail in the following section.

List of functions

Proceed through this section for a list of all the available functions, their syntax, and examples of how to use them.

ADDFILE

Appends the current field data to an external file. The data could contain formulas, fixed text in quotes or any number of other text strings. A value of True (1) is returned if ADDFILE successfully writes the file. If not, False (0) is returned.

Syntax:

ADDFILE("filename" , "data")

filename

The filename of the external file (in quotes) to place data. If the file already exists, new data will be appended to the end of the file. If no file exists with that name, a new file will be created.

data

The data to write to the file, also in quotes.

Example

Consider the following formula:

```
ADDFILE("Logfile.csv", SPOOLTEXT(6,13,30,1) & "," &  
SPOOLTEXT(81,3,10,1) & "," & SPOOLTEXT(85,81,8,1) & "!*n*!")
```

In this formula, the SPOOLTEXT function returns data from the spool file and then inserts it into a file called “Logfile.csv.” The macro !*n*! is used to insert a new line. The content of this file is:

```

1 Bluemega Technology,20/05/2003,42776.04
2 Bluemega Technology,20/05/2003,42776.04

```

You can use this function to create a report file containing a summary of the original spool file. For example, suppose you create an invoice form and you have a spool file or CSV file that contains 100 invoices. Now you want to create an external file that contains the name of the customer, the date and the monetary total of the invoice. This can be accomplished by adding a field in the form (probably outside of the printed area) with a formula using the ADDFILE function. After printing, you’d have a separate text file containing the customer’s name, date and monetary total of each invoice.

ALLSUBSTITUTE

Substitutes text within a text string with other user-defined text. Use ALLSUBSTITUTE when you want to replace all occurrences of a text sequence with alternate text. This function is similar to a word processor’s search and replace feature.



If the text defined as *old_text* is not found in the string, the *new_text* value will not be inserted.

Syntax:

ALLSUBSTITUTE(*text*, *old_text*, *new_text*)

text

The text string you wish to modify.

old_text

The text you want to replace.

new_text

The new text you want to insert.

Example

Consider the following formula:

```
ALLSUBSTITUTE( SPOOLTEXT(81,3,10,1), "/", " - ")
```

This formula first reads the spool file using the SPOOLTEXT function. The string returned by the SPOOLTEXT function is “20/05/2003”.

The formula is modified as follows:

```
ALLSUBSTITUTE( “20/05/2003”), "/", " - ")
```

The function substitutes all “/” characters with “ – “. Thus, the string “20/05/2003” is converted to “20 – 05 – 2003”

Here is another formula using the ALLSUBSTITUTE function:

```
ALLSUBSTITUTE( SPOOLTEXT(44,8,44,1), " ", "")
```

This formula reads the spool file with the SPOOLTEXT function and the formula is modified:

```
ALLSUBSTITUTE( “12345 55555 12345678901 99”, " ", "")
```

By replacing all “space” characters with nothing, this formula removes the spaces between characters of the string and you will receive the following result: “12345555551234567890199”.

AND

This Boolean function returns a True (1) value only if both supplied logical arguments are true. If either (or both) is false, then a False (0) value is returned.

Syntax:

```
AND( logical_cond1, logical_cond2 )
```

logical_cond1

are two conditions that may also be complex formulas.

logical_cond2

Example

Consider the following formula:

```
IF(AND(EXACT(SPOOLTEXTL(1,29,1,35),"*"),FIND("#",SPOOLTEXTL(1,29,92,35))), "Line starting with '*' and containing '#", "Detail line")
```

This formula reads 35 lines—line by line—and then applies the formula to each one. For each line, the formula reads the first character –

SPOOLTEXTL(1,29,1,35) – and compares this character with ‘*’ – EXACT(SPOOLTEXTL(1,29,1,35),"*"). Then the formula reads the line – SPOOLTEXTL(1,29,92,35) – and searches for the character ‘#’ – FIND("#",SPOOLTEXTL(1,29,92,35)) – in the line. If the line starts with the character ‘*’ AND if the line contains the character ‘#’, then the first part of the IF function is executed – "Line starting with ‘*’ and containing ‘#’"; otherwise the second part of the IF function is executed – "Detail line".

As you can see, the first condition of the AND function could return 0 (zero) or 1, because Exact is a Boolean function.

But the second condition of the AND function contains the FIND function. FIND could return 0 (zero) or a positive value that can be greater than zero. It is the reason why the AND function considers that zero is FALSE and a value greater than zero is TRUE.

If you need to test more than 2 conditions, you can nest AND function:

```
AND( AND ( x1 > 10 , x1 < 20 ) , AND ( y1 > 15 , y1 < 30 ) )
```

Where x1 and y1 are valid numeric values.

CHAR

Returns the character that corresponds with the ANSI number specified.

Syntax:

CHAR (number)

number

Any number between 1 and 255 representing the ANSI value of a desired character.

Example

Consider the following formula:

```
CHAR(27) & CHAR(27) & "%-20006X"
```

CHAR(27) is the escape character for a PCL printer. Using this character begins the escape sequence. This formula would be interpreted by the printer as follows:

```
<esc><esc>%-20006X
```


You can use this function to send an escape sequence to call the JetCAPS MICR Dimm, for example. The JetCAPS MICR Dimm has a feature to convert a numeric value into a string:

'452.00' is converted into 'Four hundred and fifty two'.

To call this feature, you need to include a PCL escape sequence. So you start this sequence with CHAR(27).

CHOOSE

Returns a specific value from a list of numbers. This value is determined through a user-supplied index number. If the value *indexInList* is not in the valid range, a blank string is returned.

Syntax:

CHOOSE(*indexInList*, *itemList*)

indexInList

A number that corresponds with the desired list item contained in *item_list*.

itemList

A list of numbers, formulas, or text separated by commas. This value can contain a maximum of 29 items.

Example

Consider the following formula:

```
CHOOSE(VALUE(SPOOLTEXT(93,1,1,1)), "Mrs.", "Ms", "Mr.")
```

Using the SPOOLTEXT function, this formula first reads the current spool file. In this instance, the string returned by the function is '3.' Since the CHOOSE function requires a numeric value as the first parameter, the string must be converted from text to numeric.

Because of this, the formula would be modified as follows:

```
CHOOSE( 3, "Mrs.", "Ms", "Mr.")
```

In this example, "Mr." would be printed. If the first parameter were to be changed to "1," then "Mrs." would be printed. And using "2" would print the value "Ms."

CHOOSELIST

Returns a value from a list of numbers as specified by a key string.

Syntax:

CHOOSELIST(KeyString, itemList, [returnValue])

KeyString

A value, formula or string within the itemList.

itemList

A list of up to 100 numbers, or text separated by commas.
Each item a strong or value to be returned if found.

returnValue

An alternate text string or value to be returned in the event the KeyString is not found in the itemList.

Chooselist returns itemList1 if the keyString is found in itemList.

Example

Consider the following formula:

```
CHOOSELIST(SPOOLTEXT(1,1,2,1),"IN","Invoice","DO","Delivery
Order","PO","Purchase Order","Other")
```

This formula uses the SPOOLTEXT function to read the spool file. In this instance, the parameters of SPOOLTEXT specify that we read on line 1, column 1, 3 characters on 1 line. Then the string that is returned is 'IN'.

The formula then becomes:

```
CHOOSELIST( "IN", "IN", "Invoice", "DO", "Delivery Order", "PO",
"Purchase Order", "Other")
```

Now the formula will compare the first parameter to each group of items.

First parameter "IN" is compared in the following manner:

- To "IN": if equal then "Invoice" is returned
- Then to "DO": if equal then "Delivery Order" is returned
- Then to "PO": if equal then "Purchase Order" is returned
- "Other" is returned if not found.

This function can also be used in the Document Validation or Global Page Calculation formula to detect which form or which you want to use.

CLEAN

Removes all non-printable characters from either text or the result of a formula derived through external files.

Syntax:

CLEAN(text)

text

Any information from which you want to remove non-printable characters.

Example

Consider the following formula:

CLEAN(CHAR(7) & SPOOLTEXT(6,13,30,1) & CHAR(13))

First the formula creates a temporary string starting with character 7 and the content of SPOOLTEXT function (Capella Technologies), ending with character 13.

So the formula becomes:

CLEAN("?Capella Technologies?")

CHAR(7) and CHAR(13) are replaced by '?' character because it is not possible to display these characters. This string will now print "Capella Technologies."

CODE

This function returns the ANSI value of the first character in a specified text string.

Syntax:

CODE(text)

text

The text from which you want the ANSI code equivalent of the first character.

Example

Consider the following formula:

CODE(SPOOLTEXT(1, 31, 92, 1))

The formula first reads the spool file. The returned string is the line number 31. This line starts with '*' character. So the formula is modified as:

CODE("*Your PO #: 123456789 ")

The CODE function returns the ANSI value of the first character. The character '*' has an ANSI value equal to 42.

Suppose you want to detect if the first character of the line is '*', you can create two different formulas. But the result will be the same.

Formula 1 (using IF and EXACT) and formula 2 (using IF and CODE)

```
IF( EXACT( SPOOLTEXT( 1, 31, 1, 1), "*" ), "Yes", "No")
IF( CODE( SPOOLTEXT( 1, 31, 1, 1) ) = 42, "Yes", "No")
```

The returned string will be "Yes" in both cases if the line starts with '*'.

CSVTEXT

Returns both text and numerical data (in quotes) from a designated field of the current CSV file. If the value specified by field is not in range, an empty string will be returned. This function is similar to the SPOOLTEXT function except it is used to retrieve data from CSV files rather than spool files.

Syntax:

CSVTEXT(field)

field

The number of the field in the list (first field = 1).

Example

Examples of this function are essentially provided throughout this section. Whenever you see the SPOOLTEXT function being used to retrieve data from spool files, you can use the CSVTEXT function to get data from a CSV file.

DIR

DIR is a Boolean function that first checks to see if a specific file exists, and then returns either a value of True (1) if it does, or False (0) if it does not.

Syntax:

DIR(text)

text

The name of a file (in quotes) to check for.

