

Elixir

Vitesse®

User Guide

Elixir Technologies Corporation
721 East Main Street
Ventura, CA 93001

Copyright © 2008 by Elixir Technologies Corporation.
All rights reserved.

VT200801V200

All product names and trade names used herein are trademarks of their respective owners.

To report any errors you might find, please send a note to errata@elixir.com. Please note: e-mails are used to correct future versions of this guide and may not receive an individual reply. For technical support, please contact Customer Support.

Welcome

This document provides background and procedural information for using Vitesse. Because business requirements and user preferences can vary for each company, the material contained herein should be viewed as a general guide only.

As you step through the material, please remember that your specific business needs will govern selection from the various options and preferences described. Similarly, certain selections will require that some tasks be repeated, while other selections will allow you to omit tasks entirely.

Who Should Use this Guide?

Users should already be familiar with Microsoft Windows operations, printer and font management operations, and PC peripherals, and have access to reference manuals for related PC software and hardware.

About This Guide

Vitesse user guide is organized as follows:

Chapter 1: Introduction, provides an overview of the Vitesse project modes, functions and features that facilitate creating projects and resources for VIPP printing. It also provides information on the new features introduced in Vitesse 2.00.

Section I: Vitesse Basics

Chapter 2: Understanding the Vitesse Workspace, provides an overview of the Vitesse workspace and describes activities and tasks for using the workspace.

Chapter 3: Understanding the Vitesse Directory Structure, explains the Vitesse directory structure, the scope of resources used in Vitesse, and associated tasks for setting up working folders.

Section II: Using VIPP Resources

Chapter 4: Working with Forms, provides information on the form resource and explains tasks for using forms, setting form properties, and caching resources.

Chapter 5: Working with Segments, provides information on using the segment resource and provides tasks for creating a segment, setting segment properties, caching a segment, and dynamically resizing segments.

Chapter 6: Working with Fonts, provides information on the different font styles available in Vitesse. It gives a detailed overview on the font types, mappings, and their significance in VIPP printing.

Chapter 7: Working with Images, describes how to add images to a project and set image properties.

Section III: Vitesse Project Tools

Chapter 8: Using the Drawing Tools, provides information on drawing tools. Included are instructions for adding drawing objects in the form and setting associated properties. It also describes the basic operations for editing/modifying objects inserted in a design.

Chapter 9: Using the Text Tools, provides activities and tasks for using text and graphic text objects and setting associated properties. It also gives a detailed overview on using rich text from external projects, spell checker and dictionary files, finding and replacing text, and for setting the default text properties.

Chapter 10: Applying Transformations, describes how to apply different formatting options on field, text, and graphic text objects.

Chapter 11: Working with Colors, describes activities and tasks for working with colors, color palettes, and patterns.

Section IV: Creating a VIPP Project

Chapter 12: Creating a Native Mode Project, describes activities and tasks for creating VIPP native mode projects.

Chapter 13: Creating a Database Mode Project, describes activities and tasks for creating VIPP database mode projects.

Chapter 14: Creating a Line Mode Project, activities and tasks for creating VIPP line mode projects.

Chapter 15: Creating an XML Mode Project, activities and tasks for creating VIPP XML mode projects.

Section V: Basic Project Setup

Chapter 16: Setting Document Properties, describes activities and tasks for setting properties associated with a Vitesse project.

Chapter 17: Defining Paper and Print Setup, describes activities and tasks for defining paper and print setup.

Chapter 18: Defining the Logical Page Setup, describes activities and tasks for adding a logical page to a design and setting logical page properties.

Section VI: Working in Database Mode

Chapter 19: Understanding Database Mode Data, provides an overview of the database mode data stream and explains the database data structure in detail. It also gives a detailed overview on printing documents using the ZSORT feature.

Chapter 20: Working with Data in Database Mode, describes activities and tasks for working with data in database mode projects.

Section VII: Working in Line Mode

Chapter 21: Understanding Line Mode Data, provides an overview of the line mode data stream and explains the data stream attributes in detail. It also describes activities and tasks for working with data files and the *Data View* window in Vitesse.

Chapter 22: Working with Data in Line Mode, describes activities and tasks for working with data in line mode projects.

Section VIII: Working in XML Mode

Chapter 23: Understanding XML Mode Data, provides an overview of the XML mode data stream and explains the XML data structure in detail.

Chapter 24: Working with Data in XML Mode, describes activities and tasks for working with data in XML mode projects.

Section IX: Using Data Driven Objects in a Project

Chapter 25: Using Frames, describes activities and tasks for working with frames in Vitesse.

Chapter 26: Using Variables, describes activities and tasks for working with variables in Vitesse.

Chapter 27: Adding Conditions, explains condition objects and conditional logic that you can apply to design and data objects inserted in a project.

Chapter 28: Adding Charts, describes activities and tasks for adding charts and setting chart properties. It also provides information on mapping data to charts using different techniques for various types of data files and project modes.

Chapter 29: Using Actions, describes activities and tasks for working with action objects in Vitesse.

Chapter 30: Adding Barcodes, describes activities and tasks for adding a barcode to a design and setting barcode properties. It also gives a detailed overview on specifying the barcode data using barcode, field, text, and graphic text objects.

Section X: Referencing Resources

Chapter 31: Using Forms in Projects, describes activities and tasks for inserting and referencing forms in Vitesse.

Chapter 32: Using Segments in Projects, describes activities and tasks for inserting and referencing different segment types in Vitesse.

Chapter 33: Referencing Images, describes how to reference image resources in a project design and set image properties of the image references.

Section XI: Preparing the VIPP Project for Production

Chapter 34: Previewing and Testing the Project, describes activities and tasks for previewing and testing projects in Vitesse. The chapter also provides information about performing standard operations in Vitesse, including opening and saving resources.

Chapter 35: Importing and Exporting Projects, provides information regarding the import and export operations of the Vitesse design projects and describes activities and tasks related to them.

Chapter 36: Vitesse Administrative Options, describes activities and tasks for opening Vitesse in administration and user mode. It also provides information about operating Vitesse in the administration mode.

Appendix A: Error and Warning Message Handling, lists and explains error and warning messages that can be encountered while performing specified Vitesse operations, along with explanations for each occurrence. Also included are messages and handling routines for VIPP source code errors.

Glossary, lists the basic Vitesse terms used in this guide.

Conventions

Conventions are the same or similar to Windows conventions, including:

- Display conventions
- Typographic conventions

Display Conventions

Elixir products adhere to Microsoft Windows conventions for using menus, menu commands, dialog boxes, command buttons, icons and mouse. See your Windows manual for more information.

Typographic Conventions

This guide uses the following typographic conventions to identify special information.

Convention	Information Type/Example
< > (angle brackets),	Keystrokes <i>Ex:</i> <Enter> Key combinations are denoted by a plus sign between keys. <i>Ex:</i> <Shift> + <F1>.
Bold type	- Field names within an explanation. <i>Ex:</i> Use rules set by selected Start on option. - Dialog areas referenced in tasks. <i>Ex:</i> Select an item from the List of used Fonts . Click the Options tab. - Tool names referenced in dialogs/windows. <i>Ex:</i> Make sure the Text tool is selected. - Button names with text written on them. <i>Ex:</i> Click OK to save settings. - Any text that you must type. <i>Ex:</i> Enter LPT1 for the following option.
<i>Italic</i> type	- References to other guides. <i>Ex:</i> <i>Elixir Scout Guide</i> . - References to other chapters. <i>Ex:</i> <i>Chapter 10: eDelivery</i> . - Toolbar names. <i>Ex:</i> Click any tool on the <i>Drawing</i> toolbar. - Window names. <i>Ex:</i> <i>Message</i> window displays. - Directory paths. <i>Ex:</i> <i>drive:\Programs Files\Elixir Technologies</i> .
Quotation marks	- References to tasks, activities or concepts. <i>Ex:</i> See “Adding Drawing Objects” section in <i>Chapter 5: Working with Drawing Tools</i> .

Symbols Used in This Guide

Look for the following symbols as you read through this guide:

Symbol	Information Type
	Note This symbol calls your attention to additional information.
	Shortcut This symbol shows a shortcut for a procedure you just learned.
	Tip This symbol highlights a helpful tip.
	Warning This symbol emphasizes an important note and/or warning.
	Migrating Users This symbol emphasizes a note for users migrating from previous versions of Vitesse.

Getting the Answers You Need

Elixir Technologies provides the following resources to help you learn and use Vitesse:

- Customer Support
- Elixir Training
- Help
- Related Guides

Customer Support

If you purchased this product directly from Elixir Technologies, you can contact the Customer Support Center for your region at the number listed below:

North and South America:

- +1 805 641 5900 ext. 3
Monday – Friday, 7:00 am to 4:00 pm Pacific Standard Time; if closed, press 2 to reach Elixir’s extended Support coverage.
- Elx_support@elixir.com

Asia Pacific:

- Asia_support@elixir.com

Europe:

- +44 (0) 207 993 4811
Monday – Friday, 6:00 am to 4:00 pm Central European Time
- +1 805 5900 ext. 3
Monday – Friday, 4:00 pm to 2:00 am Central European Time
- Europe_support@elixir.com

If you have a problem with your Elixir product, contact the Elixir Customer Support Center for your region. An Elixir Product Specialist will answer your call and ask for the following information:

- Your name, organization, telephone number and address.
- Elixir product name and version number.
- A complete description of the problem, including any error messages printed or displayed on your monitor.

Elixir Website Support

You can also obtain product support by accessing the Elixir web page at <http://www.elixir.com> and selecting **Support**.

The Elixir website allows you to:

- Select your preferred support method.
- Identify support centers.
- Access web support.
- Register Elixir products.
- Renew Elixir licenses.
- Learn through the User Group how others are using Elixir products.
- Download the latest patches.
- Contact Elixir Partner Support.
- Ask a question of Elixir Support. To submit a question, select Questions> Submit a Question here. Please complete the question form.
- Find answers in the list of Frequently Asked Questions.
- Access the Elixir Knowledge Base.

Elixir Training

Elixir Technologies offers training for its full-range of Windows-based products. For more information, contact:

Elixir Learning
(805) 641-5900 ext.4

Using Help

Elixir software uses the Microsoft Windows Help program to provide help for all product functions.

What's This? Help

Selecting a dialog control such as an input field, radio button or drop-down list, and then pressing Shift + F1 displays a pop-up window containing information about that control. You can also get this information by right-clicking the control and selecting What's This? from the pop-up window or clicking  on the dialog title bar if available.

Help Pop-ups and Jumps

The help can be opened using Help>Contents from the application menu bar. Within the Help topic text, words or phrases in blue are drop-down hotspots and underlined words or phrases in blue are jumps or hyperlinks. Clicking a hotspot displays images and links directly below the hotspot within the same topic. Clicking a jump displays an associated Help topic within the same window.

Related Guides

If you require further information about a specific topic or wish to obtain product background information, reference the *Vitesse Getting Started Guide* which is part of the Vitesse documentation package.

User Notes:

Contents

Welcome	I
Who Should Use this Guide?	I
About This Guide	I
Conventions	IV
Display Conventions	IV
Typographic Conventions	V
Symbols Used in This Guide	VI
Getting the Answers You Need	VII
Customer Support	VII
Elixir Website Support	VIII
Elixir Training	VIII
Using Help	VIII
Related Guides	IX
Chapter 1: Introduction	37
Vitesse Overview	37
Before You Start	39
Creating a Preliminary Project	39
Defining and Locating Test Data	39
Understanding the VIPP Modes	40
What is Native Mode?	40
What is Line Mode?	40
What is Database Mode?	41
What is XML Mode?	42
What's New in Vitesse 2.00?	43
Form and Segment Behavior	43
Frame Behavior	43
Restricted Mode Operation	43
Predefined Media and Fields	44
Batch Conversion of Documents	44
Section I: Vitesse Basics	
Chapter 2: Understanding the Vitesse Workspace	47
Getting Familiar with the Workspace	47
Using the Design Area	48
Design View	48
Source View	49
Using the Project Tree View Window	49

Media-----	50
MasterDoc -----	51
Resources -----	51
Using the Property View Window -----	51
Using the Data View Window -----	52
Using the Message View Window -----	53
Clearing Messages-----	54
Saving Messages-----	54
Setting the Default View Options -----	54
Arranging the Workspace Windows -----	55
Using Workspace Tools -----	56
Using the Menu Bar -----	56
Using the Toolbar-----	57
Viewing the Status Bar -----	57
Placing Objects in the Design Area -----	58
Using the Ruler -----	58
Displaying and Hiding the Ruler -----	58
Setting the Default Ruler Option -----	58
Changing the Unit of Measurement-----	58
Using the Grid-----	59
Displaying and Hiding the Grid -----	60
Setting the Default Grid Option -----	60
Automatically Aligning Objects to the Grid -----	60
Setting the Default Snap to Grids Option -----	61
Setting the Grid Color-----	62
Setting the Grid Style -----	63
Setting the Grid Spacing-----	63
Using Guides -----	64
Turning the Guides On and Off-----	64
Setting the Default Guide Option-----	65
Moving Guides to the Design Area -----	65
Removing Guides from the Design Area -----	66
Automatically Aligning Objects to the Guides -----	66
Setting the Default Snap to Guides Option-----	66
Setting the Guide Color-----	66
Setting the Guide Style-----	67
Using the Crosshair-----	68
Displaying and Hiding the Crosshair -----	68
Setting the Default Crosshair Option -----	68
Moving Objects using the Keyboard-----	69
Setting the Nudge Options -----	69

Zooming within the Design Area -----	70
Specifying Custom Zoom Value -----	71
Moving the Design Area View -----	71
Using the Full Screen View -----	72
Refreshing the Project -----	72
Navigating the Workspace -----	72
Choosing Commands -----	73

Chapter 3: Understanding the Vitesse Directory Structure 75

Understanding Resource Scope -----	75
Local Resources -----	75
Shared Resources -----	75
Global Resources -----	75
Understanding the Vitesse Directory Structure -----	76
Vitesse Folder -----	77
ClientsFolder -----	77
Client Folder -----	77
Project Folder -----	77
Data Files -----	78
Config -----	78
xgf -----	78
FormLib -----	78
ImgLib -----	78
SegLib -----	78
MisLib -----	78
Setting up the Working Folder -----	79
Using Scope Specific View in the Project Tree -----	79
Opening Vitesse Projects in Scope Specific View -----	81
Setting Default Folder Preferences -----	81
Adding a Directory -----	82
Deleting a Directory -----	83
Rearranging the Directory List -----	83

Section II: Using VIPP Resources

Chapter 4: Working with Forms 87

Using a Form -----	87
Creating a New Form -----	87
Setting Form Properties -----	89
Setting the Form Size -----	89
Setting the Form Orientation -----	90
Specifying Form Details -----	91

Caching Resources -----	93
Caching a Form -----	93
Chapter 5: Working with Segments	95
Using a Segment -----	95
Creating a New Segment -----	96
Setting Segment Properties -----	97
Setting the Segment Size -----	97
Setting the Segment Origin -----	98
Clipping and Caching a Segment -----	99
Specifying Segment Details -----	100
Dynamic Behavior of Segments -----	101
Working with Postscript Segments -----	101
Viewing Postscript Segment as a Faded Template -----	102
Setting Transparency of White Areas in a PostScript Segment -----	104
Chapter 6: Working with Fonts	105
Font Types Supported in Vitesse -----	105
Managing Font Styles in Vitesse -----	106
Creating a New Font Style -----	106
Editing a Font Style -----	109
Using Legacy Fonts -----	109
Mapping a Legacy Font to a VIPP Font -----	110
Viewing Mapped Fonts in Vitesse -----	111
Mapping VIPP Fonts to True Type or PostScript Fonts in Vitesse -----	112
Setting a Default Font -----	114
Setting Font List Viewing Options -----	115
Chapter 7: Working with Images	117
Adding an Image -----	117
Setting Image Properties -----	118
Replacing an Image -----	118
Previewing the Image as a Faded Template -----	119
Setting Transparency of White Areas in an Image -----	120
Caching an Image -----	121
Previewing Images in Low Resolution -----	121

Section III: Vitesse Project Tools

Chapter 8: Using the Drawing Tools 125

Drawing Objects-----	125
Using the Line Tool-----	127
Drawing a Line -----	127
Positioning a Line -----	127
Repositioning a Line -----	129
Using the Box Tool -----	131
Drawing a Box-----	131
Positioning a Box-----	131
Creating a Rounded Box -----	132
Setting Corner Radius -----	132
Splitting a Box-----	133
Repeating a Box -----	134
Using the Path Tool -----	136
Drawing a Path -----	136
Positioning a Path -----	136
Setting Path Nodes -----	137
Adding a Path Node-----	138
Deleting a Path Node -----	139
Positioning a Path Node -----	139
Using the Polygon Tool -----	141
Drawing a Polygon-----	141
Positioning a Polygon-----	141
Selecting a Polygon Shape -----	142
Using the Static Table Tool-----	144
Drawing a Static Table -----	144
Positioning a Static Table -----	144
Setting the Number of Rows and Columns -----	145
Setting the Table Cell Spacing -----	146
Setting Common Object Properties -----	148
Setting the Object Size-----	148
Setting the Box Size-----	148
Setting the Path Size -----	149
Setting the Polygon Size-----	149
Setting the Static Table Size -----	150
Setting Object Line Width -----	150
Setting the Line Width-----	151
Setting the Box Line Width-----	152
Setting the Path Line Width-----	153

Setting the Polygon Line Width -----	154
Setting the Static Table Line Width-----	155
Setting the Object Line Style-----	156
Setting the Line Style -----	157
Setting the Box Line Style-----	158
Setting the Path Line Style -----	159
Setting the Polygon Line Style-----	160
Setting the Static Table Line Style -----	161
Assigning Properties to Multiple Selected Objects -----	161
GEP Keys-----	162
Basic Operations-----	162
Selecting an Object-----	162
Select Design Area Objects -----	163
Select Project Tree Nodes -----	163
Copying an Object-----	164
Cutting an Object -----	164
Pasting an Object -----	164
Deleting an Object-----	165
Resizing an Object-----	165
Moving an Object-----	165
Renaming an Object-----	166
Sequencing Project Tree Nodes -----	166
Aligning Objects-----	166
Align Left-----	167
Align Right -----	167
Align Bottom-----	167
Align Top-----	167
Center Horizontally-----	167
Center Vertically-----	167
Evenly Space Horizontally -----	168
Evenly Space Vertically -----	168
Make Same Width-----	168
Make Same Height -----	169
Arranging Objects -----	169
Bring to Front -----	170
Send to Back -----	170
Bring Forward-----	170
Send Backward-----	170
Undoing and Repeating Actions-----	172

Chapter 9: Using the Text Tools

173

Using Text Tools -----	173
Inserting a Text Object -----	173
Inserting Text by Clicking and Typing -----	174
Creating a Text Box -----	174
Inserting a Graphic Text Object -----	174
Editing Text -----	175
Positioning Text -----	175
Setting the Text Size -----	177
Using the Merge Data Feature -----	178
Creating an XML file with Pre-defined Variables -----	178
Loading an XML file with Pre-defined Variables -----	180
Merging Data with Static Text -----	181
Autosizing Text Objects -----	183
Formatting Text Objects -----	183
Setting Text Background Attributes -----	183
Setting Line Spacing -----	184
Setting Paragraph Spacing -----	185
Aligning the Text -----	185
Indenting Text -----	186
Setting Text Tabs -----	187
Setting Tab Stops -----	187
Adding Tab Stops to a List -----	188
Deleting Tab Stops from a List -----	188
Modifying a Tab Stop in a List -----	188
Selecting the Tab type -----	188
Setting the Default Tab Value and Alignment for the Text -----	189
Formatting Graphic Text Objects -----	189
Setting Graphic Text Rotation -----	190
Using Rich Text from External Projects -----	191
Importing Rich Text Format Text -----	191
Exporting Text to a Rich Text Format Text File -----	192
Joining Text Objects -----	193
Using the Spell Checker -----	193
Specifying Dictionary Files for Spell Checking -----	194
Adding a Dictionary File -----	194
Deleting a Dictionary File -----	195
Rearranging the Dictionary Files List -----	195
Setting Default Text Attributes -----	195
Setting the Default Text Justification -----	195

Setting the Default Paragraph Spacing -----	196
Setting the Default Tab Value and Alignment-----	196
Setting the Default Text Indentation -----	197
Setting the Spell Checker Depth-----	197
Setting the Data Switch -----	197
Finding Text-----	198
Replacing Text -----	199
Chapter 10: Applying Transformations	201
Transforming Strings-----	201
Formatting a Number String -----	201
Adding a New Format Mask -----	202
Extracting a Fixed Length Sub-String from Text -----	203
Extracting a Variable Length Sub-String from Text -----	204
Stripping Characters and Blanks -----	205
Formatting a Time String -----	206
Changing String Case -----	207
Concatenating Strings-----	208
Formatting Strings as Barcodes-----	209
Chapter 11: Working with Colors	211
Using Colors -----	211
Selecting a Foreground Color-----	213
Selecting a Background Color -----	214
Working with Color Palettes -----	215
Loading a Palette -----	215
Adding a New Color to the Color Palette -----	215
Adding a Color Using the Color Spectrum-----	215
Adding a Color Using the Color Model -----	217
Editing a Color -----	219
Deleting a Color -----	220
Saving a Palette -----	221
Using Patterns -----	221
Applying Line Patterns -----	222
Applying Fill Patterns-----	223
Setting Default Color and Pattern Attributes -----	224
Setting Default Foreground Color-----	224
Setting Default Background Color -----	224
Setting Default Line Patterns -----	224
Setting Default Fill Patterns -----	225

Section IV: Creating a VIPP Project

Chapter 12: Creating a Native Mode Project 229

Using the Native Mode -----	229
Creating a New Native Mode Project-----	229
Selecting the Paper Size -----	230
Selecting the Paper Orientation-----	231

Chapter 13: Creating a Database Mode Project 233

Using the Database Mode -----	233
Creating a New Database Mode Project -----	233
Selecting the Paper Size -----	234
Selecting the Paper Orientation-----	235
What is a Sample Data File?-----	236
Selecting a Sample Data File -----	237
Specifying a Data File Delimiter -----	238

Chapter 14: Creating a Line Mode Project 239

Using the Line Mode -----	239
Creating a New Line Mode Project -----	239
Setting up a Data Grid -----	240
Selecting the Paper Size -----	242
Selecting the Paper Orientation-----	243
Setting the Page Margins-----	244
What is a Sample Data File?-----	245
Selecting a Sample Data File -----	245
Defining the Data-----	246
Specifying the Record Prefixes-----	247
Specifying the Field Type-----	248
Specifying Fixed Length Data-----	248
Specifying Delimited Data-----	248
Specifying a Delimiter-----	249
Working with Page Delimiters -----	250
Specifying Page Delimiters -----	251
Matching the Page Delimiter -----	252
Printing the Page Delimiter Line -----	253
Looping the Page Delimiter Line -----	254
Setting Document Flow Options -----	256
Specifying Printer Control Characters in Data -----	257
Using the XML Mode -----	259
Creating a New XML Mode Project-----	259
Selecting the Paper Size -----	260

Selecting the Paper Orientation-----	261
Section V: Basic Project Setup	
Chapter 16: Setting Document Properties	265
Specifying Project Details -----	265
Setting the Project Information -----	265
Maintaining Change History -----	266
Setting Regional Preferences-----	266
Applying Regional Preferences to Numeric Formats-----	266
Specifying Default Regional Preferences -----	269
Working with Regional Languages -----	271
Setting Character Set Encoding Options -----	271
Printing Multiple Copies of a Document-----	271
Setting the Document Printing Options-----	272
Chapter 17: Defining Paper and Print Setup	275
Defining the Media Properties -----	275
Adding a Paper-----	275
Adding a Predefined Paper -----	276
Setting Paper Properties -----	278
Changing the Paper Size-----	278
Changing the Paper Orientation -----	279
Using Printing Options-----	280
Setting Printing Options -----	280
Selecting the Paper Type -----	281
Setting the Paper Weight -----	281
Setting the Paper Color -----	282
Using Document Structuring Conventions-----	282
Printing a Constant Form on the Back of the Paper-----	282
Printing Several Pages on One Paper (N-UP) -----	282
Setting Default Paper Properties-----	286
Setting the Default Paper Size -----	286
Setting Default Paper Orientation -----	286
Setting Default Media Settings -----	287
Chapter 18: Defining the Logical Page Setup	289
About the Logical Page -----	289
Adding a Page-----	290
Setting the Page Properties -----	290
Setting the Page Name -----	290
Selecting a Paper-----	291

Selecting the Paper Side for Printing -----	291
Selecting the Page Orientation-----	292
Setting the Page Margins-----	293
Adding a Page Border -----	294
Creating a Zebra Effect-----	295
Setting up a Data Grid -----	296
Adding the Page Numbers -----	297
Formatting the Page Numbers -----	297
Setting the Page Number Color -----	298
Setting the Page Number Font-----	298
Specifying the Starting Page Number -----	299
Positioning the Page Numbers-----	299
Setting the Default Page Properties -----	300
Setting the Default Margins-----	300
Setting the Default Data Grid-----	300

Section VI: Working in Database Mode

Chapter 19: Understanding Database Mode Data 303

Understanding Database Mode Data -----	303
Inserting a Data File-----	304
Specifying a Delimiter-----	305
Using the Data View Window for Database Mode Projects--	305
Auto-sizing Data Columns -----	306
Understanding the Database Data File Structure-----	307
Creating ZSort -----	308
Printing Documents Using ZSort-----	309
Example 1: Using ZSORT for Simplex Printing-----	310
Example 2: Using ZSORT with Insert blank pages option	311
Example 3: Using ZSORT with Shorten last stack option-	312
Example 4: Using ZSORT with Record Processing set to	
Twice-----	313
Example 5: Using ZSORT for Duplex Printing-----	314
Trimming Fields-----	315

Chapter 20: Working with Data in Database Mode 317

Mapping Data-----	317
Understanding Database Processing Order -----	318
Mapping Fields -----	319
Inserting a Field -----	320
Using Normal Fields-----	320
Specifying the Normal Field Data -----	321

Using the Data File -----	321
Using Variables -----	322
Using Constant Data -----	324
Aligning the Normal Field Contents -----	324
Defining the Cell Margins -----	325
Setting Field Justification -----	325
Using Image Fields -----	327
Specifying an Image Resource -----	327
Using the Data File -----	327
Using Variables -----	329
Using an Existing Project Resource as Field Data -----	329
Aligning the Image Field -----	330
Using Segment Fields -----	332
Specifying a Segment Resource -----	332
Using the Data File -----	332
Using Variables -----	333
Using an Existing Project Resource as Field Data -----	334
Using the Dynamic Sizing Behavior in a Segment Field ---	334
Using Static Text Fields -----	336
Formatting Static Text Field Data -----	336
Specifying Static Text Field Data -----	337
Setting Common Field Properties -----	337
Positioning a Field -----	337
Setting Field Absolute Position -----	338
Setting Field Relative Position -----	339
Setting the Field Size -----	341
Adjusting the Field Size to its Contents -----	343
Setting Field Rotation -----	344
Scaling Image and Segment Fields -----	345

Section VII: Working in Line Mode

Chapter 21: Understanding Line Mode Data 349

Understanding Line Mode Data -----	349
Listing Mode -----	350
Record Processing Entry (RPE) Mode -----	350
Defining RPE Fields Vertically -----	350
Defining RPE Fields Horizontally -----	350
Specifying Fixed Field Data -----	351
Specifying a Delimiter for Delimited Data -----	352
Working with Page Delimiters -----	353
Specifying a Page Delimiter -----	353

Specifying the Normal Field Data-----	401
Using the Data File -----	401
Using Fixed Field Data -----	402
Using the Field Number -----	403
Using Variables -----	403
Using Constant Data -----	405
Aligning the Normal Field Contents -----	405
Creating Dynamic Tables using Field Borders -----	407
Specifying Cell Margins -----	408
Specifying Cell Margin Line Width-----	409
Specifying Cell Margin Line Style-----	410
Specifying OutLine Color and Pattern for Cell Margins ---	412
Specifying Fill Color and Pattern for Cell Margins-----	413
Setting Field Justification-----	414
Using Image Fields-----	416
Specifying an Image Resource-----	416
Using the Data File -----	416
Using Variables -----	417
Using an Existing Project Resource as Field Data -----	418
Aligning the Image Field -----	418
Using Segment Fields-----	421
Specifying a Segment Resource -----	421
Using the Data File -----	421
Using Variables -----	422
Using an Existing Project Resource as Field Data -----	423
Using Static Text Fields -----	423
Formatting Static Text Field Data -----	424
Setting Common Field Properties -----	425
Positioning a Field-----	425
Setting Field Absolute Position-----	425
Setting Field Relative Position-----	426
Setting the Field Size-----	429
Adjusting the Field Size to its Contents -----	430
Setting Spacing between Fields-----	431
Setting Field Rotation-----	433
Scaling Image and Segment Fields-----	433
What is a Data Group?-----	434
Inserting a Data Group in Line Mode Projects -----	436

Section VIII: Working in XML Mode

Chapter 23: Understanding XML Mode Data 441

Understanding XML Mode Data-----	441
Inserting a Data File -----	441
Using the Data View Window-----	442
Using the Data Window for XML Mode Projects -----	443
XML Data Tree-----	444
Viewing Multiple Tag Occurrences -----	445
XML Data File Structure-----	446
Tags -----	446
Attributes -----	447
Values-----	447

Chapter 24: Working with Data in XML Mode 449

Mapping XML Data to the Project Design -----	449
Understanding XML Data Processing Order -----	451
Associating a Data Tag to a Page -----	451
Mapping Records-----	453
Inserting a Record in XML Mode-----	455
Mapping a Record to a Data Tag-----	456
Using the Record Property Sheet-----	456
Using Drag and Drop -----	457
Printing a Record on a New Page -----	459
Record Processing-----	461
Mapping Fields -----	461
Inserting a Field in XML Mode -----	461
Mapping a Field to a Data Tag Value -----	462
Field Processing -----	464
Using Normal Fields-----	465
Specifying the Normal Field Data -----	466
Using the Data File-----	466
Using Variables-----	466
Using Constant Data -----	467
Aligning the Normal Field Contents-----	467
Creating Dynamic Tables using Field Borders -----	468
Setting Field Justification -----	469
Using Image Fields-----	471
Specifying an Image Resource -----	471
Using the Data File-----	472
Using Variables-----	473

Using an Existing Project Resource as Field Data -----	473
Aligning an Image Field -----	474
Using Segment Fields -----	476
Specifying a Segment Resource -----	476
Using the Data File -----	476
Using Variables -----	478
Using an Existing Project Resource as Field Data -----	478
Using the Dynamic Size Behavior in a Segment Field ----	479
Using Static Text Fields -----	481
Formatting Static Text Field Data -----	481
Specifying Field Data -----	482
Merging XML Data with Static Text -----	482
Setting Common Field Properties -----	483
Positioning a Field -----	483
Setting Field Absolute Position -----	484
Setting Field Relative Position -----	485
Setting the Field Size -----	488
Adjusting the Field Size to its Contents -----	489
Setting Spacing between Fields -----	489
Setting Field Rotation -----	490
Scaling Image and Segment Fields -----	491

Section IX: Using Data Driven Objects in a Project

Chapter 25: Using Frames 495

Understanding Frames -----	495
Understanding Relative and Static Objects -----	496
Working with Frames in Vitesse -----	497
Understanding Segment Overflow Behavior -----	498
Creating a Frame -----	498
Creating a Frame Using the Property Sheet -----	499
Using Borders to Preview Frames -----	500
Specifying Frame Border Width -----	500
Specifying Frame Border Style -----	501
Changing Frame Placement and Size -----	502
Relatively Positioning Objects in Frames -----	505
Using Text or Field Objects in Frames -----	506
Using Conditions in Frames -----	508
Enabling Segment Overflow -----	508
Enabling Frame Overflow in Segments -----	509
Enabling Frame Overflow in Objects referencing the	

Segment-----	509
Enabling Frame Overflow in SegRef-----	510
Enabling Frame Overflow in Segment Fields-----	510
Using SegRef in Frames-----	510
Using Frame Numbers-----	511
Working with Frames in XML Mode-----	512
Chapter 26: Using Variables	513
What is a Variable?-----	513
Understanding Variable Types-----	513
Creating a New Variable-----	514
Creating a String Variable-----	514
Example: Using a string variable as a field value-----	515
Creating a Number Variable-----	516
Example: Using a number variable as a barcode value-----	516
Creating an Array Variable-----	517
Example: Using an array variable for creating DDG Charts-----	519
Creating a Boolean Variable-----	520
Working with System Variables-----	520
Example: Using system variable BACK to switch forms for a sheet front and back-----	524
Using Persistent Variables-----	525
Using Variables for Referencing Forms-----	526
Using Variables for Referencing Segments-----	528
Using Variables for Referencing Images-----	529
Using Variables in Forms and Segments-----	530
Chapter 27: Adding Conditions	533
What are Conditions?-----	533
Defining a Condition Statement-----	534
Operands-----	534
Operators-----	534
Comparison Operators-----	534
Logical Operators-----	535
Working with Page Level Conditions-----	536
Working with Record Level Conditions-----	537
Adding Conditions to the Project Tree-----	537
Inserting a Condition-----	537
Specifying a Condition Test-----	538
Specifying Detailed Condition Tests using Logical Operators-----	539

Using the AND and OR Logical Operators -----	539
Using the NOT Logical Operator-----	540
Using the ElseIf Condition -----	540
Deleting a Condition -----	541
Using Data in Conditions-----	542
Mapping Data to a Condition in Line Mode Project -----	542
Mapping Data at the Page Level Condition – Non-Prefixed Data-----	542
Mapping Data at the Record Level Condition – Non-Prefixed Data-----	544
Mapping Data to a Condition in Line Mode Project – Prefixed Data-----	546
Mapping Data to a Condition in Database Mode-----	548
Mapping Data to a Condition in XML Mode -----	550
Using Variables in a Condition -----	552
Using System Variables in a Condition -----	554
Using Constants in a Condition-----	556
Dynamically Managing Form and Segment Content using Conditions -----	557
Forms and Segments Used as Resources in Projects -----	557
Segments and Forms Used as Stand-Alone Projects-----	558

Chapter 28: Adding Charts 561

Understanding Charts -----	561
Working with Page Level Charts -----	562
Working with Record Level Charts -----	562
Inserting a Chart -----	562
Setting Chart Properties-----	563
Naming a Chart -----	563
Selecting Chart Type-----	563
Positioning a Chart -----	564
Setting Chart Absolute Position -----	564
Setting Chart Relative Position -----	565
Setting the Chart Size -----	568
Setting the Chart Spacing-----	568
Applying the 3 D Effect -----	571
Setting the Chart Background Color -----	572
Setting a Color Table for the Chart-----	573

Setting a Chart Legend -----	575
Setting the Chart Border-----	576
Setting the Chart Border Width -----	576
Setting the Chart Border Color -----	576
Setting the Chart Border Pattern -----	576
Setting the Chart Text and Scale Color-----	576
Setting Chart Fit Sizing Option-----	577
Setting the Space Between Bars-----	578
Printing Data Values -----	578
Setting the Value Format-----	579
Merging Values in a Chart -----	579
Maintaining Chart Ratio-----	580
Setting Range for the Y-axis-----	580
Setting the Pie Chart Properties -----	581
Selecting the Pie Chart Display Type -----	581
Customizing the Pie Chart Labels-----	582
Setting the Label Dash Color -----	582
Setting the Label Dash Width -----	582
Setting the Label Offset -----	582
Setting the Slice Burst Value -----	582
Setting the Pie Chart Radius -----	583
Mapping Data to Charts -----	584
Using DDG Index to Map Fields as Chart Data-----	584
Specifying Fields as Labels and Values -----	584
Storing Field Label and Value Pairs -----	585
Using Predefined Variables as Chart Data-----	587
Using Data from the Data File (Manual Data) -----	588
Manually Mapping Field Data-----	588
Mapping Data to a Page Level Chart – Non-Prefixed Line mode Data -----	589
Mapping Data to a Record Level Chart – Prefixed Line mode Data -----	591
Mapping Data to a Chart - Database Mode Data-----	593
Mapping Data to a Chart – XML Mode Data -----	595
Manually Mapping Variables-----	597
Manually Mapping Constants -----	598

Chapter 29: Using Actions

599

Understanding Actions-----	599
Inserting an Action -----	600
Setting Action Properties -----	601
Creating a Booklet-----	601
Setting Document Control Printing Options -----	603
Selecting the Page Mode -----	603
Selecting the Duplex Mode -----	604
Selecting the Orientation -----	604
Setting Document Set Options-----	605
Specifying Finishing Operations -----	606
Selecting the Bind Option -----	606
Selecting the Jog and Offset Option -----	606
Selecting the Staple Option -----	607
Selecting the Collate Option -----	607
Setting Page Control Options -----	608
Inserting Inline VIPP Code-----	612
Modifying Variables Using Actions -----	612
Setting a Variable Value-----	613
Creating a Variable at Runtime-----	615
Incrementing a Variable -----	618
Decrementing a Variable -----	618
Adding a Variable -----	619
Adding a Value to the Variable-----	619
Adding an Existing Variable to the Variable -----	620
Subtracting a Variable -----	621
Subtracting a Value from the Variable -----	621
Subtracting an Existing Variable from the Variable -----	621
Assigning Field Value to a Variable -----	622
Assigning Value in Line Mode -----	622
Choosing a Data Line – Using Conditions -----	623
Choosing a Data Line – Using Line Number -----	626
Choosing a Data Field – Using Fixed Field Data -----	626
Choosing a Data Field – Using Delimited Data -----	627
Assigning Value in Database Mode-----	628
Assigning Value in XML Mode -----	630
Setting Variables as PDF Bookmarks-----	631
Setting Bookmark Properties-----	631

Chapter 30: Adding Barcodes **635**

- Understanding Barcodes ----- 635
 - Associating the Barcode Fonts with the Barcode Types --- 638
 - Adding a Barcode ----- 639
- Setting Barcode Properties ----- 639
 - Naming a Barcode----- 639
 - Positioning a Barcode----- 639
 - Setting the Barcode Absolute Position ----- 640
 - Setting the Barcode Relative Position ----- 641
 - Setting the Barcode Size----- 643
 - Selecting the Barcode Type----- 644
- Specifying the Barcode Data ----- 644
 - Applying Barcodes Using the Barcode Object ----- 645
 - Using Variables as Barcode Data ----- 645
 - Using Constants as Barcode Data ----- 646
 - Using Merge Data ----- 647
 - Applying Barcodes Using Field, Text, and Graphic Text Objects ----- 647
 - Example: Using Field Data as Barcode Value ----- 648
 - Example: Using Static Text as Barcode Value ----- 649

Section X: Referencing Resources

Chapter 31: Using Forms in Projects **653**

- Using Forms in Projects ----- 653
 - Embedded Forms ----- 653
 - Inline Forms ----- 653
 - Linked Forms ----- 653
 - Referenced Forms ----- 654
- Inserting Forms ----- 654
 - Inserting a Form as a Project Resource ----- 654
 - Inserting a Form in the Project Design ----- 655
- Setting Properties of a Form Reference ----- 657
 - Changing the Referenced Form ----- 657
 - Using Variables to Reference a Form ----- 657
- Setting Properties of an Inline Form ----- 658

Chapter 32: Using Segments in Projects **659**

Understanding Segment Types -----	659
Embedded Segments -----	659
Inline Segments -----	659
Linked Segments -----	659
Referenced Segments -----	660
Begin Page Segments -----	660
End Page Segments -----	660
Repeat Segments -----	660
Inserting Segments -----	661
Inserting a Segment as a Project Resource -----	661
Inserting a Segment in the Project Design -----	662
Setting Properties of a Segment Reference -----	664
Positioning a Referenced Segment -----	664
Aligning a Segment Reference -----	665
Scaling a Referenced Segment -----	667
Rotating a Referenced Segment -----	668
Changing the Referenced Segment -----	668
Using Variables to Reference a Segment -----	669
Dynamically Sizing Segments -----	670
Using Repeat Segments -----	672
Working with Segments in an XML Mode Project -----	673
Using Tag Start Segments -----	673
Using a Tag Orphan Segment -----	674
Using a Tag End Segment -----	674
Specifying Vertical Spacing between Segments -----	675

Chapter 33: Referencing Images **677**

Referencing Images -----	677
Inserting an Image in a Project Design -----	677
Setting Properties of an Image Reference -----	678
Positioning an Image Reference -----	679
Resizing an Image Reference -----	681
Rotating an Image Reference -----	682
Scaling an Image Reference -----	683
Aligning an Image Reference -----	684
Changing the Referenced Image -----	685
Using Variables to Reference an Image -----	685

Section XI: Preparing the VIPP Project for Production

Chapter 34: Previewing and Testing the Project 689

Opening Existing Resources -----	689
Opening Recently Used Resources -----	691
Previewing a Project -----	691
Proof-Printing a Project -----	692
Print Previewing Multi-Up Projects -----	693
Setting Memory Size for Optimization for View -----	696
Working with Source Code -----	697
Viewing the Source Code -----	698
Searching in the Source Code -----	699
Finding Text in the Source Code -----	699
Replacing Text in the Source Code -----	699
Editing the Source Code -----	700
Compiling VIPP Code -----	700
Reverting to the Last Compiled State -----	701
Exporting Source Code -----	701
Setting Default Source Code Options -----	702
Setting Colors for Source Code Objects -----	702
Setting Source View Background Color -----	703
Setting a Default Font for the Source Code -----	704
Saving Resources -----	704
Saving a Project -----	705
Saving a Resource -----	706
Saving Projects in Vitesse Zip Format -----	706

Chapter 35: Importing and Exporting Projects 707

Importing Projects -----	707
Setting Import Preferences -----	707
Setting Text Merge Preferences -----	708
Using Comments in the Imported Project -----	708
Using Preferences from a DEF File -----	709
Importing a Project -----	709
Exporting Projects -----	712
Setting the Export Preferences -----	712
Setting the Default Measurement Unit -----	712
Setting a Comment Level for the Exported Project -----	713
Using Elixir Comments -----	713
Converting TIFF Files -----	713
Setting the Buffer Size -----	714

Setting Maximum Number of Forms -----	714
Exporting a Vitesse Project-----	715
Packaging Other Files with a Project -----	716
Exporting a Project -----	717
Managing Unmapped Fonts during Export -----	718
Converting True Type Fonts to the PostScript Format -----	718
Packaging the PostScript Fonts into the VPC-----	718
Chapter 36: Vitesse Administrative Options	719
Vitesse User Types -----	719
Administrator-----	719
User-----	719
Opening Vitesse in Administration Mode-----	720
Opening a File in Vitesse Administration Mode -----	720
Setting Administrator Preferences-----	721
Setting up the Working Folder-----	721
Limiting Vitesse Functions-----	722
Setting Default Vitesse Mode-----	723
Setting Permissions for the Export Folder -----	724
Setting up Predefined Media-----	725
Limiting Resource Insertion-----	727
Setting Resource Loading Options -----	728
Limiting Resource Deletion -----	729
Enabling Field Insertion in Forms and Segments -----	730
Setting VIPP Projects Import Options -----	732
Opening Vitesse in User Mode -----	733
Opening a File in Vitesse Administration Mode -----	733
Appendix A: Error and Warning Message Handling	735
Source Code Error Handling -----	735
Error Messages-----	735
Warning Messages-----	735
Informational Messages -----	735
Viewing Source Code for Entire Application or External Resources -----	736
Handling Compilation Errors -----	736
Reverting to the Last Valid Compiled State -----	737
Error Messages-----	737
Import Error Messages-----	737
Warning Messages -----	739
Import Warning Messages-----	739

Export Warning Messages -----	740
Text Warning Messages -----	745
Glossary	749
Index	753

User Notes:

Chapter 1: Introduction

This chapter provides an overview of the Vitesse functions and features that facilitate creating projects and resources for VIPP printing. It also includes information on the following:

- Vitesse: From Document Design to Printed Page
- Creating a Preliminary Design
- Defining and Locating Test Data
- Understanding the VIPP modes
- What's New in Vitesse 2.00

Vitesse Overview

Vitesse is a Windows-based, visual design environment for creating, editing, and formatting variable data projects and resources for Black and White, Highlight, and Full color VIPP (Variable Data Intelligent PostScript Printware) printing. VIPP is an open language Xerox print stream, which is based on PostScript and works by extending PostScript programming language, enabling you to efficiently create Xerox printer applications. Vitesse simplifies the document designing process as it provides a Windows-based visual design environment.

With Vitesse, you can use XML, database resident, and legacy line data map the data in a project using either the drag and drop operation, or by typing in data specifications. You can also design forms and segments using objects such as lines, boxes, and images; create data-driven graphics, dynamic texts flows, and apply conditional logic to your design.

You can import and edit VIPP projects and project resources visually or by editing the source code. Vitesse provides a VIPP source view for viewing, editing, and compiling code changes in a single, integrated environment for newly designed or imported projects.

Existing hand-coded VIPP applications; Xerox native print resources such as forms (FRM and FSL), fonts (FNT), and image files (IMG and LGO) for migration from LCDS and Metacode environments; and VIPP resources (FRM, SEG, JDT, XJT, and DBM) can also be imported, edited, and saved using Vitesse.

Vitesse fully supports VIPP Native (NM), Database (DBM), Line data (JDT), and XML (XJT) project modes, and provides for creation and modification of VIPP resources such as forms and segments.

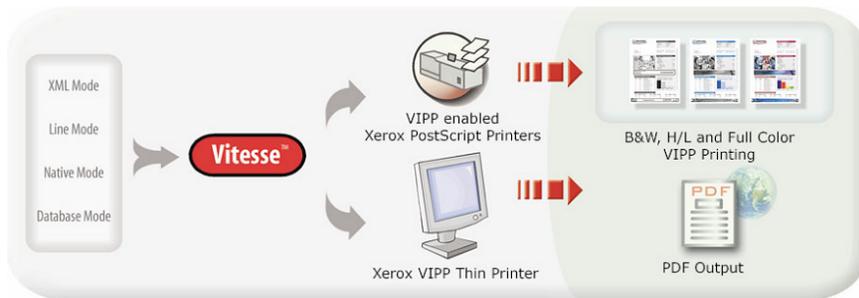
Depending on the project mode, a VIPP project designed in Vitesse can be exported as a VIPP Project Container (VPC), as well as DBM, JDT, NM, or XJT.



VPC enables you to archive all project resources into one file. Later, you can export a VPC to any location. Once exported, the VPC automatically allocates all resources to their respective VIPP directory. Contrarily, exporting projects in JDT, XJT, or DBM format does not pack the resources, indicating that you need to track each resource file individually.