Example 1

Consider the following formula:

```
DIR ( "..\spool1.txt" )
```

The file name is “..\spool1.txt”. If this file exists, then the formula returns “1.” Otherwise, the formula returns “0.”

You can use this function to validate a SPOOLImage or CSVImage field, which are used to read data from spool and CSV files, respectively. The data read is considered the name of the image you wish to dynamically print. If the image does not exist on the server, the print engine will generate an error. To avoid this error message, you can provide a validation formula in the SpoolImage or CsvImage.

Example 2

The SpoolImage has a calculation formula such as SPOOLTEXT(1,1,20,1). If the SPOOLTEXT function returns “my_picture.bmp”, the SpoolImage field will try to open this image and print it. But since this image is not available on the server, you will receive an error message.

Now you put a validation formula for the same SpoolImage field. This formula is:

```
DIR ( SPOOLTEXT( 1, 1, 20, 1 ) ).
```

The SPOOLTEXT function returns the same value (“my_picture.bmp”). Then the DIR function returns “0” if the image does not exist, or a “1” if it does.

Because this formula is in the validation of the field, the calculation formula is not evaluated when the validation formula returns “0.” The calculation formula is only evaluated if the validation formula returns “1.”

EURO

Converts a number to a Euro (€) equivalent as established by the European Union (EU).

Syntax:

```
EURO( value, currency, precision )
```

value

The currency value you want to convert.

currency

The letter string, corresponding to the code for the source currency.

precision

The number of decimals to provide in the returned value.

Code currency:

"FRF", "ITL", "FL", "NLG", "DEM", "BEF", "LUF", "FIM", "ATS", "ESP",
"IEP", "PTE", "GRD"

Example

Consider the following formula:

```
EURO( VALUE( SPOOLTEXT(25,73,8,1) ), "DEM", 2)
```

Using the SPOOLTEXT function, the formula reads the spool file and returns "50763.55" in quotes. The VALUE function can then be used to convert the result into a numeric variable (without quotes). Using the DEM rate, the EURO function converts the value into a Euro equivalent, and returns the converted value rounded with 2 digits.

EXACT

Compares two text strings and returns a value of True (1) if they are exactly the same, or False (0) if they are not. This function is case-sensitive.

Syntax:

```
EXACT( text1, text2 )
```

text1

The first text string. This value can be a formula that returns a string.

text2

The second text string. This value can also be a formula that returns a text string.

Example

Consider the following formula:

```
IF( EXACT( SPOOLTEXT(71,83,3,1), "USD"), "US Dollars", "Other")
```

SPOOLTEXT function reads in the spool file and returns a string in quotes. Using SPOOL1.TXT, SPOOLTEXT function returns “USD”. The EXACT function is comparing the SPOOLTEXT result (“USD”) with “USD”. If they are identical, then the first part of the IF function is returned, and “US Dollars” will be printed. If they are different, then the second part of the IF function is returned. “Other” will be printed.

The EXACT function is only used to compare text strings. If you want to compare numeric values, you have to use the symbol “=.”

EXIT

Cancels a print operation after the current page, resulting in an error code being returned. This function is typically used in a validation formula, when incorrect data is detected in the spool data file. A blank string is always returned by this formula.

Syntax:

EXIT(returnCode, text)

returnCode

The exit code returned by the print engine to the calling application.

text

The error message to be displayed on the console or written in the log file.

Example

Consider the following formula:

```
EXIT( 1, “Error message” )
```

The EXIT function cancels the process, but the print engine will continue to print the current page.

FIND

FIND function finds one text string within another text string and returns the number of the starting position of that search string. FIND is case sensitive and does not allow wildcard characters.

The default value of start_pos is “1,” and will return the first character in the search string (depending on start_pos) if left blank.

Syntax:

FIND (find_text, within_text [, start_pos])

find_text

The text you want to find.

within_text

The text string containing the text you want to find.

start_pos

Specifies the character at which to start the search.

Example

Consider the following formula:

```
FIND( "Dis", SPOOLTEXT( 1, 72, 92, 1) )
```

First the formula reads the spool file using the SPOOLTEXT function and returns a string, then the formula is:

```
FIND( "Dis", " 51263.55 50763.55 Dis 15 % -7614.53" )
```

FIND searches for the word "Dis" in the string, and returns the position where it finds the first letter that matches with "Dis". In this case, the returned value would be 41. Since this is a numeric value (without quotes), you can use this function as a numeric parameter in yet another function.

For example, the LEFT function accepts two parameters; the first one is a string, the second is a numeric value. You can replace the second parameter by a FIND function.

```
LEFT ( "This is a sample text", FIND(" text", "This is a sample text") )
```

The part of the function FIND(" text", "This is a sample text") returns 17. So the new formula is:

```
LEFT ( "This is a sample text", 17 )
```

FORMAT

Returns a string formatted in accordance with user-defined instructions. A value of "#" indicates an *insignificant* zero, and "0" indicates a *significant* zero.

Syntax:

FORMAT(text, format)

text

The text to be formatted.

format

A string that contains the format of the converted string.

Example

Consider the following formula:

```
FORMAT( SPOOLTEXT( 13, 72, 8, 1 ), "### ##0,00" )
```

The formula reads data in the spool file with the SPOOLTEXT function and returns a string in quotes. Since this string must be numeric, a value of "51263.55" is returned. Now the new formula is:

```
FORMAT( "51263.55" , "### ##0,00" )
```

Using the format parameter as reference, the spool data is then reformatted as "51 263,55". If the format parameter were changed to "000 000.00," the resulting value would be "051 263.55."

The FORMAT function is intended solely for altering the format of a string value. To change the format of a numerical value, use the TEXT function.

FULLDATE

FULLDATE returns the current date.

Syntax:

```
FULLDATE( format, language )
```

format

The number that represents the format of the date you want to use.

language

The value that represents the language to use to print the date.

List of available date formats:

- | | |
|----------------------|----------------------|
| 1 = yymmdd | 11 = dd mmm...m yyyy |
| 2 = mm/dd/yy | 12 = yy/mm/dd |
| 3 = mm-dd-yy | 13 = yyyyymmdd |
| 4 = Mmm dd, yyyy | 14 = mm/dd/yyyy |
| 5 = Mmm...m dd, yyyy | 15 = mm-dd-yyyy |
| 6 = dd Mmm yy | 16 = dd/mm/yyyy |

7 = dd Mmm yyyy	17 = dd-mm-yyyy
8 = dd/mm/yy	18 = Mmm dd yy
9 = dd-mm-yy	19 = Mmm...m dd yy
10 = dd mmm...m yy	20 = yyyy/mm/dd

List of available languages:

- 1 = English
- 2 = Spanish
- 3 = French

Example

Consider the following formula:

```
FULLDATE ( 5, 1 )
```

The result of this formula will differ depending on what date you print the form. In this instance, suppose the form is printed on the 22nd of May in the year 2005. Using the formula above (format = 5 and language = 1) the following date would be printed: “May 22, 2005.”

Now consider the following formula used on the same date:

```
FULLDATE ( 14, 1 )
```

This formula will print “05/22/2005.”

FULLTIME

FULLTIME returns the current time.

Syntax:

```
FULLTIME ( format )
```

format

The number that represents the format of the time you want to use.

List of available time formats:

- 1 = HH:MM:SS 12 hours (with space before if HH < 10)
- 2 = HH:MM:SS 12 hours AM/PM (with space before if HH < 10)
- 3 = HH:MM:SS 24 hours (with space before if HH < 10)
- 4 = HH:MM 12 hours (with space before if HH < 10)
- 5 = HH:MM 12 hours AM/PM (with space before if HH < 10)
- 6 = HH:MM 24 hours (with space before if HH < 10)
- 7 = HH:MM:SS 12 hours (with '0' before if HH < 10)

- 8 = HH:MM:SS 12 hours AM/PM (with '0' before if HH < 10)
- 9 = HH:MM:SS 24 hours (with '0' before if HH < 10)
- 10 = HH:MM 12 hours (with '0' before if HH < 10)
- 11 = HH:MM 12 hours AM/PM (with '0' before if HH < 10)
- 12 = HH:MM 24 hours (with '0' before if HH < 10)

Example

Consider the following formula:

FULLTIME (2)

The result of this formula depends on when you print the form. Suppose that you print the form at 3:27:38 PM. So the format is 2 (HH:MM:SS 12 hours AM/PM).

“ 3:27:38 PM”

Finds different examples

- | | |
|------------------------------|------------------------------|
| FULLTIME (1) “ 3:27:38” | FULLTIME (7) “03:27:38” |
| FULLTIME (2) “ 3:27:38 PM” | FULLTIME (8) “03:27:38 PM” |
| FULLTIME (3) “15:27:38” | FULLTIME (9) “15:27:38” |
| FULLTIME (4) “ 3:27” | FULLTIME (10) “03:27” |
| FULLTIME (5) “ 3.27 PM” | FULLTIME (11) “03:27 PM” |
| FULLTIME (6) “15:27” | FULLTIME (12) “15:27” |

IF

The IF function evaluates a specified condition and returns either a value of True (1) or False (0), depending on the result.

Syntax:

IF(logical_test, true_cond, false_cond)

logical_test

Any value or formula that can be evaluated as either True (1) or False (0).

true_cond

The value or formula that is returned if logical_test is determined to be True (1).

This value is not calculated if logical_test returns False (0).

false_cond

The value or formula that is returned if logical_test is False (0).

This value is not calculated if logical_test returns True (1).

Example

Consider the following formula:

```
IF(EXACT(SPOOLTEXT(1,29,1,1), "*", "Text line", "Detail line"))
```

The SPOOLTEXT function returns the first character of the line 29. If this character is equal to '*', then the first part of the IF function is calculated and then printed. If this character is not equal to '*', then the second part of the IF function is calculated and then printed. If logical_test returns a value of "True," the false_cond is not evaluated.

To create a formula with more than one IF function, you can nest more than one IF function, as shown in the following formula:

```
IF( X < 5, "X < 5", IF( X < 10, "X < 10", "X > 10" ) )
```

INT

INT returns an integer portion of a number.

Syntax:

```
INT ( value, dec_separator )
```

Value

The real number you want to round down to an integer.

dec_separator

Indicates which character is used as a decimal separator (optional). By default, the decimal separator is '.'

Example

Consider the following formula:

```
INT( VALUE( SPOOLTEXT( 13,72,8,1 ) ), "." )
```

The SPOOLTEXT function reads data in the spool file and returns a string of "51263.55". The VALUE function must then be used to remove the quotes and convert the string into a number. The new formula would now be as follows:

INT(51263.55, ".")

The INT function simply rounds down a number and returns 51263 as a whole number (without quotes). If you want to correctly round a number, you must instead use the ROUND function.

ISBLANK

ISBLANK determines whether or not a specific text string is empty. If the text is blank, a value of "1" is returned. Otherwise, "0" is returned.

Syntax:

ISBLANK(text)

text

Any text string or formula that returns a string.

Example

Consider the following formula:

```
IF(ISBLANK(SPOOLTEXT(85,81,8,1)), "To continue",  
SPOOLTEXT(85,81,8,1) )
```

The formula reads data from the spool file and detects whether or not the returned string is empty. If the string is empty, then "To continue" is printed. If the string is not empty, then the content of the string is printed.

Suppose you have an invoice form with multiple pages and the invoice total on the last page, and you wish to print special information on all pages of the invoice except the page with the total. Since you know that the total only exists on the last page of the spool file, you can add a field with a formula that detects if the total is empty. If it is, the special information is printed.

ISNUMBER – ISNONTEXT – ISTEXT

These three functions work together to test the outcome of a calculation. ISNUMBER determines if a parameter is a valid number, ISNONTEXT verifies that a parameter is not a text string, and ISTEXT checks if a parameter is a text string.

When combined with IF, CHOOSE or CHOOSELIST functions, they provide a useful method for locating errors in formulas.

Syntax:

ISNUMBER(parameter)
 ISNONTEXT(parameter)
 ISTETEXT(parameter)

parameter

The value you want to test. Value can be a numeric value or a text string.

Example

Consider the following formula:

ISTETEXT(CHOOSE(VALUE(SPOOLTEXT(92,46,1,0)), 5.5, "20.6"))

The SPOOLTEXT function returns a string. Before the returned value can be used by the CHOOSE function, it must be converted into a numeric value. Depending on the value read from the spool file, the CHOOSE function returns 5.5 (numeric) or "20.6" (string). ISTETEXT will then test whether or not the returned value is a string or a number.

- If the CHOOSE function returns 5.5, then ISTETEXT function returns "0" or False.
- If the CHOOSE function returns "20.6", then ISTETEXT function returns "1" or True.

LEFT

LEFT returns a specific character (or characters) from the *beginning* of a text string.

Syntax:

LEFT(text, num_char)

text

The text string or formula that contains the characters you want to extract.

num_char

Specifies the number of characters you want to extract. This value must be greater than or equal to zero.

If num_char is greater than the length of the text, the entire string is returned. If num_char is omitted, a default value of "1" is assumed.

Example

Consider the following formula:

```
LEFT(SPOOLTEXT(1,31,92,1), 1)
```

Using the SPOOLTEXT function, the formula reads data from the spool file. The content of line 31 is returned and the new formula would be as follows:

```
LEFT("Your PO #: 1234567 Our PO #: 876", 1)
```

The LEFT function will then return the first left character of the string, which in this case is ‘*’.

LEN

LEN returns a value representing the precise number of characters (including spaces) in a specific text string.

Syntax:

```
LEN( text )
```

text

The text whose length you wish to determine.

Example

Consider the following formulas:

```
LEN(SPOOLTEXT(12,72,10,1))  
LEN(TRIM(SPOOLTEXT(12,73,10,1)))
```

First the SPOOLTEXT function reads data from the spool file and returns the following value: “ 51263.55.” Note the space at the beginning of the result string. The SPOOLTEXT function removes spaces from the end of the string, but not the beginning. To remove spaces at the beginning of the string, use the TRIM function.

The new formulas are as follows:

```
LEN( “ 51263.55” )  
LEN( TRIM ( “ 500.00” ) ) LEN( “500.00” )
```

The returned values are then 9 and 6.

LOWER

LOWER returns a specified text string after converting it to lowercase.

Syntax:

LOWER (text)

text

The text string or formula you want converted to lowercase.

Example

Consider the following formula:

```
LOWER ( SPOOLTEXT ( 53, 13, 30, 1 )
```

The SPOOLTEXT function returns a string from the spool file. In this case, the returned string is “CAPELLA TECHNOLOGIES.” So, the LOWER function would convert the string to lowercase and the final result of the formula would be “capella technologies”.

MID

This function returns a specified number of characters from a text string starting at a user-defined position.

Syntax:

MID(text, start_num, num_char)

text

The text string containing the characters you want to extract. It could be a formula.

start_num

The position of the first character you want to begin extracting text. If the value of start_num is greater than the length of the text, or less than “1,” MID returns an empty string.

num_char

Specifies the number of characters you want MID to return from the text. If num_char is less than the length of text, but start_num plus num_char exceeds the length of text, MID returns the characters up to the end of text.

Example

Consider the following formula:

MID(SPOOLTEXT(1,32,92,1), 17, 25)

The SPOOLTEXT function returns a string value from the spool file.

“12345678901234567890123456789012345678901234567”

“ 8108900 HP Laserjet Printer 50 ...”

The formula now extracts 25 characters, starting at position 17.

MODULO

MODULO divides a number by a user-specified value and returns a result with the same sign as the divisor.

Syntax:

MODULO (text, divisor)

text

The text number you wish to divide.

divisor

The number by which you want to divide the number defined in text. An error is returned if the divisor is “0.”

Example

Consider the following formula:

MODULO(“1234567489048120”, 97)

MODULO must be used when the number corresponding to the text is greater than 2,147,483,647. If the number corresponding to the text is not greater than 2,147,483,647, you can use the operator “%.”

Suppose you have a string equal to “456,” you can calculate the modulo 54 using two different formulas.

Formula 1

MODULO(“456”, 54)

The result would be the string “24.”

Formula 2

VALUE(“456”) % 54

The result would be the number 24.

NOT

NOT reverses the value of its argument. If the expression defined by a logical argument is False, NOT will return True. If it is True, using NOT will return False.

The NOT function is useful for making sure a certain value is not equal to one particular value.

Syntax:

NOT (logical)

logical

A value or expression that can be evaluated as True (1) or False (0).

Example

Consider the following formula:

```
IF(NOT(EXACT(SPOOLTEXT(71,83,3,1), "USD") ), "Other", "US  
Dollars")
```

The formula begins by reading the spool file. The data is then compared to "USD," and the result of the comparison is reversed. In other words, if the result is True (data is equal to "USD"), the NOT function would return False. Conversely, if the result is False (data is not equal to "USD"), the NOT function returns True.

OR

OR will return a value of True (1) if at least one of the two logical arguments is true. A False (0) value will only be returned if all arguments are false.

Arguments within this function must evaluate to either True (1) or False (0). Complex formulas can return False (0), True (1) or greater than "1" (True).

Syntax:

OR (logical_cond1, logical_cond2)

logical_cond1

Two conditions and may be complex formulas.

logical_cond2

Example

Consider the following formula:

```
IF(OR(EXACT(SPOOLTEXT(1,29,1,1),"*"),FIND("#",SPOOLTEXT(1,29,92,1))=10 ), "Line starting with * or has # character at position 10", "Detail line")
```

The SPOOLTEXT function reads the first character of line 29 in the spool file.

This character is then compared with “*,” and if equal returns a value of True. In the second part of the OR function, SPOOLTEXT reads line 29 and searches for the character “#.” If this character is found at position 10 (as specified in this particular formula), then the second part of the OR function returns True. As you can see, in this case the OR function returns True only if the first character of the specific line is “*” and if there is a “#” character at position 10.

The OR function can also be nested to test more than 2 conditions, as shown in the following formula:

```
OR( OR( x1 > 10 , x1 < 5 ), OR ( y1 > 15 , y1 < 10 ) )
```

Where x1 and y1 are valid numeric values.

AND and OR functions can also be used together in the same formula.

RAND

RAND returns an evenly-distributed random number less than “1” and greater than or equal to “0.”

Syntax:

```
RAND()
```

Example

To generate a random number greater than or equal to 0 (but less than 100) you would use the following formula:

```
INT ( RAND() * 100 )
```

To generate a random number greater than 50 but less than 100:

```
INT( (RAND() * (100 – 50 + 1) ) +50 )
```

READFILE

READFILE reads an external ASCII file into the current data field.

Syntax:

```
READFILE( "filename", keepformat, charset )
```

filename

The name of an external file (in quotes) to be inserted at the current position in the formula.

keepformat

A numeric value defining if the text should keep its current format, as determined by the following:

“0” if CR-LF in the data must be removed.

”1” if the text should keep its current new lines.

charset

A numeric value specifying which character set the data file is using, as determined by the following:

“1” for files using the Windows character set.

”2” for files using the PC850 character set.

”3” for files using the Roman8 character set.

Example

Consider the following formula:

```
READFILE( SPOOLTEXT(6,4,20,1) & ".txt", 1 )
```

In this example, the data located on line 4 and column 6 contains the name of the bank. But since there is no information on this bank in the spool file, an ASCII file containing the relevant information can be created. As the SPOOLTEXT function retrieves the name of the bank, the READFILE function can read the external file corresponding to that bank. So, the new formula would be as follows:

```
READFILE ( "bank 1.txt", 1 )
```

The “1” parameter tells the READILFE function that the CR-LF (new line) must be kept when printing the result.

The READFILE function is useful for including dynamic information that is not present in the spool file when printing. When used in conjunction with the WRITEFILE function, numbers to indicate sequential checks

could be stored in an external file. This file could then be read to determine the current check number and increment additional checks accordingly.

REPLACE

This function replaces a specified number of characters in a text string with a different text string.

Syntax:

REPLACE (text, start_num, num_char, new_text)

text

The text in which you want to replace some characters.

start_num

The position of the character in text that you want to replace with new_text. This value must be greater than “0.” If this value is greater than the text length, it will be assumed to be the length of the text.

num_char

The number of characters you want to replace in the text string. If this value is “0,” new_text will be inserted and no characters in the original text string will be replaced.

new_text

The actual text that will replace the designated characters in the text string.