VIPP uses Dynamic Document Composition (DDC) to format and print line printer and database data using PostScript features at production speeds. DDC is based on real time composition of the document at the printer or imaging device. In keeping with Dynamic Document Composition (DDC), VIPP projects and resources from Vitesse are typically transported over the network to the PostScript print controller, and then merged with the data file by the Xerox VIPP compiler.

Additionally, the Vitesse VIPP project code can be sent to a Xerox VIPP Thin Printer for distillation and conversion to PDF. The VIPP Thin Printer can be resident on the PostScript print controller or any server or client machine running Windows NT, Windows 2000, or later versions. The image below depicts how projects designed in Vitesse are exported to destination printers for generating the required output:



Vitesse also extends Xerox PDF workflow functionality by providing the following features:

- Drag and drop workflow for creating PDF bookmarks.
- PDF bookmark creation in all three VIPP modes.
- Creation of single-level bookmarks or sub-bookmarks for hierarchical bookmark levels within a PDF.
- Bookmark-driven PDF splitting.

Before You Start

Prior to creating your projects in Vitesse, you need to familiarize yourself with prerequisite activities that are required for creating a design. Some basic activities include: defining the input; setting the project defaults; specifying document attributes; identifying the content, layout, and structure of the design and associated data; and identifying scenarios for subsequent design testing, etc. The sections below briefly explain the process of creating and implementing projects in Vitesse.

Subsequent chapters provide detailed descriptions of the activities and tasks involved when using Vitesse for designing applications.

Creating a Preliminary Project

You should prepare a preliminary structure of the required output before building the actual design. You can create a pencil drawing depicting the physical layout of each page or identify a proximate archived project to subsequently import and serve as the basis for a new project. In either case, your first step should establish how each document page will look. This includes object placement, arrangement of images/logos and borders, layout of headers and footers, font selection, and simplex vs. duplex printing considerations, etc.

Once you have settled on a preliminary design, you should next specify the design preferences and document properties, as appropriate for subsequent development and testing. This activity includes specification of default text, paper, color options, document name, description, etc.

Defining and Locating Test Data

After specifying the initial design, you should next specify the parameters for testing the design. You can create a test data file and define parameters for placing in the design by using a text editor or by excerpting relevant data from a production data file.

In either case, records and fields in a test data file should structurally and accurately represent the production data, and follow the organization's business rules including all possible scenarios and data combinations. Typically, an effective test data contains only enough records and associated fields necessary to reflect the cohesiveness of the various design combinations, test the required results, and enable you to print preview the results. You can test and preview the document design competency with test data, forgoing the usually larger files typical of the actual production data.

Understanding the VIPP Modes

Vitesse supports the following VIPP modes for designing and testing your projects:

- Native mode
- Line mode
- Database mode
- XML mode

These modes or project types are based on the various VIPP input data formats and are discussed in the following sections.

What is Native Mode?

Native mode is used for creating, positioning, formatting, and printing static objects such as lines, boxes, text, images, forms, and segments directly on a physical page.

Applications created in the Native mode support fully composed Native mode data streams. For example, you can create college application forms or promotional flyers using this mode, as well as other documents. Native mode applications do not support variable data.

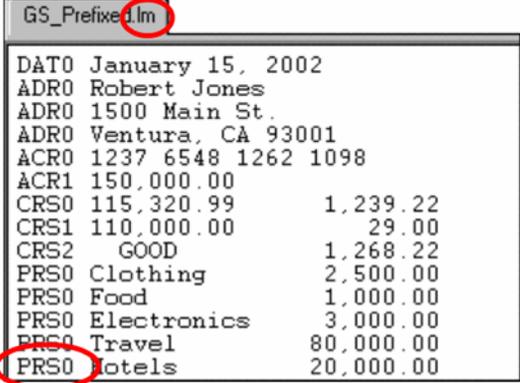
What is Line Mode?

Line mode enables you to work with legacy type data streams. Legacy type data streams are characteristic of transactional documents, such as telephone bills, credit card statements, banking statements, and sales reports. These data streams are also referred to as “line” or “print ready” data. One example is LCDS (Line Conditioned Data Streams). Line mode projects comprise line data in ASCII format that is line numbered or RPE prefixed. Moreover, the data can either have delimited or non-delimited fields within records. Line mode enables you to read, process, and format the line printer data either line-by-line (records) or column-by-column (fields).

Using Line mode, you can also control the page composition, color, and formatting of the line printer data by a JDT (Job Descriptor Ticket) file. This file contains the VIPP commands used to define the required presentation of the line printer data. The .LM extension is recommended for Line mode data files, but is not required, as VIPP and Vitesse can use other extensions.

The following image is an example of a fixed length, prefixed line mode data file:

Line mode data files can use the .LM extension.



Each record starts with a prefix.

```
GS_Prefixed.lm
DAT0 January 15, 2002
ADR0 Robert Jones
ADR0 1500 Main St.
ADR0 Ventura, CA 93001
ACR0 1237 6548 1262 1098
ACR1 150,000.00
CRS0 115,320.99      1,239.22
CRS1 110,000.00      29.00
CRS2 GOOD           1,268.22
PRS0 Clothing        2,500.00
PRS0 Food            1,000.00
PRS0 Electronics    3,000.00
PRS0 Travel          80,000.00
PRSO Hotels          20,000.00
```

What is Database Mode?

Database mode is used to process data files containing delimited records. The delimited data file is typically extracted from a relational database. You can use the Database mode to create applications, such as promotional flyers, insurance contracts, etc. Records in a data file may be of fixed or variable length, and must have the same sequence and number of fields. The first line in the data file contains field names. Remaining lines contain data in the order of field names. Delimiters are used to separate fields within records. Data files used in the Database mode can use the .DBF extension.

The following image is an example of a variable length data file in the Database mode:

Database mode data files can use the .DBF extension.

```
qsdat.dbf
TITLE:NAME:FNAME:LNAME:ADDRESS1:ADDRESS2:ZIP:DATE
Ms.:Jonina Robinson:Jonina:Robinson:54 Maple St.:Ventura, CA
Ms.:Martina Adams:Martina:Adams:12 Birch Rd.:Ventura, CA:930
Mr.:Brian Evans:Brian:Evans:3807 Bergen Dr.:Ventura, CA:9310
Mr.:Douglas Hanson:Douglas:Hanson:56 Oak St.:Ventura, CA:930
```

The first line in the file lists the field names and their sequence in the record.

":" represents the field delimiter.

What is XML Mode?

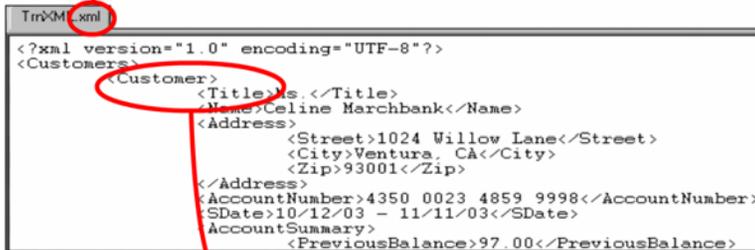
XML mode uses XML data files in ASCII format. Data in the XML files is organized in a tree structure. Nodes in the tree structure represent tag names, attribute names, and associated values. The entire data in the data file is enclosed within start and end tags.



Delimited and/or prefixed data files are not supported in XML mode.

XML mode requires an XJT (XML Job Ticket) file for processing XML data. An XJT file contains instructions on how to process and arrange the XML data into a document. It contains all global layout definitions for the document (orientation, forms, medias, frames, fonts, colors, etc.) and an XML Processing Definition (XPD) table that describes specific actions to be performed on specific XML tags. XML Mode files use the .XML extension.

XML mode data files use
.XML extension.



```
<?xml version="1.0" encoding="UTF-8"?>
<Customers>
  <Customer>
    <Title>Ms.</Title>
    <Name>Celine Marchbank</Name>
    <Address>
      <Street>1024 Willow Lane</Street>
      <City>Ventura, CA</City>
      <Zip>93001</Zip>
    </Address>
    <AccountNumber>4350 0023 4859 9998</AccountNumber>
    <SDate>10/12/03 - 11/11/03</SDate>
    <AccountSummary>
      <PreviousBalance>97.00</PreviousBalance>
```

Tags represent Record and Field names,
text between the nodes represents the data.

Sample of XML mode data file.

What's New in Vitesse 2.00?

Vitesse 2.00 introduces new and advanced features that can be accommodated in the designs. Also, some of the existing features and object behaviors have been enhanced in the new version. The sections below briefly explain the features and associated behaviors.

Form and Segment Behavior

Vitesse 2.00 enables you to manage the content of segments and forms through the use of conditional logic. Additionally, you can dynamically resize segments based on the objects used in the segment design.

Frame Behavior

Vitesse 2.00 supports the overflow of segments across multiple frames. In previous Vitesse versions, if the contents of a segment object did not fit in one frame, the rest of the contents were printed outside the frame. With this new overflow behavior, if the contents of a segment do not fit in one frame, the rest of the contents automatically flow to the next frame. If the current frame is the last frame on the page, then the overflow behavior automatically causes a page break and a new page is created.

Restricted Mode Operation

You can operate Vitesse 2.00 in administration and/or user modes. Using the Administration mode, you can operate Vitesse as an administrator, utilizing all the functionalities provided in Vitesse. Moreover, you can use the **Administrator** category in the *Preferences* dialog (discussed in later chapters) to choose features that you want to be available in the Vitesse user mode.

User mode provides a customized Vitesse workspace, depending on the preferences set by the administrator. The administrator can restrict options such as creating, opening, saving, or exporting projects in Vitesse user mode. The administrator can also setup rights for each project resource by setting preferences in the Administration mode. For example, restrictions can be set for loading form resources in a project.

Predefined Media and Fields

Vitesse 2.00 enables you to predefine media settings by specifying paper properties in an XML file. By placing the media file at the path *drive:/Program Files/Vitesse/Config*, only papers defined in the XML file are inserted in a project. Similarly, you can also set up a predefined variable list to add variables in forms and segments.

Batch Conversion of Documents

You can export multiple .EVD files in a single application instance using the Vitesse batch conversion feature. This process is invoked using the command prompt. You can also choose to log all of the messages generated during the conversion process.

Section I: Vitesse Basics

User Notes:

Chapter 2: Understanding the Vitesse Workspace

This chapter provides an overview of the Vitesse workspace and describes activities and tasks for using the workspace. Information includes the following:

- Getting Familiar with the Workspace
- Using the Design Area
- Using the Project Tree View Window
- Using the Project Tree, Property, Data, and Message View Windows
- Arranging the Workspace Windows
- Using Workspace Tools
- Placing Objects in the Design Area
- Moving Objects using the Keyboard
- Zooming within the Design Area
- Moving the Design Area view
- Using the Full Screen View
- Refreshing the Project
- Navigating the Workspace
- Choosing Commands

Getting Familiar with the Workspace

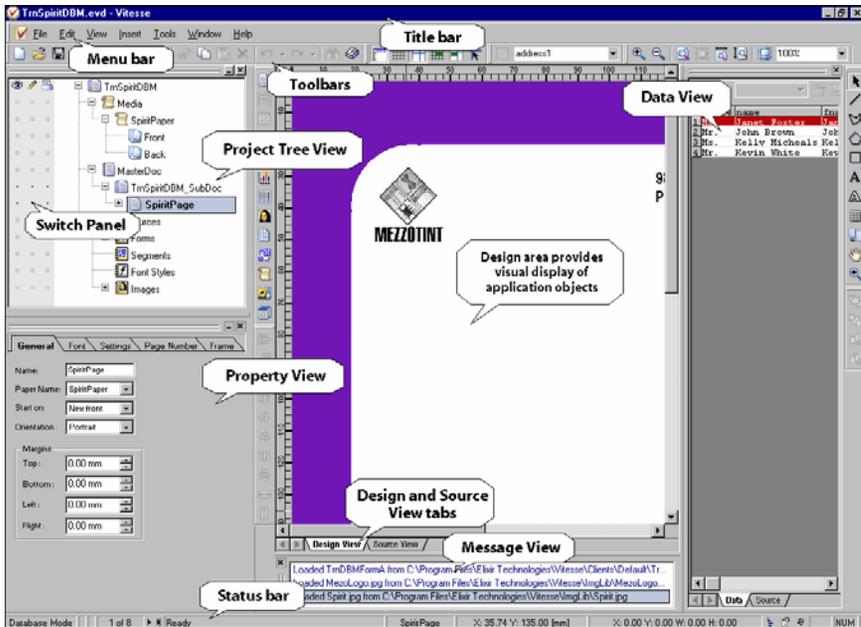
When you open a project in Vitesse, the workspace and all associated components display. Application objects are created and maintained in the workspace.

The workspace contains the following windows:

- Design area
- Project Tree View window
- Property View window
- Data View window
- Message View window

You can customize the position and size of the windows per your requirements. The workspace windows can be displayed as either floating or docked and can also be hidden from view.

The image below identifies each part of the Vitesse workspace:



The Vitesse workspace.

The following sections explain each window in detail.

Using the Design Area

The design area provides a visual representation of the active application. Vitesse provides two tabs for viewing the design area in the following two modes: Design View and Source View. You can scroll the design area vertically and horizontally using the scroll bars in both views.

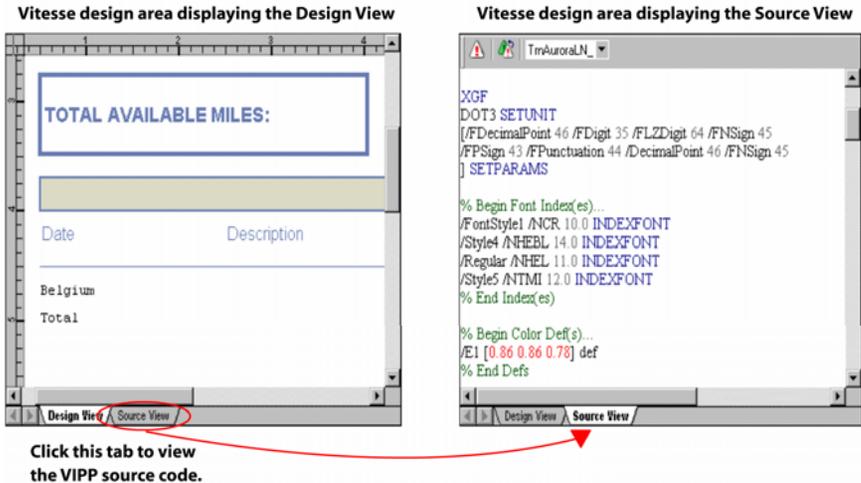
Design View

Design View is the default view when a project is opened in Vitesse. You can add drawing objects to this view, visually map objects and data, resize, and move objects as required. Mapped objects have corresponding entries automatically created in the project tree. Object properties that you set are also immediately reflected in the Design View.

You can, therefore, design a document and view its composed structure merged with the variable data simultaneously in the Design View.

Source View

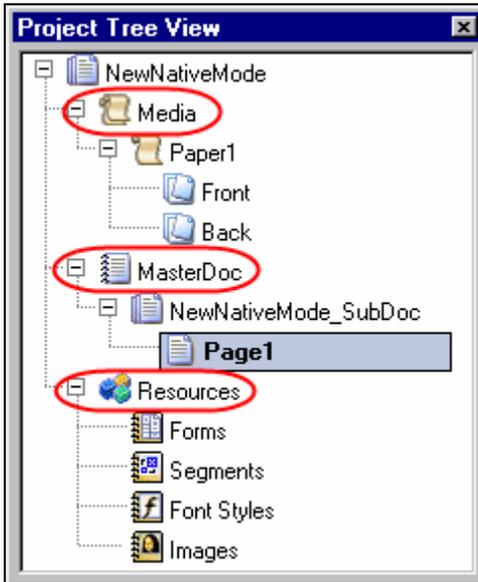
Clicking the **Source View** tab generates the VIPP source code for the active application. Application or external resource source code can be viewed either by selecting the **SubDoc** node or the required Project tree resource node and then switching from **Design View** to **Source View**, or by selecting options from the drop-down list provided in the Source View.



Design area can be viewed using the Design View and Source View tabs.

Using the Project Tree View Window

By default, the *Project Tree View* window is located at the top left side of the design area. This window contains a project tree that provides a hierarchical representation of the Vitesse objects used within a project. The structure of the tree varies, depending on the type of resource or project mode you are working on.



Project Tree View window displaying the default nodes.

The root (top most) node in the project tree specifies the VIPP project type and name. Sub-nodes under the root node are the default high-level nodes and contain further sub-nodes explained in the sections below. Objects that are inserted in the design area are also added to the project tree as object nodes. You can set properties for each object node using the *Property View* window. You can also use the project tree to select, add, and/or remove resources and objects from the project. During printing, Vitesse processes objects in the order they are placed in the project tree.



Default project tree nodes cannot be deleted.

Media

The **Media** node enables you to insert physical paper(s) to your design. Each inserted paper (or **Paper** node) can have different property settings such as the paper type, orientation, size, etc. You can set the properties for a particular paper by selecting the **Paper** node, and then specifying the associated properties in the *Property View* window. Each **Paper** node further contains a **Front** and **Back** node that indicate the front and back side of the selected paper respectively. You can insert a form on either side of the paper.

MasterDoc

The **MasterDoc** node contains the **SubDoc** and **Page** node by default. These nodes represent the logical page(s) and their associated properties used in a design. You can set the logical page definitions at the MasterDoc, SubDoc, and Page level using the *Property View* window. The **MasterDoc** node and consequently the **Page** node also contain nodes for all objects that are mapped to the logical page(s).

Resources

The **Resources** node contains the sub-nodes **Fonts**, **Segments**, **Font Styles**, and **Images** that represent all of the resources (forms, segments, fonts, and images) used or referenced in the project.

You can display or hide the *Project Tree View* window by selecting Window>Project Tree from the main menu.

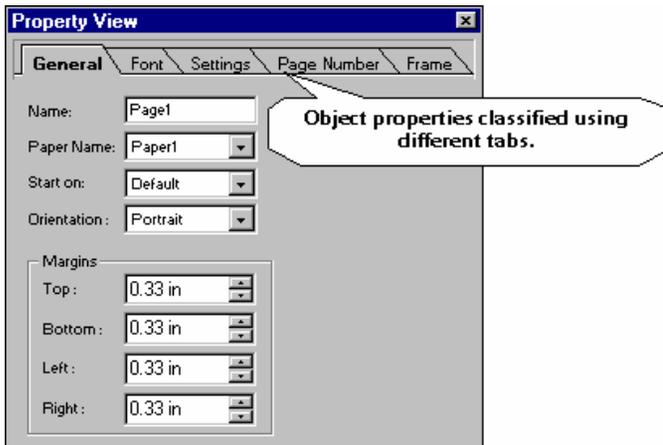


You can also display or hide the *Project Tree View* window either by:

- Clicking  on the status bar.
- Or
- Pressing <Ctrl>+<1> on the keyboard.

Using the Property View Window

Once an object is added to the design, you can specify the attributes/properties using the *Property View* window. The *Property View* window is located at the bottom left of the workspace by default. The properties displayed in the *Property View* window change as different objects are selected from either the design area or the project tree. Each object can have different attributes/properties that are categorized using different tabs in the window.



The *Property View* window displaying the *General* properties associated with a *Page* object.

To specify any property, select the relevant tab and make your changes using the available options.

The *Property View* window can be opened or closed by selecting the *Window > Properties* menu option.



You can also display or hide the *Property View* window either by:

- Clicking  icon on the status bar.
- Or
- Pressing **<Ctrl> + <2>** on the keyboard.

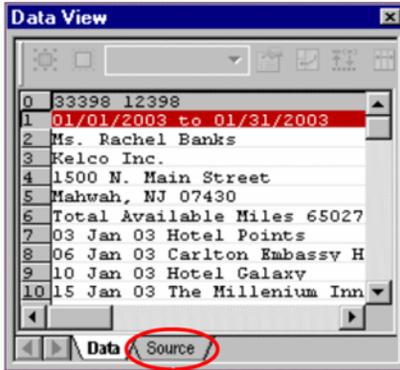
Using the Data View Window

Vitesse also supports the use of data files for composing data-driven applications. The data files can be viewed in the *Data View* window so you can drag and drop required data directly into the design area. This window is context-sensitive and displays data files and associated options relating to specific project modes (i.e., Line, Database, or XML mode).

The *Data View* window is located at the right side of the workspace by default and contains two tabs:

- The **Data** tab shows the data file in a row and column layout for Line Mode and Database Mode projects. For an XML Mode project, the data is displayed in a tree structure.
- The **Source** tab displays the source code of the data file. You can edit the file only if it is not in read-only mode.

Data View window displaying the data in a grid layout.



Source View window displaying the data for editing.



Click on the Source tab to edit the data file.

Data View window can be viewed using the Data View and Source View tabs.

To view the *Data View* window if it is not already open, select Window>Data from the menu bar.



You can also view the *Data View* window either by:

- Clicking  on the status bar.
- Or
- Pressing <Ctrl> + <3> on the keyboard.

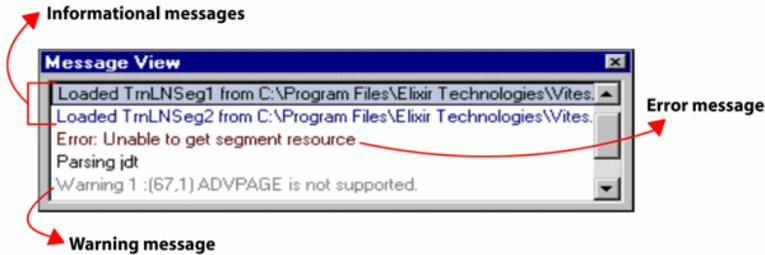
Using the Message View Window

The *Message View* window is located at the bottom of the workspace by default. This window displays the processing and compilation errors, warnings, and informational messages generated by the application. Messages are displayed when either a VIPP application is being imported or source code is being edited and compiled.

Error messages display in black text, warnings display in gray text, and informational messages display in blue text. You can close the *Message View* window by selecting Window>Messages menu option to allow more space to work. However, the window will automatically redisplay whenever a message is generated by the application.



You can also view or hide the *Message View* window by pressing <Ctrl> + <4> on the keyboard.



The Message View window displaying informational, error, and warning messages in their respective colored text.

Clearing Messages

To clear the messages generated by the application, right-click in the *Message View* window, and then select **Clear** from the pop-up menu.

Saving Messages

To save the messages generated by the application:

- 1 Right-click in the *Message View* window area.
- 2 Select **Save** from the pop-up menu to open the *Save* dialog.
- 3 Search **Save in** for the required folder where you want to save the message file.
- 4 Specify the file name in the **File name** entry box.
- 5 Click **Save** to save the file.



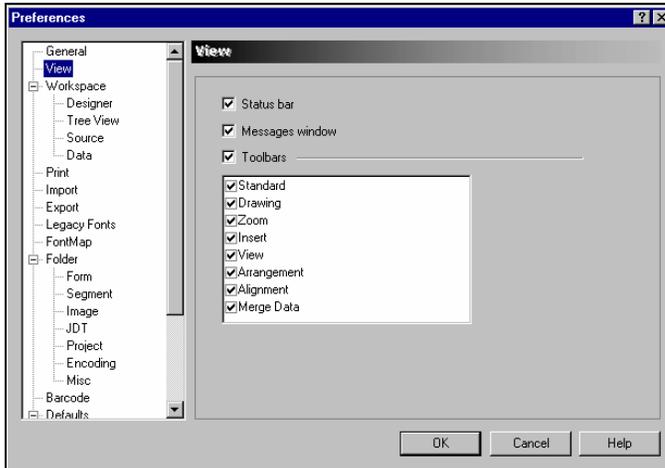
You can create a message log file in any required format.

Setting the Default View Options

You can customize Vitesse to display or hide the *Message View* window, tool bars, and status bar every time you open a project. To specify the viewing options:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **View** category.

- 3 Select the options you want to show or hide each time the application is opened. Options include **Status bar**, **Messages window**, **Toolbars**, and associated toolbar options in the list.



Preferences dialog –View Category.

- 4 Click **OK**.



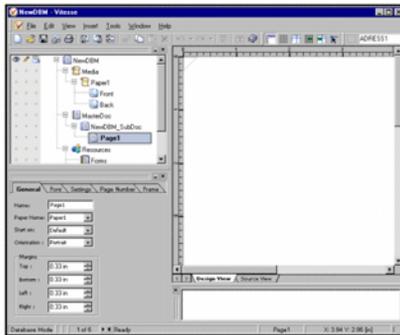
The items not selected will not be displayed the next time you open a project. However, you can view them any time during the application design using the **View** menu or the **Window** menu.

Arranging the Workspace Windows

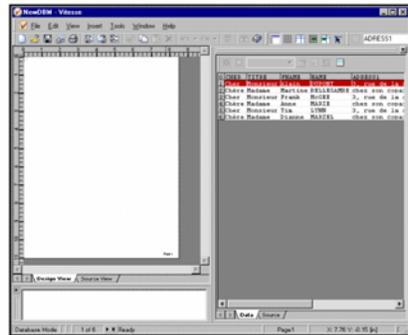
Vitesse lets you arrange the workspace windows in an orderly manner using the main menu options. To arrange the design area windows, select the required arrangement option from the Windows menu.

- **Default State** – displays all of the workspace windows docked to the default locations.
- **Data/Design State** – displays the *Data View* window and the design area docked in the workspace.
- **Data/Design/Tree State** – displays the *Data View* window, the design area, and the *Project Tree View* window as docked in the workspace.

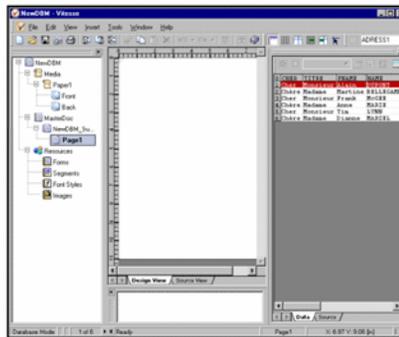
An illustration of these views is as follows:



The default view displaying all the windows docked at their default locations.



The Data/Design State displaying the data view and the design view.



The Data/Design/Tree State displaying the data view, design view and the Project tree.

The Message View window displays at the bottom of the workspace in all three states.

Using Workspace Tools

Vitesse toolbars provide easy access to most commonly used application commands. You can perform standard MS-Windows operations to hide or display a toolbar or dock a toolbar at the edge of the workspace.

Using the Menu Bar

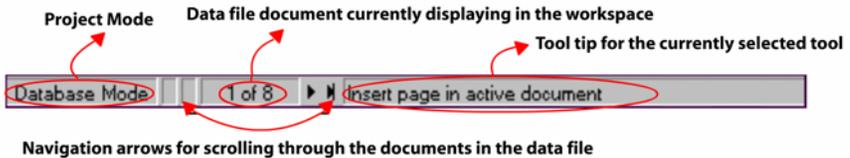
A menu bar with drop-down menus displays directly below the application title bar. These drop-down menus provide both standard MS-Windows and application-specific options. Selecting a menu item performs the associated action or displays a submenu or a dialog.

Using the Toolbar

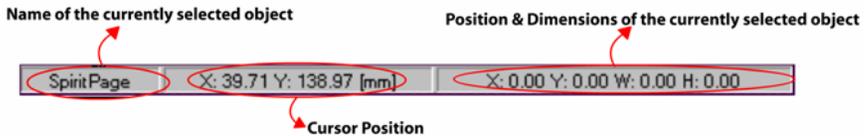
Directly below the menu bar are toolbars with buttons that provide access to the most commonly used menu commands and tools. You can hide or display all available toolbars by selecting **View>Toolbars>toolbar name** from the main menu. You can also dock the toolbars in the workspace.

Viewing the Status Bar

The status bar displays along the lower edge of the Vitesse workspace. When you move the mouse pointer over the design area, the status bar displays the current position of the mouse pointer and the zoom level. The status bar also displays the tool tip when the mouse pointer is placed over a command.



In addition, when an object is selected or being drawn in the design area, the status bar also displays the tool tip of the object being drawn, the height, width, and the current coordinates of the mouse pointer. This information helps when drawing objects in the design area.



At the extreme right of the status bar are three buttons, , , and  to show or hide the *Project Tree View* window, the *Property View* window, and the *Data View* window .



To view all of the buttons on the status bar, ensure the Vitesse workspace window is in the maximized state.

Placing Objects in the Design Area

Vitesse provides various tools that help you draw and place objects accurately in the design area. These tools are discussed below:

Using the Ruler

Vitesse provides horizontal and vertical rulers to help visually place objects in the design area. Rulers display along the top and left side of the design area. When you move an object or cursor in the design area, lines indicating the object or cursor position appear on the rulers. By default, the unit of measurement for the rulers is Dot 3 (dots per inch). You can show or hide the rulers and change the unit of measurement used in the rulers.

Displaying and Hiding the Ruler

To display or hide the rulers, select View>Ruler from the menu bar.



You can also display or hide the ruler either by:

- Clicking  on the *View* toolbar.
- Or
- Pressing <Ctrl> + <R> on the keyboard.

Setting the Default Ruler Option

To display or hide the rulers every time you open a project:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Ruler** option under the **View** area as required, and then click **OK**.

Changing the Unit of Measurement

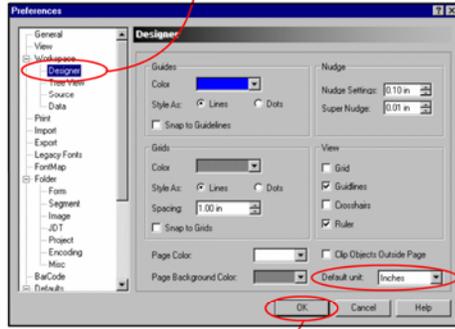
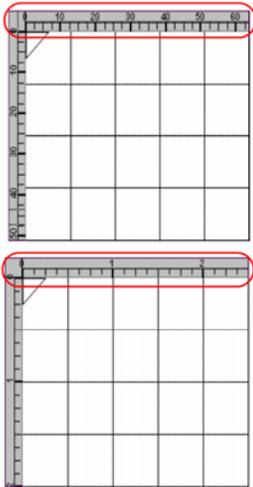
You can also change the unit of measurement for the rulers. To specify the required unit of measurement:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Specify the required unit of measurement from the **Default unit** drop-down list.
- 4 Click **OK**.

The design area displays the rulers with the specified settings.

1. Default unit of measurement is Dot3. To change the unit of measurement, open the Preferences dialog.

2. Select the Designer category to display the associated options.



3. Select the required unit (e.g. inches) from the Default unit drop-down list.

4. Clicking OK changes the unit of measurement from Dot 3 to Inches.

Image showing steps for changing the default unit of measurement.



To change the unit of measurement, you can also right-click the rulers displaying in the design area, and then select the required unit from the pop-up menu.

Using the Grid

You can also use the grid while creating an application. A grid can display either as a set of lines or dots in the design area. Grids help in marking ruler positions across and down the design area so you can align objects accurately in the design. The grid, however, is only a tool to assist you visually with the application design; it does not display when you print or preview your document.

You can show or hide the grid; snap objects to the grid; and you can modify the grid spacing and color.

Displaying and Hiding the Grid

To display or hide the grid tool, select View>Grid from the menu bar.



You can also turn the grid on or off either by:

- Clicking  on the *View* toolbar.
- Or
- Pressing <Ctrl> + <G> on the keyboard.

Setting the Default Grid Option

To display or hide the rulers every time you open a project:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Grid** option under the **View** area, and then click **OK**.

Automatically Aligning Objects to the Grid

Vitesse enables you to automatically align objects to the grid by providing the snap to grid feature. Snapping offers a way of precisely placing an object on the design area by having the object affix itself to the intersections of grid lines or dots.



We recommend keeping the grid on before activating the snap to grid feature so that you can see the snap points.

To turn on the snap to grid feature, click  on the *View* toolbar. You can use the same method to turn snapping off.

The object snaps to the next grid point.

Cursor moves the object to a position between the grid points.

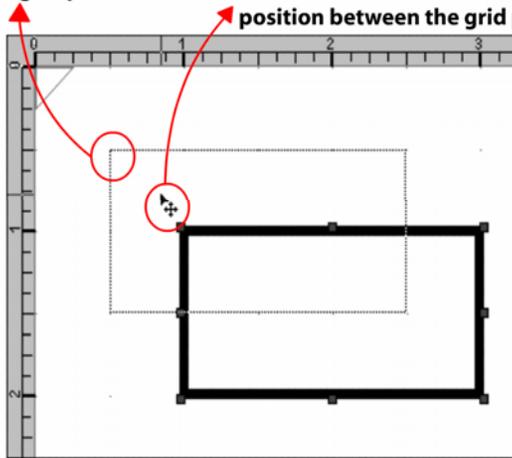


Image shows the box alignment to the grid using the snap to grid feature.

Setting the Default Snap to Grids Option

To turn the snap to grid feature on or off every time you open a project:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Snap to Grids** option under the **Grids** area, and then click **OK**.

Setting the Grid Color

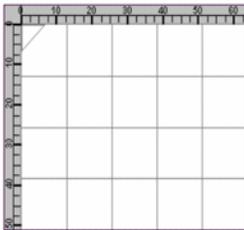
You can also specify the grid color using the following steps:

- 1 Select Tools>Preferences from the main menu to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 For setting the grid color, click the **Color** drop-down under the **Grids** area; a color palette with command buttons for choosing the appropriate color is displayed. Select the required color.
- 4 Click **OK**.

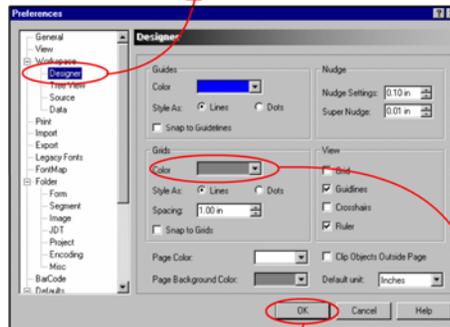
The grid updates with the specified color.

1. Grids displaying in grey color.

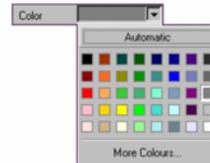
To change the color, open the Preferences dialog.



2. Select the Designer category to display the associated options.



3. Select the required color (e.g. blue) from the drop-down color palette.



4. Clicking OK changes the grid color from grey to blue.

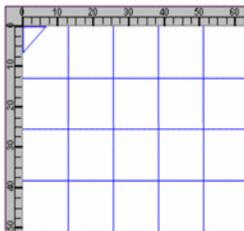
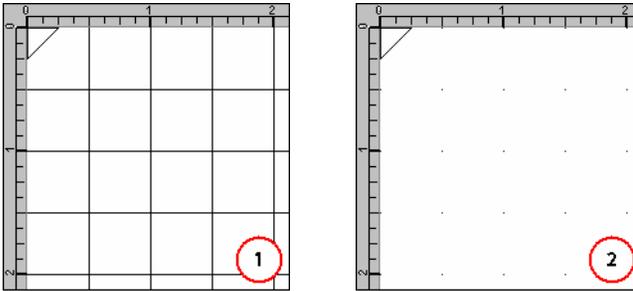


Image showing steps for setting the grid color.

Setting the Grid Style

You can choose to display the grid as either intersecting lines or dots in the design area. To specify the grid style:

- 1 Select **Tools>Preferences** from the main menu to display the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 As appropriate, select any one of the following **Style As** options available under the **Grids** area:
 - **Lines** – displays the grid as intersecting lines.
 - **Dots** – displays the grid as dots.
- 4 Click **OK**. The grid displays in the selected style.



Design area displaying grid style set to: (1) Lines (2) Dots.

Setting the Grid Spacing

Vitesse lets you customize the spacing between the grid points, as per the specified unit of measurement.

To specify the grid spacing:

- 1 Select **Tools>Preferences** from the main menu to display the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Specify the required spacing in Inches in the **Spacing** entry box under the **Grids** area.



You can also set the unit of measurement by right clicking the upward and downward arrows in the **Spacing** entry box and selecting **Units>unit name** from the pop-up menu.

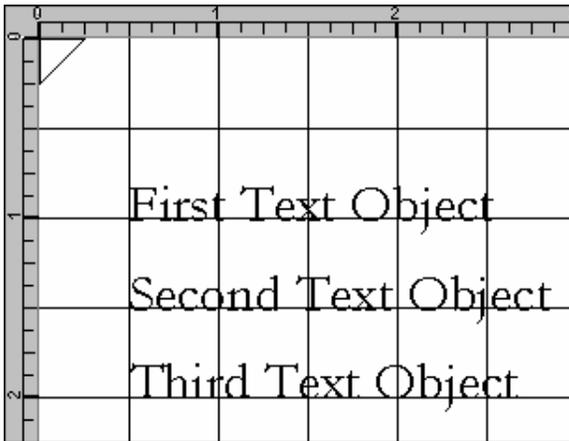


The **Spacing** value can be specified in Inches only.

4 Click **OK**.

The grid displays as per the specified spacing value.

Setting the grid spacing visually helps with placing objects in accordance with the grid lines. For example, if you want to place text objects vertically at a distance of 0.5 inches with each other, you can set the grid spacing to 0.5 inches.



Design area displaying text objects placed accurately on the grid lines with spacing value set to 0.5 Inches.

Using Guides

Guides are horizontal and vertical lines that you can drag from the rulers onto the design area. You can move, hide, and remove the guides. You can also snap objects to guides, change guide color, and choose to display guides as solid or dotted lines. Guides are used as visual aids when placing objects and do not display while previewing or printing.

Turning the Guides On and Off

To turn the guides on or off, select View>Guides from the menu bar.



You can also turn the guide on or off either by:

- Clicking  on the *View* toolbar.
- Or
- Pressing <Ctrl> + <D> on the keyboard.

Setting the Default Guide Option

To display or hide the guides every time you open a project:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Guidelines** option under the **View** area, and then click **OK**.

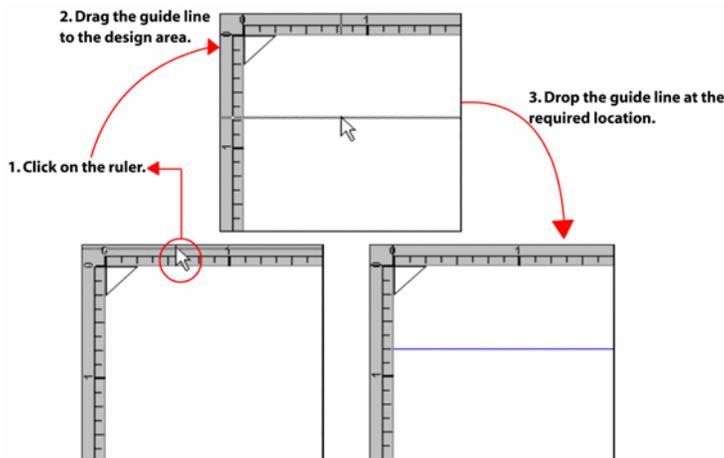
Moving Guides to the Design Area

When you turn on the guides feature, the guides do not display in the design area like the ruler or the grids do. To view the horizontal guide, click at any point on the horizontal ruler and drag the line to the required position in the design area. Similarly, to view the vertical guide, drag the line from any point on the vertical ruler to the required location in the design area. You can drag as many horizontal and vertical guides as appropriate.



Before using the guides option, you should make sure the rulers are displaying in the design area.

The image below illustrates the process of viewing guides in the design area:



Removing Guides from the Design Area

To remove a guide from the design area, click the guide and then drag it back to the ruler.

Automatically Aligning Objects to the Guides

Vitesse also provides the snap to guide feature for automatically aligning objects to the guides. This enables you to place objects precisely in the design area.

To turn on the snap to guide feature, click  on the *View* toolbar. You can use the same method to turn snapping off again.



Snapping to guides takes precedence over snapping to the grid in places where guides fall between the grid lines.

Setting the Default Snap to Guides Option

To turn the snap to guide feature on or off every time you open a project:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Snap to Guidelines** option under the **Guides** area, and then click **OK**.

Setting the Guide Color

You can also specify the guide color using the following steps:

- 1 Select **Tools>Preferences** from the main menu to display the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 For setting the guide color, click the **Color** drop-down under the **Guides** area; a color palette with command buttons for choosing the appropriate color is displayed. Select the required color.
- 4 Click **OK** to update the guides with the specified color.

Setting the Guide Style

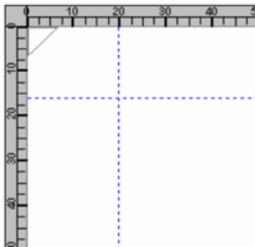
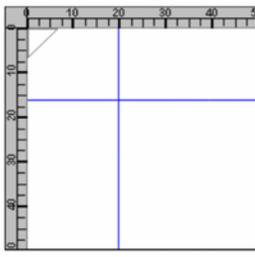
You can choose to display the guides as either solid or dotted lines in the design area. To specify the guide style:

- 1 Select Tools>Preferences from the main menu to display the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 As appropriate, select one of the following options available under the **Guides** area:
 - **Lines** – displays the guides as solid lines.
 - **Dots** – displays the guides as dotted lines.
- 4 Click **OK**.

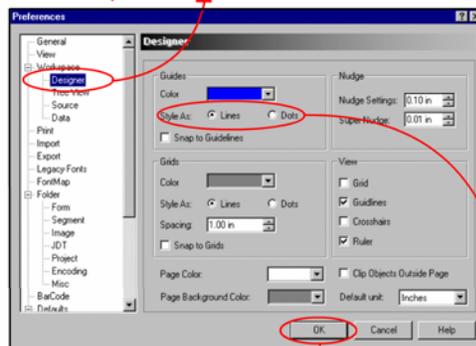
The guide displays in the selected style.

1. Guides displaying in line style.

To change the style, open the Preferences dialog.



2. Select the Designer category to display the associated options.



3. Select the Dots option.



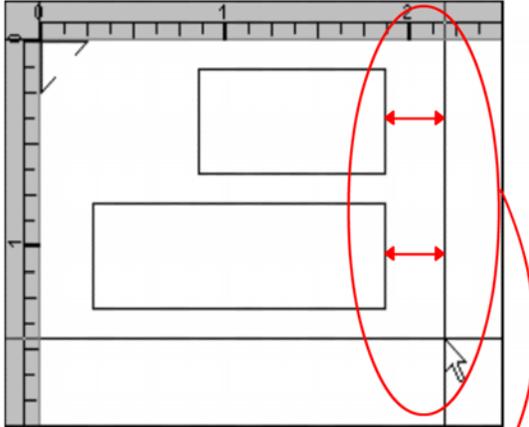
4. Clicking OK changes the style from lines to dots.

Image showing steps for setting the style for guides.

Using the Crosshair

The Crosshair tool displays as a pair of lines intersecting at the cursor. These lines move with the cursor and help in placing objects relative to each other in the design area.

You can also use the crosshair to assist in aligning objects visually, as shown below:



Using crosshair to verify the alignment of objects visually.

Displaying and Hiding the Crosshair

To display or hide the crosshair, select View>Crosshair from the menu bar.



You can also turn the crosshair on or off either by:

- Clicking  on the *View* toolbar.
- Or
- Pressing <Ctrl> + <L> on the keyboard.

Setting the Default Crosshair Option

To display or hide the crosshair every time you open a project:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 Select or deselect the **Crosshairs** option under the **View** area, and then click **OK**.

Moving Objects using the Keyboard

Once an object is placed in the design area, you can move it anywhere in the design area. To precisely place objects in the design area, you can also make use of the arrow keys on the keyboard.

To move an object using the keyboard:

- 1 Select the required object in the design area.
- 2 Press the appropriate arrow key to move the object in the required direction with a specified displacement. This is also referred to as nudge in Vitesse.
- 3 Press <Shift> along with the appropriate arrow key to move the object in the required direction with a finer displacement. This is referred to as super nudge in Vitesse.

Setting the Nudge Options

You can set the increment value (nudge setting) by which you want to displace an object.

To specify the nudge settings:

- 1 Select Tools>Preferences from the main menu to display the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **Designer** category under the **Workspace** node.
- 3 In the **Nudge** area, specify a value in Inches for the following:
 - **Nudge Settings** – sets the value by which an arrow key displaces the selected object in the appropriate direction.
 - **Super Nudge** – sets the smallest value by which an arrow key displaces the selected object in the appropriate direction.



The **Nudge** values are specified only in Inches.

- 4 Click **OK**.

Zooming within the Design Area

To view the entire design on the screen, or to view just a particular area of your design at higher magnification, you can use the zoom tool. Vitesse provides various options for setting the magnification level. To do so, select **View>Zoom>required option** from the menu bar. The options are:

- **Zoom In** – magnifies the entire design area view. You can also magnify the view by clicking  on the *Zoom* toolbar.
- **Zoom Out** – reduces the entire design area view. You can also zoom out of the view by clicking  on the *Zoom* toolbar.
- **Zoom Back** – reverts back to the last magnification level.
- **Zoom Fit** – scales the design view to fit completely in the given design area. You can also zoom to full extent by clicking  on the *Zoom* toolbar.
- **Zoom to Object** – magnifies the selected object in the design area. You can also zoom to the selected object by clicking  on the *Zoom* toolbar.
- **Zoom to Width** – scales the design view to fit the design area width. You can also use the **Zoom to Width** option by clicking  on the *Zoom* toolbar.
- **Zoom to Height** – scales the design view to fit the design area height. You can also use the **Zoom to Height** option by clicking  on the *Zoom* toolbar.

The View>Zoom menu also provides sub-menu options to zoom in or out using pre-defined values: 50%, 100%, 250%, or 500%.

Click  on the Zoom toolbar to zoom to the selected object.

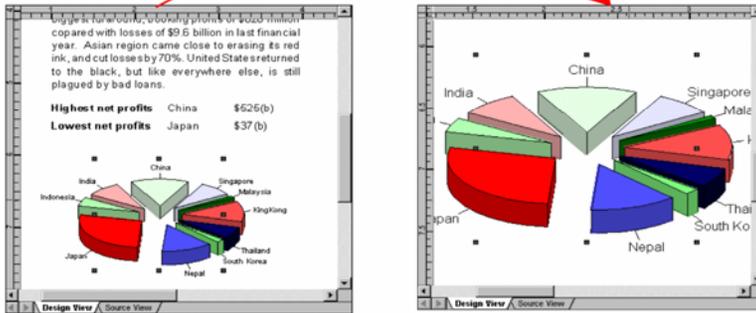


Image showing the magnified pie chart view using the Zoom to Object option.

Specifying Custom Zoom Value

To zoom in or out of the design area to a specified custom value:

- 1 Select View>Zoom>Custom from the menu bar to open the *Custom Zoom* dialog.
- 2 Specify the required **Zoom** percentage value. You can either select the values available in the drop down list or type a new value.
- 3 Click **OK**.