Example

Consider the following formula:

```
REPLACE( SPOOLTEXT(1,31,20,1), 10, 1, “Number” )
```

The SPOOLTEXT function reads in the spool file and returns a string:

```
1234567890123456789
```

```
“*Your PO #: 1234567”
```

The REPLACE function will remove 1 character at position 10. And this character will be replaced by the new string “Number”. So the result of this formula is:

“*Your PO Number: 1234567”

REPT

This function repeats designated text a specified number of times. REPT is useful for filling a text string with instances of text.

Syntax:

REPT(text, num_times)

text

The text string you want to repeat.

num_times

A positive number specifying the number of times the text will be repeated. If this value is “0,” an empty string will be returned. If the value is not an integer, the additional digits will be ignored.

Example

Consider the following formula:

```
REPT("*",12-LEN(TRIM(SPOOLTEXT(82,36,9,1)))) &
TRIM(SPOOLTEXT(82,36,9,1))
```

The formula begins by reading the spool file with the SPOOLTEXT function and returns the following string: “ 1500.00”. But notice the 2 spaces at the beginning of the string? The TRIM function is then used to remove the two extraneous leading spaces.

Once the formula calculates the length of the string (minus the 2 spaces), and depending on the length, the formula repeats “*” character. So, the new formula is:

```
REPT("*", 12-LEN("1500.00") ) & "1500.00"
LEN("1500.00") is equal to 7, so the new formula is:
REPT("*", 5 ) & "1500.00"
```

And the result is “*****1500.00”

The REPT function is used to generate a formatted string with the same length depending on the length of the input string. By using the REPT and LEN functions, you can ensure the output string will always have the same length.

RIGHT

RIGHT returns a specific character (or characters) from the *end* of a text string.

Syntax:

RIGHT(text, num_char)

text

The text string or formula containing the characters you want to extract.

num_char

Specifies the number of characters you want extracted from the end of the text string. This value must be greater than or equal to “0.” If greater than the length of text, the entire string will be returned. A value of “1” will be assumed if num_char is not supplied.

Example

Consider the following formula:

```
RIGHT ( SPOOLTEXT( 1,32,92,1), 1)
```

The formula starts reading the spool file using the SPOOLTEXT function. It returns a string (line 32). Then the new formula is:

```
RIGHT(“HP Laserjet Printer 50 U 10 15 187.50 140.62 7031.25 2”, 1)
```

The RIGHT function will return the first left character of the string, ‘2’.

RLMC

Similar to the MODULO function, RLMC divides a specified number by a divisor. However, unlike MODULO, RLMC utilizes a constant value of “97.” The purpose of this function is to support a French check application.

Syntax:

RLMC(text)

text

The text number (containing at least 31 digits) which you want to be divided by a value of “97.”

Example

Consider the following formula:

```
RLMC("9541023458741265415894125450238")
```

As you can see, the text contains 31 digits. And the returned code is "09".

ROUND

This function rounds a number to a user-specified number of digits.

Syntax:

```
ROUND( number, num_digits )
```

number

The number you want to round.

num_digits

Specifies the number of digits to round the number. When a value of "0" is used, the number will be rounded to the nearest integer.

Example

Consider the following formula:

```
ROUND(VALUE(SPOOLTEXT(13,72,8,1)) *  
VALUE(SPOOLTEXT(47,72,2,1)) / 100, 2)
```

The formula reads two variables in the spool file and returns the strings "51263.55" and "15."

Each string is converted into a numeric variable (minus quotes), using the VALUE function. The new formula is:

```
ROUND( 51263.55 * 15 / 100, 2 )
```

The print engine now calculates the formula and produces the following new formula:

```
ROUND ( 7689.5325 , 2)
```

Finally, the ROUND function rounds the value to 2 decimal numbers and the end result is 7689.53.

More Examples

```
ROUND ( 564.4568, 2 ) 564.46  
ROUND ( 894.15, 1 ) 894.2
```



```
ROUND ( 9871.615, 0 ) 9872  
ROUND ( 148.999, 2 ) 149.00
```

SEARCH

This function searches for a specific character or text string and then returns a numerical value representing the location of its first occurrence. SEARCH allows for the use of wildcard characters.

Syntax:

```
SEARCH( find_text, within_text [, start_pos] )
```

find_text

The text you want to find. If *find_text* is empty, SEARCH matches the first character defined by *start_pos* in the designated text string. The following wildcard characters can be used to perform the search:

“?” matches any single character.

”*” matches any sequence of characters.

within_text

The text you wish to search.

start_pos

Specifies the character at which to start the search. If this value is omitted, a default setting of “1” is assumed.

Example

Consider the following formula:

```
SEARCH("PO?#", SPOOLTEXT(1,31,92,1) )
```

The formula starts reading the spool file and the SPOOLTEXT function returns a string in quotes “*Your PO #: 1234567 Our PO #: 876”. The new formula is:

```
SEARCH("PO?#", "*Your PO #: 1234567 Our PO #: 876" )
```

The question mark (?) replaces only 1 character. Then the SEARCH function searches “PO?#” and returns the position of the first match character, which in this case would be “7.”

SPOOLTEXT

Returns data from the current data spool page (in quotes). New lines are automatically inserted as `!*n*!` with multiple lines. Starting blanks are removed for numeric fields, and trailing blanks are removed except for barcode fields.

Syntax:

`SPOOLTEXT(col, line, length, height)`

col

The starting column of the data area (first col = 1).

line

The starting line of the data area (first line = 1).

length

The number of columns (characters) for the data area.

height

The number of lines for the data area.

Example

Consider the following formula:

`SPOOLTEXT (6, 13, 30, 4)`

```
000001111111111222222222233333333334
567890123456789012345678901234567890
12
13 Capella Technologies
14 3 Oldfield
15 Irvine, CA 92618
16 USA
17
```

The area to retrieve starts at column 6 and line 13; the number of characters is 30 and the number of lines is 4. The `SPOOLTEXT` function reads the data and stores the result (with quotes) and the special macro `!*n*!` is inserted to indicate new line. The print engine stores the result in memory in the following manner:

`"Capella Technologies!*n*!3 Oldfield!*n*!Irvine, CA 92688!*n*!USA"`

When printed, this data would appear as follows:

Capella Technologies
3 Oldfield
Irvine, CA 92618
USA

Caution should be used when attempting to make calculations from the result of a SPOOLTEXT function. Since the print engine stores the new line as a macro, it might be necessary to instead use the SPOOLTEXTL function to perform a line-by-line calculation.

Another limitation of the SPOOLTEXT function concerns space characters at the beginning of a retrieved value. For example, suppose you use SPOOLTEXT to retrieve data from a spool file, and the print engine stores the following result in memory:

```
" 500.00!*n*! 4546.50!*n*! 38115.25"
```

As you can see, space characters at the end of each part of the string have been removed, but each string starts with space characters. If you attempted to use the VALUE function to convert each string to a number, the formula would be as follows:

```
VALUE(" 500.00!*n*! 4546.50!*n*! 38115.25" )
```

But the result of this formula would be a value of 500.00. This is because characters after the first string (spaces) are not valid digits, so the conversion would halt after the first value. To convert every part of the formula, you must use the SPOOLTEXTL function.

This limitation can also be seen when using the TRIM function, as illustrated in the following formula:

```
TRIM( SPOOLTEXT(23,69,10,3) )  
TRIM(" 500.00!*n*! 4546.50!*n*! 38115.25" )
```

In this instance, the returned string would be as follows:

```
"500.00!*n*! 4546.50!*n*! 38115.25"
```

You cannot use the TRIM function if SPOOLTEXT contains more than one line. Instead, you must use the SPOOLTEXTL function.

SPOOLTEXTL

SPOOLTEXTL returns data from the current spool page and then applies a formula to each line of data. This allows for the application of a formula on a column of values using a single field and a single formula.

Syntax:

SPOOLTEXTL(col, line, length, height)

col

The starting column of the data area (first col = 1).

line

The starting line of the data area (first line = 1).

length

The number of columns (characters) for the data area.

height

The number of lines for the data area.

You can use this function to calculate the product of data in one valid column area with data from another valid column area. If you need to use two SPOOLTEXTL functions in the same formula, the height parameter must be the same in both instances.

Example

Consider the following formula:

```
ROUND(VALUE(SPOOLTEXTL(25,69,8,3))*VALUE(SPOOLTEXTL(
34,69,5,3)) / 100,2)
```

This formula uses the SPOOLTEXTL function to read data line-by-line and then apply the formula to each one. This results in three new formulas, as shown in the following revised formula:

```
ROUND(VALUE(SPOOLTEXT(25,69,8,1))*VALUE(SPOOLTEXT(34,
69,5,1)) / 100,2)
& "!"*n!" &
ROUND(VALUE(SPOOLTEXT(25,70,8,1))*VALUE(SPOOLTEXT(34,
70,5,1)) / 100,2)
& "!"*n!" &
ROUND(VALUE(SPOOLTEXT(25,71,8,1))*VALUE(SPOOLTEXT(34,
71,5,1)) / 100,2)
```

As you can see, the SPOOLTEXTL function has been converted to SPOOLTEXT and the remainder of the formula is applied to the retrieved data.

NOTE: Remember, if you use more than 1 SPOOLTEXT function in a formula then the number of lines (height parameter) must be the same.

You can also mix SPOOLTEXT and SPOOLTEXTL functions, but only if SPOOLTEXT reads one line.

SPOOLTEXTS

This function searches for specific text within a defined region of the current spool file. If located, the text is returned in quotes and new lines are automatically inserted as `!*n*!` with multiple lines. An empty string is returned if the specified text is not found.

Syntax:

SPOOLTEXTS(*text*, *startcol*, *startline*, *stopcol*, *stopline*, *coloffset*, *lineoffset*, *length*, *height*)

text

A string, in quotes, to be search in the specified spooltext area.

startcol

The starting column for the data search area (first col = 1)

startline

The starting line for the data search area (first line = 1)

stopcol

The ending column for the data search area

stopline

The ending line for the data search area

coloffset

The column offset from the location, where the text was found

lineoffset

The line offset from the location, where the text was found

length

The number of columns (characters) to take for the SPOOLTEXT()

height

The number of lines to take for the SPOOLTEXT()

Example

Consider the following formula:

SPOOLTEXTS("ship", 1, 72, 25, 75, 13, 1, 6, 1)

Using the SPOOLTEXTS function, this formula reads data from the spool file from line 7 and column 1 to line 75 and column 25. From this area, the SPOOLTEXTS function searches for the word "ship."

If the word "ship" is found, the SPOOLTEXTS function stores the line and column number of the word. Suppose the word is found on line X and column Y, the SPOOLTEXTS function is then converted into a SPOOLTEXT function, as shown in the revised formula:

SPOOLTEXT(13+X, 1+Y, 6, 1)

The last four parameters of the SPOOLTEXTS function indicates that if information is found in the first area, the returned string is located on the spool file 13 characters after the column position, 1 line after the line position.

As you can see, the SPOOLTEXTS function is useful for retrieving information that might not always be in the same location within the spool file.

STOREN

This function stores a numeric value in a variable. This variable data can then be used in formulas using [\$N#] special field names where "#" is the index number. If the specified data is within the valid index range, and is a number, a value of "1" is returned. Otherwise, a function of "0" is returned.

Syntax:

STOREN(index, value)

index

The numeric variable number where the data has to be stored (0 to 99).

value

The numeric variable value to store (floating point value).

Example

Consider the following formula:

STOREN (44, VALUE (SPOOLTEXT (85, 81, 8, 1)))

Using the SPOOLTEXT function, the formula reads the spool file and returns the following string: “42776.04.”

The returned string is then converted into a numeric value using the VALUE function. The STOREN function stores this result in the numeric array at index 44. This numeric array is available during the printing process, and can be called through a special formula [\$N44].

STORES

Similar to STOREN except this function stores an entire string for use as a variable. This variable data can then be used in formulas using [\$S#] special field names where “#” is the index number. If the specified data is within the valid index range, and is a string, a value of “1” is returned. Otherwise, a function of “0” is returned.

Syntax:

STORES (index, string)

index

The numeric variable number where the data has to be stored (0 to 99).

string

The string variable to store (in quotes).

Example

Consider the following formula:

```
STOREN ( 44, SPOOLTEXT ( 6, 13, 30, 1 ) )
```

The SPOOLTEXT function reads the spool file and returns the string “Capella Technologies.” Then the STORES function stores this result in the string array at index 44. This string array is available during the entire printing process, and can be called by using the special formula [\$S44].

SUBSTITUTE

This function searches through a text string, locates the first occurrence of user-specified text and then replaces it with a new text string.

Syntax:

SUBSTITUTE (text, old_text, new_text)

text

The text string that contains the characters or text you wish to replace.

old_text

The text or characters to be replaced. If this text string is found, its first occurrence will be replaced by *new_text*. If it is not found, *new_text* will not be inserted.

new_text

The text that you want to replace *old_text*.

Example

Consider the following formula:

```
SUBSTITUTE ( SPOOLTEXT ( 86, 81, 8, 1 ), ".", "," )
```

The SPOOLTEXT function reads the spool file and returns the string "42776.04." Using this value, the resulting formula would be as follows:

```
SUBSTITUTE ( "42776.04", ".", "," )
```

The SUBSTITUTE function would then search for all instances of the "." character in the string and replace it with a "," character. The final result of this formula would be "42776,04."

SUMCOL

SUMCOL is a math function that calculates the sum of values in a column.

Syntax:

```
SUMCOL ( SPOOLTEXT( ... , ... , ... , ... ) [, format ] )
```

spooltext

(...) is a column of valid numbers.

format

One of the following values:

"1" indicates the text to be converted has a United States format ("," for thousand separator and "." for decimal place). This is the default value.

“2” indicates the text to be converted has a European format (“.” for thousand separator and “,” for decimal place).

Example

Consider the following formula:

```
ROUND ( SUMCOL ( SPOOLTEXT ( 25, 69, 8, 3 ) ), 2 )
```

The SPOOLTEXT function reads the spool file and returns the values “500.00,” “4546.50” and “38115.25.”

The SUMCOL function would then add these values and the following formula would result:

```
ROUND ( 43161.75000, 2 )
```

The ROUND function further modifies the value, resulting in 43161.75.

If the spool file were to have a European format, the formula would have to be as follows:

```
ROUND ( SUMCOL ( SPOOLTEXT ( 25, 69, 8, 3 ), 2 ), 2 )
```

The SPOOLTEXT function returns the values “500,00,” “4546,50” and “38115,25.”

In this example, the SUMCOL function indicates a European format. The format attribute only indicates the input format, and the output format is always in US format. So, the resulting formula would be as follows:

```
ROUND ( 43161.75000, 2 )
```

The final result would be 43161.75. If you want the final value to be in a European format, use either the FORMAT or TEXT functions.

TEXT

TEXT function converts a value to text in a user-specified format.

Syntax:

```
TEXT ( value, format )
```

value

A numeric value or a formula that evaluates to a numeric value.

format

A string defining the number of decimals for the output string (using '0'), and the symbol to use as decimal separator.

The format text must be in quotes. To fully differentiate the thousand and decimal separators, be sure to define both in format string. If no format is defined, the default format is '0' (no decimals).

Example

Consider the following formula:

```
TEXT( ROUND ( SUMCOL ( SPOOLTEXT ( 25, 69, 8, 3 ) ), 2 ), "000
000,00" )
```

The SPOOLTEXT function reads the spool file and returns the values "500.00," "4546.50" and "38115.25." The SUMCOL function adds these values and produces the following revised formula:

```
TEXT( ROUND ( 43161.75000, 2 ), "000 000,00"
```

The ROUND function modifies the number and rounds it down to only 2 decimal places, resulting in the following formula:

```
TEXT( 43161.75, "000 000,00" )
```

The final result of this formula is the string "43 161,75".

TRIM

TRIM removes leading and trailing spaces in a text string.

Syntax:

```
TRIM ( text )
```

text

The text string from which you want spaces removed.

Example

Consider the following formula:

```
TRIM ( SPOOLTEXT ( 81, 81, 12, 1 ) )
```

The SPOOLTEXT function reads the spool file and returns the string " 42776.04." Using this value, the new formula would be as follows:

TRIM (" 42776.04")

The TRIM function removes the spaces and the final result is "42776.04."

TRIMC

TRIMC removes user-specified leading and trailing characters from a text string.

Syntax:

TRIMC (text, character)

text

The text string you wish to modify.

character

The specific character you wish to delete from the beginning and/or end of the string.

Example

Consider the following formula:

TRIM (SPOOLTEXT (75, 83, 18, 1) , "*")

The SPOOLTEXT function reads the spool file and returns the string "*****42776.04*****." Using this value, the new formula would be as follows:

TRIM ("*****42776.04*****" , "*")

The TRIM function removes all instances of the "*" character, so the final result would be "42776.04."

UPPER

UPPER modifies a specific text string and converts all lower-case characters to uppercase.

Syntax:

UPPER (text)

text

Either a text string or formula you wish to convert to uppercase.

Example

Consider the following formula:

```
UPPER ( SPOOLTEXT ( 6, 13, 30, 1 )
```

The SPOOLTEXT function reads the spool file and returns the string “Capella Technologies” from the specified location. The UPPER function would then convert the string to uppercase and the final result would be “CAPELLA TECHNOLOGIES.”

VALUE

VALUE converts a text string representing a number into a numerical equivalent.

Syntax:

```
VALUE ( text, [format ] )
```

text

The text (in quotes) containing the text you want to convert.

format

A digit value: 1 means the text to convert has a US format (',' is the thousand separator and '.' is the decimal separator). 2 means the text to convert is in European format ('.' is the decimal separator and ',' is the thousand separator). The default value is 1.

- This function is used to make calculations with spool data. Because the SPOOLTEXT function returns a text string, it is mandatory to convert the text string before making calculation.
- VALUE function returns a number in floating data type.
- The returned value is always in US format, even when indicating a European input format.

Example

Consider the following formula:

```
VALUE ( SPOOLTEXT ( 15, 73, 6, 1 ) )
```

The SPOOLTEXT function reads the spool file and returns the strong “500.00.” Thus, the new formula would be as follows:

```
VALUE ( “500.00” )
```

The result is 500.0000000 because the VALUE function returns a number in floating data type. You have to use a VALUE function with a ROUND or a TEXT function, to remove insignificant zero.

If the formula is:

```
VALUE ( SPOOLTEXT ( 15, 74, 6, 1 ) )
```

The SPOOLTEXT function reads the spool file and returns the string "500,00." So, the new formula would be as follows:

```
VALUE ( "500,00" )
```

Because the data you read has a European format and because you forgot to specify the input format, the output numeric value is 500000.00000. In this case you have to modify the original formula:

```
VALUE ( SPOOLTEXT ( 15, 74, 6, 1 ) , 2 )  
VALUE ( "500,00" , 2 )  
500.000000
```

As you can see, the output numeric value is in US format. Usually you use a TEXT or a ROUND function to correctly format the European output number.

WRITEFILE

WRITEFILE writes the current field data to an external file.

Syntax:

```
WRITEFILE( "filename", "data")
```

filename

Filename of the external file (in quotes) where the data is to be written. If this file already exists, it will be overwritten by the new data. Otherwise, it will be created.

data

The specific data (in quotes) to be written to the file.

- Include `!*n*` macro command in data to insert CR-LF in file.
- Returns 1 if the file is created and data successfully written.

Example

Consider the following formula:

```
WRITEFILE( "Counter.txt", TEXT( VALUE( [ $$1 ] ) + 1, "00000000" ) )
```

Before calculating the formula, the [\$S1] special field is replaced by the appropriate value. If the string stored in [\$S1] is “1000002”, the formula would be as follows:

```
WRITEFILE( "Counter.txt", TEXT( VALUE( "1000002" ) + 1,  
"00000000" ) )
```

The VALUE function is then calculated.

```
WRITEFILE( "Counter.txt", TEXT( 1000002 + 1, "00000000" ) )  
WRITEFILE( "Counter.txt", TEXT( 1000003, "00000000" ) )  
WRITEFILE( "Counter.txt", "1000003" )
```

The string “1000003” would then be written to the external file called “COUNTER.TXT”.