You can also specify a custom zoom value in the text box provided in the *Zoom* toolbar.

Moving the Design Area View

When the design area is magnified, you may not be able to see all of it. The pan tool lets you move the design area to change the view without having to change the magnification.

To move/pan the design area view:

- 1 Click  on the *Drawing* toolbar.
- 2 Click anywhere in the design area and drag to the required view.

Using the Full Screen View

To display as much of your design as possible on the screen, you can switch to the full-screen view. In this view, Vitesse removes most of the screen elements, such as the menu bar, toolbars, and scroll bars. It only displays the design area, the *Project Tree View* window, and the *Property View* window.

To switch to the full screen view, select View>Full Screen from the menu bar or click



on the *Zoom* toolbar.

To turn off the full-screen mode and switch to the previous view, click  on the *Full Screen* toolbar, or press <Esc> on the keyboard. The *Full Screen* toolbar displays only in the full-screen mode.

Refreshing the Project

You can refresh the document design to reflect the latest changes in the design area by selecting View>Refresh from the menu bar. You can also refresh your design by pressing <F5> on the keyboard.

To refresh the design changes along with the property settings specified in the *Property View* window, select View>Refresh All from the menu bar. You can also perform the same operation by pressing <Shift> + <F5> on the keyboard.

Navigating the Workspace

Vitesse workspace components are organized according to function and use. You can navigate any component using the mouse.

- **To use the menu bar:**

Click the appropriate menu, and then select the required option from the drop-down list.

- **To show the toolbar:**

From the *View* menu, select Toolbars>*toolbar name*. A check mark next to the required toolbar indicates that it has been selected.

- **To use a toolbar:**

Click the required icon on a toolbar to perform an operation.

- **To use the right-click menu:**

Position the cursor on the menu or design area, right-click the mouse, and then select the required option.

Choosing Commands

A command instructs Vitesse to perform a specific action. You can choose a command by either:

- Clicking a toolbar button.
- Selecting a menu command.
- Using a shortcut menu.
- Using a shortcut key.

In the chapters that follow, command selection is described using the most common method for the described task. However, any related command selection can be used to accomplish the same instruction.

User Notes:

Chapter 3: Understanding the Vitesse Directory Structure

This chapter explains the Vitesse directory structure, the scope of resources used in Vitesse, and associated tasks for setting up working folders. Information covers:

- Understanding Project Resource Scope
- Understanding the Vitesse Directory Structure
- Setting up the Working Folder
- Using Scope Specific View in the Project Tree
- Setting Default Folder Preferences

Understanding Resource Scope

A resource's scope refers to the accessibility of a resource in various Vitesse projects. There are three types of resource scopes defined in Vitesse:

- Local
- Shared
- Global

Local Resources

Local resources are contained only in projects in which they are created or referenced. By default, all resources created or referenced within a project are local to that project.

Shared Resources

Shared resources are shared between groups of similar projects that reside in the same directory. For example, a department logo may be shared across all the projects of the department but not across the entire company. In this case, you can store the logo as a shared resource between the department projects to avoid storing it individually in every project.

Global Resources

Global resources are available to every project that you create in Vitesse. For example, a company logo assigned with a global scope is available across all the projects within a company. The purpose is to avoid storing the logo individually in every project.

Sharing resources saves storage space, ensures consistency, and simplifies maintenance. Any changes that you make in a resource assigned a shared or global scope is applied to all the projects that it is being referenced. We suggest that you keep the commonly used resources as either shared or global, whichever is applicable.

Resources with different scopes reside at different locations according to the Vitesse directory structure (explained in “Understanding the Vitesse Directory Structure” task below). Once you have defined the scope of a resource, you can manually place it at the required directory location.

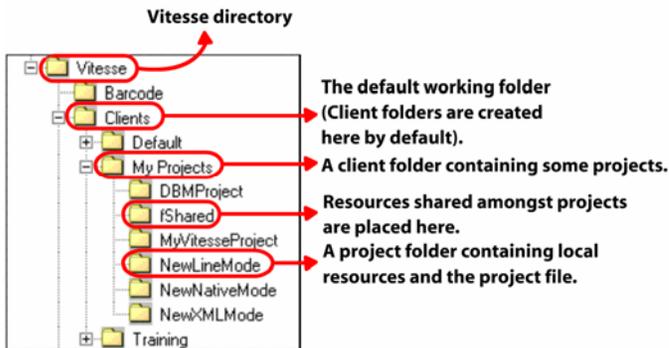


Local resources take precedence over the shared and global resources. For example, if a resource referenced in a project is present in the local, shared, and global scope, the local resource will be used. Similarly, shared resources take precedence over global resources.

Understanding the Vitesse Directory Structure

Vitesse formats and composes documents using the Dynamic Document Composition model (DDC). DDC is based on real time composition of a document at the printer or the imaging device. In keeping with this model, the resources are often referenced in a design and do not form a part of the project file. The referenced resources (images, segments, and forms) are located and accessed from the different Vitesse directories, which are explained later in this section.

Vitesse follows the same folder structure used in VIPP for organizing the project files and the resources referenced in projects. The following image shows the basic structure of the Vitesse folders/directories:



The Vitesse directories are explained as follows:

Vitesse Folder

The main **Vitesse** directory is located at *drive:\Program Files\Elixir Technologies\Vitesse*, where *drive* is the Vitesse installation drive. The **Vitesse** folder contains the application components (e.g. *.dll, *.ocx, *.exe files etc.); resource, client, and project folders as discussed below:

ClientsFolder

The **Clients** folder is located at *drive:\Program Files\Elixir Technologies\Vitesse\Clients*. **Clients** is the default working folder in which the client folders associated with projects are created and maintained.



You can change the working folder by using the procedure explained in the “Setting up the Working Folder” section later in this chapter.

Client Folder

All Vitesse projects essentially relate to a client specified at the time of project creation. Vitesse maintains these clients as folders; each client folder further contains multiple project folders that consist of applications created in Native, Line, Database, and XML modes. A client folder helps maintain a set of projects that share common resources. Besides the project folders, a client folder also contains an **fShared** folder; all of the shared resources used within the projects are located in the **fShared** folder.



Whenever a new client folder is created, an **fShared** folder is automatically created within the client folder. New client folders can be created at the time of project creation. For more information, refer to *Section 4: Creating a VIPP Project*.

Vitesse creates a **Default** folder within the **Clients** folder that can be used as a default client when creating projects. The **Default** option is always available in the list of clients whenever you create a new project. You can also create and maintain your own client folders for different projects. In the image above, **MyProjects** is a custom client folder that contains the project folders.

Project Folder

When you create a project, Vitesse automatically makes a project folder in the specified client directory. The name of the project folder is identical to the name specified at the time of project creation. For example, in the image, **NewLineMode** is the project folder created in the **MyProjects** client directory. The resources created or referenced within this project are local and also located in the same project folder.

Data Files

In the **Data Files** folder, you can place the actual data files used in the print production process.

Config

The **Config** folder contains the Vitesse configuration files and the sample data files for projects.

xgf

The **xgf** folder contains the predefined sample VIPP applications that you can import in Vitesse.

FormLib

The stand-alone form resources are created in the **FormLib** directory by default. In this directory, you can also place the form resources that are shared globally among projects located within different client folders.

ImgLib

In the **ImgLib** directory, you can place the image resources that are shared globally among projects located within different client folders.

SegLib

The stand-alone segment resources are created in the **SegLib** directory by default. In this directory, you can also place the segment resources that are shared globally among projects located within different client folders.

MisLib

In the **MisLib** directory, you can maintain miscellaneous resources that are accessed globally by projects.



The FormLib, ImgLib, SegLib, and MisLib directories are also referred to as the global resource libraries.

Setting up the Working Folder

Clients folder is the default working directory in which all of the Vitesse projects are created. You can also create and save your projects at different locations other than the default working folder.

To set up the working folder:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **General** category to view the associated options.
- 3 Specify the required directory path in the **Working Folder** entry box. You can also click  adjacent to the **Working Folder** entry box to open the *Browse for Folder* dialog to specify the location.
- 4 In the *Browse for Folder* dialog, browse and select the required directory.
- 5 Click **OK** to return to the *Preferences* dialog.

The full path of the specified directory displays in the **Working Folder** entry box.

- 6 Click **OK**.

Projects will now be created at the specified working directory.



You can also open an existing project from any other location on the system. Selecting File>Save from the menu bar saves the existing project in the specified working folder.

Using Scope Specific View in the Project Tree

Vitesse lets you view the project tree and the project resources on the basis of their resource scope. The view is created depending on the locations from where the resources are being accessed in a design. For example, if a project uses a form that is located in the **FormLib** directory, the form is displayed as a global resource. Similarly, if you use the **fShared** directory for referencing a form, it is displayed as a shared resource.

To view the project resources based on the resource scope, select View>Project Tree>Scope Specific View from the menu bar.

1. Project Tree View displaying the resources classified on the basis of resource type.



2. Select Project Tree>Scope Specific View from the View menu.



3. Project Tree View displaying the resources classified on the basis of resource scope.



Image shows the steps involved in setting the scope specific view.

In the scope specific view, the following three nodes identifying the three scopes are displayed under the **Resources** node:

- **Local** – contains the resources that are accessed from the project folder itself.
- **Shared** – contains the resources that are accessed from the **fShared** folder of the respective client directory.
- **Global** – contains the resources that are accessed from the global resource libraries.

You can toggle between the two views through this menu item. By default, all of the resources in the project are displayed without their scope specific view and classified according to their types such as forms, segments, fonts, and images.



You can also turn the scope specific view on or off by right-clicking in the project tree and selecting **Tree>Scope Specific Resource** from the pop up menu.

When a resource is created or inserted in a project, it is assigned a local scope by default. To change the scope of a resource after it has been created, you have to manually cut the resource from its current location and paste in the required folder.

Opening Vitesse Projects in Scope Specific View

To turn on the scope specific view of resources whenever a Vitesse project is opened:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Tree View** category under the **Workspace** node to view the associated options.
- 3 Select the **Scope specific resource view** option.
- 4 Click **OK**.

Setting Default Folder Preferences

When you import a VIPP project in the Vitesse environment, the resources referenced within the VIPP project are also imported. However, the VIPP source code includes only the resource name(s) of the referenced resource(s) and does not provide the full path where the resource is located. Therefore, prior to importing a VIPP project in Vitesse, you need to define a list of directories/folders for the required resources so that Vitesse locates the resources referenced in the imported project at the specified paths. These directories are also used as default paths for the resources.

The **Folder** category in the *Preferences* dialog provides options for specifying the directories. This category consists of the different resource types that may have been used in an imported VIPP project. The resource types are identified by their respective sub-categories under the **Folder** node in the *Preferences* dialog and explained below:

- **Form** – lists the paths of the form resources used in a VIPP project. VIPP projects use VIPP forms with the *.frm extension.
- **Segment** – lists the paths of the segment resources used in a VIPP project. VIPP projects use VIPP segments with the *.seg extension.
- **Image** – lists the paths of the image resources used in a VIPP project.



For more information on forms, segments, and images, refer to *Section 2: Using VIPP Resources* in this guide.

- **JDT** – lists the paths of the JDT resources used in a VIPP project. A JDT (Job Descriptor Ticket) resource is a file that controls the page composition of a print job in a VIPP project. These files have *.jdt as their extension.
- **Project** – lists the paths of the VIPP project files. A VIPP Project File contains all of the information needed for a single project and is saved as a *.vpf file.
- **Encoding** – lists the paths of font encoding files used in a VIPP project.
- **Misc** – lists the paths of any miscellaneous resources used in a VIPP project other than those mentioned above.

For each resource type, you can add new folders, delete existing folders, and reorder the folders. The process for managing the folders is exactly the same for all of the resource types, as mentioned in the following tasks.

Adding a Directory

To add a new directory for a resource:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the category for the required resource under the **Folders** node. The associated directory list displays the list of available folders.
- 3 Click  to add a row for specifying a new directory in the list.
- 4 Click  in the newly added row to open the *Browse for Folder* dialog.
- 5 In the *Browse for Folder* dialog, browse and select the required folder.
- 6 Click **OK** to return to the *Preferences* dialog.
The full path of the specified folder displays in the newly added row.
- 7 Click **OK** to add the specified folder to the list.

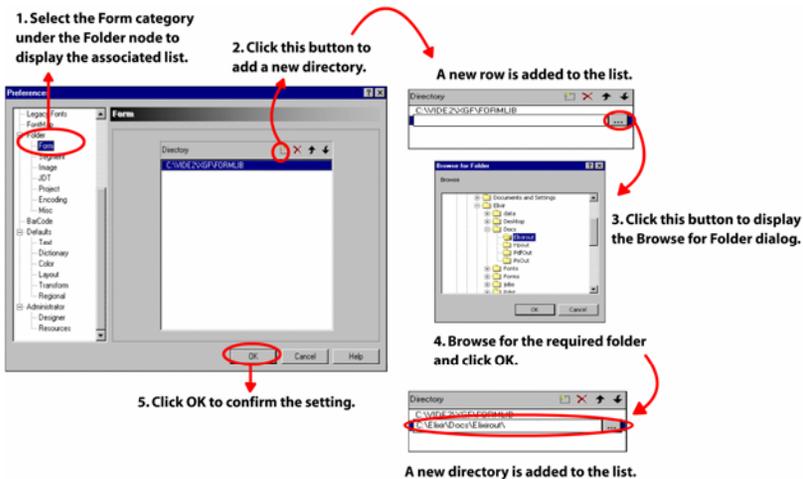


Image showing the steps involved in adding a new directory to the Form category. Adding a directory to the remaining categories will follow the same procedure.

Deleting a Directory

To delete a selected directory from the list:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the required resource category under the **Folders** node.
- 3 Select the required directory, and then click .

Rearranging the Directory List

You can also reorder the directory list for a resource type. Vitesse looks for the required resources starting from the first location specified in the list. If the resource is not found, it moves down to the next location in the list, and so on. We recommend keeping the folders that are more likely to contain the required resources at the top of the list. This will help in locating the resources quickly.

To reorder the list of directories:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the required resource category under the **Folders** node.
- 3 Select the directory to be moved.
- 4 As appropriate, click  to move the selected folder up one level in the list or click  to move the selected folder down one level in the list.
- 5 Click **OK**.

User Notes:

Section II: Using VIPP Resources

User Notes:

Chapter 4: Working with Forms

This chapter provides information on the form resource and explains tasks for:

- Using a Form
- Setting Form Properties
- Caching Resources

Using a Form

A form is a single page resource that is designed using static objects such as images, segments, composed text data, charts, barcodes, tables, lines, paths, polygons, and boxes. A form is used as a background overlay in different applications. For variable data documents, the overlay is merged with the variable data for the final output. You can insert forms to the **Front** and **Back** sides of a **Paper** or to **Page** objects, depending upon the design requirements. Vitesse enables you to create forms in the Elixir Vitesse Format (*.evf). You can insert linked, embedded, inline, and referenced forms in Vitesse, and set or change the form properties. You can also import VIPP Forms (*.frm) to a Vitesse project.



Refer to *Chapter 31: Using Forms in Projects* for details on inserting forms in designs. For more information on importing VIPP forms, refer to *Chapter 35: Importing and Exporting Projects*.

Creating a New Form

To create a new form:

- 1 Select File>New from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl>+<N> on the keyboard.

- 2 From the **New** tab, select the **Form** icon  , and then click **New**.
The *Form* dialog is displayed with the **General** tab selected.

- 3 In the **General** tab, set the following form specifications under the **Project** area:
 - **Name** – specifies a unique name identifying the form. The name can be 30 characters in length and can include alphanumeric characters.
 - **Path** – displays the path of the default form folder. By default, the form is saved in `drive:\Program Files\Elixir Technologies\Vitesse\FormLib\`. You can also click  to browse and select the required location using the *Browse for Folder* dialog.

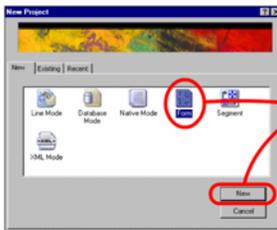


Client directories contain applications created in Native, Line, Database, and XML modes. As forms are stand-alone resources used as backgrounds in different designs, they are not associated with client directories. Therefore, the **Client** drop-down list remains disabled. For more information on client directories, refer to *Chapter 03: Understanding the Vitesse Directory Structure*.

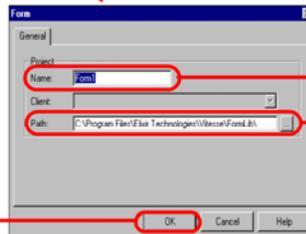
- 4 Click **OK**.

The new form resource is created and the corresponding form node is created under the **Resources>Forms** node in the project tree.

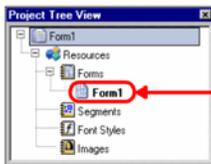
1. Click  on the Standard toolbar to open the New Project dialog.



2. Select **Form** and click **New** to open the Form dialog.



3. Specify a form name.
4. Specify the path where you want to save the form.

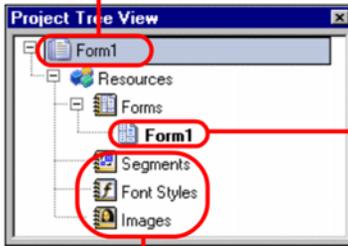


5. Click **OK** to create the new form resource.

Image showing the steps for creating a new form.

The root node in the project tree represents the project title, which in this case, is the form name itself. The project title node identifies the form as a stand-alone resource or project. No properties are associated with this node.

Form project node.



The newly created form resource. References to the resources such as fonts, segments, and images are added under this node.

Segment, Fonts, and Image resources inserted in the form are added under these nodes.

The project tree displaying the newly created form node.

You can design the newly created form using the drawing tools and other objects. Segments, fonts/font styles, and images can also be added to a form.

Setting Form Properties

Once a form resource is created, you can set the associated form properties. When setting the form properties, you should keep in view the final project design which the form will be used. For example, if the form is to be inserted on a page with the A4 standard size, then the recommended form size is A4 as well.

Setting the Form Size

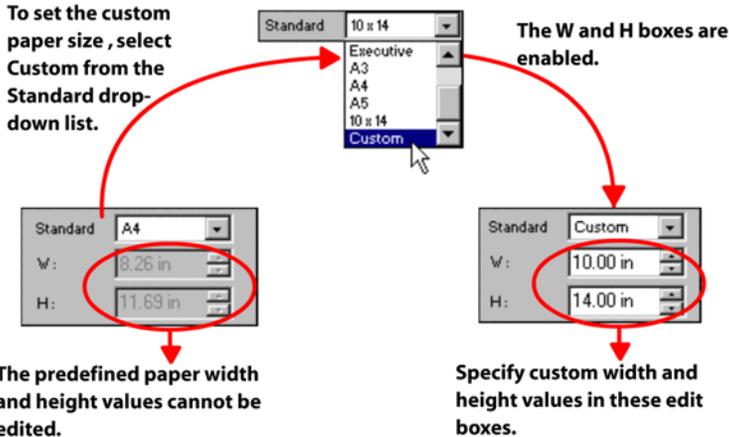
To set the form size:

- 1 Select the required form node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 Select the required form size option from the **Standard** drop-down list.
The **W** and **H** entry boxes display the width and height of the form resource.
- 4 To set a form size other than the predefined values, select the **Custom** option from the **Standard** drop-down list. This option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the form size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting Units> *unit name* from the pop-up menu.

- 5 Enter the required custom form width and height, and then click **OK**.
The specified size is applied to the form.



The image shows the process of setting the custom form size.

Setting the Form Orientation

Form orientation defines the layout position of a form for printing. Vitesse supports four orientation options discussed below.

To set the form orientation:

- 1 Select the required form node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Orientation** drop-down list, select one of the following orientation options:
 - **Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - **Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
 - **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.
- 4 Click **OK**.

The selected form displays with the specified orientation.



The two settings, **Inverse Landscape** and **Inverse Portrait**, do not reflect in the design area until you insert the form in a page. For information on inserting forms in a design, refer to *Chapter 31: Using Forms in Projects*.

Specifying Form Details

You can view or specify the form identification information including the form title, short description about the form, form creator and generator name, copyright details, version number, creation and modification dates in the *Property View* window.



The information that you specify in the **Properties** tab is displayed as comments in the generated VIPP project source code.

To specify the form details:

- 1 Select the required form node from the project tree.
- 2 From the *Property View* window, select the **Properties** tab. This tab contains the project information that is organized as a **Key** (item) and **Value** pair.
- 3 Double-click any cell under the **Key** and **Value** columns to enter the form details.



You cannot modify the **Copyright** and **Version** information.

- 4 To add a new key-value pair above or below a selected row, click  in the **Properties** tab.
- 5 Enter the additional information as required in the newly added key and value pair.



You can also delete a selected key-value pair by selecting the required row, and then clicking  in the **Properties** tab.

1. Click  to add a new key-value pair.

2. Double-click the newly added row to enter information.

3. The new information is reflected in the source code.

Properties

Key	Value
Title	New Vitesse Project
Description	Simple VIPP Project
Creator	Vitesse
Generator	Vitesse
Keywords	bill, statement
Creation Date	10:30:34 Thursday, Ja...
Copyright	Copyright (C) 2003 Eli...
Version	Mobilkom.RC8
Last Modified	
<Key10>	<Value>

Source View

```

%%!
%%Title: New Vitesse Project
%%Description: Simple VIPP Project
%%Creator: Vitesse
%%Generator: Vitesse
%%Keywords: bill, statement
%%Creation Date: 10:30:34 Thursday, January 26, 2006
%%Copyright: Copyright (C) 2003 Elzar Technologies Corp
%%Version: Mobilkom.RC8
%%Last Modified:
%%EndComments

XGF
DOT3 SETUNIT
[/FDecimalPoint 44 /FDigit 35 /FLZDigit 64 /FNSign 45
 /FPSign 43 /FPunctuation 46 /DecimalPoint 44 /FNSign 45
] SETPARAMS
  
```

The image shows the process of specifying form details.

Caching Resources

In a typical printing process, the resources used in a document are loaded to the printer memory, and then printed. In some cases, a document may include heavy resources that are used repetitively. Loading such heavy resources to the printer memory each time they occur in the project design makes the printing process cumbersome.

To overcome this issue and maintain the efficiency of the printing process, Vitesse offers the resource-caching feature. This feature enables you to store and retrieve certain resources from the printer cache. Once a resource is available in the cache, the printer does not load it from the system memory. This eliminates the surplus loading time and therefore, makes the printing process efficient. Vitesse allows you to specify the caching option explicitly for the form, segment, and image resources.



We recommend using the cache option only for the frequently used heavy resources. Caching too many resources unnecessarily may overload the printer cache.

Caching a Form

To enable resource caching for a form resource, select the **Cache** option from the **General** tab of the *Property View* window. Selecting this option does not have any effect on the interface. It only enables you to load and retrieve forms from the printer cache while printing.

User Notes:

Chapter 5: Working with Segments

This chapter provides information on using the segment resource and provides tasks for:

- Creating a New Segment
- Setting Segment Properties
- Caching a Segment
- Dynamic Behavior of Segments
- Working with Postscript Segments

Using a Segment

A segment is a VIPP Native mode or PostScript resource that can be reused in different locations in a document and/or a form. Vitesse enables you to create segment files in the Elixir Vitesse Segment (*.evs) format. A Vitesse segment is designed using static objects such as images, composed text data, charts, and barcodes. Drawing objects, such as lines, paths, polygons, and boxes are also used.

You can import PostScript forms, Encapsulated PostScript files (EPS), or PS format files as segments in Vitesse, using the **Insert Segment** option from the menu bar or the *Insert* toolbar. These files can be inserted as segments, but cannot be edited in Vitesse (*.ps, *.eps, *.prn). Vitesse also lets you import VIPP Segments (*.seg) in the Vitesse design. A segment can be referenced several times in a page.



One difference between forms and segments is that segments can be rotated and resized within a form or a document.

You can insert linked, embedded, inline, referenced, begin page, end page, and repeat segments in Vitesse designs, and set or change the segment properties.



Refer to *Chapter 32: Using Segments in Projects* for details on inserting segments in designs. For more information on importing VIPP segments, refer to *Chapter 35: Importing and Exporting Projects*.

Creating a New Segment

To create a new segment:

- 1 Select File>New from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl>+<N> on the keyboard.

- 2 From the **New** tab, select the **Segment** icon , and then click **New**.

The *Segment* dialog is displayed with the **General** tab selected.

- 3 In the **General** tab, set the following form specifications under the **Project** area:

- **Name** – specifies a unique name identifying the segment. The name can be 30 characters in length and can include alphanumeric characters.
- **Path** – displays the path of the default form folder. By default, the segment is saved in *drive:\Program Files\Elixir Technologies\Vitesse\SegLib*. You can also click  to browse and select the required location using the *Browse for Folder* dialog.



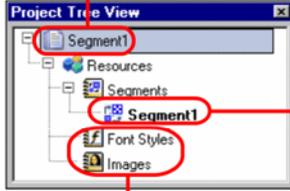
Client directories contain applications created in Native, Line, Database, and XML modes. As segments are stand-alone Vitesse resources used in page designs, they are not associated with client directories. Therefore, the **Client** drop-down list remains disabled. For more information on client directories, refer to *Chapter 03: Understanding the Vitesse Directory Structure*.

- 4 Click **OK**.

The new segment resource is created and a corresponding segment node is created under the **Resources>Segments** node in the project tree.

The root node in the project tree represents the project title, which in this case is the segment name itself. The project title node identifies the segment as a stand-alone resource or project. No properties are associated with this node.

Segment project title node.



The design objects and references to resources in the segment are added under this node.

Fonts and images included in the segment are added to the respective resource nodes.

The project tree displaying the newly created segment node.

You can design the newly created segment using the drawing tools and other objects. Fonts/font styles and images can also be added to a segment.

Setting Segment Properties

Once a segment resource is created, you can set the associated segment properties. The specified properties reflect in the project design whenever the segment is referenced.



PostScript segments cannot be edited in the Vitesse environment and therefore, the properties associated with them are different, as explained in “Working with PostScript Segments” section later in this chapter.

Setting the Segment Size

To set the segment size:

- 1 Select the required segment node under the **Resources>Segments** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Size** area, enter the required segment width and height in the **W** and **H** entry boxes.

The specified size is applied to the segment.



You can also set the unit of measurement for the segment size by right clicking in the **W** and **H** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

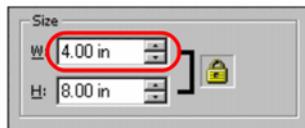
To increase or decrease the segment size in equal proportion, click  in the **Size** area. The following image shows how to resize a segment in equal proportion.

1. Clicking  specifies to maintain an aspect ratio of 1:2 (3":6") for the segment.



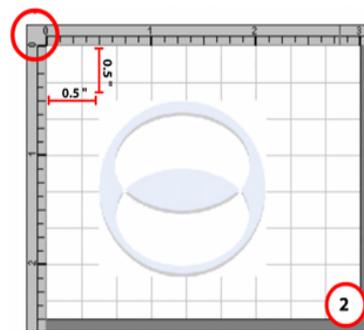
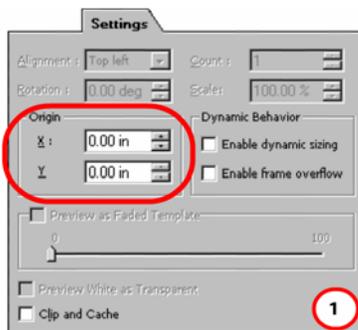
2. Specify the required width value (e.g. 4").

3. The height adjusts to 8" maintaining the aspect ratio of 1:2 (4":8").



Setting the Segment Origin

Objects are positioned in the design area with respect to an origin. The point of origin is defined by the two coordinates (x, y). In the Vitesse design environment, origin is the top left corner of the design area. The value is set to (0, 0) by default for all project designs.



1) Property View window - Settings tab for a segment resource specifying the origin set to (0, 0) by default, 2) Design area ruler displaying the origin set to (0, 0) and an image placed at (0.5, 0.5) w.r.t. the origin.

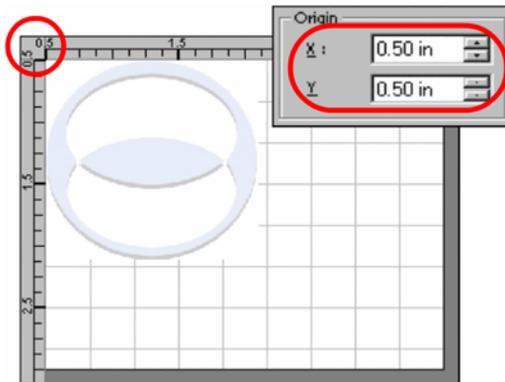
Vitesse enables you to specify a different point of origin when designing segments. To set the origin for a segment:

- 1 Select the required segment resource under the **Resources>Segments** node in the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Origin** area, specify the following:
 - **X** – specifies the horizontal coordinate of the origin, per specified unit of measurement.
 - **Y** – specifies the vertical coordinate of the origin, per specified unit of measurement.



You can also set the unit of measurement for the segment origin by right clicking in the **X** and **Y** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The values set for the x and y coordinates collectively specify the origin of the segment. The positioning of the design objects inserted in the segment corresponds to the specified origin. The following figure shows the origin of a segment set to (0.5, 0.5) and an image resource placed at (0.5, 0.5) with respect to the new origin:



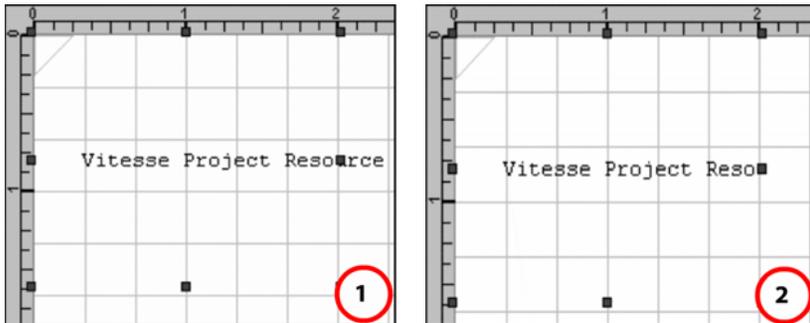
Clipping and Caching a Segment

To enable resource caching for a segment resource, select the **Clip and Cache** option from the **Settings** tab of the *Property View* window.



For more information on caching resources, refer to the “Caching Resources” section in *Chapter 4: Working with Forms*.

The **Clip and Cache** setting is effective for segment designs that exceed the page borders. A segment that contains an object or a resource that falls outside the segment boundary is clipped to display only the contents within the segment borders. Therefore, by means of clipping, only the design contents within the segment boundary are cached to the printer memory.



Design area displaying a project: 1) Using a segment with the Clip and Cache option not selected, 2) Using a clipped segment.

Specifying Segment Details

You can view or specify the segment identification information including the segment title, short description about the segment, segment creator and generator name, copyright details, version number, creation, and modification dates in the *Property View* window.



The information that you specify in the **Properties** tab is displayed as comments in the generated VIPP project source code.

To specify the segment details:

- 1 Select the required segment node under the **Segments** node from the project tree.
- 2 From the *Property View* window, select the **Properties** tab. This tab contains the project information that is organized as a **Key** (item) and **Value** pair.
- 3 Double-click any cell under the **Key** and **Value** columns to enter the form details.



You cannot modify the **Copyright** and **Version** information.

- 4 To add a new key-value pair above or below a selected row, click  in the **Properties** tab.
- 5 Enter the additional information as required in the newly added key and value pair.



You can also delete a selected key-value pair by selecting the required row, and then clicking  in the **Properties** tab.

Dynamic Behavior of Segments

Vitesse enables you to use pre-defined fields, charts, and conditions in segments. It also lets you dynamically size the segments according to the variations in a segment design. To dynamically size the segments, you need to specify the associated options available in the **Settings** tab under the **Dynamic Behavior** area. This feature takes effect only when these settings are also specified for the segment referenced in a project design. The process of using the dynamic behavior of segments is explained in *Chapter 32: Referencing Segments*.



Dynamic behavior of segments is not applicable to Line mode projects.



Enable frame overflow option is discussed in *Chapter 25: Using Frames*.

Working with Postscript Segments

VIPP is a PostScript (PS) based Printer Definition Language (PDL). Vitesse enables you to use PostScript (*.ps), Encapsulated PostScript (*.eps), and print files (*.prn) as segments in design projects. You can use document processing applications (such as Word, FrameMaker, Excel, and PageMaker) to create/edit a PostScript resource file. You can use these files in a Vitesse project design. However, editing of PostScript segments is not allowed in Vitesse.

To insert a PostScript segment:

- 1 From the *Project Tree View* window, select the **Segments** node under the **Resources** node.
- 2 Select **Insert>Segment** from the menu bar. The *Insert Segment* dialog is displayed with the **New** tab selected.



You can also open the *Insert Image* dialog either by:

- Clicking  on the *Insert* toolbar.
- Or
- Right-clicking the **Segments** node and selecting **Insert>Segment** from the pop up menu.

- 3 Select the **Existing** tab.
- 4 Browse to the required location and select the required segment resource.
- 5 Click **Open** to insert the selected segment as a resource in the project.

A segment node is created under the **Resources>Segments** node identifying the newly added segment resource. The node name is identical to the actual segment file name and also displays the associated extension.



The **General** and **Properties** tabs are not available for the PostScript segment resources in the *Property View* window as you cannot edit these resources in the Vitesse environment.

Viewing Postscript Segment as a Faded Template

Vitesse enables you to lighten the segments that are added to a project. This feature is useful when you need to use a segment resource as a background on a part of your page. The fading effect is applied to the segment resource only in the design and does not affect the original file.

To view a postscript segment as faded:

- 1 Select the required segment resource under **Resources>Segments** to view the associated properties in the *Property View* window.
- 2 From the **Settings** tab, select the **Preview as Faded Template** option; this enables the slider.

- 3 Move the **Preview as Faded Template** slider to the required shade value (between 0 % and 100 %). As you move the slider, corresponding fade behavior is reflected in the design area.

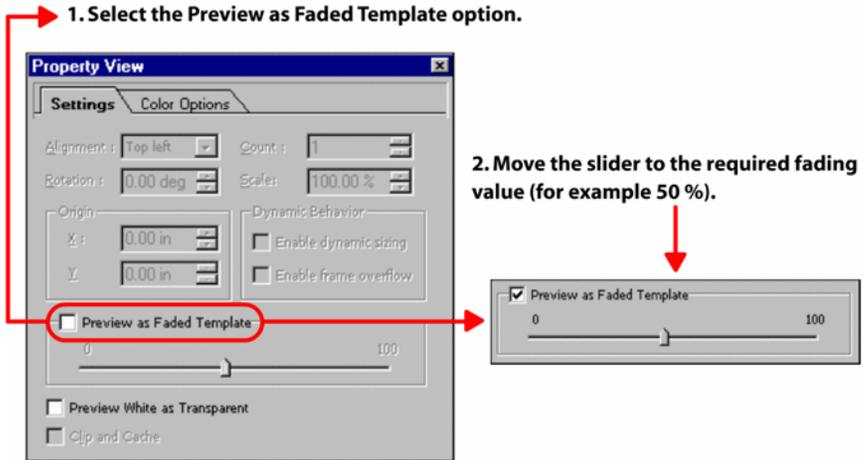
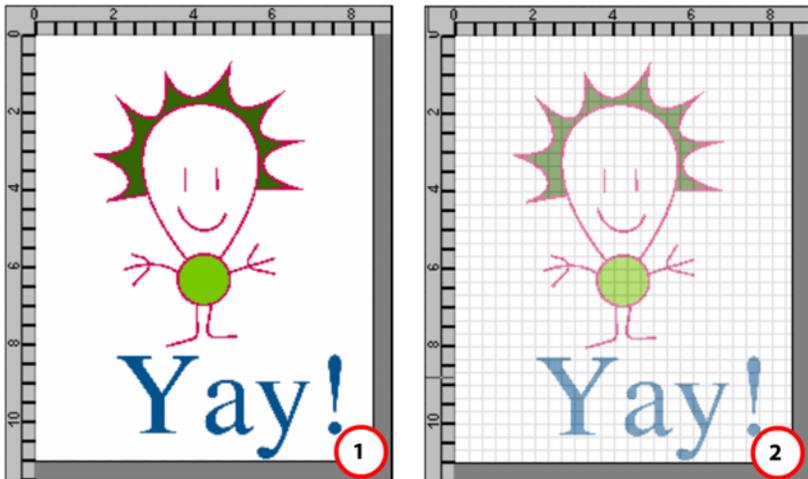


Image shows the steps to set the Preview as Faded Template option.

The segment lightens according to the specified adjustment as shown below:



Design area displaying the PostScript segment resource: 1) Original, 2) Faded to nearly 50 %.

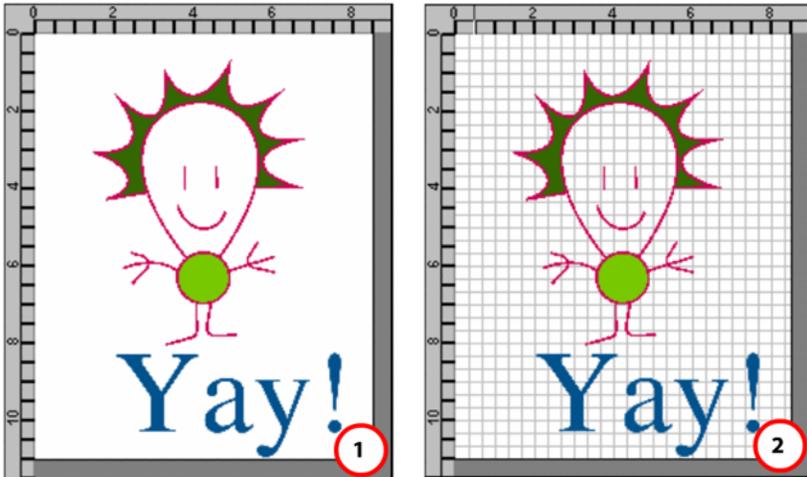
Setting Transparency of White Areas in a PostScript Segment

If a segment is referenced in the design such that it overlaps an existing object, you can set the overlapping segment area as transparent so that the overlapped object becomes visible. This feature is applicable only to segments that have a white background.

The white area of the segment is set as transparent such that the overlapped object is visible through the white portion of the segment.

To view the white part of a segment as transparent:

- 1 Select the required PostScript segment node under the **Resources** node.
- 2 From the **Settings** tab of the *Property View* window, select the **Preview White as Transparent** option.



Design area displaying: 1) The segment resource, 2) The grid lines visible through the white part of the segment only using the Preview White as Transparent option.



You can also achieve the same effect by arranging the objects according to the design requirements. For more information, refer to the “Arranging Objects” section in *Chapter 8: Using the Drawing Tools*.

Chapter 6: Working with Fonts

This chapter provides information on the different font styles available in Vitesse. It gives a detailed overview on the font types, mappings, and their significance in VIPP printing. Tasks include:

- Font Types Supported in Vitesse
- Managing Font Styles in Vitesse
- Using Legacy Fonts
- Mapping a Legacy Font to a VIPP Font
- Mapping VIPP Fonts to True Type or PostScript Fonts in Vitesse
- Setting a Default Font
- Setting Font List Viewing Options

Font Types Supported in Vitesse

Vitesse supports VIPP, PostScript, and TrueType fonts that you can use while composing documents. However, when using the PostScript or TrueType fonts in a project that is to be printed on a VIPP printer, you must specify their VIPP font substitutes.

The following are the explanations for the font types supported in Vitesse:

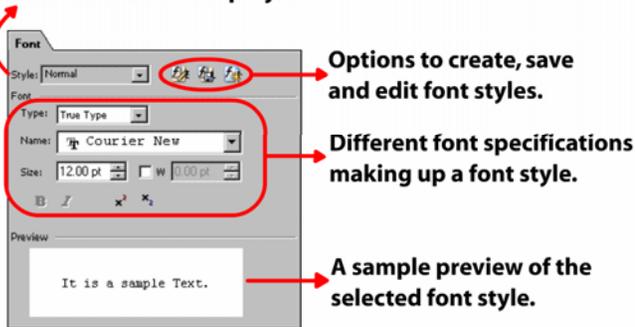
- **PostScript Fonts** – Vitesse uses the PostScript fonts that are installed on your system. These fonts can be applied to your design and can be exported as resources later.
- **True Type Fonts** – TrueType fonts can be applied to the design. You can convert these fonts as PostScript fonts and package them as resources during the export process.
- **VIPP Fonts** – VIPP fonts are references to PostScript fonts, and are pre-mapped to PostScript fonts on the printer controller that runs VIPP. These are virtual fonts and therefore, need to be mapped to a True Type or PostScript Type 1 font for rendering in the Vitesse designer.

For final printing, all supported fonts are substituted by VIPP font names using the **Font Map** (described in the “Mapping VIPP Fonts to True Type or PostScript Fonts in Vitesse” section later in the chapter). The font names are then matched with the VIPP fonts available on the destination printer where the final print output is composed. Vitesse also enables you to convert the True Type fonts to the PostScript format and export them as a project resource. For more information on exporting fonts as a project resource, refer to *Chapter 35: Exporting and Importing Projects*.

Managing Font Styles in Vitesse

A font style is a Vitesse resource created by setting different font specifications (e.g. type, name, size, etc.). The font styles in a Vitesse project are represented by the **Font Styles** node under the **Resources** node in the *Project Tree View* window. The **Font Styles** node does not have any sub-nodes. When it is selected in the project tree, the options associated with it are displayed in the *Property View* window. Using these options, you can create different font styles for a project, and then apply the required styles to the page, text, graphic text, field, and/or chart objects. Tasks for creating, modifying, and saving a font style are discussed below.

Drop-down list of font styles defined for the current project.



The Font options associated with the Font Styles node.

Creating a New Font Style

Vitesse enables you to create and save font styles for applying to the page, text, graphic text, chart, and field objects used in a project design. The font styles that you create in a certain project are available for use only in that project.



Four font styles are available for use in all newly created projects by default. These are **Normal**, **PNFT**, **Style1**, and **Style2**. You cannot edit the pre-defined **Normal** style.

To create a new font style:

- 1 Select the **Font Styles** node under the **Resources** node in the project tree. The **Font** tab is displayed in the *Property View* window.
- 2 In the **Font** tab, click  to create a new font style.
- 3 In the **Style** entry box, enter the required name for the newly created font style.
- 4 Select the required font **Type** from the following available options:
 - **True Type** – sets True Type as the font type.
 - **PostScript** – sets PostScript as the font type.
 - **VIPP** – sets VIPP as the font type.

Depending on the selected font type, a list of corresponding fonts display in the **Name** drop-down list.

- 5 Select the required font from the **Name** drop-down list.



The **Name** drop-down list displays a preview for each selected True Type or PostScript font. To include the font name with the preview, select **Show font names in font combo box** in the **General** category of the *Preferences* dialog.

- 6 Specify the required font size in the **Size** entry box.
- 7 To restrict the font size to a specific width value, select the **W** option adjacent to the **Size** entry box.
- 8 Specify the required font width in the entry box associated with the **W** check box.



You can also set the unit of measurement for the font size and width by right clicking the upward or downward arrows in the **Size** and **W** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

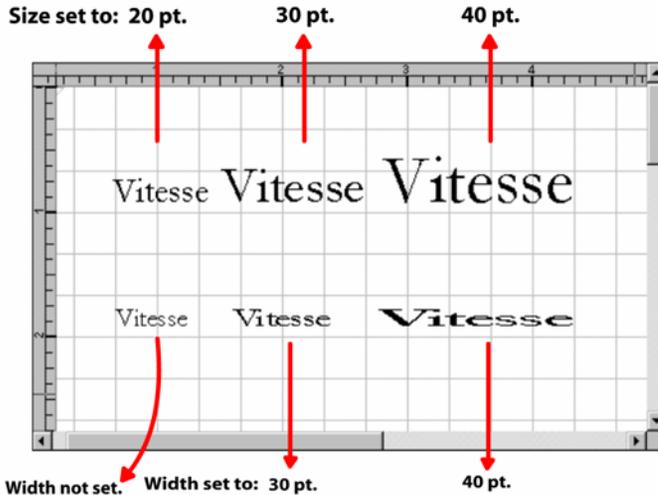


Image showing a font with different sizes and widths.

- 9 Click any one of the following to specify if the font style is to be used as a superscript or a subscript:
- **B** – applies the bold style to a font.
 - **I** – applies the italic style to a font.
 - **U** – applies the underline style to a font.

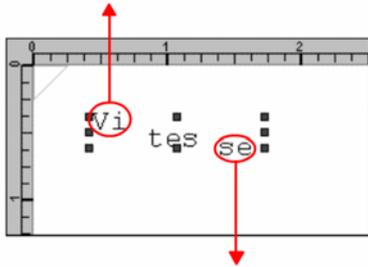


By selecting the **Show font family members in font combo box** option in the **General** category of the *Preferences* dialog, Vitesse displays the regular, bold, and italicized styles for each font in the **Name** drop-down list. The **B**, **I**, and **U** features are not available when this option is selected. The availability of these features is also dependent on the type of font selected.

-  – applies the superscript style to a font.
-  – applies the subscript style to a font.

Text entered using either of the superscript or subscript style will display as a superscript or subscript respectively.

Superscript style



Subscript style

Image shows the superscript and subscript settings applied to a text object.



The **Preview** area provides a sample preview of the specified font style settings.

- 10 Click  to save the font settings as a new font style.

You can apply the newly created font style on the required page, text, graphic text, field, and/or chart object using the **Font** tab in the *Property View* window associated with the object. You can also create, edit, and save the font styles using options associated with these objects. The font styles defined for a project, but not applied anywhere, will not be retained after the project is closed.

Editing a Font Style

To edit an existing font style, select the font style from the **Style** drop-down list, and then click . Repeat **Steps 3–10** in “Creating a New Font Style” task to make your changes.

Using Legacy Fonts

Vitesse enables you to import VIPP projects and Xerox native resources (FRM and FSL) containing legacy fonts (for example, FNT and Elixir fonts). However, VIPP printers require PostScript fonts for printing. Vitesse, therefore, allowing you to map legacy fonts to VIPP fonts to make the Xerox formats portable to the VIPP environment. On import, a legacy font used in a design is substituted by the VIPP font mapped to it. The mapping is defined using the **Legacy Font Mapper**.

The **Legacy Font Mapper** is displayed in the **Legacy Fonts** category of the *Preferences* dialog. Vitesse installs some common VIPP font names that are already matched with the legacy fonts. It also enables you to define new font mappings, and delete or edit the already defined mappings to meet your font needs.