This function can be used with the READFILE function to read and write a file that contains a counter. For example, you can store the next check number used by a check printing application in an external file. Then you can read this file to know the current check number, and then increment that number and write it into the same external file.

Dynamic Text Formatting

Special macro commands can be inserted into text formulas to modify the appearance of the text merged into fields. These formatting commands begin with the command “!*” and end with “*!”. These text formatting commands can be used individually or together within the same text string.

Commands

NOTE: Any text formatting macro command must be in quotes in the formula, like any other text.

!*n*!

New Line command. When found in merged text, a new line is automatically created.

!*v:ptSize*!

Point size command, where the ptSize is the numerical value of the new font point size. This can be applied to all installed fonts.

!*s:S1,S2*!

Font size command. S1 could be R (regular), B (bold), I (italic) or A (all—bold and Italic). S2 can also be R (regular), U (underlined), S (strikeout) or A (all—underline and strikeout). The values of S1 and S2 could also be equal to X if you don't want the attribute changed.



Be careful whenever using this macro command. When changing text attributes, you need to verify that the attribute you specify is included in the compiled form. Although the compilation process normally exports font information, FormPort Designer does not know what font to export when formulas dynamically change the attributes, and will assume the different font properties indicate a new font. A work-around for this problem is to add a text object outside the page using the font you want to be included during the compilation.

!*f:name_of_font*!

Change font command. Where name_of_font is the name of the new font you want to use. Only fonts installed in FormPort Designer are valid.

!*c:c1,c2,c3*!

Change font color command. Colors are defined by standard RGB values between 0 and 255. In this command, c1 is the RED value, c2 is for GREEN and c3 is BLUE.

Text Formatting Example

It is now time to see an example of these text formatting commands in action. Consider the following sentence:

It **must** be *possible* to create a *static* text object like this

Word	Font and Attributes
It	Arial regular black 12
must	Arial regular red 20
be	Arial regular black 12
possible	Arial italic black 14
to	Arial regular black 12
create	Times New Roman regular black 14
a	Arial regular black 12
static	Arial bold/italic black 9
text	Arial regular black 12
object	Arial regular black 14 underlined
like this	Arial regular dark green 12

As you can see, the formula to implement this appearance is as follows:

```
"It !*v:020*!!*c:255,000,000*!must!*v:012*!!*c:000,000,000*! be
!*v:014*!!*s:IR*!possible!*v:012*!!*s:RR*! to !*f:Times New Roman*!
!*v:014*!create !*f:Arial*!!*v:012*! a !*v:009*!!*s:AR*!static!*v:010*!
!*s:RR*! text !*v:014*!!*s:RU*!object !*v:012*!!*s:RR*!
!*c:000,128,000*!like this"
```


Special Fieldnames

Some special fieldnames exist within FormPort Designer for the purpose of retrieving information from the form itself, the environment, the data file, etc. These special fieldnames always begin with the “\$” character and can be used just like regular field names.



NOTE

Be sure special fieldnames are surrounded by square brackets.

[\$FORM]

This fieldname is used to get the form short name but without the path.

[\$TITLE]

Used to get the title of the form. An alternative to this would be [\$SHORTDESC].

[\$AUTHOR]

Used to retrieve author information from the form.

[\$REVISION]

Used to get the revision number.

[\$DESDATE]

Retrieves the date of the last save.

[\$DESCRIPTION]

Used to get the description of the form. An alternative to this would be [\$LONGDESC].

[\$COPY]

This command retrieves the current copy number—but only in software multi-copy mode. This mode differs from the hardware multi-copy mode that is specified through either the Printer Information or Page Setup.

[\$SPAGE]

Used to get the currently-merged spool page number. Since the print engine is not capable of knowing how many spool pages are in a file at the beginning of the printing process, you cannot use a formula such as “page x/y” (whereas ‘x’ indicates the current page [\$SPAGE]) since the value of ‘y’ cannot be determined.

[\$PAGE]

Used to get the currently-used document page number. [\$PAGE] returns 1 if page 1 of the form is used, and so on.

[\$PSIDE]

Used to retrieve the paper page side. [\$PSIDE] returns a value of “1” if the front side of the paper, and a value of “2” indicates the back. This fieldname is only available when operating in duplex mode.

[\$DATANAME]

Used to get the *short* name of the data file without path or extension.

[\$DATAFILE]

Used to get the *extended* data file name with path and extension.

[\$ENDOFDATA]

Determine if the current page of the spool file is the last. A value of “1” is returned if the current page is the last one. If not, a value of “0” is returned.

[\$U1] to [\$U9]

Stores variables defined using the -U#=parameter command line option.

[\$S1] to [\$S99]

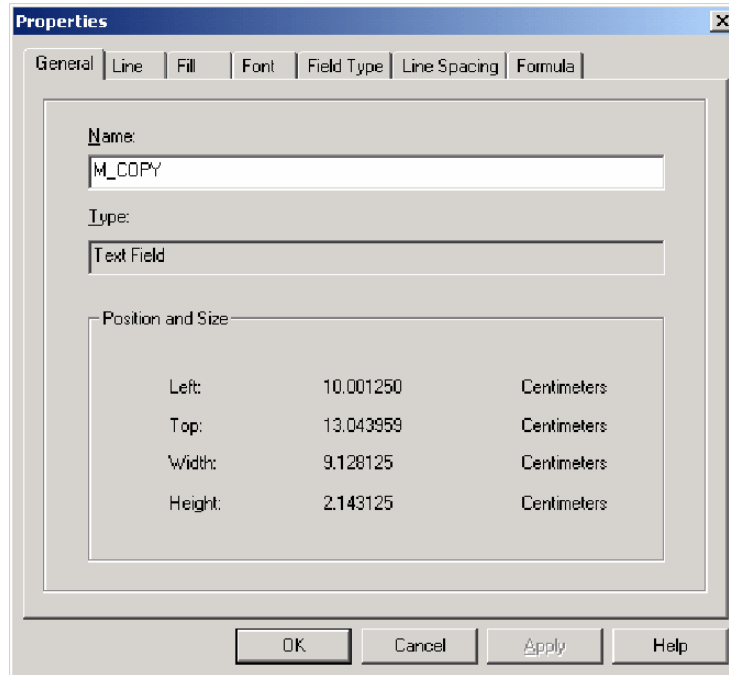
stores text string variables defined using STORES function.

[\$N1] to [\$N99]

stores text string variables defined using STOREN function.

Special Names for Fields

To use data within a field to perform calculations on a printing page, you can define special names through the property sheet of that object, as showing in the following image:



These special names all begin with a “M_” prefix, and are described in detail below.

M_COPY

This name is used to set the number of copies printed for the current page. The number of copies can then be applied to both printer multi-copy and software multi-copies.

M_DUPLEX

Used to set the status of duplex mode. The formula of this field must return “0” to deactivate the duplex mode, “1” to activate (long edge) duplex mode, or “2” to activate (short edge) duplex mode.

M_INTRAY

Used to set the input paper tray for the current page. The returned numeric value defines the input tray number.

M_OUTTRAY

Sets the output paper tray for the current page. The returned numeric value defines the output tray number.

Since the result of these fields can affect the current page, they're all calculated before sending the page header. This means that Field Filling Order is not used for these fields and the print engine will calculate them regardless of field order.

Chapter 6: Troubleshooting

This chapter contains a collection of common problems and their solutions.

General Problems

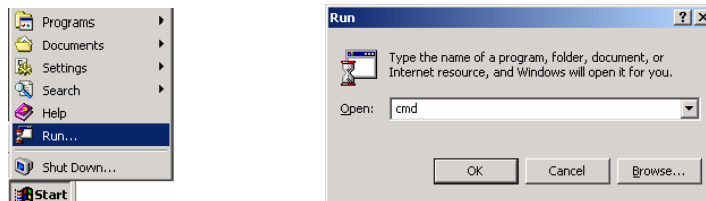
Poor Quality of Imported Bitmaps

Although FormPort Designer allows you to import a number of different bitmap image formats, all bitmap images are converted to PCX format for use within the application. Depending on the specifications of the imported image, this conversion may alter the appearance of the image object. For best results, convert any bitmap images you plan to import to the PCX format using a paint or photo editing application.

Manually Printing a Form

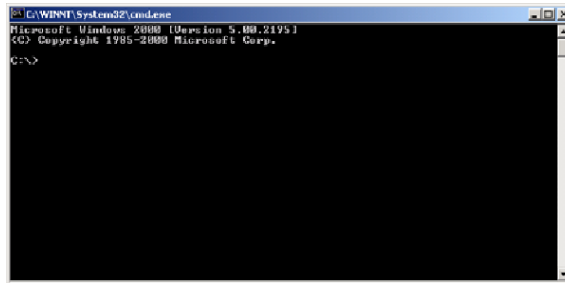
Manually printing is a useful method of testing a form and the associated spool file since FormPort Designer does not report errors when printing. To print manually, the form must first be compiled. The procedure for manually printing forms depends on which operating system you are using.

Note the directory and the name of the form and then open a DOS session by clicking on **Start** and then **Run** from the Windows menu.



In the **Run** dialog, type “**cmd**” and click **OK**.

A DOS session window will open, as shown in the following image:



Change directories to where you placed the compiled document. For example, if you placed the compiled form in the FormPort Designer directory, you'd type the following in the DOS window:

cd \Program Files\CapTech\FormPort Designer

And then press the **Enter** key.

To print the file, enter the following command:

c:\Program Files\CapTech\FormPort Designer\Bin\prt6.exe sample.fpf



Be sure to substitute the file "sample.mff" with the name of your file.