There are four major columns in the Legacy Font Mapper including: **Legacy**, **VIPP**, **Height**, and **Use Height**.

Mapping a Legacy Font to a VIPP Font

To add a new mapping to the Legacy Font Mapper:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also display the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

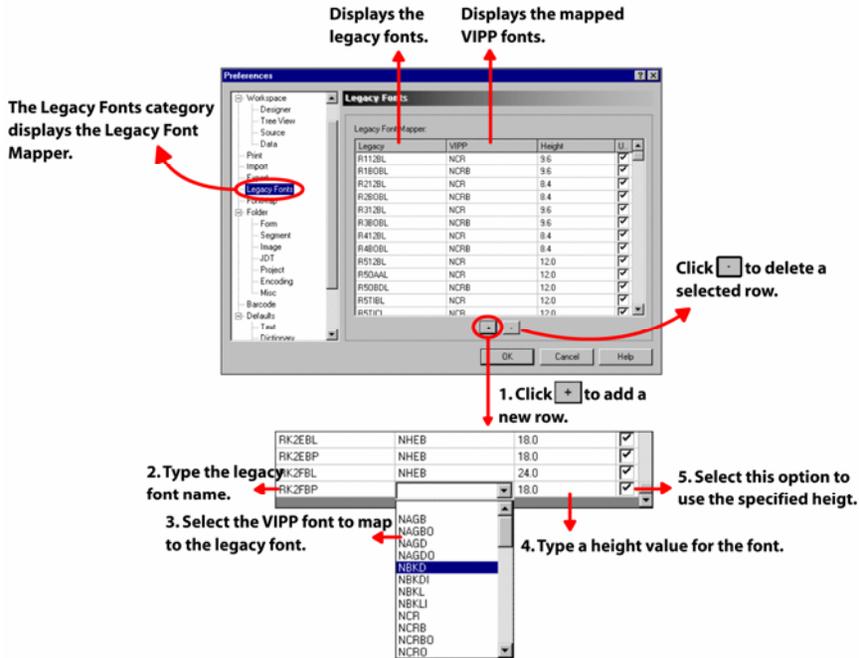
- 2 In the *Preferences* dialog, select the **Legacy Fonts** category to view the **Legacy Font Mapper**.
- 3 Click  to add a new row at the bottom of the table to define a new mapping.
- 4 In the **Legacy** column, double-click the associated cell and enter the name of the legacy font in place of which you want to use another font.
- 5 In the **VIPP** column, select the associated cell and select a VIPP font from the drop-down list to map to the legacy font specified in Step 4.
- 6 In the **Height** column, double-click the associated cell and specify the required height for the selected VIPP font.
- 7 Select the **Use Height** column cell to apply the specified height (above) to the VIPP font. Otherwise, the default font height is applied.



Specifying the correct height while mapping the legacy fonts is important, as height is critical in legacy fonts. In order to render fonts correctly, you must specify the VIPP font height, which must be the same as that of the legacy font.

- 8 Click **OK** to save changes to the newly defined mapping.

The following image displays the different components of the Legacy Font Mapper. It also illustrates the steps involved in defining a new font mapping.

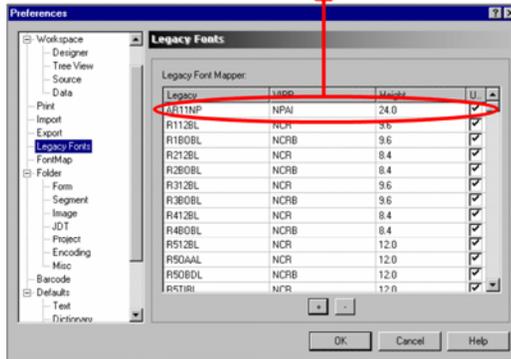


The font mappings already defined can be edited by double clicking the required cell. You can also delete a mapping by selecting a font map entry and clicking [X].

Viewing Mapped Fonts in Vitesse

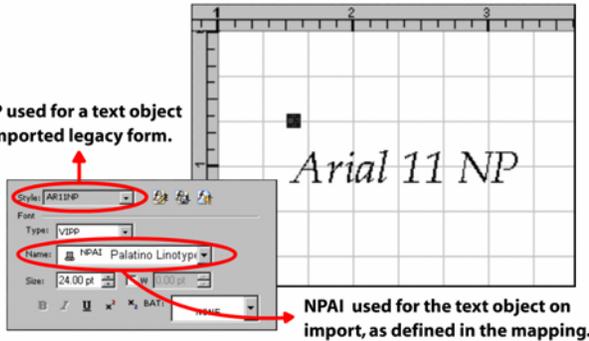
When you import a VIPP project containing objects that have legacy fonts applied to them, the mapping you define using the **Legacy Font Mapper** takes effect such that the legacy fonts used in the imported project are replaced by the specified VIPP fonts. The image below shows a mapping defined for a legacy font. It also displays a form (using the mapped legacy font) imported to Vitesse.

A legacy font 'AR11NP' mapped to a VIPP font 'NPAI'.



Design area displaying a legacy form (FRM) imported to Vitesse (EVF).

AR11NP used for a text object in the imported legacy form.



NPAI used for the text object on import, as defined in the mapping.

Mapping VIPP Fonts to True Type or PostScript Fonts in Vitesse

When designing projects or resources in Vitesse, you can use either True Type or PostScript Type 1 fonts. These fonts are used only for viewing and therefore, are referred to as rendered fonts. When the project or resource is complete, it is exported to the VIPP format for printing. The export process uses a **Font Map** to equate the rendered font used in design with a VIPP font name. The printer then uses the VIPP font name to locate a printer (PostScript) font for printing.

For example, if you use the **True Type** font **Courier New** in your project, the project will reference the **VIPP** font **NCR** when exported for printing.



You can also explicitly convert True Type fonts to VIPP formats during the export process instead of mapping them to VIPP fonts. For more information, refer to *Chapter 35: Importing and Exporting Projects*.

The **Font Map** available in the **FontMap** category of the *Preferences* dialog is used to associate rendered fonts with printer fonts. There are five major columns in the **Font Map**, namely: VIPP, PostScript, True Type, B(bold), and I(italic). Vitesse installs the most common VIPP font names that are already matched with their True Type equivalent in the **Font Map**. The **Font Map** is editable, enabling you to change the default mappings if necessary to meet with your font requirements.

You can install PostScript fonts on your operating system and add them to the **Font Map** so that Vitesse uses the same font for rendering as used by the printer. If both True Type and PostScript fonts are available, Vitesse will use the PostScript fonts for displaying.

To add a new font mapping:

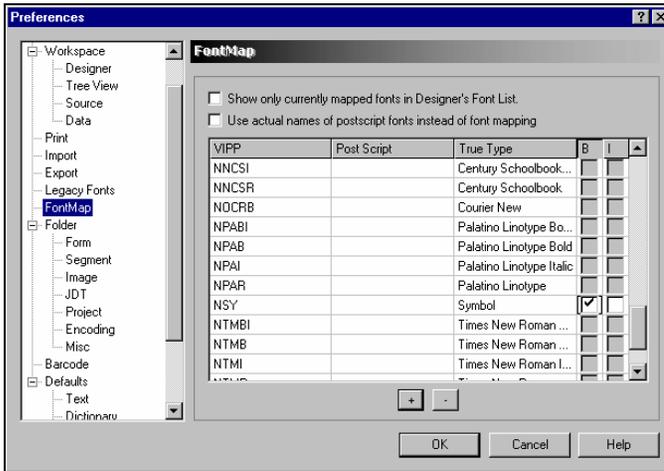
- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also display the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 In the *Preferences* dialog, select the **FontMap** category to view a list of the already defined font mappings.
- 3 Click  to add a new row at the bottom of the list.
- 4 Double-click the new empty cell in the **VIPP** column and enter a VIPP font name.
- 5 Double-click the cell in the **PostScript** column and select a font from the drop-down list to associate the VIPP font to a PostScript font. The drop-down list only includes the PostScript fonts that have already been installed on your operating system.
- 6 Double-click the cell in the **True Type** column and select a font from the drop-down list to associate the VIPP font to a True Type font.
- 7 If the True Type font that you have selected is not already bold or italicized, the **B** and **I** check boxes become active and you can set the font to bold and/or italic.
- 8 Click **OK**. The default font mappings are used the next time you use them in your project.



Preferences dialog with the FontMap category and default font mappings.

Setting a Default Font

Whenever you create a new text, graphic text, chart, or field object, a default font is applied to the text/data that it contains. You can specify the most commonly used font(s) in your design as the default font for the application to avoid applying it to the required objects repeatedly. See the image below that explains the steps for specifying a default font.

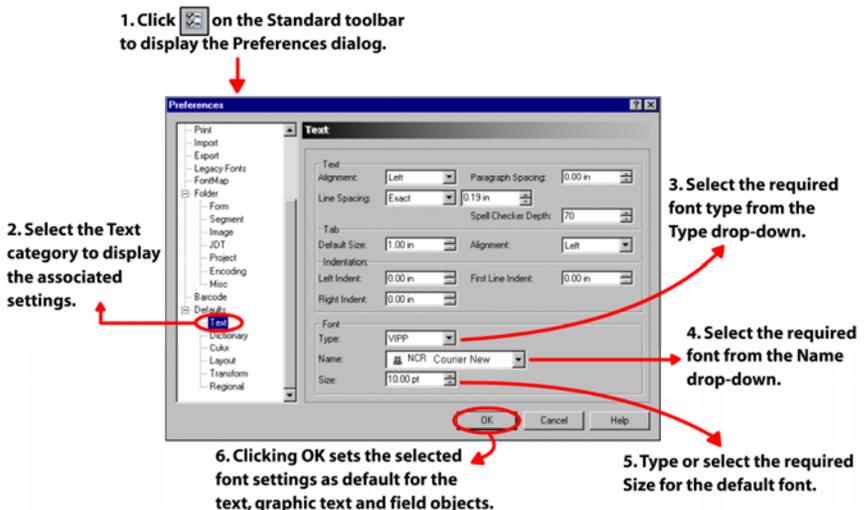


Image shows the steps involved in setting a default font.

Setting Font List Viewing Options

To specify the default font viewing options:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 In the *Preferences* dialog, select the **FontMap**.
- 3 As appropriate, select the required options from the following:
 - **Show only currently mapped fonts in the Designer's Font List** – displays only the mapped fonts available from the **Name** drop-down list in the **Font** tab.
 - **Use actual names of postscript fonts instead of font mapping** – uses the PostScript font name in the source code for a mapped font applied to an object. Not selecting this option specifies to use the VIPP font name.



The font name used in the source code will be used for printing at the print controller.

- 4 Click OK to apply the specified viewing options for the font list.

User Notes:

Chapter 7: Working with Images

This chapter describes how to add images to a project and set image properties. The chapter covers the following tasks:

- Adding an Image
- Changing the Image
- Previewing the Image as a Faded Template
- Caching the Image
- Previewing Images in Low Resolution

Adding an Image

Vitesse enables you to insert images in stand-alone forms and segments resources, and/or applications created in different modes. Images can be inserted in different formats, such as *.bmp, *.jpg, *.jpeg, *.tif, *.pcx, *.png, etc.



GIF and TIFF files with LZW compression are not supported in Vitesse.

Images are always added as resources. To use the images in forms, segments, and/or applications, you need to reference the image in the required project. For more information on referencing images, refer to *Chapter 33: Referencing Images*.

To add an image to a project:

- 1 From the *Project Tree View* window, select the **Images** node under the **Resources** node.
- 2 Select Insert>Image from the menu bar. The *Insert Image* dialog is displayed with the **Existing** tab selected.



You can also open the *Insert Image* dialog either by:

- Clicking  on the *Insert* toolbar.
- Or
- Right-clicking the **Images** node and selecting Insert>Image from the pop up menu.

- 3 Browse to the required location and select the required image.
- 4 Click **Open** to insert the selected image as a resource in the project.
An image node is created under **Resources>Images** node identifying the newly added image resource. The node name indicates the actual image file name and associated extension.



Project Tree View window displaying the newly inserted image resource.

After adding the image, you can reference it in the project design. Referenced images display under the **Page** node or the appropriate form and segment nodes in the project tree.

Setting Image Properties

Once an image resource is created, you can set the associated image properties. The specified image properties reflect in the project design whenever the image is referenced.

Replacing an Image

To replace an image with a new image:

- 1 Select the required image node under the **Images** node from the project tree.
- 2 In the **General** tab of the *Property View* window, click  to display the *Open* dialog. The text already displaying in the **Path** entry box specifies the location of the image resource.
- 3 Search **Look in** for the required target folder.
- 4 Select the image file, and then click **Open**.
- 5 The previously added image is replaced with a new image and the **Path** entry box is updated with the newly specified path.



If the image to be replaced is already being referenced in the project design, it will not be replaced. However, the selected image file is added to the project as a new image resource under the **Images** node. You can delete the reference and then replace the image resource in such situations.

Previewing the Image as a Faded Template

Vitesse enables you to lighten the images added to a project. This can be of use when you would like to use an image as a background in your project design. The fading effect is applied to the image only in the design and does not affect the original image file.

To preview an image as faded:

- 1 Select the required image resource to view its associated properties in the *Property View* window.
- 2 From the **General** tab, select the **Preview as Faded Template** option, which enables the slider.
- 3 Move the **Preview as Faded Template** slider to the required shade value (between 0 % and 100 %). As you move the slider, corresponding fade behavior is reflected in the design area.

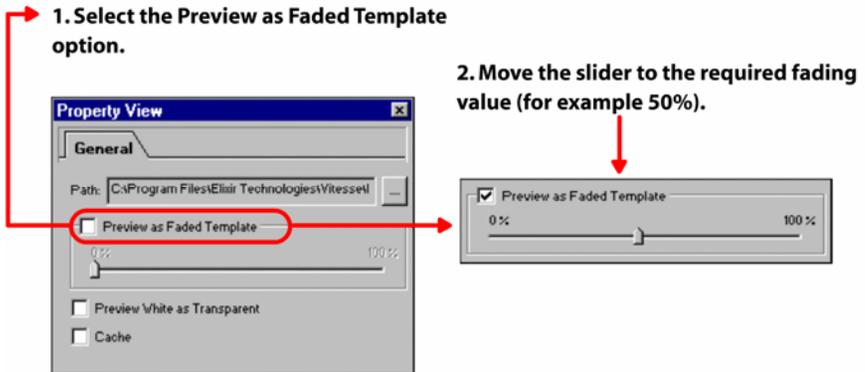
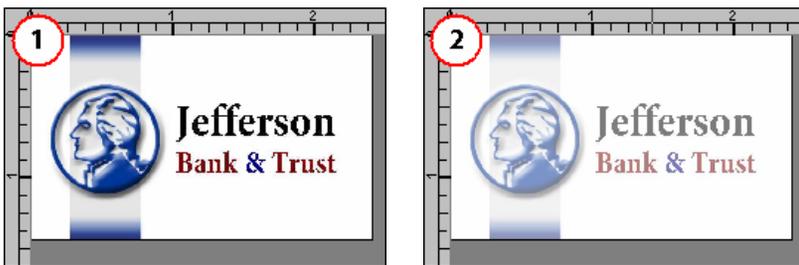


Image shows the steps to set the Preview as Faded Template option.

The image resource lightens according to the specified adjustment as shown below:



Design area displaying the image resource: 1) Original, 2) Faded to nearly 50 %.

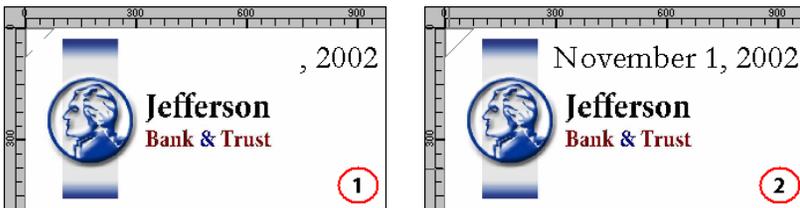
Setting Transparency of White Areas in an Image

If an image is referenced in the design such that it overlaps an existing object, you can set the overlapping image area as transparent so that the overlapped object becomes visible. This feature is applicable only to images that have a white background.

The white area of the image is set as transparent such that the overlapped object is visible through the white portion of the image.

To view the white part of an image as transparent:

- 1 Select the required image node under the **Resources** node.
- 2 From the **General** tab of the *Property View* window, select the **Preview White as Transparent** option.



Design area displaying the image resource used in a design: 1) The date field is overlapped by the image 2) The date field is visible through the image using the Preview White as Transparent option.



This option is specified in the properties associated with the image resource, but it is reflected only when the image is referenced in the project design and overlaps an object. For more information on referencing images, refer to *Chapter 33: Referencing Images*.



You can also achieve the same effect by specifying an arrangement of objects according to the design requirements. For more information, refer to the "Arranging Objects" section in *Chapter 8: Using the Drawing Tools*.

Caching an Image

To enable resource caching for an image, select the **Cache** option from the **General** tab of the *Property View* window. Selecting this option does not have any effect on the image itself. It only enables you to load and retrieve images from the printer cache while printing.



We recommend using the cache option only for the frequently used heavy images. Caching of images makes the printing process efficient.



For more information on caching resources, refer to the “Caching Resources” section in *Chapter 4: Working with Forms*.

Previewing Images in Low Resolution

Vitesse allows you to use high-resolution images in a project design to achieve the required print quality. High-resolution images are heavy resources and can make the previewing and scrolling process inefficient due to the extra time required for loading them in the system memory.

To keep the previewing of projects efficient and smooth even when you are using high resolution images:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.
- 2 Select the **General** category to view the associated options.
- 3 Select the **Use low resolution 96 dpi images for optimized preview in the Designer** option.
- 4 Click **OK**.

By setting this option, Vitesse creates a 96 dpi version of the high resolution images used in project designs. The 96 dpi images are used only for previewing documents; it does not affect the resolution of the final print output.



When you export the project, the image is packaged with the original resolution value.

User Notes:

Section III: Vitesse Project Tools

User Notes:

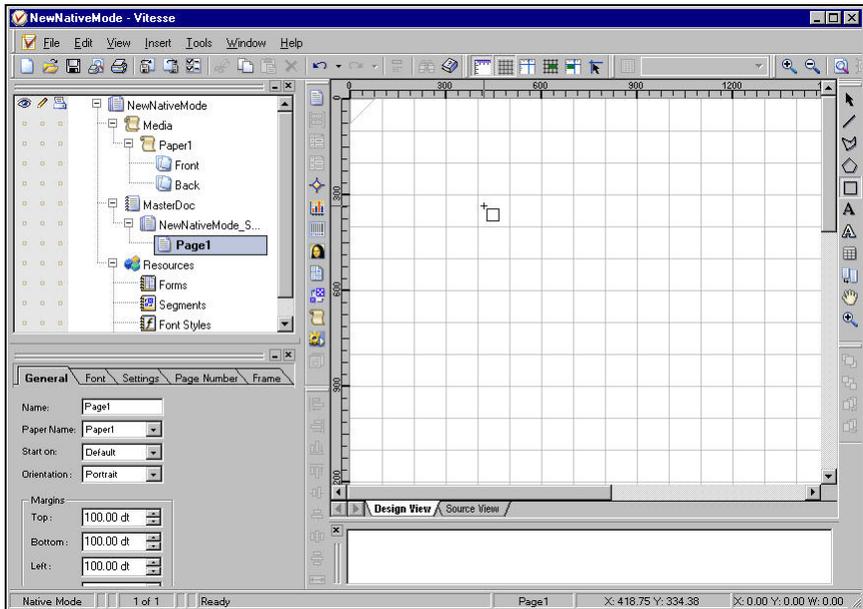
Chapter 8: Using the Drawing Tools

This chapter provides information on drawing tools. Included are instructions for adding drawing objects in the form and setting associated properties. It also describes the basic operations for editing/modifying objects inserted in a design.

- Drawing Objects
- Setting Common Object Properties
- Basic Operations

Drawing Objects

Vitesse provides several object types for drawing shapes such as lines, boxes, polygons, paths, and tables. All drawing operations are performed by clicking and dragging the mouse in the design area. When you click a tool on the *Drawing* toolbar, the mouse pointer changes depending on your selection. You can move the mouse pointer in the design area to draw the selected object in the form. The image below shows the design area with the box tool selected.



Selecting the box tool changes the mouse pointer.

You can also add images, barcodes, and charts to a design using the tools on the *Insert* toolbar.



There are seven drawing tools available on the *Drawing* toolbar in Vitesse.

Tools	Purpose
	Select tool – enables you to select objects for editing and make multiple copies of the object in the design area. This tool is selected by default.
	Insert Line – enables you to draw line objects in the design area.
	Insert Path – enables you to draw path objects in the design area.
	Insert Polygon – enables you to draw polygon objects in the design area.
	Insert Box – enables you to draw box objects in the design area.
	Insert Text – enables the insertion of a text object in the design area.
	Insert Graphic Text – enables the insertion of a graphic text object in the design area.
	Insert Table – enables the insertion of a table object in the design area.
	Insert Frame – enables the insertion of a frame object in the design area.
	Pan View – enables you to move the design area in a magnified view.
	Zoom tool – enables you to magnify the design area.



For more information on using the text tools, refer to *Chapter 9: Using Text Tools*.

All of the tools described in this section can also be used in the magnified view mode.



Drawing tools are available only in Native, Database, and XML modes. For Line mode projects, drawing objects are enabled only inside the forms or segments that are inserted in the project. You cannot directly insert the drawing objects on a page.

Using the Line Tool

You can use the line tool to draw lines in the design area and set or change line properties such as position, size, color, style, etc. See the “Setting Common Object Properties” section later in the chapter which describes properties that apply to all of the drawing objects.

Use the following tasks to draw a line, view, and set line object properties.

Drawing a Line

To draw a line in an existing or a new document:

- 1 Click the line tool, , on the *Drawing* toolbar. The mouse pointer changes from  to .
- 2 Click and drag the mouse anywhere on the design area to draw the line.
The line is added to the design area and a corresponding line node is added under the **Page** tree node in the *Project Tree View* window.

You can set the line object **Name** using the **General** tab of the *Property View* window.

Positioning a Line

To position a line object to the required location in the design area:

- 1 Select the line node from the project tree.



If you click away from the line object in the design area, it may be difficult to select it again. This is because the line is within a bounding box and selecting the line may select the bounding box instead. To easily select the line, click on its node in the project tree.

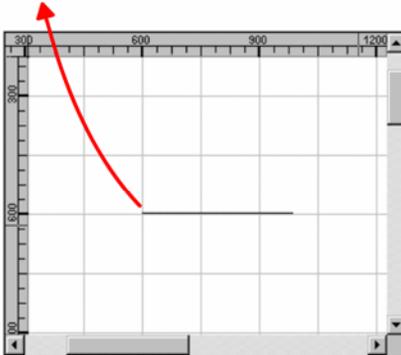
- 2 In the **General** tab of the *Property View* window, set the following **Start Point** information:
- **X** – positions the line with the specified horizontal starting point.
 - **Y** – positions the line with the specified vertical starting point.



You can also set the unit of measurement for the start points by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

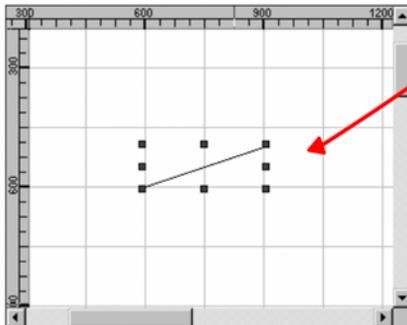
The image below shows the effects of changing the horizontal and vertical starting points of the line object.

1. Initial line position with default horizontal and vertical start coordinates.



2. In the **General** tab of the *Property View* window, set the line object start coordinates.

Start Point:	
X:	600.00 dt
Y:	600.00 dt
End Point:	
X:	900.00 dt
Y:	500.00 dt



3. Line object position is modified based on the specified horizontal and vertical start coordinates.



You can also set the line position either by dragging the object handles located at the start point of the line or with handles that are adjacent to the start point. These settings are automatically updated in the **X** and **Y** edit boxes for the **Start Point** coordinates of the line.

Repositioning a Line

To reposition a line to the required value:

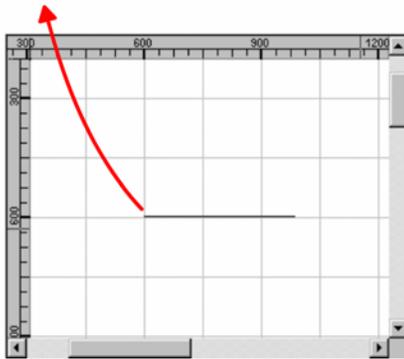
- 1 Select the line node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **End Point** information:
 - **X** – repositions the horizontal end point of the line with the specified horizontal value.
 - **Y** – repositions the vertical end point of the line with the specified vertical value.



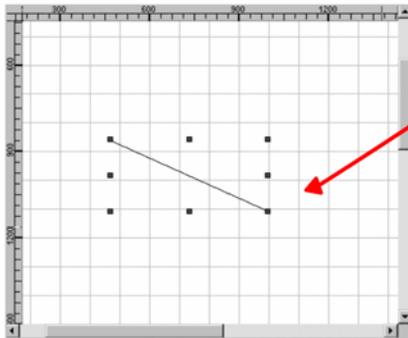
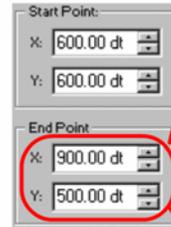
You can also set the unit of measurement for the end points by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**>*unit name* from the pop-up menu.

The image on the next page shows the effects of changing the horizontal and vertical end points of the line object.

1. Initial line position with default horizontal and vertical start coordinates.



2. In the General tab of the Property View window, set the line object end coordinates.



3. Line object position is modified based on the specified horizontal and vertical end coordinates.



You can also set the line size by dragging the object handles located at the end point of the line or with handles that are adjacent to the end point. These settings are automatically updated in the **X** and **Y** edit boxes for the **End Point** coordinates of the line.



- To set the line color and default outline patterns, use the **Color** tab. For more information on using color, refer to *Chapter 11: Working with Colors*.
- To set the size, outline width, and style, use the **Settings** tab. See the “Setting Common Object Properties” section later in the chapter.

Using the Box Tool

The box tool enables you to draw box objects in the design area. Once a box is added to a document, you can view and set box properties, including position, size, style, background and foreground color, etc. See the “Setting Common Object Properties” section later in the chapter which describes properties that apply to all of the drawing objects.

Use the following tasks to draw a box, view, and set box object properties.

Drawing a Box

To draw a box in an existing or a new document:

- 1 Click the box tool, , on the *Drawing* toolbar. The mouse pointer changes from  to .
- 2 Click and drag the mouse anywhere on the design area to draw the box.
The box is added to the design area and a corresponding box node is added under the **Page** tree node in the *Project Tree View* window.

You can set the box object **Name** using the **General** tab of the *Property View* window.

Positioning a Box

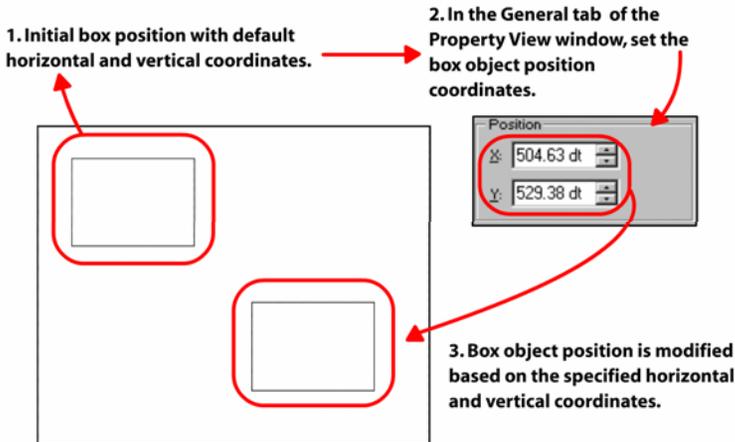
To position a box object to the required location in the design area:

- 1 Select the box node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Position** information:
 - **X** – positions the box with the specified horizontal value, per specified unit of measurement.
 - **Y** – positions the box with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The image below shows the effects of changing the horizontal and vertical **Position** coordinates of the box object.



Creating a Rounded Box

You can specify the box corner types as either rounded or squared by setting the corner radius in the *Property View* window.

Setting Corner Radius

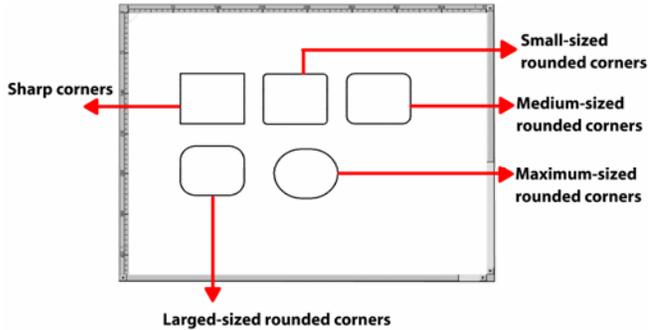
To set rounded corners for the box object:

- 1 Select the required box node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Rounded Box** option; this activates the **Radius** entry box.
- 4 Specify the required corner **Radius** value. A value of zero creates square corners.

As you increase the radius, the corners of the selected box object are rounded off to the specified value. Each rounded box corner is a quarter of a circle. The customized radius value is the distance from the center of the circle to the arc that makes the rounded corner.



You can also set the unit of measurement for the radius by right clicking the upward or downward arrows in the entry box, and then selecting **Units**>*unit name* from the pop-up menu.



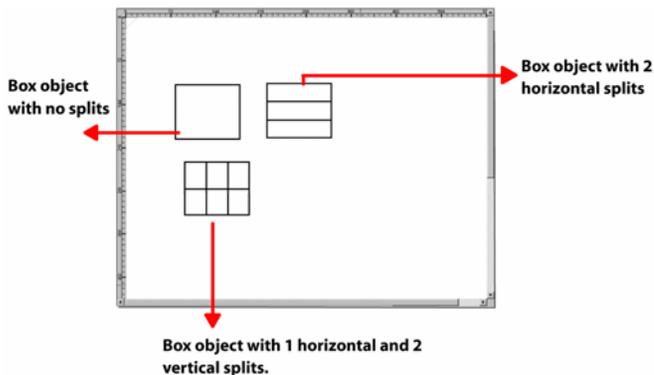
Examples of different box corners.

Splitting a Box

You can split a box object horizontally and vertically.

To split the box object:

- 1 Select the required box node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Split** option; this activates the **Horiz** and **Vert** entry boxes.
- 4 Specify the following:
 - **Horiz** – sets the number of horizontal split lines in the box object.
 - **Vert** – sets the number of vertical split lines in the box object.



Examples of different box splits.

Repeating a Box

Vitesse provides a feature for making multiple copies of a box object. Multiple copies have the same properties as the source box object. Any change that you make in the source object is also inherited by the box object copies. You can also set horizontal and vertical offsets to easily view multiple copies of the source box object and place the copies at different positions, relative to the original box. Copies of the box object cannot be selected. To select the copies, you need to select the original box object.

To create multiple copies of the box object:

- 1 Select the required box node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Repeat** option; associated options are enabled.
- 4 Specify the number of box copies in the **Count** entry box.
- 5 To view the box copies, set the following offset values:
 - **X-Offset** – specifies the horizontal offset of the box copies.
 - **Y-Offset** – specifies the vertical offset of the box copies.

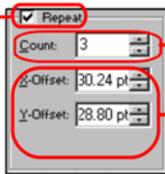
The first box copy is placed relative to the original box object based on the specified horizontal and vertical offsets. The second box copy is placed relative to the first box copy and so on.



You can also set the unit of measurement for the horizontal and vertical offsets by right clicking the upward or downward arrows in the entry box, and then selecting **Units**>unit name from the pop-up menu.

The image below shows an example of creating three box copies in the design area.

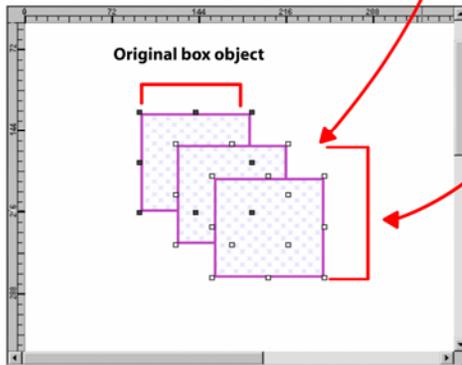
1. In the **Settings** tab of the **Property View** window, select the **Repeat** option .



2. Specify the number of **box copies** in the **Count** entry box.

3. Set the horizontal and vertical **offset values** of the **box copies**.

4. **Box copies** are created at the specified horizontal and vertical **offsets**.



- To set the outline and fill color of the box object along with the outline and fill patterns, use the **Color** tab. For more information on using color, refer to *Chapter 11: Working with Colors*.
- To set the size, outline width, and style, use the **General** and **Settings** tabs respectively. See the “Setting Common Object Properties” section later in the chapter.

Using the Path Tool

The path tool enables you to draw path objects in the design area. Once a path is added to a document, you can view and set path properties, including position, size, style, background and foreground color, nodes, etc. See the “Setting Common Object Properties” section later in the chapter which describes properties that apply to all of the drawing objects.

Use the following tasks to draw a path, view, and set path object properties.

Drawing a Path

To draw a path in an existing or a new document:

- 1 Click the path tool,  on the *Drawing* toolbar. The mouse pointer changes from  to .
- 2 Click in the design area to set the starting point, and then drag the mouse pointer to draw a line.
- 3 Position the mouse pointer where you want the first line segment to end, and click.
- 4 Continue clicking to set endpoints for subsequent line segments.
- 5 Double-click at the required position to finish drawing the path.

The path is added to the design area and a corresponding path node is added under the **Page** tree node in the *Project Tree View* window.

You can set the path object **Name** using the **General** tab of the *Property View* window. The path object is always enclosed within a bounding box and selecting the path selects the bounding box.

Positioning a Path

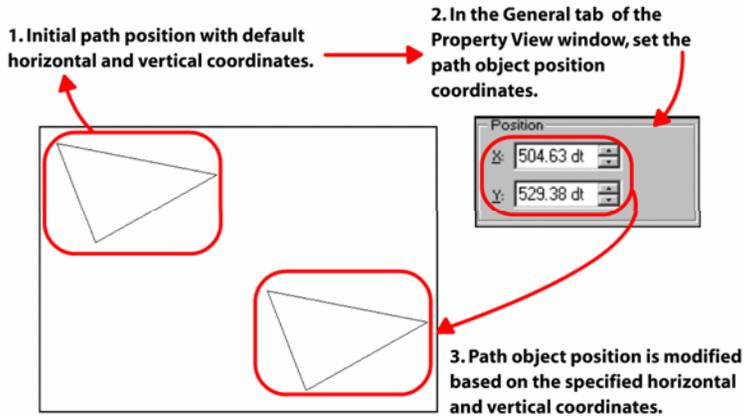
To position a path object to the required location in the design area:

- 1 Select the path node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Position** information:
 - **X** – positions the path with the specified horizontal value, per specified unit of measurement.
 - **Y** – positions the path with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

The image below shows the effects of changing the horizontal and vertical **Position** coordinates of the path object.



Setting Path Nodes

When you draw with the path tool, you click to create points on straight line segments. You can adjust the line segments by adjusting points on the line. The points are also referred to as path nodes. Once you have added the path object in the design area, you can modify the path object by adding or deleting the path nodes. The **Nodes** tab in the *Property View* window displays the number of nodes (depending on the selected path object) and their associated horizontal and vertical position coordinates under the **X** and **Y** table columns.



A row in the **Nodes** tab table identifies the corresponding node of the path object.

Adding a Path Node

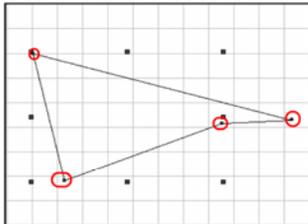
To add a path node to an existing path object:

- 1 Select the path node from the project tree.
- 2 From the *Property View* window, select the **Nodes** tab.
- 3 Click **Add**. A new row is appended to the table and a corresponding node is inserted in the path object. The new node is inserted at the point where the last node was added in the design area.
- 4 To add another node preceding or following an existing node in the path object, select an existing row in the table, and then click **Add**. The new node is added to the selected path object at the required position in the design area.



To view the newly added nodes, you need to adjust the horizontal and vertical coordinates of the new node in the **Nodes** tab. See the “Positioning a Path Node” section later in this chapter.

1. Initial path object contains four nodes with respective horizontal and vertical coordinates.



X	Y
606.25	500.00
706.25	968.75
1200.00	703.13
1200.00	703.13

2. Select the 2nd row in the table, and then click Add to add a row below the first row.

X	Y
603.13	500.00
325.00	1228.13
728.13	1018.75
1496.88	1246.88
1631.25	768.75

The new row identifies the new node.

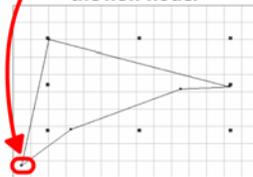


Image shows the process of adding a path node.

Deleting a Path Node

To delete a path node from an existing path object:

- 1 Select the path node from the project tree.
- 2 From the *Property View* window, select the **Nodes** tab.
- 3 Select the required row in the table, and then click **Del.**

Selected row is deleted and corresponding node is also removed from the path object.

Positioning a Path Node

Apart from positioning the entire path object, you can also individually position the path nodes of the selected path object.

- 1 Select the path node from the project tree.
- 2 From the *Property View* window, select the **Nodes** tab.
- 3 Select the required row in the table.
- 4 Double-click the cells under the **X** and **Y** columns; then enter horizontal and vertical position coordinates of the selected node.

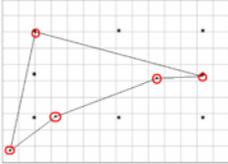
The selected node is positioned at the specified values.



You can also position a path node by dragging the required node in the design area. These settings are automatically updated in the **X** and **Y** edit boxes of the associated node in the **Nodes** tab.

The image below shows the effects of changing the **X** (horizontal) and **Y** (vertical) coordinates of a path node in the design area.

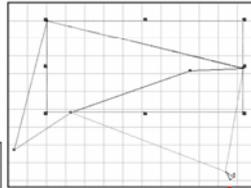
1. Initial path object contains five nodes with respective horizontal and vertical coordinates.



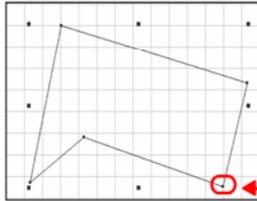
X	Y
603.13	500.00
325.00	1228.13
728.13	1018.75
1436.68 cm	1246.88
1631.25	768.75

2. Set the X or Y coordinate value in the 4th row to position the 4th node of the path object.

3. Alternatively, to position the 4th node, bring the mouse pointer near the node. The pointer changes to



4. Drag the node at the required position in the design area.



5. The selected node is positioned at the required place in the design area.



- To set the outline and fill color of the path object along with the outline and fill patterns, use the **Color** tab. For more information on using color, refer to *Chapter 11: Working with Colors*.
- To set the size, outline width, and style, use the **General** and **Settings** tabs respectively. See the “Setting Common Object Properties” section later in the chapter.

Using the Polygon Tool

The polygon tool enables you to draw polygon objects in the design area. Once a polygon is added to a document, you can view and set polygon properties, including position, size, style, background and foreground color, etc. See the “Setting Common Object Properties” section later in the chapter which describes properties that apply to all of the drawing objects.

Use the following tasks to draw a polygon, view, and set polygon object properties.

Drawing a Polygon

To draw a polygon in an existing or a new document:

- 1 Click the polygon tool, , on the *Drawing* toolbar. The mouse pointer changes from  to .

- 2 Click and drag the mouse anywhere in the design area to draw the polygon.

The polygon is added to the design area and a corresponding polygon node is added under the **Page** tree node in the *Project Tree View* window.

You can set the polygon object **Name** using the **General** tab of the *Property View* window. The polygon object is always enclosed within a bounding box (or bounding rectangle) and selecting the polygon selects the bounding box.

Positioning a Polygon

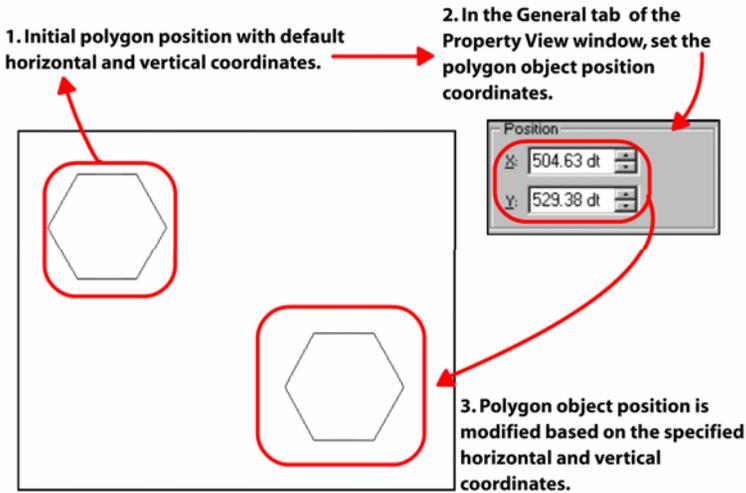
To position a polygon object to the required location in the design area:

- 1 Select the polygon node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Position** information:
 - **X** – positions the polygon with the specified horizontal value, per specified unit of measurement.
 - **Y** – positions the polygon with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The image below shows the effects of changing the horizontal and vertical **Position** coordinates of the polygon object.



Selecting a Polygon Shape

Using the polygon tool, you can draw polygons of different predefined shapes. You can also choose the number of sides of the polygon, from 3 to 999, for drawing a custom shaped polygon. The **Number of lines** area in the **General** tab of the *Property View* window displays a list of the following shapes and corresponding number of sides:

Polygon Shape	Sample	Number. of Sides
Trigon		3
Tetragon		4
Pentagon		5
Hexagon (Default selection)		6
Heptagon		7
Optagon		8
Custom	Sets a customized shape for a polygon. For example	Minimum is 3; Maximum is 999.

Polygon Shape	Sample	Number. of Sides
	 (nonagon).	

To set the polygon shape:

- 1 Select the required polygon node from the project tree.
- 2 In the **General** tab of the *Property View* window, select the required polygon shape from the drop-down list. The adjacent edit box displays the number of sides for each shape.
- 3 To define a custom polygon shape, select the **Custom** option from the drop-down list and set the required value in the adjacent entry box. As you increase the number of lines, the sides of the selected polygon object are rounded off to the specified value.



Setting **999** as the number of lines converts the polygon to a circle.

1. From the Number of lines area, select the required polygon shape or select Custom to specify the required number of lines in the polygon.

2. Polygon is updated based on the selected shape.

Design area with different polygon shapes.

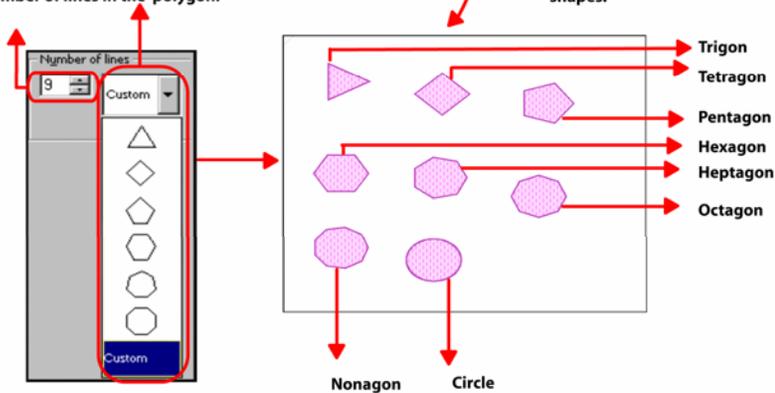


Image showing different polygon shapes.



- To set the outline and fill color of the polygon object along with the outline and fill patterns, use the **Color** tab. For more information on using color, refer to *Chapter 11: Working with Colors*.
- To set the size, outline width, and style, see the “Setting Common Object Properties” section later in the chapter.

Using the Static Table Tool

Tables are about displaying and managing content in your documents. Vitesse static table is an object that you can add to the design area/page to help organize text and other content into rows and columns. While static tables are great for displaying content in your documents, you can also use them for making quick and easy forms or to create complex page layouts in a snap.

Once the table is added to a document, you can view and set the table properties, including position, size, style, background and foreground color, cell spacing, number of rows and columns, etc. See the “Setting Common Object Properties” section later in the chapter which describes properties that apply to all of the drawing objects.

Use the following tasks to draw a static table, view, and set table object properties.

Drawing a Static Table

To draw a static table in an existing or a new document:

- 1 Click the table tool,  on the *Drawing* toolbar. The mouse pointer changes from  to .
- 2 Click and drag the mouse anywhere in the design area to draw the table.
- 3 The table is added to the design area and a corresponding table node is added under the **Page** tree node in the *Project Tree View* window.

You can set the table object **Name** using the **General** tab of the *Property View* window.

Positioning a Static Table

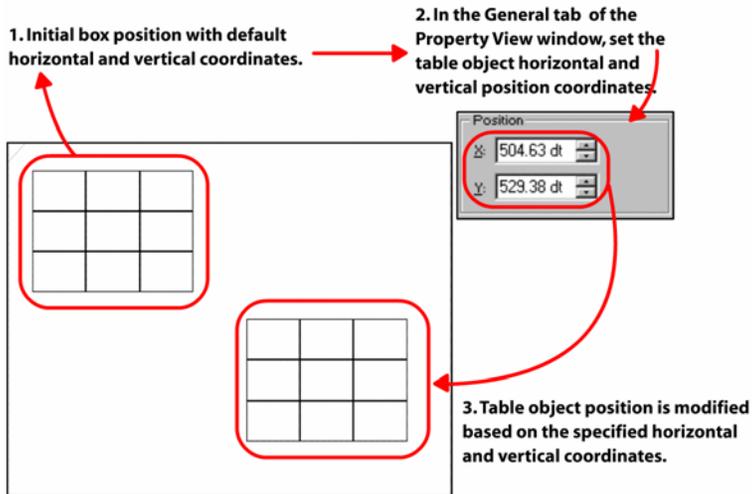
To position a table object to the required location in the design area:

- 1 Select the table node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Position** information:
 - **X** – positions the table with the specified horizontal value, per specified unit of measurement.
 - **Y** – positions the table with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The image below shows the effects of changing the horizontal and vertical **Position** coordinates of the table object.



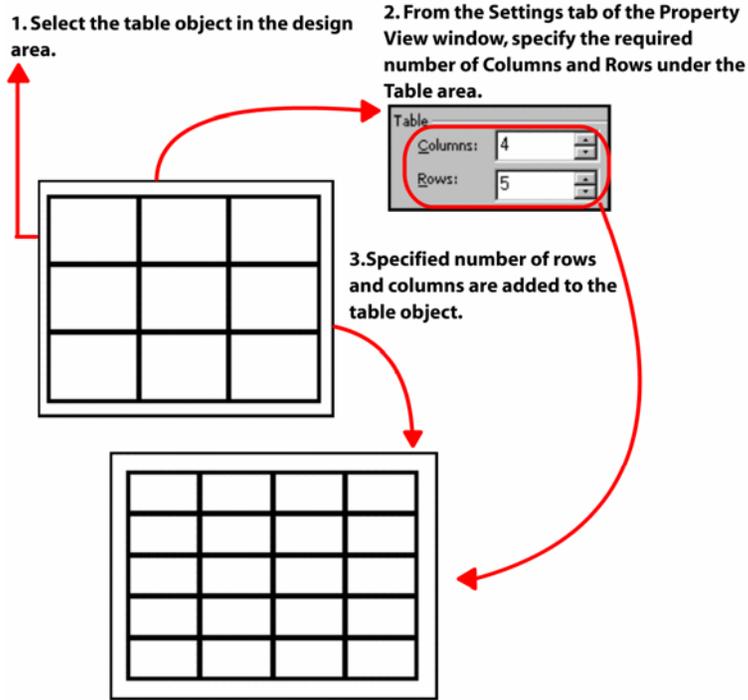
Setting the Number of Rows and Columns

Once you have added the table object in the design area, you can modify the object by deleting or adding rows and columns. The default number of rows and columns in the table is 3 each.

To add or delete the number of rows and columns in the table:

- 1 Select the table node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Table** area, set the following:
 - **Columns** – specifies the number of table columns.
 - **Rows** – specifies the number of table rows.

The selected table is modified as shown in the image below:



Setting the Table Cell Spacing

Apart from adding or deleting rows and/or columns in the table, you can also modify the table cells by increasing or decreasing their spacing from each other.

To set the table cell spacing:

- 1 Select the table node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Cell Spacing** area, set the following:
 - **Horizontal** – specifies the horizontal spacing of the cells in the table, per specified unit of measurement.
 - **Vertical** – specifies the vertical spacing of the cells in the table, per specified unit of measurement.

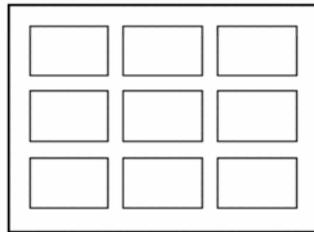
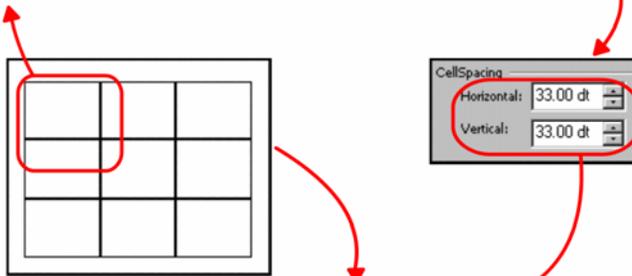


You can also set the unit of measurement for the spacing by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The cells of the selected table are spaced as shown in the image below:

1. Initial horizontal and vertical spacing is zero (default) for all the cells in the table.

2. In the Settings tab of the Property View window, set the horizontal and vertical cell spacing values (for example, 33.00dt).



3. The Horizontal and Vertical spacing values are applied to all the table cells.



- To set the outline and fill color of the table object along with the outline and fill patterns, use the **Color** tab. For more information on using color, refer to *Chapter 11: Working with Colors*.
- To set the size, outline width, and style, see the “Setting Common Object Properties” section later in the chapter.

Setting Common Object Properties

Most objects available in Vitesse share common properties such as line width, style, size, color, etc. Tasks in this section explain how to set common properties for different design objects.

Setting the Object Size

To set the object size:

- 1 Select the required object node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Size** information:
 - **W** – resizes the object with the specified horizontal value, per specified unit of measurement.
 - **H** – resizes the object with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the size by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

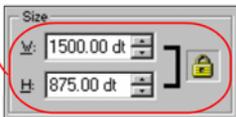


The  icon in the **Size** area is used for maintaining the aspect ratio of the drawing object. Clicking this icon and then setting the width and height increases or decreases the length and width of the object in equal proportion.

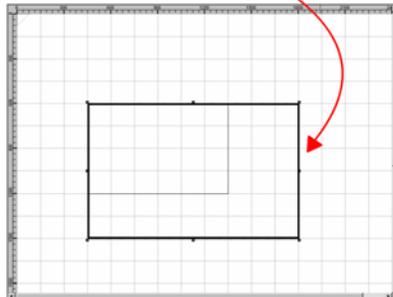
The following images show the effects of increasing the object size for the box, path, polygon, and static table objects proportionally:

Setting the Box Size

1. To increase the box width and height in equal proportions, click  in the **Size** area. Specify the width and height values in the appropriate entry boxes.



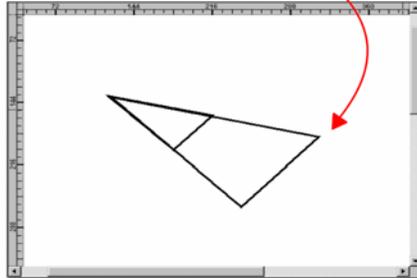
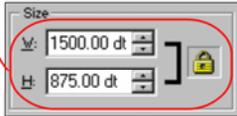
2. Box object increases proportionally based on the specified width and height values.



Setting the Path Size

1. To increase the path width and height in equal proportions, click  in the Size area. Specify the width and height values in the appropriate entry boxes.

2. Path object increases proportionally based on the specified width and height values.

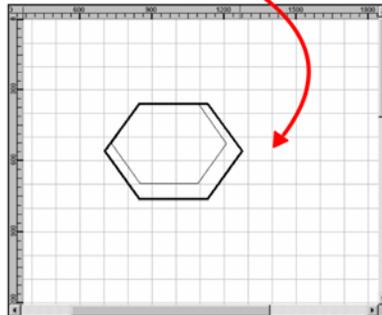
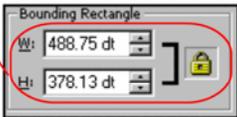


Setting the Polygon Size

To set the polygon size, use the **Bounding Rectangle** area in the **General** tab of the *Property View* window.

1. To increase the polygon width and height in equal proportions, click  in the Bounding Rectangle area. Specify the width and height values in the appropriate entry boxes.

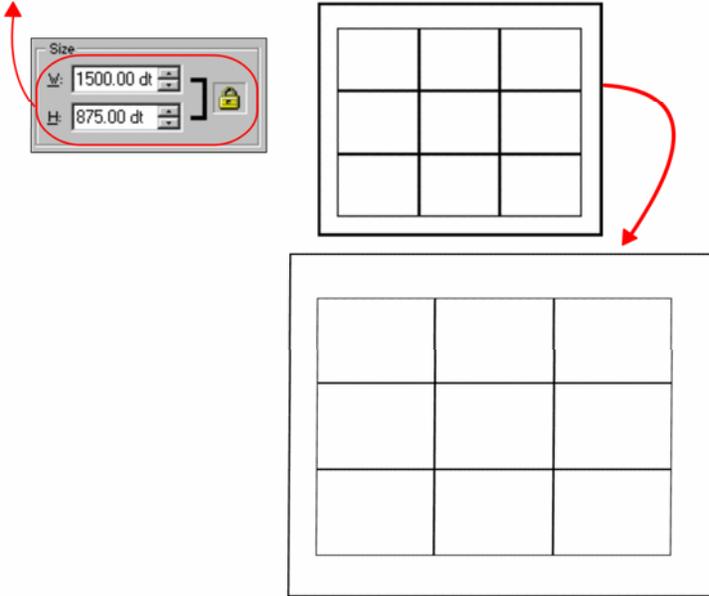
2. Polygon object increases proportionally based on the specified width and height values.



Setting the Static Table Size

1. To increase the table width and height in equal proportions, click  in the Size area. Specify the width and height values in the appropriate entry boxes.

2. Table object increases proportionally based on the specified width and height values.



Setting Object Line Width

Vitesse provides various options to change the width of an object.

To set the object outline width:

- 1 Select the required object node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Line** area, select the required width option from the **Width** drop-down list. The adjacent entry box displays the width value associated with the selected option.
- 4 To set a width other than the predefined values, select the **Custom** option from the **Width** drop-down list. This option enables the adjacent entry box.



You can also set the unit of measurement for the width by right clicking the upward or downward arrows in the entry box, and then selecting **Units>unit name** from the pop-up menu.

- 5 Enter the required line width value; width of the selected line object is modified.

The following images show the process of setting the custom outline width for the line, box, path, polygon, and static table objects:

Setting the Line Width

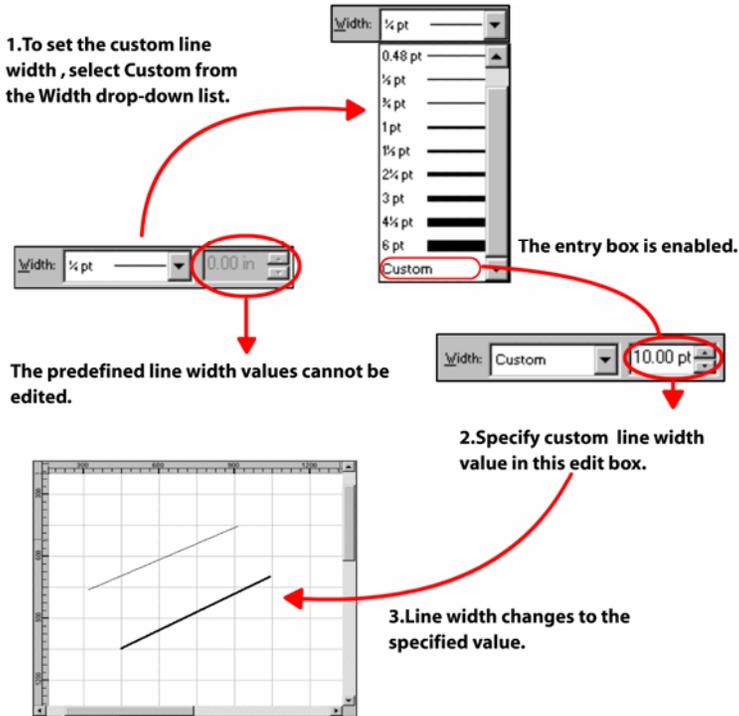
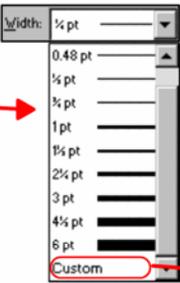


Image shows the process of changing the line object width.

Setting the Box Line Width

1. To set the custom box outline width, select Custom from the Width drop-down list.



The entry box is enabled.

The predefined line width values cannot be edited.



2. Specify custom outline width value in this edit box.



3. Box outline width changes to the specified value.

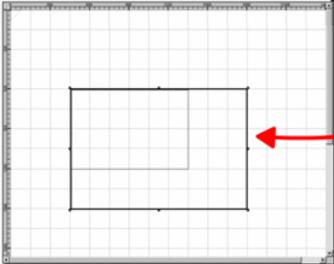


Image shows the process of changing the box object width.

Setting the Path Line Width

1. To set the custom path outline width, select Custom from the Width drop-down list.



The entry box is enabled.

The predefined line width values cannot be edited.



2. Specify custom outline width value in this edit box.



3. Path outline width changes to the specified value.

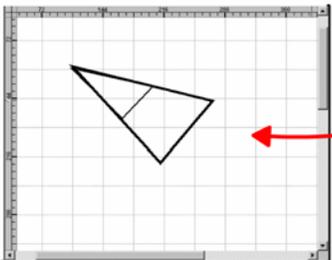
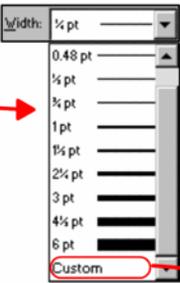


Image shows the process of changing the path object width.

Setting the Polygon Line Width

1. To set the custom polygon outline width, select Custom from the Width drop-down list.



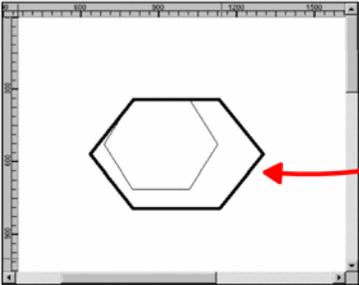
The entry box is enabled.



The predefined line width values cannot be edited.



2. Specify custom outline width value in this edit box.



3. Polygon outline width changes to the specified value.

Image shows the process of changing the polygon object width.

Setting the Static Table Line Width

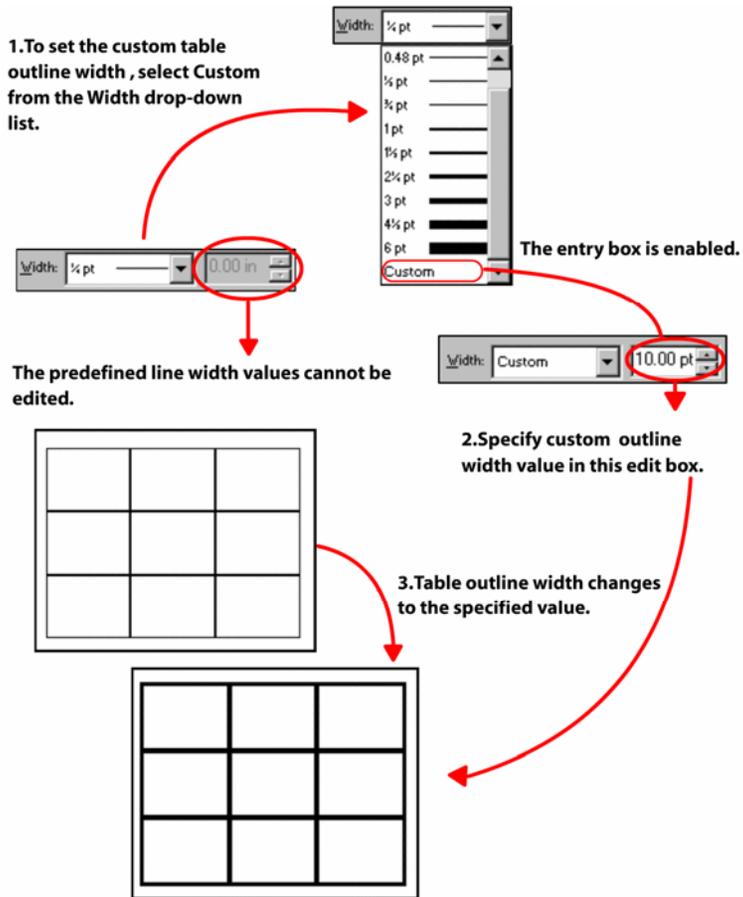


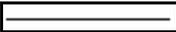
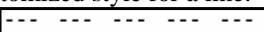
Image shows the process of changing the table object width.

Setting the Object Line Style

Along with changing the line width, you can also apply different line styles to objects. For example, you can make the line appear as dotted or dashed.

To set the object outline style:

- 1 Select the required object node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Line** area, select one of the following style options from the **Style** drop-down list:

Line Style	Sample
Solid	
Dashed	
Dotted	
Dash-Dotted	
Dash-Dot-Dotted	
Custom	Sets a customized style for a line. For example,  .

The area adjacent to the **Style** drop-down list shows the sample preview of the selected style.

The black and white boxes  in the **Line** area depict the selected line style pattern. The box in black represents the line and the box in white represents the line spacing used in the line style. The pattern keeps changing, depending on the **Style** selection. For example, if you select the dash-dotted line style, the pattern changes to . The first box is always black.

- 4 To set a style other than the predefined values, select the **Custom** option from the **Style** drop-down list.
- 5 Move the slider to set the custom line style pattern length, and then click the required box to toggle from black to white.

See images below that show how to set the custom outline style for the line, box, path, polygon, and static table objects.

Setting the Line Style

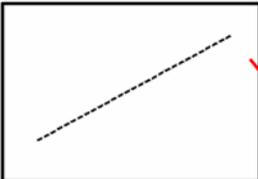
1. To set the custom line style, select Custom from the Style drop-down list.



2. Move the slider to set the required length of the pattern. The length keeps incrementing as you move the slider and is indicated by the number at the left of the slider.



3. Click the required box to toggle between black and white. Black box represents the line and white box represents the space.



4. The line style of the selected line object changes to the specified style.



Image shows the process of changing the outline style of the line object.

Setting the Box Line Style

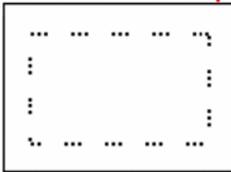
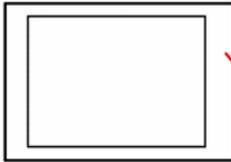
1. To set the custom outline style, select Custom from the Style drop down list.



Sample preview of specified style.

2. Move the slider to set the required length of the pattern. The length keeps incrementing as you move the slider and is indicated by the number at the left of the slider.

3. Click the required box to toggle between black and white. Black box represents the line and white box represents the space.

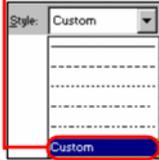


4. The outline style of the selected box object changes to the specified style.

Image shows the process of changing the outline style of the box object.

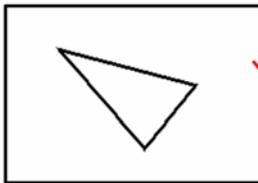
Setting the Path Line Style

1. To set the custom outline style, select Custom from the Style drop down list.



Sample preview of specified style.

2. Move the slider to set the required length of the pattern. The length keeps incrementing as you move the slider and is indicated by the number at the left of the slider.



3. Click the required box to toggle between black and white. Black box represents the line and white box represents the space.

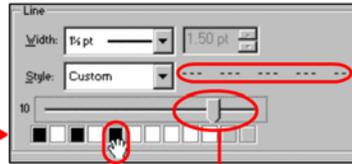


4. The outline style of the selected path object changes to the specified style.

Image shows the process of changing the outline style of the path object.

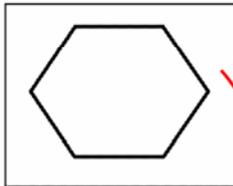
Setting the Polygon Line Style

1. To set the custom outline style, select Custom from the Style drop down list.

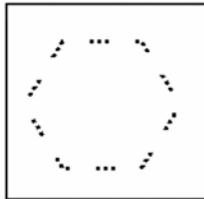


Sample preview of specified style.

2. Move the slider to set the required length of the pattern. The length keeps incrementing as you move the slider and is indicated by the number at the left of the slider.



3. Click the required box to toggle between black and white. Black box represents the line and white box represents the space.

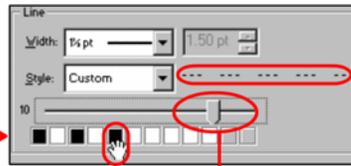
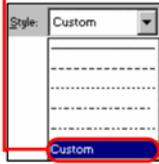


4. The outline style of the selected polygon object changes to the specified style.

Image shows the process of changing the outline style of the polygon object.

Setting the Static Table Line Style

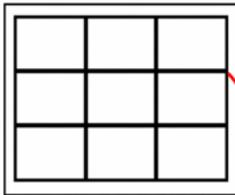
1. To set the custom outline style, select Custom from the Style drop-down list.



Sample preview of specified style.

2. Move the slider to set the required length of the pattern. The length keeps incrementing as you move the slider and is indicated by the number at the left of the slider.

3. Click the required box to toggle between black and white. Black box represents the line and white box represents the space.



4. The outline style of the selected table object changes to the specified style.

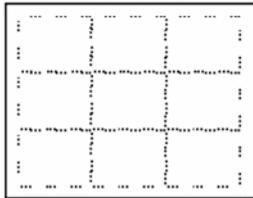


Image shows the process of changing the outline style of the table object.

Assigning Properties to Multiple Selected Objects

You can select individual objects simultaneously to manipulate them as a single object. If you select multiple static objects of different types in the design area, such as a path, table, line, text, the *Property View* window indicates a mixed selection of properties that are common to the selected objects. The *Property View* window for a mixed selection displays the object position coordinates, dimensions, line width and styles, color (outline and/or fill) and font (if text object is also selected) of the selected set of objects using the **General**, **Settings**, **Color**, and **Font** tabs respectively. Specifying any of the above mentioned properties applies to all of the selected objects.



Properties that do not apply to multiple selected objects remain disabled.

The *Project Tree View* window also highlights all nodes of the multiple selected objects.

GEP Keys

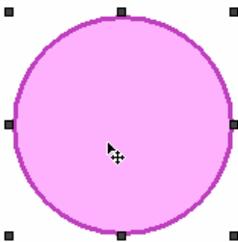
Graphic Element Property (GEP) keys are sets of similar design object graphic attributes, such as line style, line width, outline color, and fill patterns. Vitesse assigns the same GEP key to objects having identical attributes. These keys are visible and modifiable only in the source view. See your VIPP Reference Manual for more information on GEP keys.

Basic Operations

You can organize your design and design objects using basic editing commands including cut, copy, paste, and delete; and modifying commands such as selecting, resizing, aligning, and arranging. You can move objects to different locations in the design area, and specify fixed distances between multiple objects and object copies. You can also apply color to design objects or change an existing color. This section covers the basic operations involved in manipulating Vitesse design objects and setting object properties.

Selecting an Object

To modify an object, you must first select it. Vitesse provides a variety of methods for making selections, including the Selection tool, menu bar, project tree, and keyboard commands. All objects have handles at each corner and each midpoint. As illustrated in the figure below, resize handles mark a selected object and the mouse pointer changes to  in the selection mode.



Selected box object.

Select Design Area Objects

You select a design area object by clicking it. You select multiple objects by clicking objects and simultaneously pressing <Shift>. You can also select all of the objects available in the design area by selecting Edit>Select All from the menu bar or pressing <Ctrl>+<A> on the keyboard. This command enables you to cut, copy, or paste all of the selected objects from one page to another page within the same project or to different projects. See tasks for cutting, copying, and pasting objects later in the chapter.

Select Project Tree Nodes

You select a project tree node by clicking it. For each project tree node you select, corresponding objects are automatically selected in the design area, and vice-versa.

You select multiple project tree nodes by clicking project tree nodes and simultaneously pressing any one of the following keyboard combinations, as appropriate:

- <Ctrl> to select multiple Project tree nodes.
Or
- <Ctrl> + A to select all Project tree nodes.
Or
- <Shift> + <UP ARROW> to select current node and one node above.
Or
- <Shift> + <DOWN ARROW> to select current node and one node below.
Or
- <Shift> + <PAGE UP> to select current node and nodes preceding the selected node.
Or
- <Shift> + <PAGE DOWN> to select current node and nodes following the selected node.
Or
- <Ctrl> + <Shift> + <HOME> to select all nodes between current and first node.
Or
- <Ctrl> + <Shift> + <END> to select all nodes between current and last node.

Copying an Object

Copying an object places a copy of the object on the clipboard. You can paste a copy of the object on the same page or on a different page. There are several methods for copying objects. You can either:

- Select an object on the page and select **Copy** from the Edit menu.
- Select the object on the page and click  on the *Standard* toolbar.
- Right-click an object on the page and select **Copy** from the pop-up menu.
- Select the object on the page, and then press <Ctrl>+<C> on the keyboard.

You can use the same technique to manage single or multiple object selections.

Cutting an Object

Cutting an object removes it from the page and places it on the clipboard. You can cut an object and then paste it to move it from one location to another on the page. There are several methods for cutting objects. You can either:

- Select the object on the page and select **Cut** from the Edit menu.
- Select the object on the page and click  on the *Standard* toolbar.
- Right-click an object on the page, and then select **Cut** from the pop-up menu.
- Select the object on the page, and then press <Ctrl>+<X> on the keyboard.

You can use the same technique to manage single or multiple object selections.

Pasting an Object

After copying and/or cutting objects, you can paste them on the page at the required location. The paste command has no effect if the clipboard is empty. There are several methods for pasting objects. You can either:

- Cut an object from the page, and then paste the object at the required location on the page using the Paste command from the Edit menu.
- Copy an object from the page, and then paste the object at the required location on the page using the Paste command from the Edit menu.
- Cut or copy an object from the page, and then click  on the *Standard* toolbar.
- Cut or copy an object from the page, and then paste the object by right-clicking at the required location and selecting **Paste** from the pop-up menu.
- Cut or copy an object, and then press <Ctrl>+<V> on the keyboard.

You can use the same technique to manage single or multiple object selections.

Deleting an Object

You can delete objects within the design area in Vitesse. There are several methods for deleting objects. You can either:

- Select an object on the page and select **Delete** from the Edit menu.
- Select an object on the page and click  on the *Standard* toolbar.
- Right-click an object on the page and select **Delete** from the pop-up menu.
- Select an object on the page and press <Delete> on the keyboard.

You can use the same technique to manage multiple object selections.

Resizing an Object

Vitesse enables you to enlarge or reduce an object size either horizontally, vertically, or diagonally using the **Size** feature.

Use one of the following methods to change an object size:

- Select the object, and then use the resize handles to drag the mouse pointer either vertically or horizontally. To resize the object both horizontally and vertically, drag one of the corner handles. Release the mouse button at the required line length and/or width.
- Select the object, and then specify the horizontal and vertical size using the **W** and **H** entry boxes respectively in the **General** tab of the *Property View* window. The object size changes to the specified size.

You can use the same technique to manage multiple object selections.

Moving an Object

You can use one of the following methods for positioning objects in the design area:

- Select and drag the object using the mouse at the required place in the design area. The object is moved to the new location.
- Select the object and enter the coordinates in the **X** and **Y** entry boxes in the **General** tab of the *Property View* window. The object is positioned to the required location.
- Select the object, and then press the arrow keys for the direction in which you want the object to move 30 dots at a time. Press <Shift>+<arrow key> to move the selection 3 dots at a time.

You can use the same technique to manage multiple object selections.

Renaming an Object

You can also rename a selected object using the **Name** entry box in the **General** tab of the *Property View* window.

Objects can also be renamed by renaming their respective project tree nodes in the *Project Tree View* window.

To rename project tree nodes, either:

- Double-click the required tree node in the project tree window.
- Or
- Press <F2> on the keyboard.

Rename the required project tree node, and then click away from the **Name** entry box to save the name.



Main project, media, and resource nodes cannot be renamed.

Sequencing Project Tree Nodes

You can sequence both single and multiple project tree nodes to suit a required hierarchical project structure either by a drag-and-drop operation or cut, copy, and paste operation. The sequencing of project tree nodes mirrors the VIPP Source code and determines the print order.

Aligning Objects

The *Alignment* toolbar enables you to align selected objects along the horizontal or vertical axis. You can align objects vertically along the right edge, center, or left edge of the selected objects; or horizontally along the top edge, center, or bottom edge of the selected objects. Edges are determined by the bounding boxes enclosing each selected object.

Using the *Alignment* toolbar, you can also distribute selected objects so that their centers or edges are evenly spaced. You can resize selected objects so that the horizontal or vertical dimensions of all objects become common to the recently added object in the design area.



The object that you most recently add in the design area is displayed at the lowest level in the project tree.



Alignment toolbar.

Use one of the following commands to align objects in the design area:

Align Left

Select Tools>Alignment Tools>Align Left or click  on the *Alignment* toolbar to align the multiple selected objects to the left of the recently added object.

Align Right

Select Tools>Alignment Tools>Align Right or click  on the *Alignment* toolbar to align the multiple selected objects to the right of the recently added object.

Align Bottom

Select Tools>Alignment Tools>Align Bottom or click  on the *Alignment* toolbar to align the multiple selected objects to the bottom of the recently added object.

Align Top

Select Tools>Alignment Tools>Align Top or click  on the *Alignment* toolbar to align the multiple selected objects to the top of the recently added object.

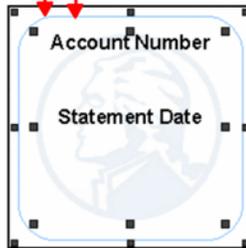
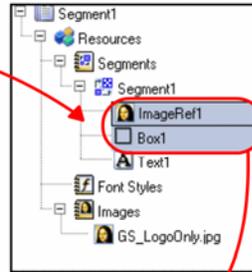
Center Horizontally

Select Tools>Alignment Tools>Center Horizontally or click  on the *Alignment* toolbar to place the multiple selected objects at the horizontal center of the recently added object.

Center Vertically

Select Tools>Alignment Tools>Center Vertically or click  on the *Alignment* toolbar to place the multiple selected objects at the vertical center of the recently added object.

Clicking  aligns the multiple selected image and box objects vertically; clicking  aligns the selected image and box objects horizontally.



The image, box, and text objects are now neatly aligned with respect to each other.

Image shows the vertical and horizontal alignment of box and image objects to make them appear neater and more defined.



Refer to *Chapter 9: Using the Text Tools* for information on the text object.

Evenly Space Horizontally

Select Tools>Alignment Tools>Evenly Space Horizontally or click  on the *Alignment* toolbar to place the multiple selected objects at equal horizontal distance.

Evenly Space Vertically

Select Tools>Alignment Tools>Evenly Space Vertically or click  on the *Alignment* toolbar to place the multiple selected objects at equal vertical distance.

Make Same Width

Select Tools>Alignment Tools>Size Same Width or click  on the *Alignment* toolbar so that the horizontal width of all the multiple selected objects match the width of the recently added object.

Make Same Height

Select Tools>Alignment Tools>Size Same Height or click  on the *Alignment* toolbar so that the vertical width of all the multiple selected objects match the height of the recently added object.

Clicking  aligns the selected objects such that height of all objects become equal to the text object.

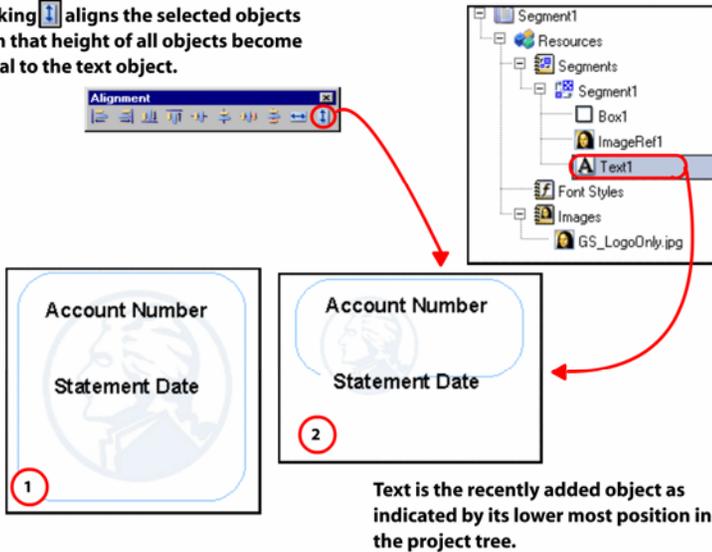


Image shows the 1) initial object heights 2) object heights made equal to the text object height.

Arranging Objects

Within the design area, Vitesse lets you arrange/stack objects based on the order in which they are created, placing the most recently created object at the top of the stack. The arrangement order of objects determines how they appear when they are overlapping. You can change the arrangement order of objects at any time using the **Arrangement Tools>options** from the Tools menu or the *Arrangement* toolbar.



Arrangement Toolbar.

The options are explained below:

Bring to Front

Select Tools>Arrangement Tools>Bring to Front or click  on the *Arrangement* toolbar to bring the selected object to the front of overlapping objects in the design area.

Send to Back

Select Tools>Arrangement Tools>Send to Back or click  on the *Arrangement* toolbar to send the selected object to the back of overlapping objects in the design area.

Bring Forward

Select Tools>Arrangement Tools>Bring Forward or click  on the *Arrangement* toolbar to bring the selected object up one step from the current position in the design area and project tree.

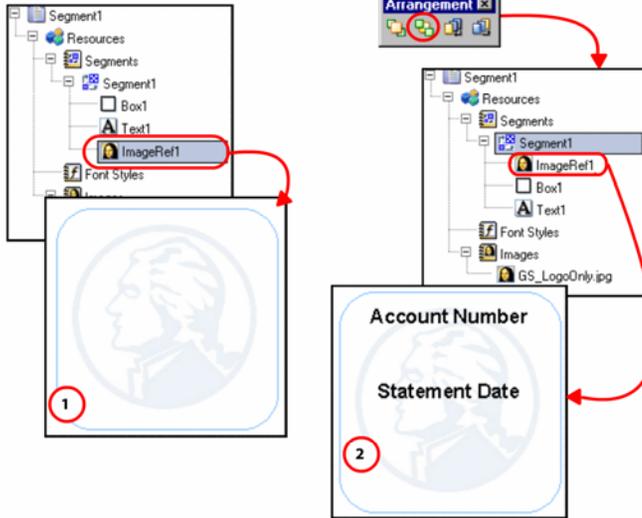
Send Backward

Select Tools>Arrangement Tools>Send Backward or click  on the *Arrangement* toolbar to send the selected object down one step from the current position in the design area and project tree.

As you arrange the objects using one of the above options, the project tree also reflects the object movements.

The image is placed over the box and text object because it was added last.

Clicking  sends the image to the background of the segment so that the text becomes visible.

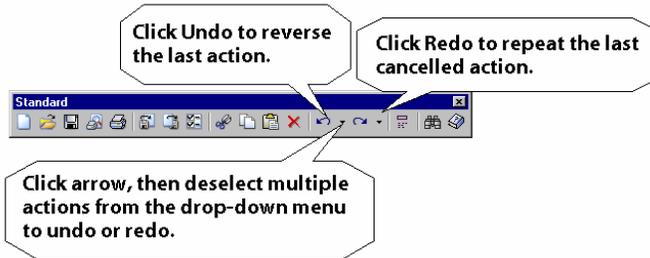


After sending the image backward, the text becomes visible in the foreground. The project tree also depicts these changes.

*Image shows that 1) the text becomes invisible because the image was added last
2) sending the image to the background using the Send to Back option brings the text to the foreground.*

Undoing and Repeating Actions

Vitesse keeps track of your editing and formatting changes. If you want to undo a change or command selection, you can usually reverse several previous actions.



You can also select the **Undo** and **Redo** options from the Edit menu.

Chapter 9: Using the Text Tools

This chapter provides activities and tasks for using text and graphic text objects and setting associated properties. It also gives a detailed overview on using rich text from external projects, spell checker and dictionary files, finding and replacing text, and for setting the default text properties. Tasks include:

- Inserting Text and Graphic Text Objects
- Positioning and Sizing Text and Graphic Text Objects
- Merging Variable Data with Text and Graphic Text Objects
- Formatting Text and Graphic Text Objects
- Setting Text Tabs
- Using Rich Text from External Projects
- Using the Spell Checker and Setting the Spell Checker Depth
- Specifying the Dictionary Files for Spell Checking
- Setting the Default Text Attributes
- Setting the Data Switch
- Finding and Replacing Text

Using Text Tools

Vitesse enables you to insert multiple text objects in the design area and set or change text properties for these objects. Vitesse also allows you to import a rich text format file and export text to a rich text format file. You can merge variable data with text for forms and segments using Vitesse. Facilities to join text objects and to use the spell checker by specifying dictionary files, finding, and replacing text are also available. Along with the text object, you can also insert a graphic text object in the design.

Use the following tasks to insert text (or graphic text), view, and set text object properties.

Inserting a Text Object

Once a document is loaded in the design area, you can insert a text object using either of the two methods described below:

- Inserting the text by clicking and typing.
- Inserting the text by clicking and dragging.

Inserting Text by Clicking and Typing

To insert a text object in an existing or new document:

- 1 Click the text tool, , on the *Drawing* toolbar.

The mouse pointer changes from  to .

- 2 Click at the required place in the design area where you want to insert the text.
- 3 Enter the required text using the keyboard. The text box is automatically created around the entered text.



You can enter text properties while the cursor is still active inside the text object. The properties will not save to the object, however, until you click outside the text object. We recommend that you finish typing the text, and then click once outside the text object. This action selects the text object. Once selected, all properties automatically apply as they are entered in the *Property View* window.

Creating a Text Box

To create a text box in the document:

- 1 Click the text tool, , on the *Drawing* toolbar.

The mouse pointer changes from  to .

- 2 Using the mouse, draw a text box of required dimensions at the appropriate location in the design area.
- 3 Enter the text in the text box using the keyboard.

The text object is added to the design area and a corresponding text node is added under the **Page** tree node in the *Project Tree View* window.

You can set the text object **Name** using the **General** tab of the *Property View* window.

Inserting a Graphic Text Object

To insert a graphic text object in an existing or new document, click the graphic text tool, , on the *Drawing* toolbar. The remaining steps for adding this object to the design area are exactly the same as for the text object (see above task).



Properties for the graphic text object are similar to that of the text object. You can set the size of the graphic text object, position the object, apply different fonts; colors and color patterns; transformations; line and paragraph spacing; auto size the text, and align the text using the properties available in the *Property View* window. For information on setting these properties, refer to the tasks (below) explained for the text object.

Editing Text

To edit an existing text within the text box, do either of the following:

- Double-click the selected text object.
Or
- Right-click the selected text object, and then select **Edit Text** from the pop-up menu.

Positioning Text

To accurately place a text object in the design area:

- 1 Select the required text object in the design area.



You can also select the **Text** node from the *Project Tree View* window to set the text object properties.

- 2 In the **General** tab of the *Property View* window, set the following **Position** information:

- **X** – positions the text object with the specified horizontal value, per specified unit of measurement.
- **Y** – positions the text object with the specified vertical value, per specified unit of measurement.



You can also set the unit of measurement for the position of the object by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The drop-down lists adjacent to the **X** and **Y** entry boxes enable you to set the relative position of text objects with respect to each other.

You can set the **X** (Horizontal) and **Y** (Vertical) object position using either Absolute or Relative positioning options.

Absolute Positioning - Places the object using the coordinates entered in the **X** and **Y** entry boxes.

Relative Positioning - Places the current object relative to the previous object. Using relative positioning lets you avoid entering the exact coordinates, yet still

enables you to position objects accurately.  displays next to object nodes that are relatively positioned; the horizontal arrow represents the **X** coordinate relative position, and the vertical arrow represents the **Y** coordinate relative position. When working with relative positioning, you need to remember that each position is in relation to the previous object. As a result, changing the position of one object also affects the position of the object relative to it.

The following is the list of options and associated definitions for both **Horizontal** and **Vertical Relative** positioning:

Start Same - positions the start of current object at the start of the previous object.

Start Center - positions the start of the current object at the center of the previous object.

Start Next - positions the start of the current object at the end of the previous object.

Center Before - positions the center of the current object at the start of the previous object.

Center Same -positions the center of the current object at the center of the previous object.

Center Next -positions the center of the current object at the end of the previous object.

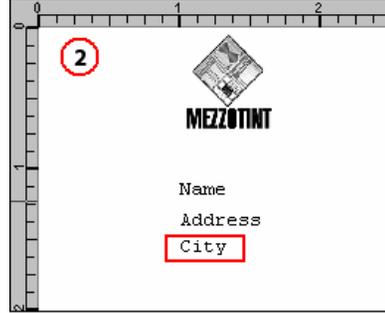
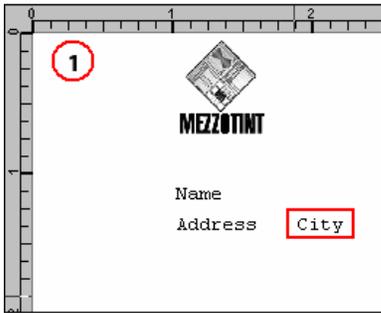
End Before - positions the end of the current object at the start of the previous object.

End Center - positions the end of the current object at the center of the previous object.

End Same - positions the end of the current object at the end of the previous object.

The image below describes different horizontal and vertical placement options:

The City Text box object is positioned at 1) X [0.00 in] – Start Next, Y [1.22 in] – Absolute 2) X [0.00 in] – Start Same, Y [1.22 in] – Start Next.



Setting the Text Size

To set the size of the text object:

- 1 Select the required text object node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Size** information:
 - **W** – sets the text object width, per specified unit of measurement.
 - **H** – sets the text object height, per specified unit of measurement.



You can also set the unit of measurement for the size by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.



The  icon in the **Size** area is used for maintaining the aspect ratio of the text object. Clicking this icon and then setting the width and height increases or decreases the length and width of the object in equal proportion. In the unlock mode, you can set any values for the width and height individually. For more information, see the “Setting Common Object Properties” section in *Chapter 8: Using the Drawing Tools*.

Using the Merge Data Feature

Vitesse supports merging data with text in forms, segments, Line, Database, and XML mode projects. You can merge data with text in either of the following ways:

- Use the Database, and/or XML mode to obtain pre-defined variables from the respective data files, and then merge the selected variable containing information with the inserted text object. Refer to *Sections 6, 7, and 8* for more information on merging data in the listed modes.



For Line mode projects, you need to insert a form or segment prior to using the data merge feature associated with the text object.

- Define variables in an XML file, and use the pre-defined variables in the XML file to merge data with text. You can define and use the XML file only as an administrator in the Vitesse Administration mode.

This section explains how to use the XML file for merging data with text.



A **variable** is a container that stores information. For more information, refer to *Chapter 26: Using Variables*.

Creating an XML file with Pre-defined Variables

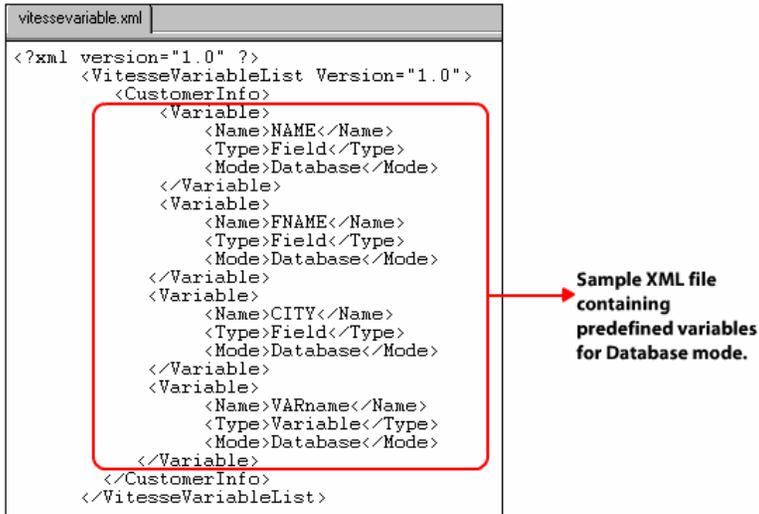
As an administrator, you can create an XML file using any text editor in which you define the variables to be used in a design, and list the required type and mode; pertaining to each variable (these are identified by their respective tags, as shown in the table below). The XML file can contain any number of variables required in the project. This file can also be updated anytime; this action does not affect the variables already used in a project. The updates in the file will be available the next time you open a project. The variables that you define are enclosed within start and end tags. The following table shows the tags used in the XML file:

Tag Name	Description
Name	Specifies the name of the variable or field. For example, City or VARCountry, etc. Variable names begin with VAR and each name must be unique. No spaces or special characters are allowed.
Type	Specifies the type of variable. For XML mode projects, the following items are also included in the XML

Tag Name	Description
	<p>file:</p> <p>Element: specifies a complete path for the XML tags. The type of variable is set to Element when the type of project mode is set to XML.</p> <p>Field: specifies the name of the field.</p> <p>Attribute: name of the attribute must contain @, as per VIPP rules. The type of variable is set to Attribute when the type of project mode is set to XML</p>
Mode	Specifies the type of project mode; either Database or XML mode.

All of the above tags are enclosed within the <Variable></Variable> tags.

The following images provide examples of the VitesseVariables.xml file for Database and XML project modes.



```

vitessevariable.xml
<?xml version="1.0" ?>
<VitesseVariableList Version="1.0">
  <CustomerInfo>
    <Variable>
      <Name>NAME</Name>
      <Type>Field</Type>
      <Mode>Database</Mode>
    </Variable>
    <Variable>
      <Name>FNAME</Name>
      <Type>Field</Type>
      <Mode>Database</Mode>
    </Variable>
    <Variable>
      <Name>CITY</Name>
      <Type>Field</Type>
      <Mode>Database</Mode>
    </Variable>
    <Variable>
      <Name>VARname</Name>
      <Type>Variable</Type>
      <Mode>Database</Mode>
    </Variable>
  </CustomerInfo>
</VitesseVariableList>

```

Sample XML file containing predefined variables for Database mode.

XML file for Database Mode.

```

VitesseVariables.xml
<?xml version="1.0" ?>
  <VitesseVariableList Version="1.0">
    <CustomerInfo>
      <Variable>
        <Name>^Customer^Id</Name>
        <Type>Element</Type>
        <Mode>XML</Mode>
      </Variable>
      <Variable>
        <Name>^Customer^@Name</Name>
        <Type>Attribute</Type>
        <Mode>XML</Mode>
      </Variable>
      <Variable>
        <Name>^Customer^City</Name>
        <Type>Element</Type>
        <Mode>XML</Mode>
      </Variable>
      <Variable>
        <Name>VARcountry</Name>
        <Type>Variable</Type>
        <Mode>XML</Mode>
      </Variable>
    </CustomerInfo>
  </VitesseVariableList>

```

Sample XML file containing pre defined variables for XML mode.

XML file for XML Mode.



You need to define separate XML files for Database and XML modes; a single XML file cannot contain occurrences of both the project modes.

After creating the XML file, place the file at the required folder; recommended location is: `drive:\Program Files\Elixir Technologies\Vitesse\Config`.