Press the **Enter** key to print the form on your screen.

```

C:\WINNT\System32\cmd.exe
Microsoft Windows [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>cd "\Program Files\Bluenega\MegaForm Design 6\Compile"

C:\Program Files\Bluenega\MegaForm Design 6\Compile>"c:\Program Files\Bluenega\MegaForm Design 6\Bin\prt6.exe" sample.nff
*****
* MegaForm Print Engine 06.09D                               Internal Design *
*****
* Copyright (c)2002, 2003 Bluenega Technology                 *
*****
Form File: sample.nff
No Data file
Output   : sample.prn

Processing Form file...
Image error: can't read palette of MEGA.PCX: corrupt file trailer
Starting Data merging
Done.

C:\Program Files\Bluenega\MegaForm Design 6\Compile>

```

In this example, the exported image has a bad header that the print engine is unable to print. This problem can be corrected by replacing the bad image.

Validating the Input File Format

One of the best ways to eliminate problems with the FormPort Designer print engine is to verify that the spool file associated with the document is valid. When working with a spool file, the most important value to know is the number of lines per page. Whenever you load a spool file into the current page, the number of lines per page is automatically detected. If necessary, this value can be manually changed.

The number of lines per page means that the print engine will read this number of lines from the spool file. If it does not find a formfeed character before the end of the page, all lines are stored. If it finds a formfeed character before the number of lines, it stops reading the spool file. It can never read more than this number of lines per page.

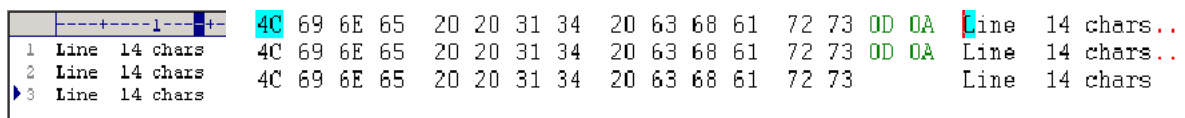
If the number of lines per page is 86, it means that every 86 lines, the print engine will make a page break. This rule is only if one page of the spool file is printing one page of the form.

Special characters

Spool files contain a number of special characters to make them function correctly. A few of these characters are described below.

CR-LF

CR-LF indicates a Carriage Return – Line Feed. These two characters indicate a new line in an ASCII editor. CR is equal to ASCII 13 (OD) and LF is equal to ASCII 10 (OA). The following figure shows a line in an ASCII editor and the exact same line in a hexadecimal editor.



```
-----+-----+-----+-----+
1 Line 14 chars 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 OD OA Line 14 chars..
2 Line 14 chars 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 OD OA Line 14 chars..
3 Line 14 chars 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 Line 14 chars
```

As you can see, to separate two lines OD-OA is inserted (CR-LF)

FF

FF indicates a FormFeed. A FormFeed character indicates a page break for the printer and the printer engine. FF is equal to ASCII 12 (OC). The following figure shows a line in an ASCII editor and the exact same line in a hexadecimal editor.

<pre> 1 Line 14 chars 2 Line 14 chars 3 Line 14 chars </pre>	<pre> 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 0D 0A 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 0D 0A 0C 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 </pre>	<pre> Line 14 chars.. Line 14 chars.. Line 14 chars </pre>
--	---	--

As you can see, there is a FormFeed character (0C) in the file. It tells the print engine to generate a page break and start a new page after the FormFeed character.

Character ASCII '00'






























This character indicates the end of a string, and will halt printing the remainder of the line when encountered by the print engine. An example of this character can be seen in the following illustrations; the one on the left is an ASCII editor and the one on the right is the same line in a hexadecimal editor.

<pre> 1 Line 14 chars 2 Line 14 chars 3 Line 14 chars </pre>	<pre> 4C 69 6E 65 20 00 31 34 20 63 68 61 72 73 0D 0A 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 0D 0A 4C 69 6E 65 20 20 31 34 20 63 68 61 72 73 </pre>	<pre> Line 14 chars.. Line 14 chars.. Line 14 chars </pre>
--	--	--












When the print engine encounters the '00' character, it stops without reading the remainder of the line.

Appendix A: FormPort Designer Shortcut Commands

For your convenience, here is a complete list of every command that has an associated icon and/or keyboard shortcut.

Icon	Keyboard	Menu Command
	ctrl+N	File→New
	ctrl+O	File→Open
	ctrl+S	File→Save
	--	File→Compile Document
	--	File→Print Preview
	ctrl+P	File→Print
	ctrl+Z	Edit→Undo
	ctrl+Y	Edit→Redo
	ctrl+X	Edit→Cut
	ctrl+C	Edit→Copy
	ctrl+V	Edit→Paste
--	Del Key	Edit→Delete
--	ctrl+A	Edit→Select All
	Alt+Enter	Edit→Properties
	--	View→Grid
	--	View→Properties Window
	--	View→Output Window
	--	View→Zoom→Zoom
	--	View→Zoom→Zoom to Selection
--	F2	Page→Spool Mapping→Spool File→Display Copied area
--	F3	Page→Spool Mapping→Spool File→Display Moved area
--	F4	Page→Spool Mapping→Spool File→Display Unused area
	--	Insert→Spool Text
	--	Insert→Spool Image
	--	Insert→Spool Barcode
	--	Insert→CSV Spool Text
	--	Insert→CSV Spool Image
	--	Insert→CSV Spool Barcode
	--	Insert→Field
	--	Insert→Rich Text
	--	Insert→Image
	--	Insert→Barcode
	--	Insert→Line
	--	Insert→Rectangle

Appendix A: FormPort Designer Shortcut Commands

	--	Insert→Round Rectangle
	--	Insert→Ellipse
	--	Format→Justification→Left
	--	Format→Justification→Center Horizontally
	--	Format→Justification→Right
	--	Format→Justification→Justify
	--	Format→Justification→Top
	--	Format→Justification→Center Vertically
	--	Format→Justification→Bottom
	--	Tools→Snap to Grid
	--	Help→About

Appendix B: FormPort Print Utility

FormPort Print is a command-line utility that provides all the same functionality as printing directly from FormPort Designer.

The command line format of this utility is as follows:

Syntax:

FORMPORT <formName> [dataFile] [options]

<formName>

Name of the compiled FormPort file to print. The extension FPF is assumed if one is not provided. This is the minimum parameter required unless the `-v` (verbose) option is defined.

[datafile]

File name of the spool or ASCII data file to be merged with the compiled form. If no value is provided, a blank form will be printed.

[options]

Optional parameter enabling more features and options.

Option Parameters

This section contains a complete list of the option parameters available when using FormPort Print.

-c#

Indicates printer-based multi-copy printing. When used, the printer will print the specified number of similar copies of each page.

Acceptable values for #: **1 to 999**

-C#

Software-based multi-copy printing. When used, FormPort Print will print the specified number of copies of each page. Each data file page will be applied # times to each form page and the [\$COPY] value used in calculations will reflect the occurrence number.

Acceptable values for #: **1 to 999**



-f

This parameter is only applicable when printing spool files.

Used to indicate multi-part sorted pages.

-i#

Overrides the forms first macro number (defined during form design).

Acceptable values for #: **1 to 32750**

NOTE: For multi-page documents, each page uses one macro number. For example, if you specify 100 as the first macro number for a 4 pages document, the first page will be stored in printer memory as macro 100, the second page as macro 101, third page as macro 102 and so on. Make sure documents loaded into printer memory do not have overlapping macro numbers.

-l

Specifies that the forms have been preloaded into the printer.



When the form **Storage** has been set to **Temporary in RAM** in FormPort Designer, this parameter has no effect and the form will be reloaded into printer memory.

-m#

Data character set. When used, indicates which character set the data file is using.

Acceptable values for #:

1 = Windows ANSI (default)

2 = PC850

3 = Roman 8

-n

Instructs the utility to perform no Post Reset (only available for OS/2). In OS/2 V3 and older, the OS/2 print manager performs a Form Feed even when the last byte is not form feed—resulting in generating a blank page after each print job. Using this parameter will prevent that from happening.

-oOutput

Indicates the output destination (file or device name.) To use, replace **Output** with the name of the destination file or device.

Examples:

-oLPT1 for a device like LPT1

-o\\SERVER\HP5SI for a LAN queue (OS/2, NT)

-oprintfile.prn to generate a print file

-ostdout for a print file sent to “stdout”

If no output destination is indicated, the output device name is taken from the MEGAOUT environment variable. If no MEGAOUT environment variable is defined, the name of the form, with the PRN extension is used.

-O"Unix process"

This option is available only in the Unix environment to indicate a Unix process, i.e. to indicate a Unix printer.

-oOutput option is only used to generate an output file.

Example:

-O"lp -dHP4000 -oraw"

In this instance, HP4000 is the name of Unix printer.

-p<model>

Indicates a printer different than the one specified while creating the form in FormPort Designer. When used, FormPort Print will automatically adapt the printing to fit the specified model (fonts, colors, graphic resolution, printer language, etc.)

<model> must be one of the following printer models:

LJ3, LJ3D, LJ3S, LJ3P: HP LaserJet III family (S=Si)

LJ4, LJ4S, LJ4L, LJ4P, LJ4PL, LJ4V: HP LaserJet 4 family (S=Si)

LJC: HP LaserJet Color

LJ5C, LJ5L, LJ5S, LJ5, LJ5P: HP LaserJet 5 family (S=Si, C=Color)

LJ6P, LJ6L: HP LaserJet 6 family

DJ1200, DJ1600: PCL5 color HP DeskJet family

LJ4000

HPPCL4: All PCL4 printers

HPPCL5: All PCL5 printers

HPPCL5E: All PCL5E printers

HPPCL5C: All PCL5C printers

HPPCL5EC: All PCL5EC printers

PROPR: IBM Proprinter dot matrixprinter (data only)

4722: IBM Financial impact printer (data only)

-q

Quiet form download. When Specified, forms are loaded in the printer but not printed.

-s#1,#2

Spool pages print range. FormPort Print will print only the spool file pages in the specified range, from pages #1 to #2.



-t

This is applicable when printing spool files.

Instructs the FormPort Print utility to return the number of spool pages processed, instead of the error code. A value of “0” is returned if an error occurs.

-u#=<value>

Defines the [\$U1] to [\$U9] User Variables for fields formulas.

Acceptable values for #: **1 to 9**

<value> is an alphanumeric string, without blanks.