Loading an XML file with Pre-defined Variables

To load the XML file with the pre-defined list of variables:

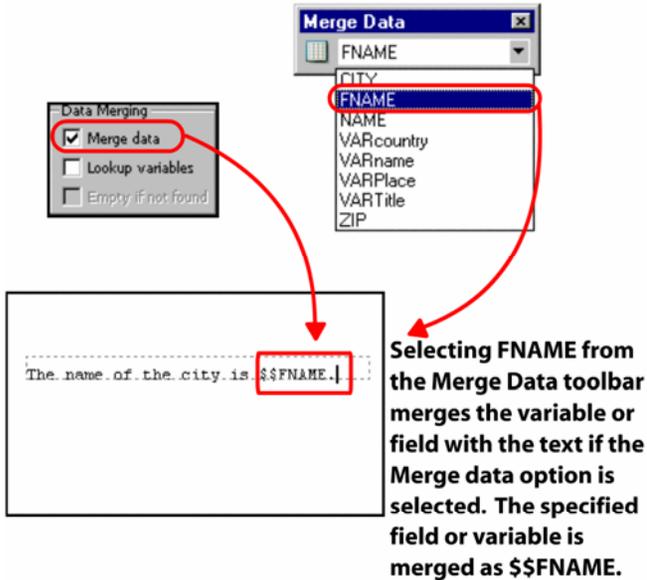
- 1 Open Vitesse in the Administration Mode. For more information, see the “Opening Vitesse in Administrator Mode” section in *Chapter 36: Vitesse Administrative Options*.
- 2 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 3 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.
- 4 Select the **Allow Field insertion in Form and Segment** option; this enables the **Fields file path** entry box.
- 5 Enter the XML file location in the **Fields file path** entry box or click  to use the *Open* dialog.
- 6 In the *Open* dialog, browse and select the required XML file.
- 7 Click **Open** and return to the *Preferences* dialog. The full path of the selected file displays in the **Fields file path** entry box.
- 8 Click **OK**.

After loading the XML file containing the variables, Vitesse allows you to merge the variables with text inserted in the form or segment. These variables become available in the *Merge Data* toolbar drop-down list, as explained in the “Merging Data with Static Text” section below.

Merging Data with Static Text

To merge the pre-defined variables or fields with static text in a form or segment:

- 1 Open a new or an existing form. For more information on creating a new form, see the “Creating a New Form” section in *Chapter 4: Working with Forms*.
- 2 Insert the text object and enter the required text.
- 3 From the **Settings** tab of the *Property View* window, select the **Merge data** option from the **Data Merging** area.
- 4 Double-click the text box and position the pointer where you want to merge the variable or field.
- 5 From the **Merge Data** toolbar, select the required field or variable from the drop-down list (for example, FNAME), and then click .



If you open the form or segment in the Database or XML mode, the \$\$FNAME will be replaced by value contained in the respective data file .

The specified variable or field is merged with the text as \$\$variable name.



You can also type \$\$*variable name*. directly in the textbox.

The \$\$*variable name*. will be replaced by the actual data when the form or segment is opened in Database or XML mode. If the **Merge data** option remains deselected, you can only view the syntax instead of the actual data.

Autosizing Text Objects

You can auto size a text object to adjust automatically to the amount of text it is displaying. This also helps prevent overlapping and extra spacing between text objects that are placed relative to each other. Vitesse provides the **Auto Size** option in the **Settings** tab of the *Property View* window to automatically resize and align the text, thus adjusting text boundary to fit the text size.

This feature is especially useful when working with variable length (delimited) fields or with fields that contain merged variables. For more information on Autosizing Variable Text Objects, see *Section 6, 7, and 8*.



When you use the **Auto Size** option for field objects, the size of the object becomes dynamic and is subject to change depending on the contents in the field. Therefore, the **Auto Size** option affects the placement of relatively positioned objects due to its changing behavior. The VIPP converter, on the other hand, requires an exact size of the field object for positioning the relative objects. Hence, this option must be disabled before exporting projects.

Formatting Text Objects

Vitesse offers different formatting features that you can apply to text and graphic text objects. You can add different background attributes to the text and set text alignment options. Additional options enable you to set word, character, line, and paragraph spacing. You can also set the indentation and tab properties of the text for precise placement on a page. Before applying the formatting options, select the **Settings** tab and choose one of the following from the **Formatting** area:

- **Normal** – uses the default text format.
- **Paragraph** – uses the paragraph text format.

Tasks in this section explain the formatting options in detail.

Setting Text Background Attributes

In Vitesse, you can select a text background and apply it to the selected text. Background attributes control the text background either by changing the color or by including an image.

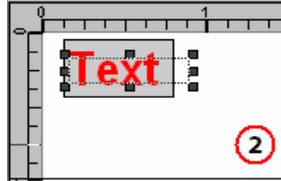
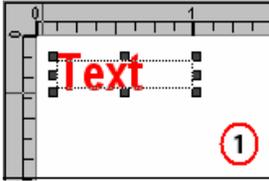
To set a text background attribute:

- 1 Select the text object in the design area.
- 2 From the *Property View* window, select the **Font** tab.
- 3 Select the required background attribute option from the **BAT** drop-down list.

The background attribute applies to the selected text object.



We recommend not using a background that makes text difficult to read.



Example: Setting Text Background Attribute to 1) NONE 2) FB_MED.



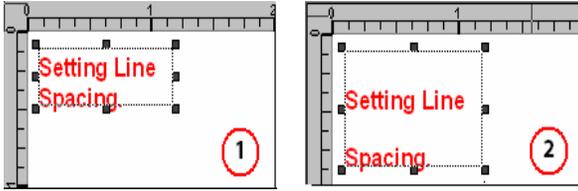
BAT keys are not available for graphic text objects. For information on the remaining options in the **Font** tab, refer to *Chapter 6: Working with Fonts*.

Setting Line Spacing

You can set the line spacing, which determines the amount of vertical space between lines of text. The **Line Spacing** area in the **Spacing** tab of the *Property View* window provides options for setting spacing between the lines.

To set line spacing, select the required text object or text lines and specify either of the following options:

- **Single** – sets single line spacing between the text lines.
- **1.5 lines** – sets line spacing to 1.5 lines between the text lines.
- **Double** – sets double line spacing between the text lines.
- **Exactly** – sets fixed line spacing exactly to an amount that you specify in the adjacent entry box. This option spaces all selected text lines evenly.
- **Multiple** – sets the line spacing that is increased or decreased by a percentage that you specify in the adjacent entry box. For example, setting line spacing to a multiple of 1.2 will increase the space by 20 percent, while setting line spacing to a multiple of 0.8 will decrease the space by 20 percent.



Example: Setting Line Spacing to 1) Single 2) Double.

Line Spacing options also apply to the graphic text object.



You can set the unit of measurement to specify the line spacing by right clicking the upward or downward arrows in the adjacent entry box, and then selecting **Units**>*unit name* from the pop-up menu.

Setting Paragraph Spacing

In addition to line spacing, you can also set paragraph spacing, thus determining the amount of space before a paragraph.

To set paragraph spacing:

- 1 Select the paragraphs between which you want to change the spacing.
- 2 From the *Property View* window, select the **Spacing** tab.
- 3 Set the required paragraph spacing in the **Pre Spacing** entry box.

The **Pre Spacing** option also applies to the graphic text object.



You can set the unit of measurement to specify the paragraph spacing by right clicking the upward or downward arrows in the adjacent entry box, and then selecting **Units**>*unit name* from the pop-up menu.

Aligning the Text

Alignment determines the position of each line of text in a paragraph relative to edges of the text box. In Vitesse, you can align text relative to the left and right edges of the text box. Text can be aligned to one edge of the text box, centered in the text box, or aligned to both edges of the text box (full justification). Use the various options provided under the **Justification** area in the **Spacing** tab of the *Property View* window to align text. Additional options to justify word and character spacing are also available.

To align text, specify either of the following:

- **Left** – click  to start text from the left-most position and extend right with normal spacing.
- **Right** – click  to start text from the right-most position and extend left with normal spacing.
- **Center** – click  to equidistance text from left and right.
- **Justify** – click  to start text from the left-most position and end in the right-most position; text is aligned evenly within the text object.
- **Word Justify** – click  to set spacing between the words in the text object.
- **Character Justify** – click  to set spacing between the characters within the same word.

For the graphic text object, you can only use the left, right, and center alignment options.

Indenting Text

Indents determine the distance between the margin of a paragraph and the beginning of the first line; where margins determine the amount of space between the border of a text box and a paragraph of text. In Vitesse, the **Indentation** area in the **Spacing** tab of the *Property View* window provides you with options for setting indentation values for the text:

- **First Line** – indents the first line of text only.
- **Left** – specifies the spacing between the left edge of the text object relative to the text inside the text object, per specified unit of measurement
- **Right** – specifies the spacing between the right edge of the text object relative to the text inside the text object, per specified unit of measurement.



Indentation options are not available for graphic text objects.



You can set the unit of measurement for indenting text by right clicking the upward or downward arrows in the entry box, and then selecting **Units>unit name** from the pop-up menu.

Setting Text Tabs

The **Tabs** area in the **Spacing tab** in the *Property View* window provides options for defining tab stop positions and alignment relative to the specified tab stop. You can also set the default tab value and alignment, extend text to the right, left, or center of the tab stop, add tab stops to a list and remove them from a list. Tabs are not applicable for graphic text objects.



A tab stop is a location on the horizontal ruler that indicates how far to indent text or where to begin a column of text.

Setting Tab Stops

To specify the tab stop position and type:

- 1 Select the required text object in the design area.
- 2 From the *Property View* window, select the **Spacing tab**.
- 3 In the **Tabs** area, specify the required tab stop position, per specified unit of measurement in the entry box and select either of the following text alignment options:
 - **Left** – aligns text to the left of the tab stop. The start position of the text is set such that it runs to the right as you type.
 - **Right** – aligns text to the right of the tab stop. The start position of the text is set such that it runs to the left as you type.
 - **Center** – aligns text to the center of the tab stop; thus centering the text on this position as you type.

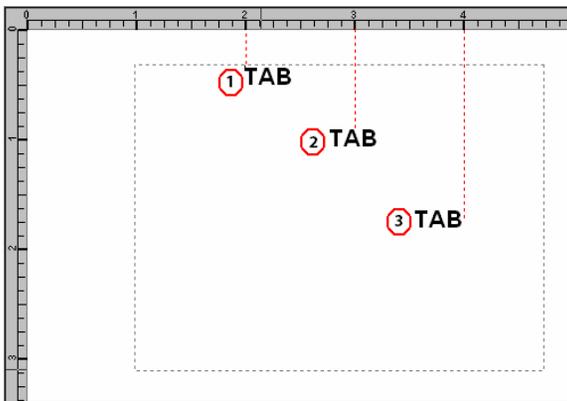


Image shows setting the tab stop position type 1) set to Left Tab stop 2) set to Center Tab stop 3) set to Right Tab stop.

Adding Tab Stops to a List

You can add a specified tab stop position and alignment option to the **Tabs** area list (or tab sequence) by clicking .

Deleting Tab Stops from a List

Select the tab stop position entry you want to remove from the list and click .

Modifying a Tab Stop in a List

To edit a tab stop added to a list:

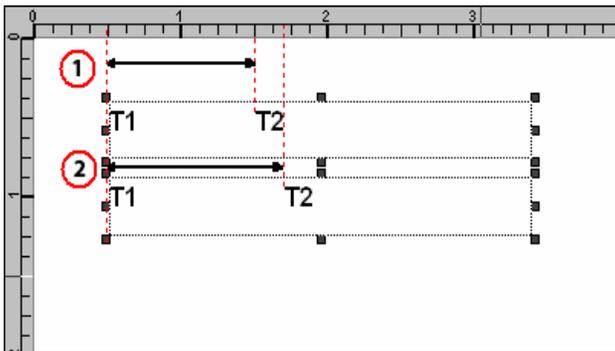
- 1 Select the required tab stop entry from the list.
- 2 Specify the new tab stop position and alignment.
- 3 Click . The selected tab stop is replaced by the modified value.

Selecting the Tab type

In Vitesse, you can also select the tab type as either:

- **Soft tab** – set to force text alignment per specified tab stop position; allows text overlapping if the previous text end point exceeds the specified stop position.
- **Hard tab** – set to align text per specified tab stop position, measured from the previous text end point; uses the default tab value if the previous text end point exceeds the specified tab stop position.

The following image illustrates the difference in functionality when selecting **Soft tab** and **Hard tab**.



- Looking at the text box marked by 1), T1 is inserted at a position set to 0.5 inch. The tab stop position is set to **Soft tab** with a tab value 1.00 inch and the alignment option set to left. Once T1 is inserted, press <Tab> on the keyboard; the cursor positions at 1.5 inch mark on the horizontal ruler. Insert T2 now. It is to be noted that when the tab stop position is set to **Soft tab**, the text T2 is aligned, measuring from the previous text T1's starting point.

- In the text box marked by 2), T1 is again inserted at 0.5 inch. The tab stop position is set to **Hard tab** with a tab value 1.00 inch and the alignment option set to left. Once T1 is inserted, press <Tab> on the keyboard; the cursor moves beyond 1.5 mark on the horizontal ruler. Insert T2 now.

It is to be noted that when the tab stop position is set to **Hard tab**, the text T2 is aligned, measuring from the previous text T1's end point.

Setting the Default Tab Value and Alignment for the Text

You can specify the default spacing between tab stops, per specified unit of measurement in the **Tab value** entry box from the **Default tab** area. You can also set alignment options relative to the specified tab stop.

To set alignment options for the **Default tab** value, specify the following:

- **Left** – aligns text to the left of the tab stop.
- **Right** – aligns text to the right of the tab stop.
- **Center** – aligns text to the center of the tab stop.



Vitesse uses tab stop position values from the tab sequence contained in the list. The default tab value is used with either **Soft** or **Hard tab** settings only when the tab stop position values in the list come to an end.

Properties specific to the graphic text object are explained in the “Formatting Graphic Text Objects” section below.

Formatting Graphic Text Objects

Once you have inserted a graphic text object in the design area, you can apply different formats to the object. The **Settings** tab in the *Property View* window provides a **Fancy** option in the **Formatting** area that is selected by default. This option further contains different formatting attributes, including the line and paragraph spacing, outline width and style, and rotation, thus meeting the needs of your document graphic presentations.



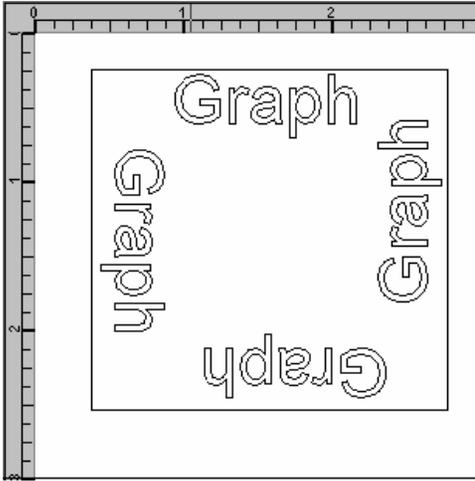
If you switch to the **Normal** or **Paragraph** option in the **Settings** tab, the **Fancy** option will no longer be available and the graphic text object will behave like a normal text object.

Setting Graphic Text Rotation

You can rotate the graphic text at the required angle by specifying the rotation value in the **Rotation** entry box from the **Setting** tab of the *Property View* window.



You can also set the rotation angle of the graphic text object by right clicking the upward or downward arrows in the **Rotation** entry box, and then selecting **Angle**>*angle name* from the pop-up menu.



Rotation examples for the graphic text object.



- For information on setting the outline style and width in the **Line** area, refer to the “Setting Common Object Properties” section in *Chapter 8: Using the Drawing Tools*. Outline style and width options are not available for the text object.
- For information on applying foreground and background colors and patterns using the **Color** tab, refer to *Chapter 11: Working with Colors*. Outline and fill pattern options are not available for the text object.
- For information on applying **Transformations**, refer to *Chapter 10: Applying Transformations*.

Using Rich Text from External Projects

Vitesse accommodates Rich Text Format (RTF) text using *WordPad* or a similar text editor. You can copy RTF text from any RTF-compliant software and paste it in the Vitesse design area. Text copied from an RTF application displays in the design area with its original presentation settings.

Importing Rich Text Format Text

Rich Text Format is the most portable file format used to exchange files cross-platform while still retaining all or much of its formatting. Importing text helps you save time in typing text.



This function is equivalent to copying text from an RTF file and pasting it in your page. You do not need to save the RTF file as a form resource.

To import text from the Rich Text Format (RTF) file:

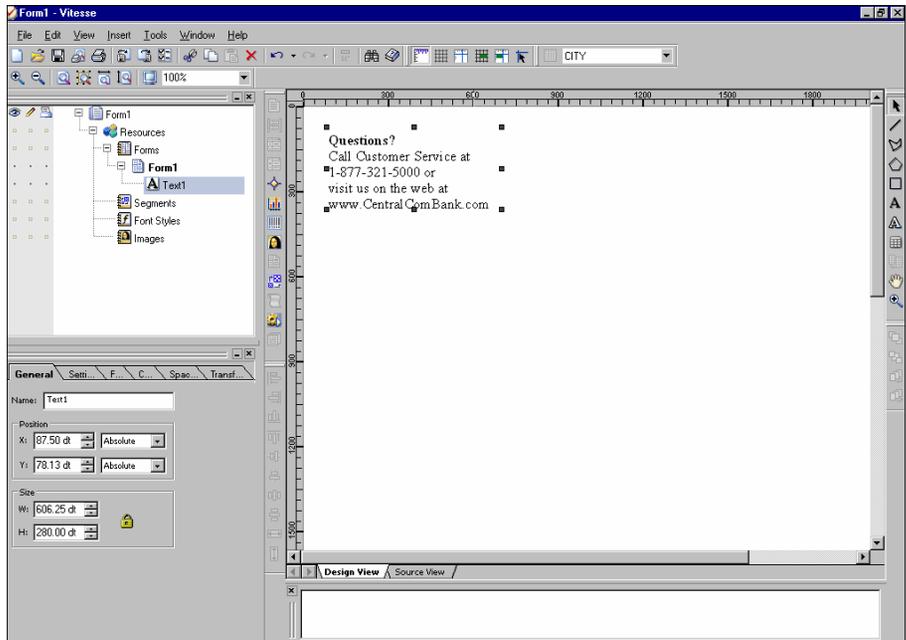
- 1 Open an existing or new project.
- 2 Insert the text object in the design area, and double-click it.
- 3 Right click the text box and select **Import** from the pop-up menu. *Import Text* dialog is displayed.



It is a good idea to draw a large text box or specify the size in the *Property View* window, prior to importing text to avoid unnecessary text wrapping.

- 4 Select **RTF Files (*.rtf)** from the **Files of type** drop-down list.
- 5 Browse to select the required RTF file from which you want to import the required text.
- 6 Click **Open** to import the selected file.

The text in the file imports into the design area.



Text has been imported from an RTF file.

Exporting Text to a Rich Text Format Text File

You can also export selected text as a Rich Text Format text file, using the following steps:

- 1 Open a new or existing project.
- 2 Insert the text box and double click it to enter text.
- 3 Right click in the text box and select **Export** from the pop-up menu.
The *Export Text* dialog is displayed.
- 4 Select **RTF Files (*.rtf)** from the **Save as type** drop-down list.
- 5 Browse to the location where you want to save the RTF file.
- 6 Click **Save**. The selected text is exported to the required RTF file in the required folder.

Joining Text Objects

Vitesse enables you to insert multiple text objects in the design area. You can join multiple text objects such that they behave as one for the purpose of moving, resizing, or rotating. Joining text objects is enabled when a multiple text selection is made.

To join multiple selected text objects, click  on the *Standard* toolbar or select Tools>Join Text from the menu bar. Selected objects are combined as a single object.



You can select multiple text objects by selecting one text object at a time and holding down <Shift> or <Ctrl> while you click each object. To deselect all of the objects, click any place but on the objects.

Using the Spell Checker

Vitesse offers the spell check feature which employs a custom dictionary to check for misspellings in a document. Spell checker displays a list of words that are similarly spelled from the dictionary. When the spell check feature identifies a word not found in the specified dictionary or dictionaries, you can select how to handle the word as mentioned in the steps below.



You can customize the way Vitesse checks spelling of text within the text objects by setting the spell check detail level. This is done by identifying the **Spell Checker Depth** value within a range of 0 -100. For more information on customizing the spell check detail level, see the “Setting Spell Checker Depth” section in this chapter.

To use the spell checker:

- 1 Insert text within a text box in the design area.
- 2 Right click the text in the text box, and select **Spell Check** from the pop-up menu.
The *Spelling* dialog is displayed.
- 3 As appropriate, select either of the following:
 - **Not in Dictionary** – identifies the incorrectly spelled word; enter the correct word.
Or
 - **Suggestion** – lists options for similarly spelled words from the dictionary; select the correct word from the list.

- 4 Do one of the following:
Click **Replace** to use the suggested word displayed by the spell checker and change the spelling of the incorrect word.
Or
Click **Ignore** to disregard the options displayed by the spell checker.
Or
Click **Cancel** to return to the design area without using the spell checker.
Vitesse displays a message confirming your selection of the above options.



The contents of suggested words is determined only by spelling, so any instances of terms that seem inappropriate in context are completely coincidental.

Specifying Dictionary Files for Spell Checking

You can customize the way Vitesse checks the spelling of text within the text objects by employing a custom dictionary. The dictionary used for referring to words for spell checking is a third party component and is acquired and placed at *drive*: `\Program Files\Vitesse\Config\Spell`. You can also spell check text in different languages by acquiring an off the shelf custom dictionary either for all languages or for a specific language. Options for adding, removing, and rearranging the dictionary list are explained in subsequent tasks.



When the spell checker encounters a word it does not recognize, it determines which words in its dictionary are similarly spelled and displays a list of those words.

Adding a Dictionary File

To add a dictionary file:

- 1 Acquire a custom dictionary and place at the recommended location *drive*: `\Program Files\Vitesse\Config\Spell`.
- 2 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 3 From the *Preferences* dialog, select the **Dictionary** category under the **Defaults** node; a list of predefined dictionaries are displayed in this category.
- 4 Click  to add a new row to the list.
- 5 Click  to open the *Select dictionary file* dialog and browse for the required dictionary file.
- 6 Select **Dictionary files (*.tlx, *.clx)** from the **Files of type** drop-down list.

- 7 Select the required file.
- 8 Click **Open** and return to the *Preferences* dialog. The full path of the selected file displays in the newly added row.
- 9 Click **OK** to add the dictionary file. The spell checker will accept words from this dictionary file as correct.



You can also double-click an existing row and specify a different dictionary. Vitesse enables you to add as many custom dictionaries as you like.

Deleting a Dictionary File

To remove a dictionary file from the list, select the required row specifying the file path, and then click  in the *Preferences* dialog.

Rearranging the Dictionary Files List

Vitesse enables you to re arrange the dictionary file list. Selecting a dictionary file row and clicking  moves the selected row up one level in the list. Similarly, selecting a dictionary file row and clicking  moves the selected row down one level in the list.

Setting Default Text Attributes

Vitesse enables you to customize the design environment by setting the default text attributes. Default settings are applied every time you open a project.

Setting the Default Text Justification

To set the default text justification:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Text** area, select a default alignment option from the **Alignment** drop-down list.
- 4 Click **OK**. Text and graphic text objects are inserted in the design area with the specified alignment the next time you place them.

Setting the Default Line Spacing

To set the default line spacing:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Text** area, select a default line spacing option from the **Line Spacing** drop-down list.
- 4 Click **OK**. Text and graphic text objects are inserted in the design area with the specified line spacing the next time you place them.



You can set default line spacing to exactly fixed line spacing by specifying an amount in the adjacent entry box. Similarly, you can also set default line spacing to multiple line spacing, which increases or decreases by a percentage that you specify in the adjacent entry box.

Setting the Default Paragraph Spacing

To set the default paragraph spacing:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Text** area, set a default paragraph spacing value in the **Paragraph Spacing** entry box.
- 4 Click **OK**. Text and graphic text objects are inserted in the design area with the specified paragraph spacing the next time you place them.

Setting the Default Tab Value and Alignment

To set the default tab value and alignment:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Tab** area, set a default tab size in the **Default Size** entry box.
- 4 Select a default tab alignment option from the **Alignment** drop-down list.

- 5 Click **OK**. Text and graphic text objects are inserted in the design area with the specified tab size and alignment the next time you place them.

Setting the Default Text Indentation

To set the default text indentation:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Indentation** area, set the default text indentation values in the **Left Indent**, **First Line Indent**, and **Right Indent** entry boxes respectively.
- 4 Click **OK**. Text and graphic text objects are inserted in the design area with the specified indentation values the next time you place them.

Setting the Spell Checker Depth

You can customize the way Vitesse checks spelling of text within text objects by setting the spell check detail level.

To set the spell checker depth:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Text** category under the **Defaults** node.
- 3 In the **Text** area, set the spell check detail level in the **Spell Checker Depth** entry box.
- 4 Click **OK**. Spell checking will be carried out with the specified detail level the next time you use the spell checker.



You can set the **Spell Checker Depth** value within a range of 0 - 100, where the value 100 indicates maximum spell check detail level, and 0 indicates minimum spell check detail level.

Setting the Data Switch

Vitesse enables you to set a default VIPP source code character combination for specifying text attribute changes. However, it is essential that this character combination does not appear within the application data or text objects. In such a situation, you can change the combination and set to a unique character combination to avoid confusion at print time.

To set the data switch:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **General** category.
- 3 Set a default VIPP source code character combination for specifying text attribute changes in the **Data Switch** entry box.
- 4 Click **OK**. The specified data switch code appears before every text attribute change in the source view.

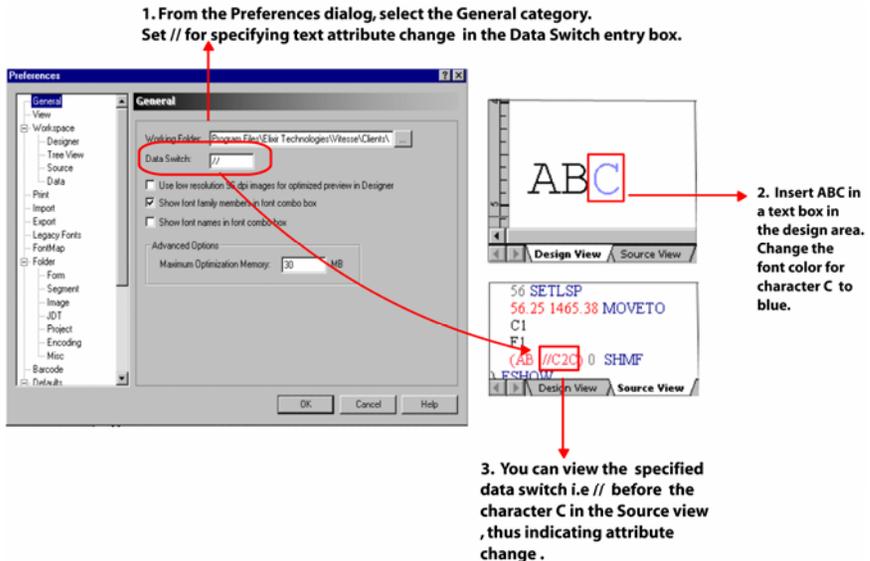


Image shows setting the data switch and viewing it for every text attribute change in the source view.

Finding Text

You can use Vitesse to quickly search for every occurrence of a specific word or phrase inserted on a page.

To find text within a page:

- 1 Select Edit>Find from the menu bar to open the *Find* dialog.
- 2 As appropriate, enter or select the following:
 - **Find what** – specifies the text you want to search for in the page.
 - **Match case** – finds occurrence(s) matching the text string in upper/lower case letters only.

- 3 Click **Find Next** in the *Find* dialog to display the next occurrence of the specified text string in the page. Once all occurrences are found, a confirmation message is displayed.



You can also press <Ctrl>+<F> on the keyboard to find a text.

Replacing Text

You can use Vitesse to automatically replace selected text in the page.

To replace selected text in the page:

- 1 Select Edit>Replace from the menu bar to open the *Replace* dialog.
- 2 As appropriate, enter or select the following:
 - **Find what** – specifies the text you want to search for in the form.
 - **Replace with** – specifies the text string to replace the text that has been found.
 - **Match case** – finds occurrence(s) matching the text string in upper/lower case letters only.
- 3 Do the following:

Click **Find Next** to display the next text string occurrence.

And/Or

Click **Replace** to replace the current occurrence text that has been found only.

Or

Click **Replace All** to replace all occurrences of the text that has been found.



You can also press <Ctrl>+<H> to find and replace text on the page.

User Notes:

Chapter 10: Applying Transformations

This chapter describes how to apply different formatting options on field, text, and graphic text objects and covers the following tasks:

- Formatting a Number String
- Extracting Sub-Strings from Text or Data Strings
- Stripping Characters and Blanks
- Formatting a Time String
- Changing String Case
- Concatenating Strings
- Formatting Strings as Barcodes

Transforming Strings

Vitesse enables you to transform variable data or constant text strings by using the various formatting options available in the **Transform** tab of the *Property View* window. You can define formats for numeric strings, time strings, and present strings as barcodes. You can also extract sub-strings, truncate extra spaces and characters, and concatenate strings. The techniques for defining and applying various formatting options on field, text, and graphic text objects are explained in subsequent tasks.

Formatting a Number String

To format/transform a string of numeric characters:

- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.



If the *Property View* window is not displaying in the workspace, select **Window>Properties** from the menu bar.

- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select the **Format number** option; adjacent drop-down list is enabled.
- 4 Select the required format mask from the **Format number** drop-down list to apply it on the selected object. The **Format number** drop-down is editable; you can also add a new format mask in the list by typing it in the entry box.

Creating Format Masks

A mask is syntax used to format the selected data. You can use the eight predefined characters specified in the **Regional** properties of the **SubDoc** node to create masks (formats) for transforming numeric strings. These characters act as placeholders for numbers, delimiters, and signs in numeric strings. For more information regarding the predefined characters, refer to *Chapter 16: Setting Document Properties*. The examples included in this chapter have regional settings set to the **US** standards.

You can also define customized characters other than the predefined meta characters to append to a numeric string, or use as a separator between numbers, and then use these characters to define format masks in the **Transform** tab.

For example, you can format a string of numeric characters 0123450 as follows:

- @ @ / @ @ / @ @ – formats the string as 12/34/50.
- \$ @ , @ @ , @ @ @ . ## – formats the string as \$ 1,23,450.00.
- ## , ## , ### – formats the string as 01,23,450.

In these examples, the characters ‘/’ and ‘,’ are used to separate numbers; ‘\$’ has been appended to a number string to signify that the value is in dollars; the mask ‘#’ prints a character, even if that character is a leading zero; and ‘@’ does not print leading zeros.

Adding a New Format Mask

To add a new format mask for formatting numbers:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 Select the **Transform** category under the **Defaults** node to view the list of number formatting masks.
- 3 Click  to add a new row to the list.
- 4 Type the new number format in the newly added row.

- 5 Click **OK** to save the new format in the list.

The existing and newly created format masks are displayed in the **Format number** drop down list in the **Transform** tab. Also, any entry that you create in the **Format number** drop-down list is added to the list of formats in the **Transform** category.



You can also perform the following operations in the list:

- To delete a selected entry, click
- To move selected entry up one level in the list, click
- To move selected entry down one level in the list, click
- To edit an existing number format, double-click the corresponding entry in the list and make your changes.



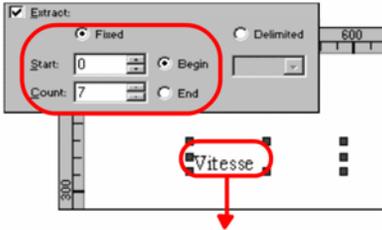
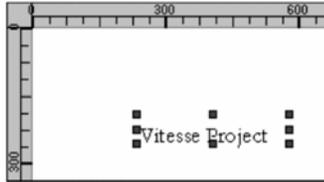
Vitesse provides different transform options for different string types. You must select a formatting option that is applicable to an object to get the required output. For example, applying **Numeric** transform to a string of characters will not produce the correct output and is indicated as an “Error” in the design area. Similarly, for other invalid transform options applied to objects, the corresponding error messages are displayed in the design area.

Extracting a Fixed Length Sub-String from Text

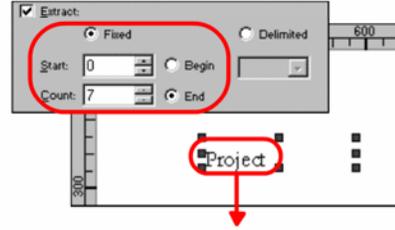
To extract a sub-string of fixed length from text:

- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 1 From the *Property View* window, select the **Transform** tab.
- 2 Select **Extract** to enable the associated options.
- 3 Select **Fixed**, and then choose one of the following.
 - **Begin** – starts the character count from the beginning of the text/data string.
 - **End** – starts the character count from the end of the text/data string.
- 4 Depending on your selection, specify the starting position of the sub-string to be extracted in the **Start** entry box. You can also set the value by clicking the upward and downward arrows in the **Start** entry box.
- 5 In the **Count** entry box, specify the length of the sub-string to be extracted.

Design area displaying a text object.



A fixed length sub-string of 7 characters extracted from the beginning of text.



A fixed length sub-string of 7 characters extracted from the end of text.

Image showing two sub-strings extracted from a single string by applying different specifications.

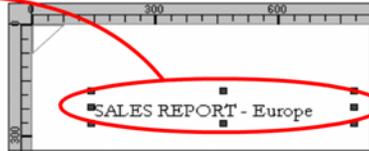
Extracting a Variable Length Sub-String from Text

To extract a sub-string of variable length containing a delimiter:

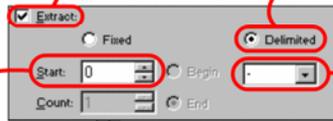
- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select **Extract** to enable the associated options.
- 4 Select **Delimited** to extract a variable length sub-string from the selected object.
- 5 In the **Start** entry box, specify the starting position of the sub-string to be extracted. You can also set the value by clicking the upward and downward arrows in the **Start** entry box.

- 6 In the associated drop-down, select the delimiter that indicates the end of the sub-string. You can also type a new character that is not available in the drop-down list.

1. Select the required object to display the associated properties in the Property View window.



2. In the Transform tab, select Extract to enable the associated settings.



3. Select Delimited to extract a variable length sub-string.

4. Specify the start position of the sub-string in the text.



5. Specify the delimiter indicating the end of sub-string.

6. Sub-string extracted starting at '0' position and ending at '-' delimiter.

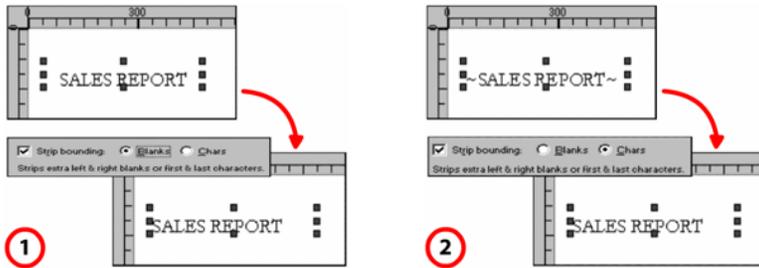
Image showing the steps involved in extracting a variable-length, delimited string.

Stripping Characters and Blanks

To truncate the extra spaces or characters from a string of text or data:

- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select **Strip bounding** to enable the associated options.
- 4 Select the required option from the following:
 - **Blanks** – truncates the starting and trailing blank spaces from the string.
 - **Chars** – truncates the starting and trailing characters from the string.

The images below show examples of applying the **Strip bounding** transform option.



Images displaying: 1) Blanks truncated from a string. 2) Tilde characters truncated from a string.

In the first example, the string “SALES REPORT” has leading and trailing space characters that are stripped off using the **Blanks** option. In the second example, the design shows a text object that has a tilde character bounding the string “SALES REPORT.” Using the **Chars** option, the two tildes are truncated from the string.

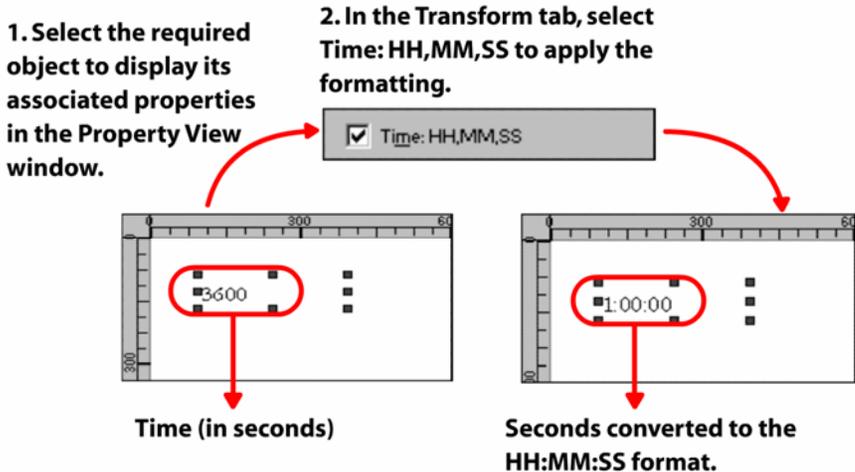
Formatting a Time String

Vitesse lets you convert time expressed in seconds to a standard time format (HH:MM:SS).

To reformat the time string:

- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select the **Time: HH,MM,SS** field to reformat the time (in seconds) to the HH:MM:SS format.

Formatting Time String example: 3600 seconds is formatted to display as 1:00:00 (1 hour) as shown in the figure below:



Changing String Case

This option enables you to switch between upper, lower, or title case for the entered text.

To change the case of a text\data string:

- 1 Select the required field, text or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select **Force case to** in order to enable the adjacent drop-down list.
- 4 From the drop-down list, select the required option from the following:
 - **Lower**– formats all characters in the string to lower case.
 - **Title Case**– formats the string such that the first character of each word is set to upper case while the remaining characters are set to lower case.
 - **Upper**– formats all characters in the string to upper case.

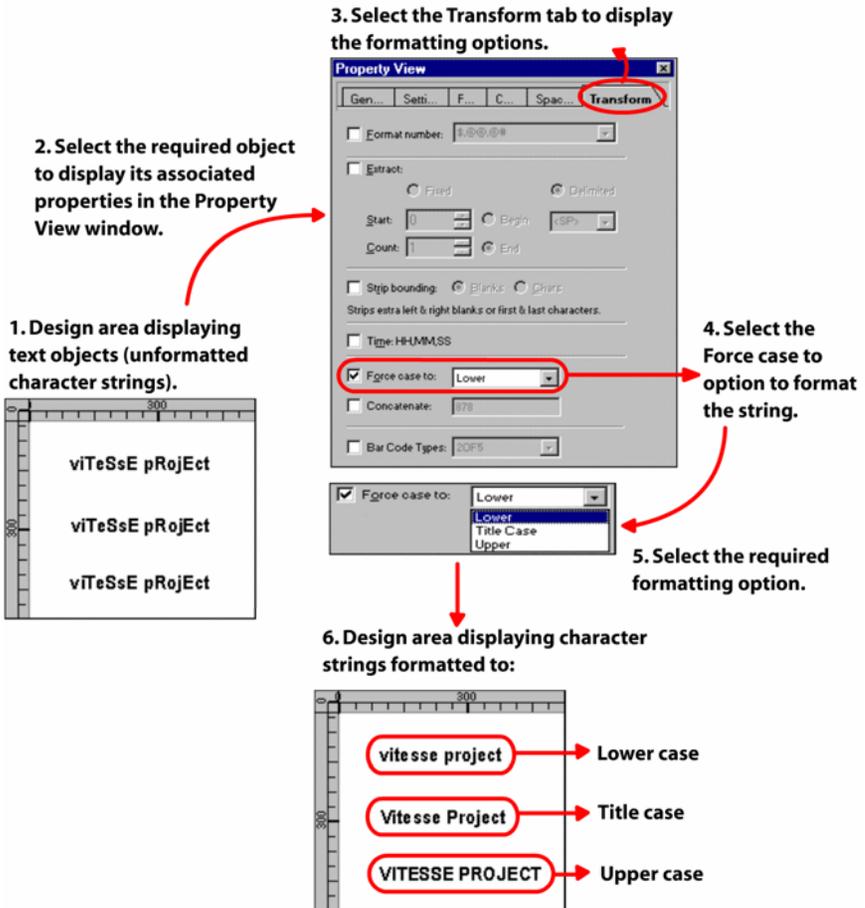


Image showing the steps involved in changing the case of strings.

Concatenating Strings

You can also concatenate constant text or variable data at the end of a field, text, or graphic text.

- 1 Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- 2 From the *Property View* window, select the **Transform** tab.
- 3 Select **Concatenate** to enable the adjacent entry box.

- Specify the required text that is to be concatenated in the associated entry box. You can also concatenate variable data by specifying the variable name (variables that are defined at the **MasterDoc** level) or the data field name (corresponding to the fields in data files) in this entry box.

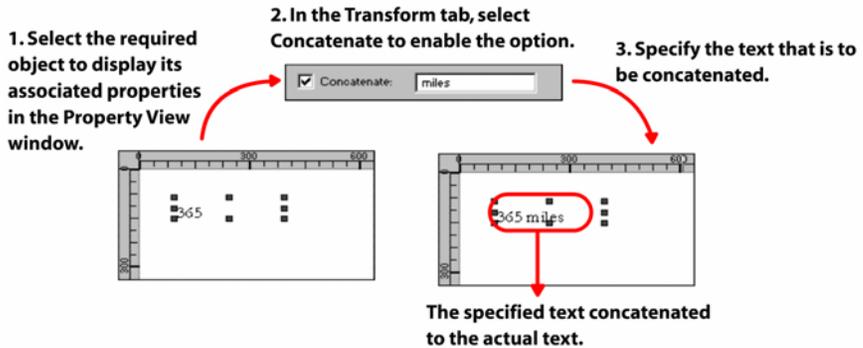


Image showing the steps involved in concatenating strings.



When specifying variable data for concatenating with a text string, also enable the **Merge data** option. For more information on merging data, refer to *Section 6, 7, and 8* in this User Guide.

Formatting Strings as Barcodes

Besides using the barcode objects, Vitesse also enables you to present the field, text, and graphic text objects as barcodes.



To view text\data in a barcode format, ensure that a barcode font type has also been applied to it.

- Select the required field, text, or graphic text object in the design area; associated properties are displayed in the *Property View* window.
- From the *Property View* window, select the **Transform** tab.
- Select **Bar Code Types** to enable the adjacent drop-down list.
- Click the **Bar Code Types** drop-down to view the list of standard barcode types supported in VIPP. Select the required barcode type to format the selected object.



Each of the standard barcode types has associated properties. For more information on a certain barcode type, refer to the relevant documentation.

Chapter 11: Working with Colors

This chapter describes activities and tasks for working with colors, color palettes, and patterns. Tasks include:

- Selecting a Foreground and Background Color
- Working with Color Palettes
- Loading a Palette
- Adding a New Color to the Color Palette
- Editing a Color
- Deleting a Color
- Saving a Palette
- Using Patterns
- Applying Line and Fill Patterns
- Setting Default Color and Pattern Attributes

Using Colors

Vitesse provides a variety of ways to create and/or modify colors. Using the default color palette (contains a list of predefined VIPP supported colors) or a customized palette, you can choose colors to apply to the foreground/border or background/interior of an object that either you are about to create, or to an object that is already on the design area. Applying a foreground color to an object fills the outline of the object with that color. Applying a background color to an object fills the interior space of the object with that color. Colors can be applied on drawing objects, tables, charts, text, and data objects using the **Color** tab in the *Property View* window.

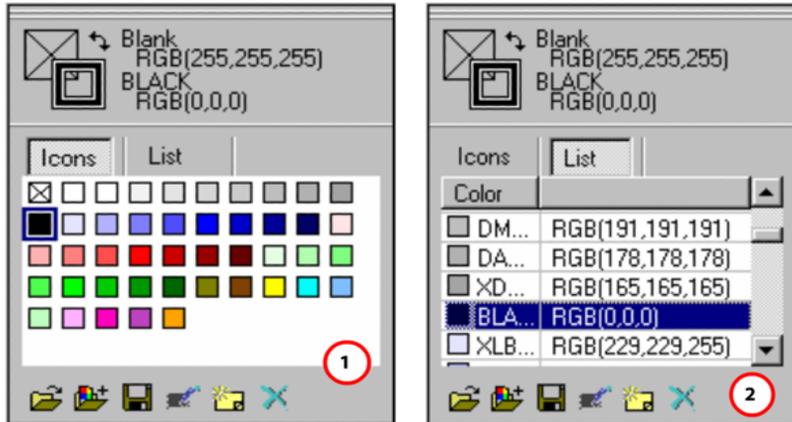


The **Color** tab is available only for drawing objects, tables, charts, text, and data objects.

The **Color** tab further contains two tabs for viewing the colors:

- **Icons** tab – displays the default color palette. You can choose a foreground and/or background color by clicking and/or right clicking the required color. Moving the cursor on a color displays the color name and associated values at the right-most corner of the palette, adjacent to the **List** tab.

- **List** tab – clicking this tab displays the default colors list in **Color** and **Value** pairs/columns, where the **Color** column identifies the color name and the **Value** column identifies the associated value in RGB colors. You can scroll the list vertically using the vertical scroll bar.



Color Property Sheet Views displaying 1) Icons tab, 2) List tab.

The top most area in the **Color** tab contains selection boxes  and  that enable you to choose whether to apply the foreground/fill and/or background/outline colors to objects. Prior to applying a color to an object, you need to ensure that the appropriate selection box is active, where an active selection box overlaps the non-active selection box. You can also click  to toggle the foreground and/or background color selections; corresponding color values display adjacent to the toggle arrows.

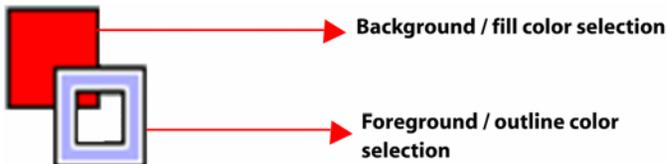


Image shows the foreground and background selection boxes where foreground selection box is the active box.

The following tasks provide information on applying foreground and background colors to objects.



The  color box in the palette identifies no color with RGB values set to 255 each.

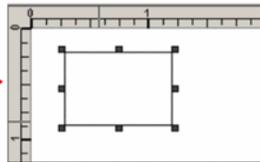
Selecting a Foreground Color

- 1 To set a foreground color for an object:
- 2 Select the required object in the design area.
- 3 From the *Property View* window, select the **Color** tab.
- 4 Select the required tab (**Icons** or **List**) for viewing colors.
- 5 Click  to fill the foreground/outer boundary of the object.
- 6 Select the required color from the color palette or list.
The object displays in the selected foreground color.

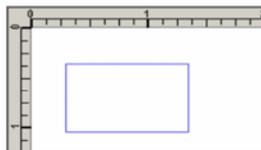


You can also apply foreground color to a selected object by clicking a color in the palette or list.

1. Select box object in the design area. The default foreground color of the box object is in black.



2. Select the Color Tab in the Property View window and click the foreground color button.



4. Object displays in the selected foreground color.



3. Select the required color from the color palette.

Selecting a foreground color for the box object.

Selecting a Background Color

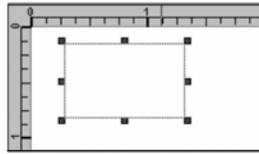
To set a background color for an object:

- 1 Select the required object in the design area.
- 2 From the *Property View* window, select the **Color** tab.
- 3 Select the required tab (**Icons** or **List**) for viewing colors.
- 4 Click  to fill the background/inside of the object.
- 5 Select the required color from the color palette or list.
The object displays in the selected background color.

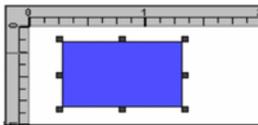
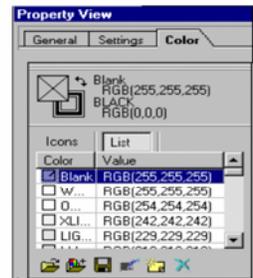


You can also apply the background color to a selected object by right clicking a color in the palette or list.

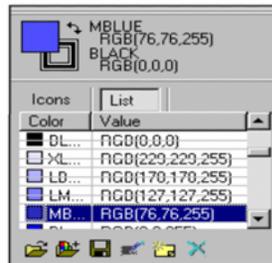
1. Select a box object in the design area. The default background color of the box is set to blank.



2. Select the Color tab in the Property View window and then select the List view. Click the Background selection button.



4. Object displays in the selected background color.



3. Select the required color from the color list.

Selecting a background color for the box object.

Working with Color Palettes

In Vitesse, you can quickly adjust the palette colors to meet the needs of your document color presentations or remove colors that are no longer used. Vitesse contains a default color palette, from which you can create and append custom colors and save as a customized palette. You can duplicate colors, delete colors from the palette, edit colors in the palette, or reload the default palette if you have replaced it.

Loading a Palette

To load a default palette:

- 1 Select the required object in the design area.
- 2 From the *Property View* window, select the **Color** tab.
- 3 Click  on the **Color** tab. The default palette replaces the current palette.

Adding a New Color to the Color Palette

You can create a new color and add it to the color palette using the *Color Selector* dialog. Colors can be created and added in the *Color Selector* dialog by either of the following features:

- Using the Color Spectrum
- Using the Color Model

Each of these methods is explained in the following sections.

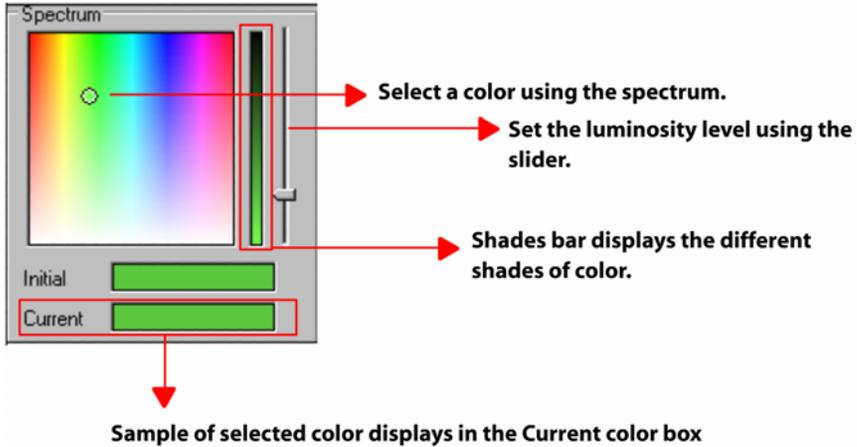
Adding a Color Using the Color Spectrum

To create a color using the spectrum and then add it to the palette:

- 1 Select the required object in the design area.
- 2 From the *Property View* window, select the **Color** tab.
- 3 Click  to open the *Color Selector* dialog.
- 4 To select a color from the **Spectrum** area, click the required color area on the spectrum. The shades available for the selected color automatically display in the adjacent shades bar.
- 5 As appropriate, select a shade by adjusting the slider next to the shades bar.



Equivalent values for the selected color also display in the **Model** area.



Selecting a color from the Spectrum.

6 Review the following:

- **Initial** – specifies the initial spectrum color.
- **Current** – specifies the current spectrum color set using the slider.
- **Gray Scale** – displays the gray color component in the selected color; the value is between 0 – 255. The slider next to the **Gray Scale** shades bar automatically moves to reflect the amount of gray color component used in the selected color.



The **Initial** color box is set to green by default. The color shade selected by adjusting the slider adjacent to the shades bar reflects the chosen color/shade in the **Current** color box.

7 Enter a unique name to identify the selected color in the **Color Name** entry box.



HCOLOR is a built in variable and requires a Highlight Color as its value. If the value for the HCOLOR specified is different from the value set as the printer HCOLOR, the printer prints the object in the color set as the printer HCOLOR.

8 Click **OK** to add the new color to the palette.



The newly created color saves in the palette only for the project and the external resources you are currently working. If you want to use this color in other projects, you need to add it to the palette again. Vitesse saves the newly created color in the ColorFile.clr file.

Adding a Color Using the Color Model

To create a color using the color model and then add it to the palette:

- 1 Select the required object in the design area.
- 2 From the *Property View* window, select the **Color** tab.
- 3 Click  to open the *Color Selector* dialog.
- 4 In the **Model** area, choose one of the following models from the **Selected Color Model** drop-down list:
 - **CIELAB** – sets the color model to CIE L*a*b.
 - **CMYK** – sets the color model to cyan-magenta-yellow-black.
 - **HSB** – sets the color model to hue-saturation-brightness.
 - **RGB** – sets the color model to red-green-blue.
- 5 Depending on the chosen model, choose the required color by either adjusting the sliders, or by specifying a numeric value in the adjacent boxes. Adjustments that you make using the sliders or entry boxes are also reflected in the **Spectrum** area.



The **Gray Scale** value is read-only and adjusts automatically in correspondence with the selected color.

- 6 Enter a unique name to identify the selected color in the **Color Name** entry box.

- 7 Click **OK** to add the new color to the palette. The following image depicts the process of creating and adding a new color to the palette.

2. Select the required model from the Selected Color Model drop-down

1- Click Insert button to open the Color Selector dialog. list.

3. Depending on the model, associated color information displays in the Color Selector dialog. Choose the required color.

4. Click OK to add the new color "Aqua" to the palette.

Adding a color to a palette using the RGB model.

The following table lists the available models and their associated color values.

The **CIELAB** model contains the following values:

- L – sets the color lightness; the value ranges between 0 – 100.
- A – sets the color chromatic component A (red to green); the value ranges between 0 – 100.
- B – sets the color chromatic component B (blue to yellow); the value ranges between 0 – 100.

The **CMYK** model contains the following values:

- C – sets the cyan component value; the value ranges between 0 – 100.
- M – sets the magenta component value; the value ranges between 0 – 100.
- Y – sets the yellow component value; the value ranges between 0 – 100.
- K – sets the black component value; the value ranges between 0 – 200.

The **HSB** model contains the following values:

- H – sets the color hue; the value ranges between 0 – 359.
- S – sets the color saturation; the value ranges between 0 – 100.
- B – sets the color brightness; the value ranges between 0 – 100.

The **RGB** values contain the following values:

- R – sets the red component value; the value ranges between 0 – 255.
- G – sets the green component value; the value ranges between 0 – 255.
- B – sets the blue component value; the value ranges between 0 – 255.

Editing a Color

You can also edit custom/user-defined colors in the color palette.

1 Select a custom color from the palette.

2 Click  to open the *Color Selector* dialog.



is enabled only for editing a custom color. For VIPP supported (default) colors, it appears in the disabled mode .

- Choose the required color from the **Spectrum** or **Model** area and set the properties specific to the selected color model. When editing a custom color selected from the palette, the **Initial** color box displays the original color and the **Current** color box shows the modified color.
- Click **OK**. The modified color replaces the selected color.



You can also double-click a custom color in the palette to edit the color using the *Color Selector* dialog.



If you double-click a default color, the *Color Selector* dialog displays default color settings. If you make any changes to the default color, Vitesse displays the *Color Selector* dialog again, enabling you to add the modified color as a new color to the palette. Default color settings cannot be modified.

- Select a custom color that you want to replace from the palette.
- Click the edit button to open the *Color Selector* dialog.
- Select the required model from the **Selected Color Model** drop-down list and choose the appropriate color.
- Click **OK** to replace the selected color with the modified color.

Editing a custom color in the palette.

Deleting a Color

You can delete a selected custom color from the palette by clicking  in the palette.



is enabled for deleting a custom color only. For VIPP supported colors, it appears as disabled .

Saving a Palette

You can save user-defined colors to the default palette by clicking  in the palette. Vitesse saves the newly created colors in the ColorFile.clr file.



The newly created color is saved in the palette only for the project and the external resources you are currently working. If you want to use this color in other projects, you need to add it to the palette again.

Using Patterns

In Vitesse, you can bring attention to drawing objects using the patterns available in the **Color** tab of the *Property View* window. You can either add a pattern to an object or you can replace an existing pattern with a different pattern.

The **Color** tab features a number of pre-defined black and white shading patterns to apply on both the object border and interior. When creating a full-color object, you can select patterns based on a color selected from the color palette in the **Color** tab.

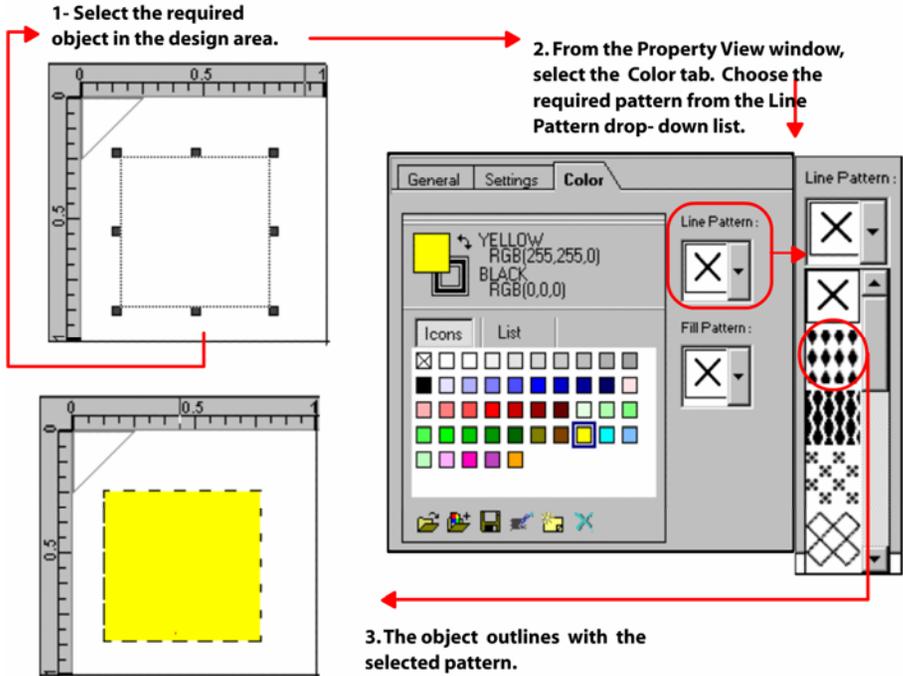
Vitesse enables you to apply line and fill patterns to design objects such as lines, paths, polygons, boxes, and tables. You can use different patterns and colors on different copies of the same object.

Applying Line Patterns

To apply a line pattern to an object:

- 1 Select the required object.
- 2 From the *Property View* window, select the **Color** tab.
- 3 Select the required pattern from the **Line Pattern** drop-down list.

The object outlines with the selected pattern.

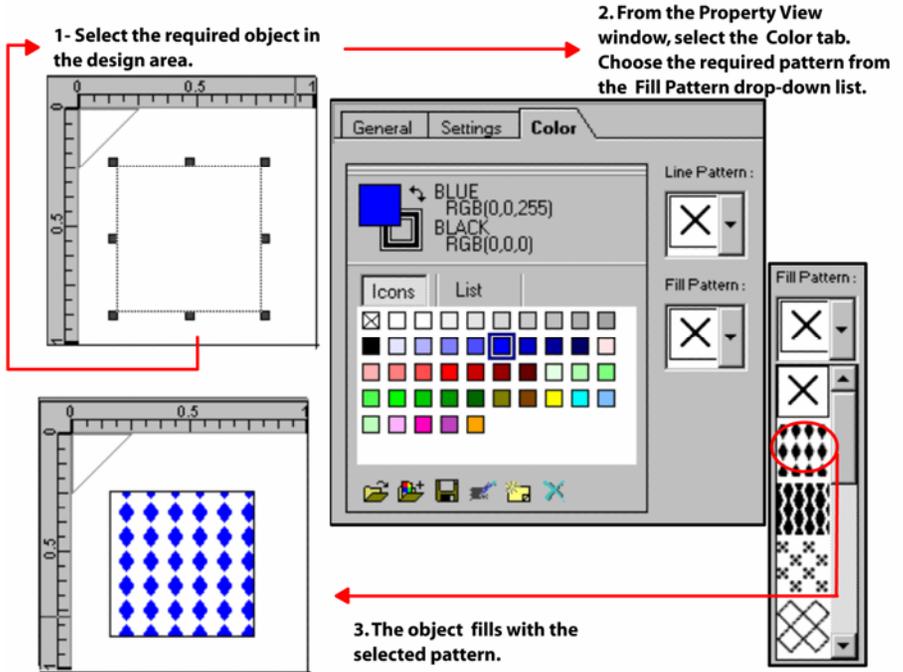


Applying a line pattern to a selected object.

Applying Fill Patterns

To apply a fill pattern to an object:

- 1 Select the required object.
 - 2 From the *Property View* window, select the **Color** tab.
 - 3 Select the required pattern from the **Fill Pattern** drop-down list.
- The object fills with the selected pattern.



Applying a fill pattern to a selected object.

Setting Default Color and Pattern Attributes

Vitesse enables you to customize the design environment by setting the default color and pattern attributes. Default settings are applied every time you open a project.

Setting Default Foreground Color

To set the default foreground color:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog .
- 2 From the *Preferences* dialog, select the **Color** category under the **Defaults** node.
- 3 Click  to apply the default foreground color.
- 4 Select the required color from the palette.
- 5 Click **OK**.

Objects are inserted in the design area with the specified foreground color the next time you draw an object.

Setting Default Background Color

To set the default background color:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Color** category under the **Defaults** node.
- 3 Click  to apply the default background color.
- 4 Select the required color from the palette.
- 5 Click **OK**.

Objects are inserted in the design area with the specified background color the next time you draw an object.

Setting Default Line Patterns

To set the default line pattern:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Color** category under the **Defaults** node.
- 3 Select a default outline pattern from the **Line Pattern** drop-down list.
- 4 Click **OK**.

Objects are inserted in the design area with the specified line pattern the next time you draw an object.

Setting Default Fill Patterns

To set the default fill pattern:

- 1 Select Tools>Preferences from the menu bar to open the Preferences dialog.
- 2 From the *Preferences* dialog, select the **Color** category under the **Defaults** node.
- 3 Select a default fill pattern from the **Fill Pattern** drop-down list.
- 4 Click **OK**.

Objects are inserted in the design area with the specified fill pattern the next time you draw an object.

User Notes:

Section IV: Creating a VIPP Project

User Notes:

Chapter 12: Creating a Native Mode Project

This chapter describes activities and tasks for creating VIPP native mode projects. Included are instructions for the following:

- Using the Native Mode
- Creating a New Native Mode Project
- Selecting the Paper Size
- Selecting the Paper Orientation

Using the Native Mode

You can use the native mode for creating, positioning, and printing static data, such as text, images, lines, boxes, forms, and segments on a physical page. A native mode project accommodates applications supporting fully composed native mode data streams. Native mode projects do not support variable data handling.

Creating a New Native Mode Project

To create a new native mode project:

- 1 Select File>New from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog by either:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <N> on the keyboard.

- 2 From the **New** tab, select the **Native Mode** icon , and then click **New**.
- 3 The *Native Mode* dialog is displayed with the **General** tab selected.
- 4 In the **General** tab, set the following **Project** specifications:
 - **Name** – specifies a unique name identifying the project. The name can be 30 characters in length and can include alphanumeric characters.
 - **Client** – lists the clients that are available for selection & creates a directory for the project in the selected client folder.



You can also create a new client by entering the appropriate name in the **Client** drop-down list. Vitesse asks for confirmation before creating the new client.

- **Path** – displays the path of the project directory in the selected client.

- 5 Click **OK**.

While creating the native mode project, you can also specify the required paper size and orientation options, as explained below:

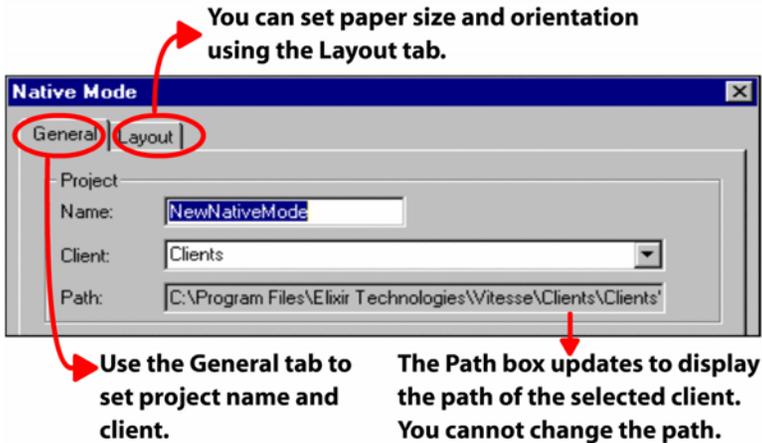


Image shows the options that can be set while creating a native mode project.

Selecting the Paper Size

You can set the physical paper size for the native mode project. You can choose either from a list of predefined paper sizes or specify a customized size.

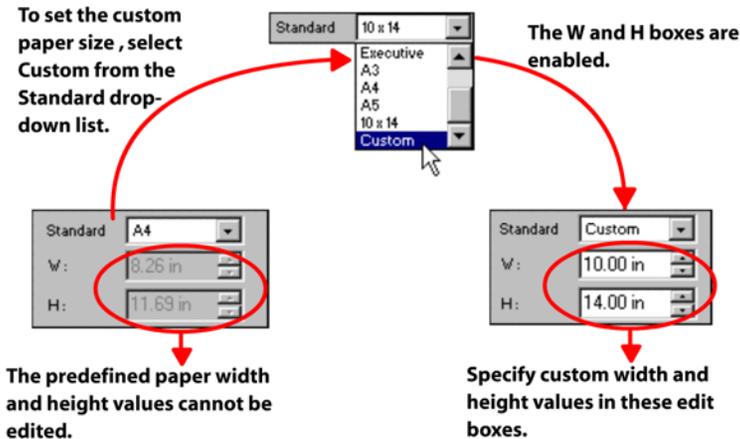
To set the paper size:

- 1 In the *Native Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select the required paper size option from the **Standard** drop-down list.
The **W** and **H** entry boxes display the width and height of the selected paper.
- 3 To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list; this option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

- 4 Enter the required custom paper width and height, and then click **OK**.
The new native mode project opens with the specified paper size.



The image shows the process of setting the custom paper size.

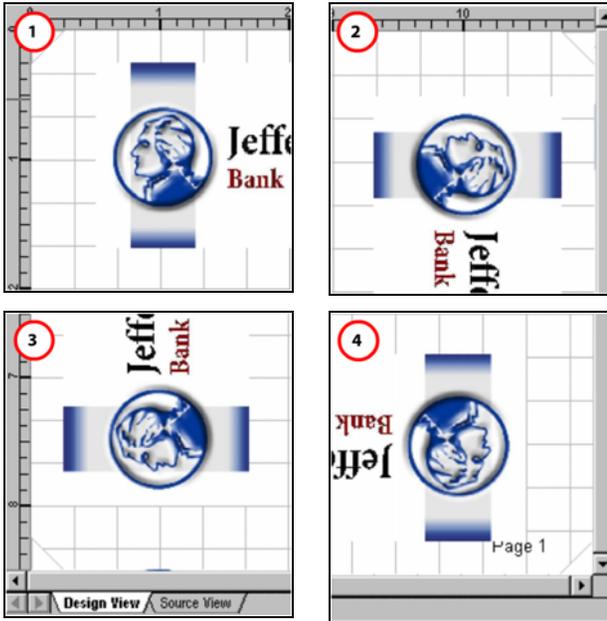
Selecting the Paper Orientation

Paper orientation defines the layout position of a physical paper for printing. Vitesse supports four orientation options, which are discussed below.

To set the paper orientation:

- 1 In the *Native Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select one of the following options from the **Orientation** drop-down list:
 - **Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - **Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
 - **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.

The new native mode project opens with the specified paper orientation.



The design area reflects the paper orientation set to (1) portrait (2) inverse landscape (3)landscape (4)inverse portrait.

- 3 Click **OK**.

Chapter 13: Creating a Database Mode Project

This chapter describes activities and tasks for creating VIPP database mode projects. Included are instructions and information for the following:

- Using the Database Mode
- Creating a New Database Mode Project
- Selecting the Paper Size
- Selecting the Paper Orientation
- What is a Sample Data File
- Selecting a Sample Data File
- Specifying a Delimiter

Using the Database Mode

Database mode projects are used for processing files containing field-delimited records. With the database mode, you can create database-enabled projects, such as mailers, promotional flyers, insurance contracts, etc. Data for these projects is extracted from a database file, which is defined in the first record. The database file specifies field names and their sequence within the record.

Creating a New Database Mode Project

To create a new database mode project:

- 1 Select File>New from the menu bar to display the *New Project* dialog.



You can also open the *New Project* dialog by either:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <N> on the keyboard.

- 2 From the **New** tab, select the **Database Mode** icon , and then click **New**.

The *Database Mode* dialog is displayed with the **General** tab selected.

- 3 In the **General** tab, set the following Project specifications:
 - **Name** – specifies a unique name identifying the project. The name can be 30 characters in length and can include alphanumeric characters.
 - **Client** – lists the clients that are available for selection and creates a directory for the project in the selected client folder.



You can also create a new client by entering the appropriate name in the **Client** drop-down list. Vitesse asks for confirmation before creating the new client.

- **Path** – displays the path of the project directory in the selected client.
- 4 Click **OK**.

While creating the database mode project, you can also specify the required paper size, orientation, and data file options, as explained below.

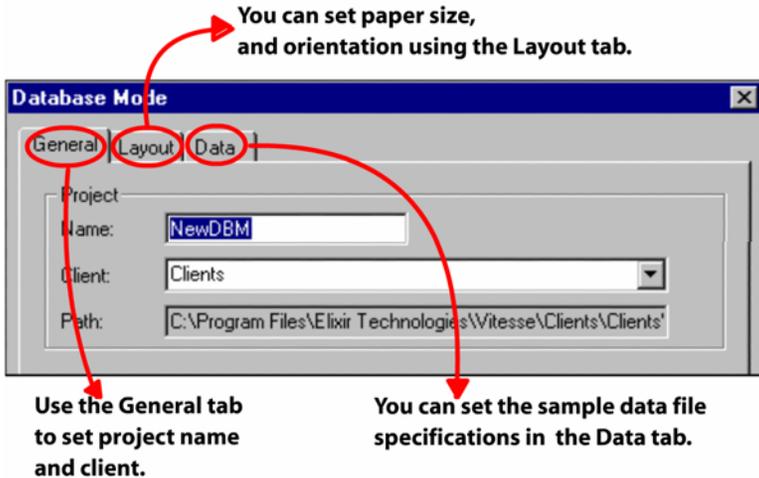


Image shows the options that can be set while creating a database mode project.

Selecting the Paper Size

You can set the physical paper size for the database mode project by choosing either from a list of predefined paper sizes or specifying a custom size.

To set the paper size:

- 1 In the *Database Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select the required paper size option from the **Standard** drop-down list.

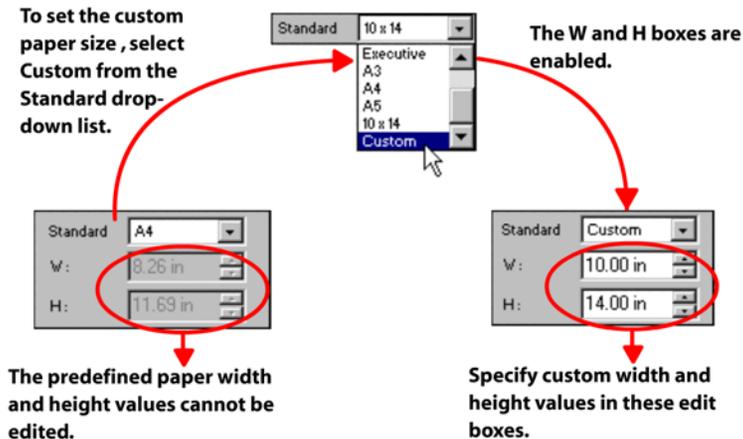
The **W** and **H** entry boxes display the width and height of the selected paper.

- To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list. This option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

- Enter the required custom paper width and height, and then click **OK**. The new database mode project opens with the specified paper size.



The image shows the process of setting the custom paper size.

Selecting the Paper Orientation

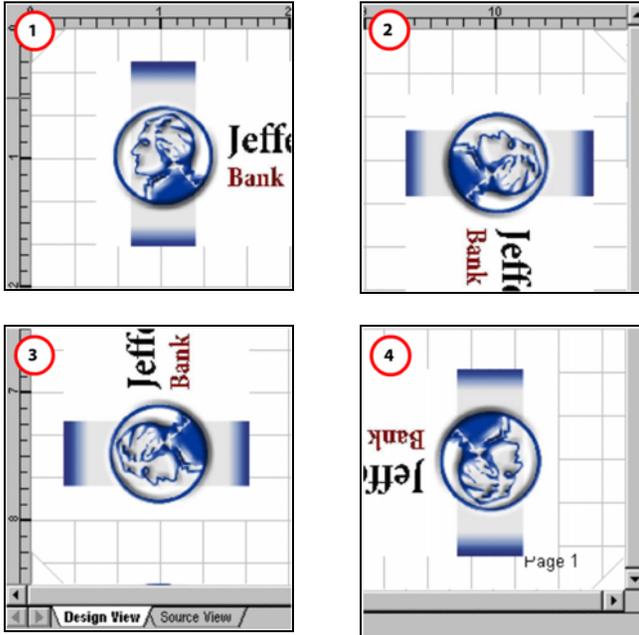
Paper orientation defines the layout position of a physical paper for printing. Vitesse supports four orientation options discussed below.

To set the paper orientation:

- In the *Database Mode* dialog, select the **Layout** tab.
- In the **Paper** area, select one of the following options from the **Orientation** drop-down list:
 - Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.

- **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
- **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.

The new database mode project opens with the specified paper orientation.



The design area reflects the orientation set to (1) portrait (2) inverse landscape (3) landscape (4) inverse portrait.

3 Click **OK**.

What is a Sample Data File?

Vitesse enables you to use a sample data file for creating and designing your projects. The sample data file should represent the original production data file and cover all the possible scenarios. However, the size of the file should not exceed 200kb.

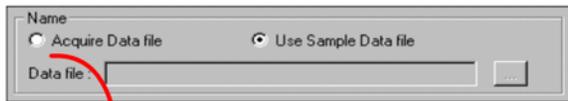
Once you have finished creating the design and exported your project as VIPP Project Container (VPC), you can simply substitute your sample data file on the printer controller with the production data file.

Selecting a Sample Data File

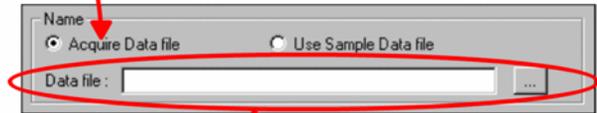
To select a sample data file:

- 1 In the *Database Mode* dialog, select the **Data** tab.
- 2 In the **Name** area, select one of the following options:
 - **Use Sample Data file** – uses the default sample data file for the project.
 - **Acquire Data file** – enables the **Data file** entry box and associated  button. You can either type the required path or click  to open the *Insert Data File* dialog for choosing the required data file specific to the project. See image below depicting the process of selecting a sample data file.

1. Select **Acquire Data file** to select a data file of your choice.

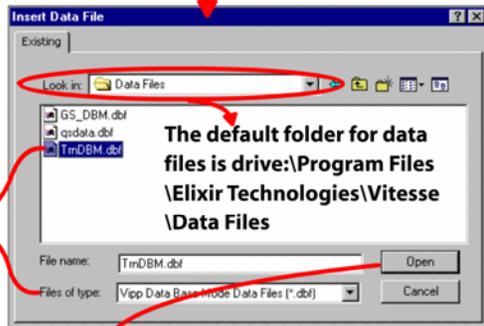


2. Data file edit-box and the associated browse button are enabled.



3. Click  .

4. Select **Vipp Database Mode Data Files** from the Files of type drop-down list to display all the database mode files and then select the required sample data file.



5. Click **Open**.

6. The path of the selected data file displays in the **Data file** entry-box.



3 Click **OK**.

Specifying a Data File Delimiter

Data files used in the database mode contain delimited data. A delimiter is used to separate fields within records; fields are of either fixed length or variable length.

To specify a field delimiter:

- 1 In the *Database Mode* dialog, select the **Data** tab.
- 2 Select the required delimiter from the **Delimiter** drop-down list.
- 3 To select a delimiter other than the predefined values, select the **Other** option from the **Delimiter** drop-down list; this option enables the **Other** entry-box.
- 4 Enter the required delimiting character in the **Other** edit-box. The delimiter can only be one character in length.
- 5 Click **OK**.

The new database mode project opens with the selected data file that contains the specified delimiter.

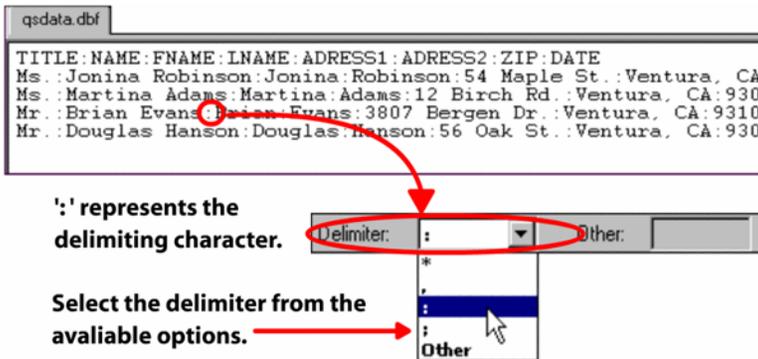


Image shows a sample database file with “:” as the specified delimiter.

Chapter 14: Creating a Line Mode Project

This chapter describes activities and tasks for creating VIPP line mode projects. Included are instructions and information on the following:

- Using the Line Mode
- Selecting a Sample Data File
- Defining the Data
- Working with Page Delimiters

Using the Line Mode

Line mode enables you to create and manage legacy type data streams. Legacy type data streams are characteristic of transactional documents, such as telephone bills, credit card statements, banking statements, and sales reports. These data streams are also referred to as “line” or “print ready” data. One example is LCDS (Line Conditioned Data Streams). Line mode projects comprise line data in ASCII format that is line numbered or RPE prefixed. Moreover, the data can either have delimited or non-delimited fields within records. Line mode enables you to read, process, and format the line printer data either line-by-line (records) or column-by-column (fields).

Creating a New Line Mode Project

To create a new line mode project:

- 1 Select File>New from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <N> on the keyboard.

- 2 From the **New** tab, select the **Line Mode** icon , and then click **New**.
The *Line Mode* dialog is displayed with the **General** tab selected.

- 3 In the **General** tab, set the following **Project** specifications:
 - **Name** – specifies a unique name identifying the project. The name can be 30 characters in length and can include alphanumeric characters.
 - **Client** – lists the clients that are available for selection and creates a directory for the project in the selected client folder.



You can also create a new client by entering the appropriate name in the **Client** drop-down list. Vitesse asks for confirmation before creating the new client.

- **Path** – displays the path of the project directory in the selected client.

4 Click **OK**.

While creating the line mode project, you can also specify the layout, data, and page delimiter settings, as explained below.

You can set paper size, orientation and margins using the Layout tab.

You can specify page break settings using the Page tab.

Use the General tab to set project name and client.

You can set the sample data file specifications in the Data tab.

Image shows the options that can be set while creating a line mode project.

Setting up a Data Grid

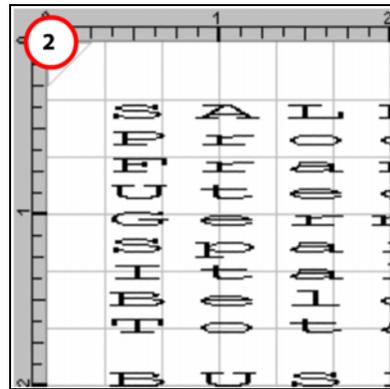
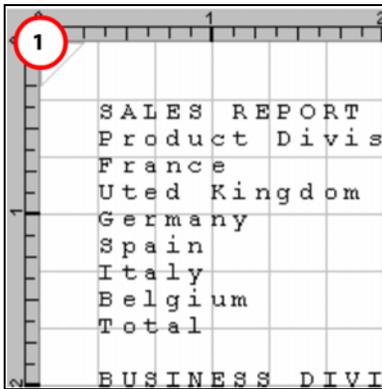
You can setup a data grid by specifying the number of characters per line (CPL) and the number of lines per page (LPP) that are mapped to a logical page.

To setup the grid:

- 1 In the *Line Mode* dialog, select the **Layout** tab.
- 2 In the **Grid** area, specify the number of characters in a line, which is to be mapped from the line data file onto the logical page in the **Character per line** entry box.

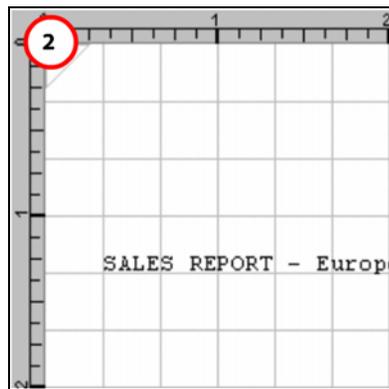


The characters per line (CPL) settings take effect when the font size is set to 0.



The design area reflects the characters per line value set to (1)80 (2)20.

- 3 Specify the number of lines per page (LPP) mapped from the line data file to the logical page in the **Lines per page** entry box. See the “Working with Page Delimiters” section for more information.



The design area reflects the line per page value set to (1) 66 (2) 10.

- 4 Click **OK**.

Selecting the Paper Size

You can set the physical paper size for the line mode project. You can choose either from a list of predefined paper sizes or specify a customized size.

To set the paper size:

- 1 In the *Line Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select the required paper size option from the **Standard** drop-down list.
The **W** and **H** entry boxes display the width and height of the selected paper.
- 3 To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list. This option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units**>*unit name* from the pop-up menu.

- 4 Enter the required custom paper width and height, and then click **OK**.
The new line mode project opens with the specified paper size.

To set the custom paper size, select Custom from the Standard drop-down list.

The W and H boxes are enabled.

The predefined paper width and height values cannot be edited.

Specify custom width and height values in these edit boxes.

The image shows the process of setting the custom paper size.

Autosizing Text Objects

You can auto size a text object to adjust automatically to the amount of text it is displaying. This also helps prevent overlapping and extra spacing between text objects that are placed relative to each other. Vitesse provides the **Auto Size** option in the **Settings** tab of the *Property View* window to automatically resize and align the text, thus adjusting text boundary to fit the text size.

This feature is especially useful when working with variable length (delimited) fields or with fields that contain merged variables. For more information on Autosizing Variable Text Objects, see *Section 6, 7, and 8*.



When you use the **Auto Size** option for field objects, the size of the object becomes dynamic and is subject to change depending on the contents in the field. Therefore, the **Auto Size** option affects the placement of relatively positioned objects due to its changing behavior. The VIPP converter, on the other hand, requires an exact size of the field object for positioning the relative objects. Hence, this option must be disabled before exporting projects.

Formatting Text Objects

Vitesse offers different formatting features that you can apply to text and graphic text objects. You can add different background attributes to the text and set text alignment options. Additional options enable you to set word, character, line, and paragraph spacing. You can also set the indentation and tab properties of the text for precise placement on a page. Before applying the formatting options, select the **Settings** tab and choose one of the following from the **Formatting** area:

- **Normal** – uses the default text format.
- **Paragraph** – uses the paragraph text format.

Tasks in this section explain the formatting options in detail.

Setting Text Background Attributes

In Vitesse, you can select a text background and apply it to the selected text. Background attributes control the text background either by changing the color or by including an image.

To set a text background attribute:

- 1 Select the text object in the design area.
- 2 From the *Property View* window, select the **Font** tab.
- 3 Select the required background attribute option from the **BAT** drop-down list.

The background attribute applies to the selected text object.

Setting the Page Margins

You can set top, bottom, left, and right page margins using the current units. The top and left margins are measured from the top left edge of the page, whereas the bottom and right margins are measured from the bottom right edge of the page. These margins form a bounding box.

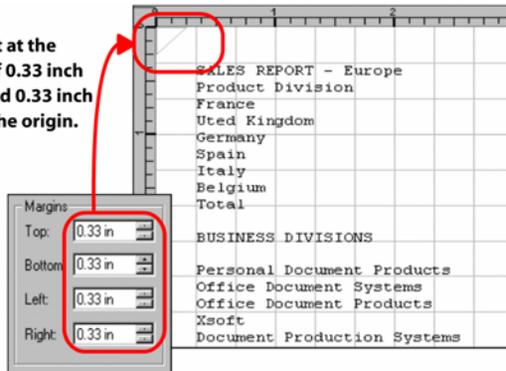
To set the margins:

- 1 In the *Line Mode* dialog, select the **Layout** tab.
- 2 In the **Margins** area, specify the **Top**, **Bottom**, **Left**, and **Right** page margins in the respective entry boxes, per specified unit of measurement.

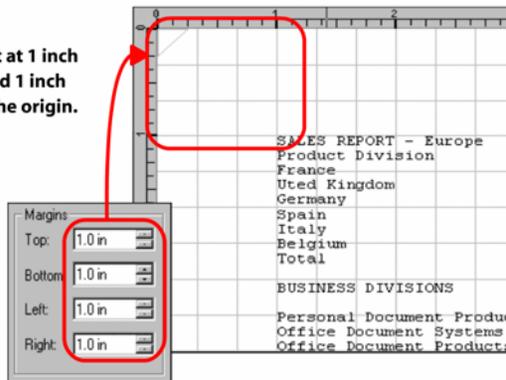


You can also set the unit of measurement for the page margins by right clicking the upward or downward arrows in the available entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The margins set at the default value of 0.33 inch from the top and 0.33 inch to the right of the origin.



The margins set at 1 inch from the top and 1 inch to the right of the origin.



Changing the margins for a line mode project.

- 3 Click **OK**.

What is a Sample Data File?

Vitesse enables you to use a sample data file for creating and designing your projects. The sample data file should represent the original production data file and cover all possible scenarios. However, the size of the file should not exceed 200kb.

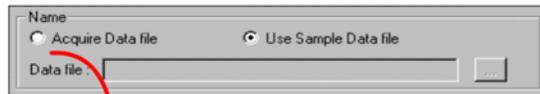
Once you have finished creating the design and exported your project as a VIPP Project Container (VPC), you can simply substitute your sample data file on the printer controller with the production data file.

Selecting a Sample Data File

To select a sample data file:

- 1 In the *Line Mode* dialog, select the **Data** tab.
- 2 In the **Name** area, select one of the following options:
 - **Use Sample Data file** – uses the default sample data file for the project.
 - **Acquire Data file** – enables the **Data file** entry box and associated  button. You can either type the required path or click  to open the *Insert Data File* dialog for choosing the required data file specific to the project. See image below depicting the process of selecting a sample data file.

1. Select **Acquire Data file** to select a data file of your choice.

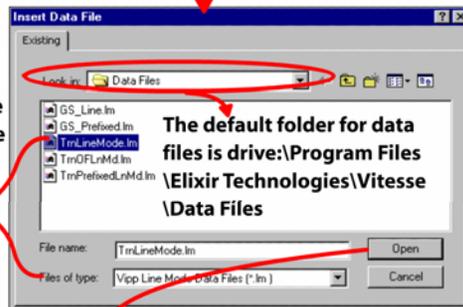


2. Data file edit-box and the associated browse button are enabled.



3. Click  .

4. Select **Vipp Line Mode Data Files** from the Files of type drop-down list to display all the line mode files and then select the required sample data file.



5. Click **Open**.

6. The path of the selected data file displays in the **Data file** entry-box.



The image shows the process of selecting a sample data file.

- 3 Specify the data file attributes. For more information, refer to “Defining the Data,” “Specifying the Field Type,” and “Specifying Delimited Data” tasks later in the chapter.
- 4 Click **OK**.

Defining the Data

Line mode data files contain line data that is either prefixed (having IDs) or non-prefixed:

- Non-prefixed data files contain records that are defined vertically using line numbers. Fields within the records are defined horizontally using line numbers, too. Vitesse uses these line numbers to locate a specific line in the data file. Non-prefixed data files are further classified as either delimited or non-delimited (fixed length).
- Prefixed data files contain records that are defined vertically using alphanumeric prefixes. Field number defines fields within records horizontally. Vitesse uses the prefixes to locate records within the data file. Prefixed data files are further classified as either delimited or non-delimited (fixed length). Prefixes can help identify the data that you want printed with similar presentation attributes such as font, color, alignment, and position, etc. For example, in a bank statement, credit lines may be prefixed by LINEC and debit lines by LINED. You can choose to print the credit lines in a different color and the debit lines in a different font.

Specifying the Record Prefixes

To define the record prefix specifications for the prefixed data file:

- 1 In the *Line Mode* dialog, select the **Data** tab.
- 2 In the **Record** area, select **Prefixed** which indicates that the selected data file contains prefixes; this option enables the **Start** and **Length** entry boxes.
- 3 Specify the start position of the prefixed record in the **Start** entry box.
- 4 Specify the length of the prefixed record in the **Length** entry box.



If you have selected a non-prefixed data file, then you do not need to set the **Prefixed** option.

Line mode data files use
.LM extension.

```
GS_Prefixed.Lm
DAT0 January 15, 2002
ADR0 Robert Jones
ADR0 1500 Main St.
ADR0 Ventura, CA 93001
ACR0 1237 6548 1262 1098
ACR1 150,000.00
CRS0 115,320.99      1,239.22
CRS1 110,000.00      29.00
CRS2 GOOD           1,268.22
PRS0 Clothing        2,500.00
PRS0 Food            1,000.00
PRS0 Electronics     3,000.00
PRS0 Travel          80,000.00
PRS0 Hotels          20,000.00
```

The prefix starts at byte
0 and is 4 characters
in length.

Record

Prefixed Start: Length:

Specifying data file prefix attributes.

- 5 Click **OK**.

Specifying the Field Type

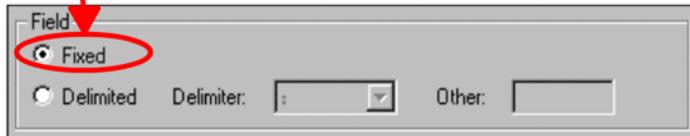
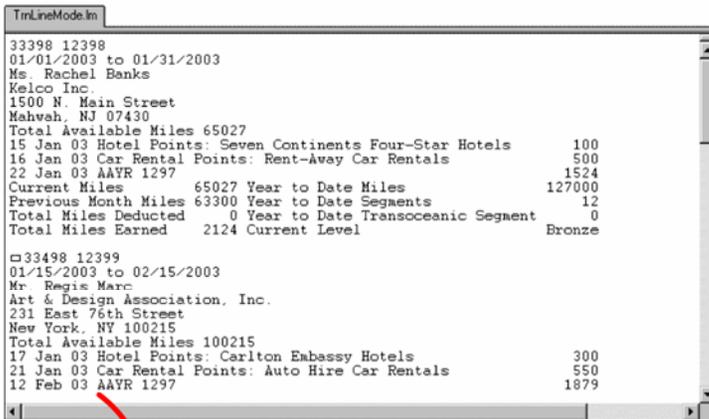
Fields within records contain information, such as a date or a name. In fixed length (non-delimited) data files (prefixed and/or non-prefixed), the fields are identified horizontally by start byte and length of the data field.

Specifying Fixed Length Data

To use fixed length fields in the data file:

- 1 In the *Line Mode* dialog, select the **Data** tab.
- 2 In the **Field** area, select **Fixed** which indicates that the data file contains fixed length fields.

The data file uses fixed length fields.



Specifying fixed length data.

- 3 Click **OK**.

Specifying Delimited Data

Fields in a record (for both prefixed and non-prefixed data files) can also be identified using delimiting characters. Every time a delimiter is encountered in the file, a new field begins, which implies that the fields can vary in length. Delimited data files are also referred to as variable length files. We recommend not using a delimiting character that is part of the data.

Specifying a Delimiter

To use delimited fields in the data file:

- 1 In the *Line Mode* dialog, select the **Data** tab.
- 2 In the **Field** area, select **Delimited** to specify that the data file contains variable length fields; this option enables the **Delimiter** drop-down list.
- 3 Select the required delimiting character from the **Delimiter** drop-down list.
- 4 To select a delimiter other than the predefined values, select the **Other** option from the **Delimiter** drop-down list; this option enables the **Other** entry-box.
- 5 Enter the required delimiting character in the **Other** edit-box. The delimiter can only be one character in length.
- 6 Click **OK**.

The new line mode project opens with the selected data file that contains the specified delimiter.

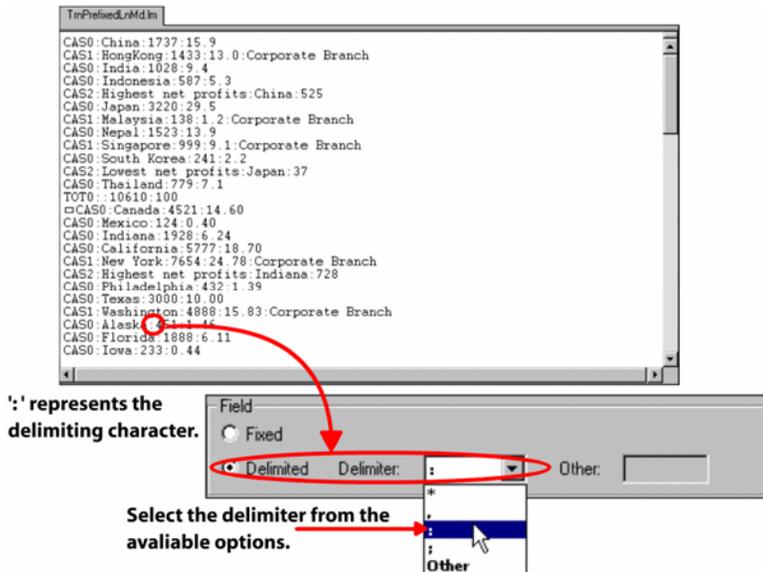


Image shows a sample database file with “:” as the specified delimiter.