-v

Returns the Print Engine version number as a return code (multiplied by 100: value from 100 to 9999). When invoked with **-v** as the first parameter, FormPort Print quits and returns the Print Engine version number multiplied by 100. Example : version 5.10 is returned as 510.

-v#

Specifies the verbose mode.

Acceptable values for #:

0 = no screen output (silent mode)

1 = command line and input/output info.

2 = command line, input/output and doc. info.

3 = command line, input/output, doc. & page info.



-w2=#

This mode can be used by the FormPort technical support center to determine the source of form problems.

Allows not taking into account the first *x* bytes in the spool file. # is a numeric variable.

-w16=#

Allows not taking into account the first *x* lines in the spool file. # is a numeric variable.

Return Value

Once the command line is executed, a numerical value indicating the status of the print job is returned.

Depending on the option parameters used, the following information will be returned instead:

- If **-v** was specified, a value representing the module number multiplied by 100 is returned.
- If **-t** was used, the number of spool pages printed by FormPort Print is returned.

If neither of those parameters is defined and an error is encountered, the run-time sends back a value greater than zero and a message describing the error is displayed.

The table below lists all potential return values:

Return Values	
0	No error encountered.
2	Read error in compiled FormPort file.
3	File specified is not a valid FormPort file.
4	Unexpected end of file in FormPort file.
5	Corrupt FormPort file.
7	Cannot open output device or file for printing.
8	Cannot open FormPort file.
9	Cannot open data file.
10	Form's validation formula failed.
11	Memory allocation error when reading FormPort form.
12	Printer-specific font file not found.
14	Corrupted font file.
20	Invalid data file pages range.
21	Invalid copies parameter.
22	Unknown target printer.
23	Unknown data character mapping ID.
24	First command line macro is out of range.
50	Next Page formula page number out of range.
51	Form validation formula looped over 10 consecutive times.
300	Memory allocation error occurred. Contact technical support for assistance.

Environment Variables

The following environment variables are also available:

MEGAOUT

This variable can contain the name of the output device where the data is to be sent. If no **-o** parameter is defined on the command line, that variable will be checked for a destination.

NOTE: MEGAOUT can be a device name like “stdout.”

MEGAOPT

This variable can contain the default settings for the FormPort Print Engine. This is especially useful when the Print Engine is called as a filter using “stdin” and “stdout” standard I/O.

Glossary

administrator

The person responsible for maintaining a computer network.

architecture

The broad design of either hardware or software which defines the structure, capabilities and limitations of the system.

case-sensitive

The ability of an operating system to distinguish between upper- and lower-case letters. For example, “A” would be considered a completely different character than “a.” This is important to know when creating or entering a system password.

client

An application that runs on a computer and relies on a server to perform some specific operation.

data

Pieces of information formatted in a precise manner. Data is what computer programs produce and/or manipulate.

database

Essentially an electronic file system, databases are a collection of information stored in a specific manner that enables computer systems to quickly and easily retrieve them.

driver

A computer program that enables a computer to communicate with a peripheral device such as a printer, scanner or network card.

duplex

When referring to printers, duplex is the ability of some printers to automatically print on both sides of a sheet of paper.

groups

Within a computer network, a collection of computers and devices as defined by an administrator.

Intranet

An internal network accessible only to those within a particular organization, such as a company or corporation.

IP address

A 32-bit numeric address that identifies each device connected to a computer network. Every device connected to the network must have a unique IP address in order to function.

Javascript

A scripting language that enables web programmers to enhance the features and functionality of their websites.

node

Any device on a computer network with a unique network address. This term applies to both computers and printers.

parameter

A characteristic or argument imposed on a program by a user.

port

An interface on a computer through which peripheral devices connect.

printer

An electronic device attached to a computer capable of rendering text and/or graphics onto paper.

print server

A computer on a network that manages one or more printers connected to the network. Any user connected to a network can print by sending their job to the print server.

server

General term for a computer or a device that manages resources on a network.

spool

A single page representation of records in a datafile that is used by FormPort Designer for data mapping.

user

General term for someone operating a computer.

WYSIWYG

What You See Is What You Get. A type of software that provides the user with a visual approximation of what the final document will look like after it has been printed.

Index

About command.....	76
ADDFILE function.....	125
Adding fonts.....	109
Align Toolbar.....	34
ALLSUBSTITUTE function	126
AND function.....	127
Angle Snap command.....	73
Arrange Icons command.....	75
Auto Select Tools.....	74
Barcode command	65
Barcode properties	55
Bottom align.....	68
Bring Forward.....	72
Bring to Front.....	72
Canvas Toolbar	34
Carriage Return – Line Feed.....	175
Cascade Windows command	75
Center Horizontally.....	67
Center Vertically.....	68
CHAR function.....	128
CHOOSE function	129
CHOOSELIST function.....	129
CLEAN function.....	130
Close All Windows command	75
Close command.....	22
CODE function	131
Compile Dialog.....	24
Compile Document As... Command.....	24
Compile Document... command	24
configuration, DIMM/flash.....	77
Copy command.....	30
CSV Field properties.....	58
CSV files.....	20
CSV Spool Barcode command	63
CSV Spool Image command.....	62

CSV Spool Text command	62
CSVTEXT function	132
Customize PDF File Name Tutorial	94
Cut command	29
Date Field command	63
Date Format properties	59
Default Properties	32
Delete command	30
DIR function	132
Display 'Moved' Area command.....	44
Display 'Unused' Area command.....	45
Display Copied Area command.....	44
Document Information.....	25
Document Setup.....	26
Drawing Toolbar	35
Dynamic Text Formatting.....	167
Edit Menu.....	29
Edit Text command.....	30
Ellipse command.....	65
E-mail tutorial	96
environment variable, FormPort Print	185
EURO function	133
EXACT function.....	134
Exit command	28
EXIT function	135
FAX Tutorial.....	100
Features and Benefits.....	19
Field command.....	63
Field Filling Order	69
File Menu	22
Fill properties	50
Filter Selection... command	31
FIND function.....	135
Fit to Height command	43
Fit to Page command	43
Fit to Width command	43
Font command	67
Font properties	53
font reference	177

Font Toolbar.....	39
FORMAT function.....	136
Format Menu.....	67
FormFeed character	176
FormPort Designer tutorial	78
FormPort Print environment variables.....	185
FormPort Print Option Parameters.....	180
FormPort Print return values.....	184
FormPort Print utility	179
FormPort Print utility syntax	179
Formula properties	57
Free Rotation.....	73
FULLDATE function.....	137
FULLTIME function	138
General properties	48
Glossary	187
Go To Page command.....	47
Grid	41
Grid Properties	73
Group	71
Group Align command	70
Group commands	71
Group Layout command	71
Group Resize command.....	69
Help contents	76
Help Menu	76
IF function.....	139
Image command.....	64
Image properties.....	54
Import Dialog.....	25
Import Document... command	25
Import Form limitations.....	117
Importing forms	116
Index Fields Tutorial.....	104
Insert Menu	48
Insert Page command.....	46
Inserting objects	110
Installation.....	21
INT function.....	140

ISBLANK function.....	141
ISNONTEXT function.....	141
ISNUMBER function.....	141
ISTEXT function	141
Justify text.....	68
Layout Toolbar.....	35
LEFT function.....	142
Left Justification	67
LEN function	143
Library/Export command.....	31
Library/Import command.....	31
Line command	65
Line properties	49
Line Spacing properties	50
Load CSV File command.....	45
Load Spool File command.....	44
LOWER function.....	143
Manually printing.....	173
MID function	144
Modifying objects.....	113
MODULO function.....	145
Multiple Accounts tutorial.....	91
New command	22
NOT function.....	146
Nudge commands.....	72
Nudge Down.....	72
Nudge Left	72
Nudge Right.....	73
Nudge Toolbar	36
Nudge Up.....	72
Numeric Field command.....	64
Numeric properties.....	60
Open Dialog.....	22
Open Files List command.....	75
Open... command.....	22
Option Parameters, FormPort Print.....	180
Options command.....	74
OR function.....	146
Overview.....	15

Page Menu	44
Page Number Field command.....	64
Page Number properties.....	60
Page Setup command.....	47
Paste command	30
Print command.....	27
Print Preview.....	27
print server, viewing	77
Properties	32, 48
Properties Window.....	41
Property Toolbar	37
RAND function.....	147
READFILE function.....	148
Rectangle command.....	65
Redo command	29
Register	76
Reload Spool File command.....	44
Remove Page command.....	46
REPLACE function	149
REPT function	150
return values, FormPort Print.....	184
Rich Text command.....	64
RIGHT function	151
Right Justification.....	67
RLMC function.....	151
Rotate commands.....	73
Rotate Toolbar	37
Round Corners properties	51
ROUND function.....	152
Round Rectangle command	65
Ruler.....	41
Save As Dialog	23
Save As... command.....	23
Save command.....	23
SCS Setup	109
SEARCH function	153
Select All command.....	31
Send Backward	72
Send Mail command	27

Send to Back	72
Snap to Grid command	73
special characters	175
Special Fieldnames	169
Spool Barcode command	62
Spool File Options command.....	45
Spool files	20
Spool Image command	62
Spool Info properties.....	56
Spool Text command	61
SPOOLTEXT function	154
SPOOLTEXTL function.....	155
SPOOLTEXTS function	157
Standard Toolbar.....	33
Status Bar	41
STOREN function.....	158
STORES function	159
Structure Toolbar	38
Style Manager	68
SUBSTITUTE function	159
SUMCOL function.....	160
Technical Support	76
Text Formatting commands	167
TEXT function	161
Tile Windows command	75
Time Field command	63
Time Format properties.....	59
Tools Menu	69
Top align	68
TRIM function	162
TRIMC function.....	163
Troubleshooting	173
Undo command.....	29
Ungroup	71
Unload CSV File command	46
Unload Spool File command	44
UPPER function.....	163
validating input file.....	175
VALUE function.....	164

View Menu.....	33
Window Menu	75
WRITEFILE function	165
WYSIWYG.....	19
Zoom command	42
Zoom Custom command.....	42
Zoom To Selection command.....	43
Zoom Toolbar	38