Working with Page Delimiters

A variable data document can consist of multiple pages of data. Page breaks are used to indicate the end of a page of data in the data file. You can set page breaks in a line mode project using one of the following options:

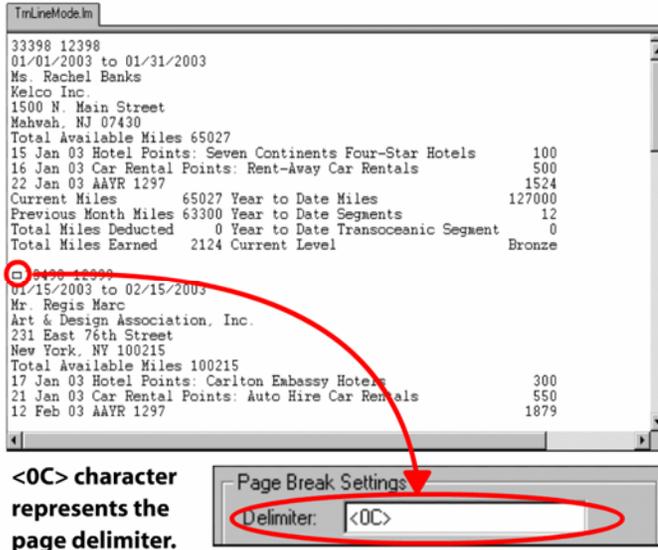
- Specifying the number of lines per page (LPP) that are mapped from the line data file to the logical page. In the line mode, reaching the LPP value causes a page break. However, page breaks occurring in the data file take higher precedence and cause a page break even if the specified LPP value has not been reached.
- Specifying Printer Control Character (PCC) byte. See the “Specifying Printer Control Characters in Data” section below for more information.
- Using explicit page delimiter such as Form feed (FF or <0C> hex). Form feed forces the printer to eject the current page and to continue printing data on another page. Form feed and PCC cannot be used at the same time. When working with PCC, form feeds are processed as print data.

The Form feed detection process works with any string contained in the data file. However, you should be careful when choosing a character as a page delimiter. For instance, selecting C as a page delimiter will start a new page every time the letter C is encountered. If your customer’s name is Chirac Christopher Black, you will then have four pages carrying “hira”, “ “, “hristopherbla”, and “k” respectively. We, therefore, recommend using a character that is not part of your actual data content as a delimiter.

Specifying Page Delimiters

To set a page delimiter:

- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 In the **Page Break Settings** area, specify the page delimiter in the **Delimiter** entry box.



Specifying the page delimiter. Any character in the file can act as the delimiting character.

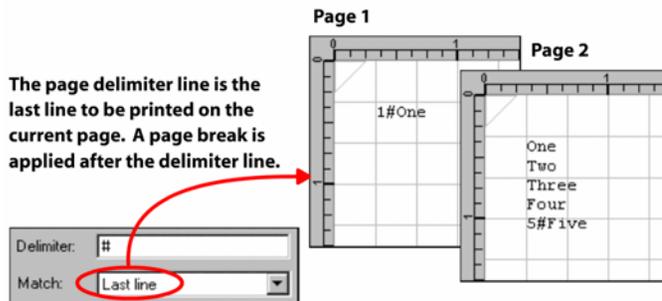
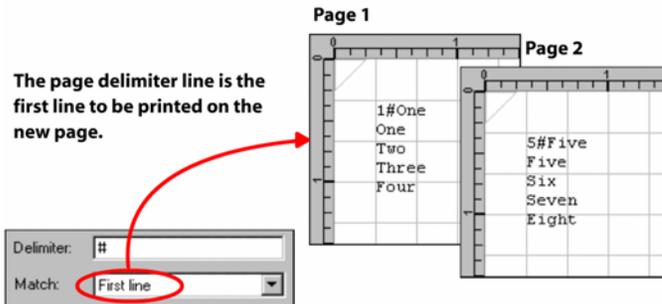
- 3 Click **OK**.

Matching the Page Delimiter

The data string on which the page delimiter is detected can be treated as either the first line of the new page, or the last line to be printed on the current page.

To choose the page delimiter match option:

- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 In the **Page Break Settings** area, select one of the following options from the **Match** drop-down list:
 - **Last line** – specifies that the data string containing the page delimiter is printed as the last line on the current page.
 - **First Line** – specifies that the data string containing the page delimiter is printed as the first line on the next page.
- 3 Click **OK**.



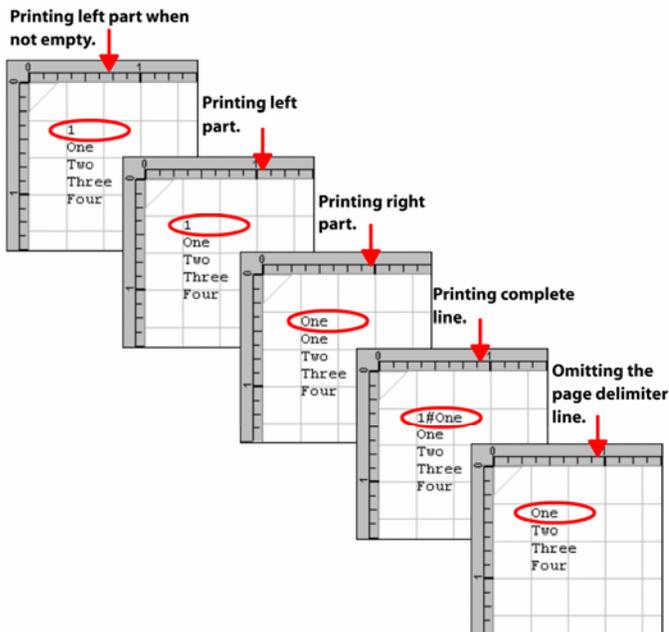
Printing the page delimiter line as the first or last line of a page. In this example, '#' represents the page delimiter.

Printing the Page Delimiter Line

Vitesse provides a number of printing options for the data string containing the delimiter. You can choose to print the entire string contents including the delimiter, or print data to the right or left of the delimiting character.

To choose the page delimiter line printing option:

- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 In the **Page Break Settings** area, select one of the following options from the **Print** drop-down list:
 - **Left when not empty** – prints part of the line that is to the left of the page delimiter if not empty.
 - **Left part** – prints part of the line that is to the left of the page delimiter.
 - **Right part** – prints part of the line that is to the right of the page delimiter.
 - **Complete line** – prints the complete line, including the page delimiter.
 - **None** – does not print the line containing the page delimiter.
- 3 Click **OK**.



The image shows various printing options for the page delimiter line. In this example, '1#One' is the page delimiter line and '#' represents the delimiter.

Looping the Page Delimiter Line

A single data line in the data file can contain multiple instances of the delimiter. You can set options to ignore any occurrences of the delimiter in the same line, after the first occurrence. You can also set options to look for all occurrences of the delimiter in a single line, causing a page break after every occurrence. Vitesse loops through the right part of the page delimiter line to look up all possible occurrences of the delimiter.

To loop the page delimiter line:

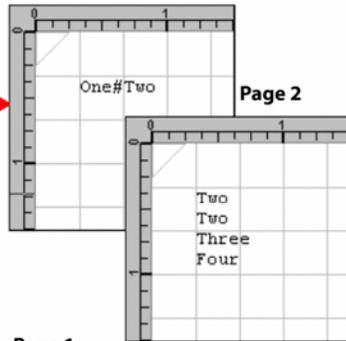
- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 In the **Page Break Settings** area, select one of the following options from the **Loop** drop-down list:
 - **Right part** – checks the page delimiter line to the right of the delimiter for more occurrences of the delimiter.
 - **None** – ignores more than one occurrences of the delimiter in the page delimiter line.
- 3 Click **OK**.

The page delimiter line '1#One#Two' contains two occurrences of the delimiter '#'.

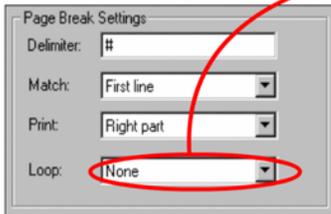
Looping the page delimiter line results in two pages, corresponding to the two occurrences of the page delimiter '#'.



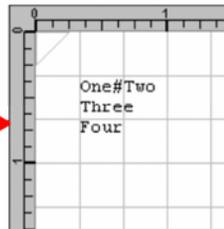
Page 1



Selecting None causes the second occurrence of the page delimiter string '#' to be ignored, resulting in a single page.



Page 1



A data line can be checked for all possible occurrences of the delimiter string or only the first occurrence.

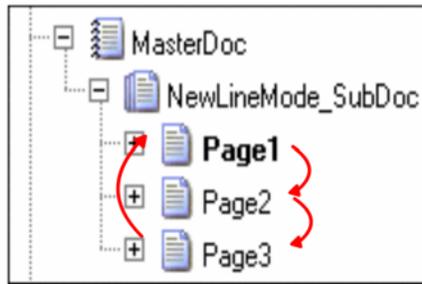
Setting Document Flow Options

When working with a document containing multiple page objects, you need to specify the sequence in which the data fills the pages.

To set the document flow:

- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 From the **Document Flow** area, select one of the following document flow options:
 - **Cyclic** – maintains cyclic data flow between the logical pages. After using the last logical page, data placement again starts from the first page. For example, for a document containing 3 logical pages and 6 data pages, the following sequence of logical pages is used 1, 2, 3, 1, 2, 3.

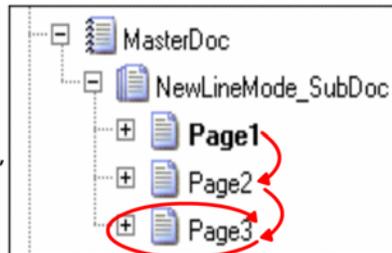
Once the last page is reached, data again starts printing from the first page and continues the same cycle until all the data is mapped.



The image displays the cyclic document flow.

- **Two Page** – specifies that only the last logical page is to be repeated for placing data; all the preceding logical pages are used only once. For example, for a document containing 3 logical pages and 6 data pages, the following sequence of logical pages is used 1, 2, 3, 3, 3, 3.

Initial pages are used only once, all the subsequent data pages are placed on the last logical page.



The image displays the data placement sequence for a two page document flow.

- 3 Click **OK**.

Specifying Printer Control Characters in Data

Printer control characters (PCC), (also called control codes or printer carriage controls) are data file characters that instruct the printer to perform a specific action. The first byte of a data line is interpreted as a printer control character by the printing device. PCC bytes control the vertical movement or spacing of a page, and are not part of the data. Vitesse reads the PCC bytes and interprets them when formatting line mode data.

To enable PCC processing, a PCC table needs to be set up that lists all the PCC bytes and the associated actions. Some PCC tables are globally predefined. For example, ANSI, IBM1403 EBCDIC and PCC definitions, etc. Vitesse supports the predefined ANSI PCC table. You can also create a custom PCC table with your own definitions. For example, you can set the: space character to print the remainder of a record on a new line; the plus character to print a record over the old line; and a one to begin a new page and so on.

To specify printer control characters in data:

- 1 In the *Line Mode* dialog, select the **Page** tab.
- 2 In the **PCC Settings** area, select one of the following PCC options:
 - **None** – disables the PCC settings.
 - **PCC** – enables PCC processing for the project. Selecting this option enables the associated drop-down list. You can select the **ANSI** option from the drop-down list to use the globally predefined ANSI PCC table.
- 3 Click **OK**.

User Notes:

Chapter 15: Creating an XML Mode Project

This chapter describes activities and tasks for creating VIPP XML mode projects. Included are instructions for the following:

- Using the XML Mode
- Creating a New XML Mode Project
- Selecting the Paper Size
- Selecting the Paper Orientation

Using the XML Mode

XML mode projects are used for processing XML (Extensible Markup Language) data files containing tags, attributes, and values. Vitesse requires a BTA (Begin Tag Action) and ETA (End Tag Action) command to process XML data. When Vitesse detects a tag in the data file, it processes the tag and associated information only if there is an associated BTA command. Otherwise, it moves on to the next tag in the data file. Subsequent chapters provide details about data processing in XML mode projects.

Creating a New XML Mode Project

To create an XML mode project:

- 1 Select File>New from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog by either:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <N> on the keyboard.

- 2 From the **New** tab, select the **XML Mode** icon , and then click **New**. The *XML Mode* dialog is displayed with the **General** tab selected.

- 3 In the **General** tab, set the following **Project** specifications:
 - **Name** – specifies a unique name identifying the project. The name can be 30 characters in length and can include alphanumeric characters.
 - **Client** – lists the clients that are available for selection and creates a directory for the project in the selected client folder.



You can also create a new client by entering the appropriate name in the **Client** drop-down list. Vitesse asks for confirmation before creating the new client.

- **Path** – displays the path of the project directory in the selected client.
- 4 Click **OK**.

While creating the XML mode project, you can also specify the required paper size and orientation options, as explained below.

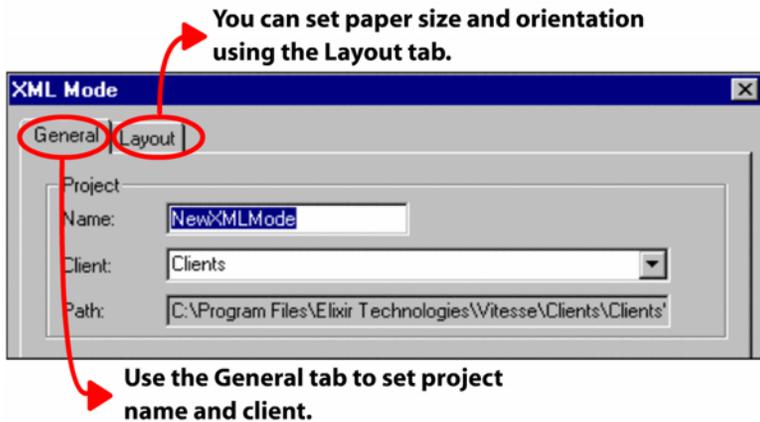


Image shows the options that can be set while creating an XML mode project.

Selecting the Paper Size

You can set the physical paper size for the XML mode project by choosing from a list of predefined paper sizes or specifying a custom size.

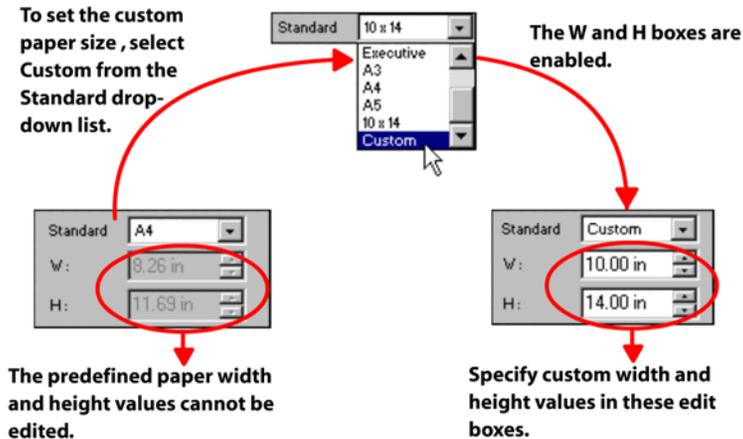
To set the paper size:

- 1 In the *XML Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select the required paper size option from the **Standard** drop-down list.
The **W** and **H** entry boxes display the width and height of the selected paper.
- 3 To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list. This option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units> unit name** from the pop-up menu.

- 4 Enter the required custom paper width and height, and then click **OK**.
The new XML mode project opens with the specified paper size.



The image shows the process of setting the custom paper size.

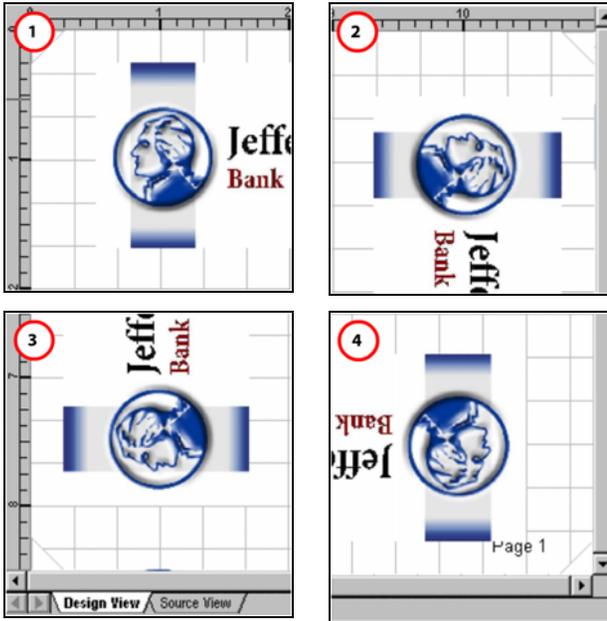
Selecting the Paper Orientation

Paper orientation defines the layout position of a physical paper for printing. Vitesse supports four orientation options discussed below.

To set the paper orientation:

- 1 In the *XML Mode* dialog, select the **Layout** tab.
- 2 In the **Paper** area, select one of the following options from the **Orientation** drop-down list:
 - **Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - **Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
 - **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.
- 3 Click **OK**.

The new XML mode project opens with the specified paper orientation.



The design area reflects the paper orientation set to (1) portrait (2) inverse landscape (3) landscape (4) inverse portrait.

Section V: Basic Project Setup

User Notes:

Chapter 16: Setting Document Properties

This chapter describes activities and tasks for setting properties associated with a Vitesse project. Tasks include:

- Specifying Project Details
- Maintaining Change History
- Setting Regional Preferences
- Working with Regional Languages
- Printing Multiple Copies of a Document

Specifying Project Details

You can view or specify Vitesse project information, such as the project title, short description of the project, project creator and generator name, copyright details, version number, and creation and modification dates in the *Property View* window by selecting the **SubDoc** node from the *Project Tree View* window.



The information that you specify in the **Properties** tab is displayed as comments in the generated VIPP project source code.

Setting the Project Information

To set the project details:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Properties** tab. This tab contains the project information that is organized as a **Key** (item) and **Value** pair.
- 3 Double-click any cell under the Key and Value columns to enter the project details.



You cannot modify the **Copyright** and **Version** information.

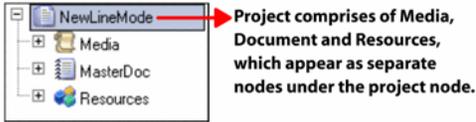
- 4 To add a new key-value pair above or below a selected row, click  in the **Properties** tab.
- 5 Enter the additional information as required in the newly added key and value pair.

Maintaining Change History

For projects that are frequently modified, a change history log can be maintained to keep track of all the changes made to the project. You can specify the date on which a project was modified, the user name, and the changes that were made in the project.

To maintain the change history:

- 1 From the project tree, select the root node that identifies the project type and name.



Project tree containing the default nodes.

- 2 From the *Property View* window, select the **Modifications** tab.
- 3 Click  to add an entry to the modifications table.
- 4 Double-click any cell under the **Date**, **User**, and **Action** columns to enter text.
- 5 Click **Apply**.

Setting Regional Preferences

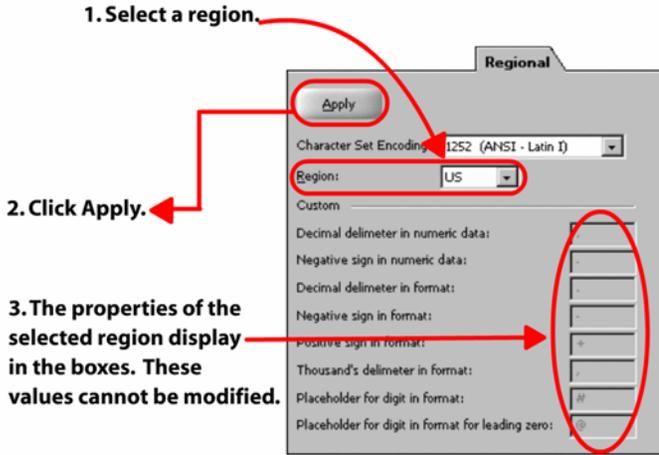
You can use regional preferences to set the appearance of numbers and currencies, according to the formats used in a particular geographical region. Vitesse provides you with predefined formats, and also enables you to create custom regional formatting options. These regional settings are used when applying transformations to numeric text and fields.

Applying Regional Preferences to Numeric Formats

You can view numeric format options for specific regions, and also specify custom numeric formats.

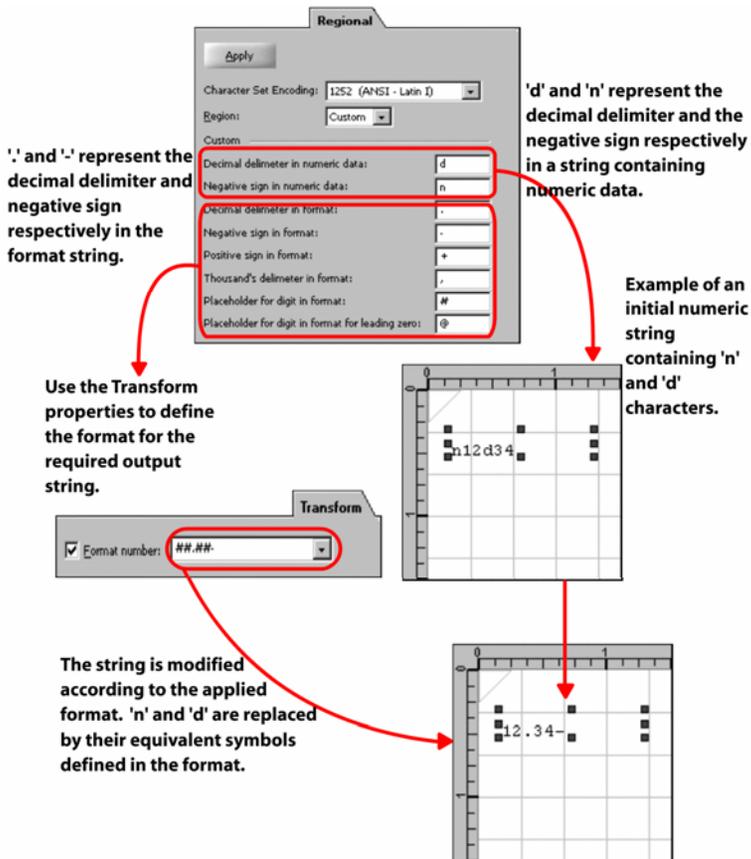
To view or set the regional numeric formats:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Regional** tab.
- 3 From the **Region** drop-down list, select one of the following numeric display options:
 - **US** – applies characters and masks defined in the US format.
 - **Europe** – applies characters and masks defined in the European format.

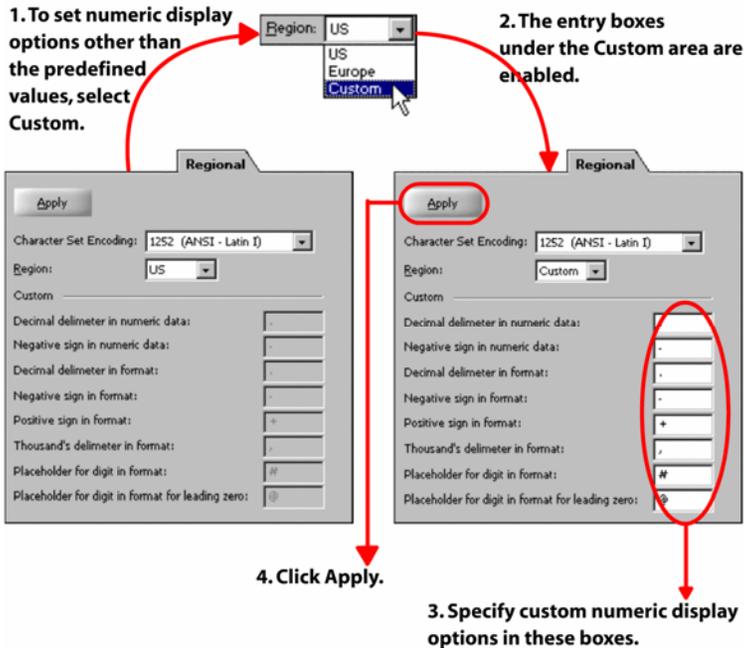


The image shows the process of setting numeric display options.

- 4 To set numeric display options other than the predefined values, select the **Custom** option from the **Region** drop-down list.
- 5 Specify the following custom numeric display options for a string containing numeric data:
 - **Decimal delimiter in numeric data** – specifies the character to be used as the decimal delimiter in a string containing numeric data.
 - **Negative sign in numeric data** – specifies the character to be used as the negative sign in a string containing numeric data.
- 6 Specify the following custom numeric display options to define the format (mask) for the required output:
 - **Decimal delimiter in format** – specifies the character to be used as the decimal delimiter in the format string.
 - **Negative sign in format** – specifies the character to be used as the negative sign in the format string.
 - **Positive sign in format** – specifies the character to be used as the positive sign in the format string.
 - **Thousand's delimiter in format** – specifies the character to be used as the thousands delimiter in the format string.
 - **Placeholder for digit in format** – specifies the character to be used as the digit placeholder in the format string.
 - **Placeholder for digit in format for leading zero** – specifies the character to be used as the leading zero digit placeholder in the format string.
- 7 Click **Apply**.



Specifying characters for defining strings and formats (masks).



The image shows the process of setting custom regional settings.



The regional numeric options are used for defining formats (masks) that are used to transform numeric text and fields. For more information on applying transformations, see *Chapter 10: Applying Transformations*.

Specifying Default Regional Preferences

To view or set the default regional numeric formats, which are applied every time you open a project:

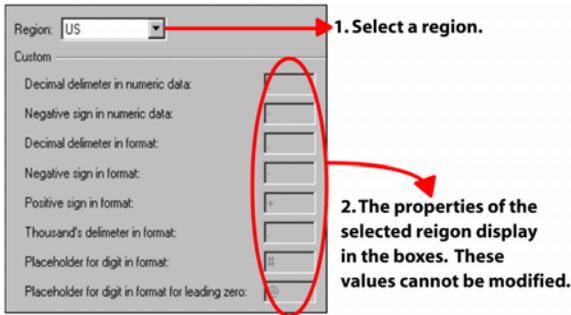
- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



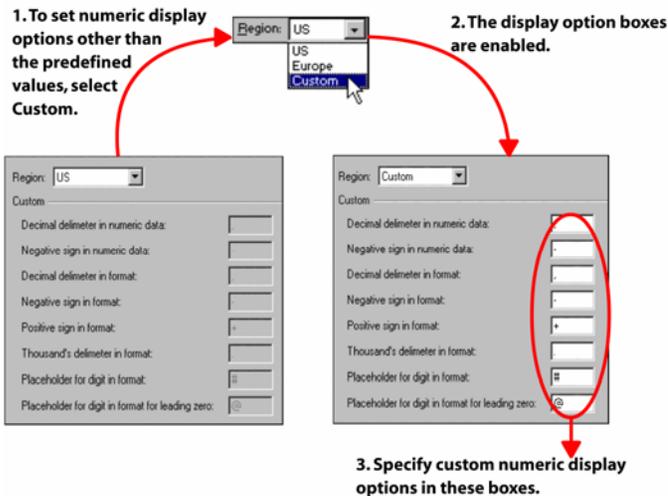
You can also open the *Preferences* dialog to set the application defaults either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 In the *Preferences* dialog, select the **Regional** category under the **Defaults** node.
- 3 Select one of the following default numeric display options from the **Region** drop-down list:
 - **US** – applies characters and masks defined in the US format.
 - **Europe** – applies characters and masks defined in the European format.



- 4 To set default numeric display options other than the predefined values, select the **Custom** option from the **Region** drop-down list.



The image shows the process of setting default regional settings.

- 5 Click **OK**.
Settings take effect the next time a project is opened.

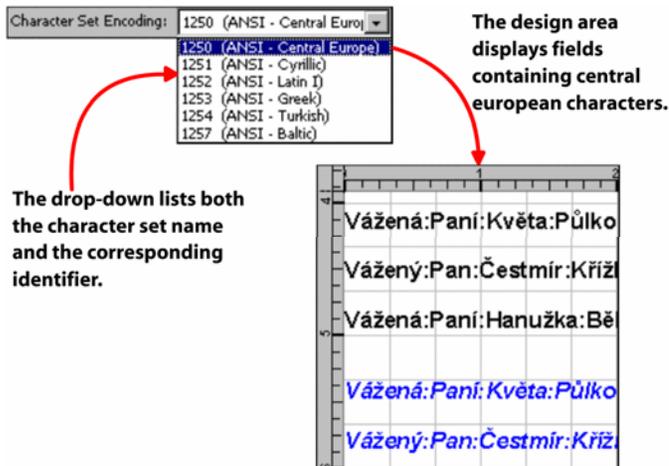
Working with Regional Languages

Vitesse provides support for regional languages including Cyrillic, Greek, Turkish, Baltic, and central European languages. You can create a document in any of these languages by setting **Character Set Encoding** options.

Setting Character Set Encoding Options

To set character set encoding:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Regional** tab.
- 3 Select the required character set option from the **Character Set Encoding** drop-down list.
- 4 Click **Apply**.



Character set encoding options enable you to create documents in different languages.

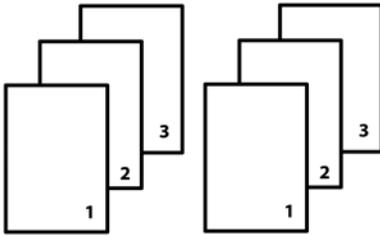
Printing Multiple Copies of a Document

Vitesse enables you to specify the number of document copies for printing. You can also specify how the printed material is to be organized by using collation options. The Cycle Copy function of VIPP allows you to achieve this. Tasks in this section provide details on setting the printing options.

Setting the Document Printing Options

To set the document printing options:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Cycle Copy** tab.
- 3 Select the **Cycle Copy** check box to enable the copy and collation options.
- 4 Specify the number of document copies to be printed in the **No of Copies** entry box.
- 5 Under the **Collation Mode** area, select one of the following options:
 - **Collate_on** – prints two or more copies of a multi-page document in normal page/binding order. For example, to print two copies of a 3-page document, pages 1, 2, and 3 are printed in this sequence as one set, and pages 1, 2, and 3 as the second set.

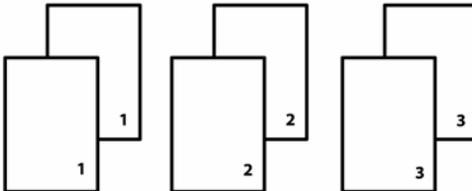


The image represents the printing sequence for two collated copies of a 3-page document.



Collate automatically sorts the printed pages and is helpful when working with a document containing multiple pages.

- **Collate_off** – prints two or more copies of a multi-page document so that all copies of the first page are printed together, then all copies of the second page are printed together, and so on.



The image represents the printing sequence for two non-collated copies of a 3-page document.

- **Collate_dbm** – prints the specified number of copies immediately after each record. This option generates the same printing sequence that is generated using the **Collate_off** option and is available only in the **Database** mode.



Collation mode is always enabled for the Native and XML mode projects.

User Notes:

Chapter 17: Defining Paper and Print Setup

This chapter describes activities and tasks for defining paper and print setup. Included are instructions for the following:

- Adding a Paper
- Adding a Predefined Paper
- Setting Duplex Printing Options
- Selecting the Paper Type
- Setting the Paper Weight
- Setting the Paper Color
- Using Document Structuring Conventions
- Printing a Constant Form on the Back of the Paper
- Printing Several Pages on One Paper (N-UP)
- Setting Default Paper Properties

Defining the Media Properties

The first step in creating variable data documents is to define the media settings.

Media settings involve defining, modifying, and selecting the type of paper the printer uses while printing, along with setting paper (or physical page) properties. The *Project Tree View* window in the Vitesse workspace contains the default **Media** node that enables you to add physical pages to your documents.

Adding a Paper

Every VIPP project contains at least one paper object, identified by the default **Paper** node under the **Media** node in the project tree. When a new project is created, one paper object is automatically added, however, you can add any number of paper objects to the project.

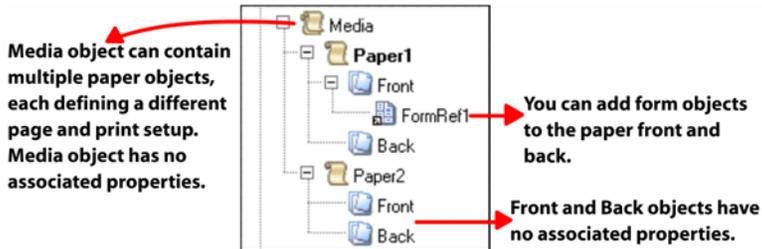
To add a paper:

- 1 Select **Insert>Paper** from the menu bar; a **Paper** node is created under the **Media** node in the project tree.



You can also add a new paper object by clicking  on the *Insert* toolbar.

Along with the **Paper** node, **Front** and **Back** nodes are also created by default under the **Paper** node; these sub-nodes denote the front and back sides of the paper. You can use overlays/forms on each side of the paper for adding static design content to the page.



You can delete a selected paper object either by:

- Selecting Edit>Delete from the menu bar.
- Or
- Clicking  on the *Standard* toolbar.

You cannot delete the first paper under the **Media** node or a paper associated with a page.

Adding a Predefined Paper

Vitesse provides the feature for using pre-defined media (paper) settings in the **Administration** mode. Using predefined settings reduces the time required for setting paper properties every time you create a project. As an administrator, you can create an XML file, in which you define the number of paper objects to be used in a design, and list the required media settings, pertaining to each paper object. You can also restrict the type of paper to be used in projects. The following image depicts the structure of an xml file containing the paper (media) settings:

```
<?xml version="1.0" ?>
- <VitesseMediaList Version="1.0">
- <Paper>
  <Name>Tray5</Name>
  <Category>Standard</Category>
  <Standard>Letter</Standard>
  <Units>Centimeters</Units>
  <Height>0.0</Height>
  <Width>0.0</Width>
  <Orientation>Landscape</Orientation>
  <Type>Drilled</Type>
  <Weight>3.45</Weight>
  <Color>BLACK</Color>
  <IncludePaperDimensions>No</IncludePaperDimensions>
</Paper>
</VitesseMediaList>
```

The <Paper></Paper> tag pair defines a paper object; all of the sub tag nodes define the properties of the paper.

The XML file can contain any number of paper definitions, as required in the project design. This file can also be updated anytime; this action does not affect the paper objects already inserted in a project.

To add a predefined paper to a design:

- 1 Using any text editor, create an xml file specifying all of the required media options such as paper name, category, standard, dimensions, orientation, type, weight, and color as identified by their respective tags in the XML file. You can also choose to generate Document Structuring Conventions (DSC) %%DocumentMedia comments using the Include Paper Dimensions tag.



DSC comments are generated in a VIPP submission file (*.sub), which is created only on exporting a project. Printers process the DSC comments in this file to determine the media requirements of the document.



You can select the required predefined media file using the *Preferences* dialog in the Vitesse Administration mode. For more information, see *Chapter 36: Vitesse Administrative Options*.

- 2 Open a new or existing project in Vitesse, and then select Insert>Paper from the menu bar to open the *Insert Paper* dialog.



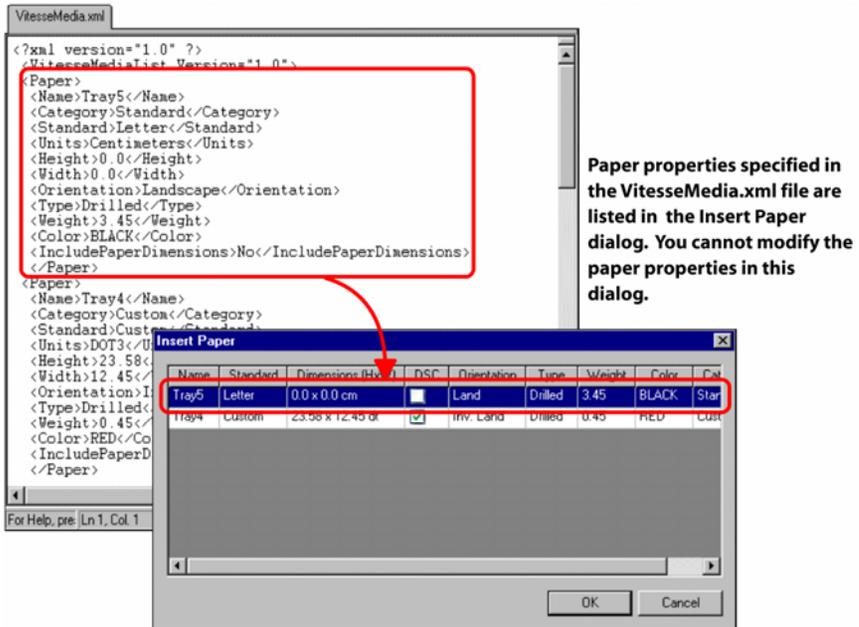
You can also open the *Insert Paper* dialog to add a predefined paper object by clicking  on the *Insert* toolbar.

- 3 From the *Insert Paper* dialog, select the required paper object to use in the design.
- 4 Click **OK**.



Using the XML file for adding predefined paper objects limits you (as a user) to choose only the paper objects listed in the *Insert Paper* dialog.

The image below shows an example of the VitesseMedia.xml file that contains two paper objects along with the associated settings. These media definitions are reflected in the *Insert Paper* dialog, which lists the paper objects and corresponding settings in columns such as **Orientation**, **Type**, **Weight**, and **Color**, etc. Paper properties cannot be modified in the *Insert Paper* dialog.



Setting Paper Properties

A paper object defines the attributes of the physical page, such as the paper size, orientation, type, weight, and color. You can also specify simplex or duplex printing options for a selected paper object.

Changing the Paper Size

You need to specify the paper type and associated dimensions as a requirement for printing the documents. Vitesse provides options to either select predefined paper types, each with different paper dimension settings, or allows you to customize the paper dimensions.

To change the paper size set at the time of creating a design:

- 1 Select the required **Paper** node from the project tree.
- 2 From the *Property View* window, select the required paper size option from the **Standard** drop-down list under the **Paper** area.

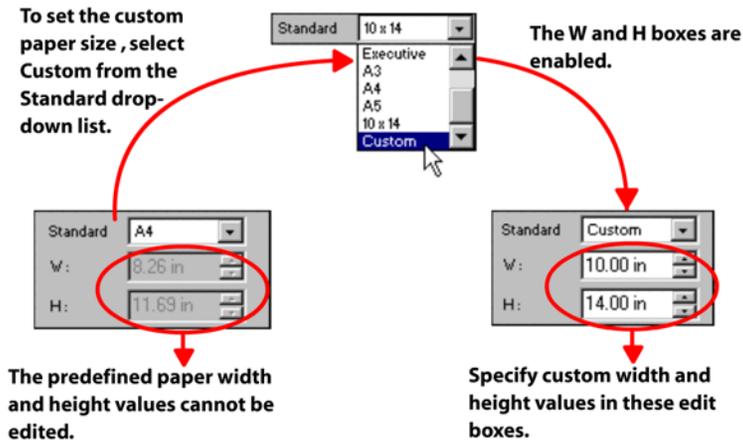
The **W** and **H** entry boxes display the width and height of the selected paper.

- To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list; this option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

- Enter the required custom paper width and height. The selected paper object is updated with the specified paper settings.



The image shows the process of setting the custom paper size.

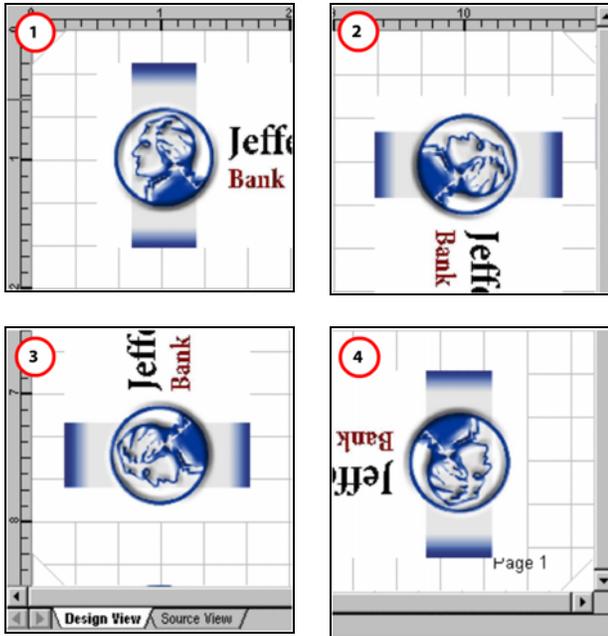
Changing the Paper Orientation

You can change the paper orientation while defining the paper properties.

To change the paper orientation:

- Select the required **Paper** node from the project tree.
- From the *Property View* window, select one of the following options from the **Orientation** drop-down list, under the **Paper** area:
 - Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.

- **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.



The design area reflects the paper orientation set to (1) portrait (2) inverse landscape (3) landscape (4) inverse portrait.

Using Printing Options

Vitesse enables you to design pages for simplex and duplex printing.

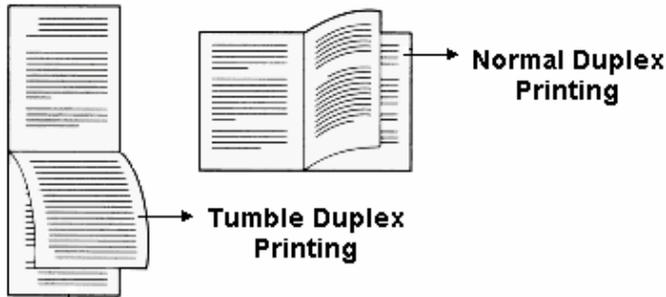
- In the simplex printing mode, only one side of the paper is used.
- In the duplex printing mode, both sides of the paper are used.

Setting Printing Options

To set the printing options:

- 1 Select the required **Paper** node from the project tree.
- 2 From the **General** tab of the *Property View* window, select one of the following printing options:
 - **Simplex** –prints documents so that only one side of the paper is used.
 - **Duplex** –prints documents so that both sides of the paper are used. Both paper sides have the same print orientation to enable side binding.

- **Tumble Duplex** – prints documents so that both sides of the paper are used. The top of one paper side and the bottom of the other side print along the same sheet edge to enable top binding.



The normal duplex printing is used for side binding, whereas the tumble duplex printing is used for top binding.

Selecting the Paper Type

The papers used for printing are available in different types, weights, and color. You can specify the type of paper you want to use for the print job at design time.

To select the paper type:

- 1 Select the required **Paper** node from the project tree.
- 2 From the *Property View* window, select one of the following options from the **Type** drop-down list, under the **Media** area:
 - **Drilled** – specifies that drilled paper is to be used for printing.
 - **Plain** – specifies that non-drilled paper is to be used for printing.

Setting the Paper Weight

Heavy weight papers increase the quality of the print job, but increase the production cost. You can set paper weight to be used for a particular document at design time. The paper weight is specified in units of grams per square meter.

To select the paper weight:

- 1 Select the required **Paper** node from the project tree.
- 2 From the *Property View* window, type or select the paper weight value in the **Weight** entry box, under the **Media** area.

Setting the Paper Color

You can set the paper color for your print job. The names for paper colors differ from paper mill to paper mill, and depending on the chosen color, the papers vary amongst the paper batches produced at different times; the final printed output can differ depending on the type of paper used.

To set the paper color:

- 1 Select the required **Paper** node from the project tree.
- 2 From the *Property View* window, select the required color option from the **Color** drop-down list, under the **Media** area.

Using Document Structuring Conventions

You can select the **Include Paper Dimensions** option from the **General** tab to generate Document Structuring Conventions (DSC) %%DocumentMedia comments. These comments specify the media requirements during a print job.

Printing a Constant Form on the Back of the Paper

When working in duplex or tumble duplex mode, you can set the back side of the paper to print with or without variable data. Vitesse provides support for printing constant forms to avoid printing variable data on the back side of the paper.

To print constant forms:

- 1 Select the required **Paper** node from the project tree.
- 2 From the **General** tab of the *Property View* window, select the **Constant Back** option to specify that a constant form is printed on the backside of the sheet without variable data.

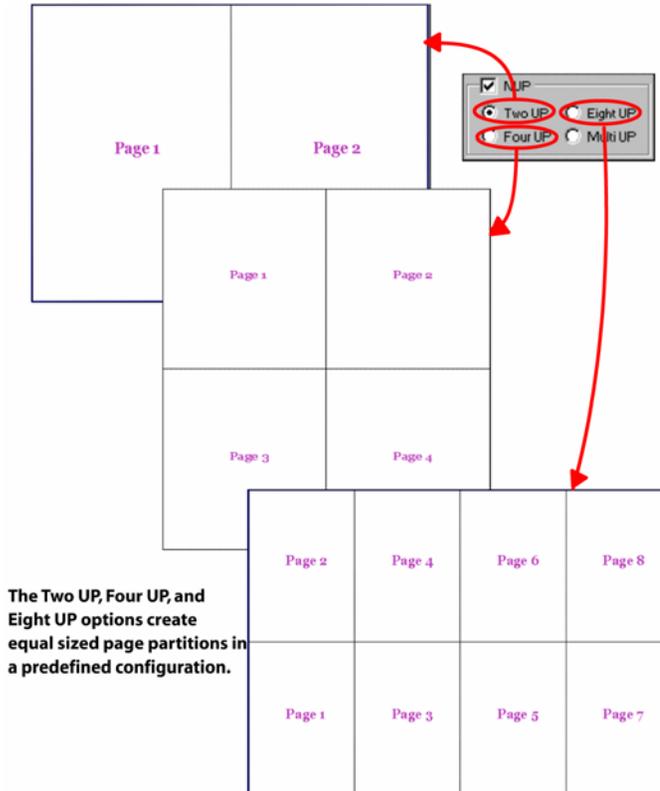
Printing Several Pages on One Paper (N-UP)

Number up (NUP) printing option enables you to place multiple logical pages comprising a document on a single sheet of paper. NUP printing is useful for saving paper on drafts and for printing booklet layouts. The sheet is divided into equal-sized areas called partitions, in which the pages are placed. NUP printing can be performed in both simplex and duplex printing mode. Vitesse provides options to use predefined partitions or create custom partitions for N-up printing.

To print several pages on one paper:

- 1 Select the required **Paper** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **NUP** to enable NUP printing.
The associated NUP options are enabled.

- 4 Select one of the following options to set up the paper partitions:
 - **Two UP** – specifies that two logical pages are to be placed on the paper while printing.
 - **Four UP** – specifies that four logical pages are to be placed on the paper while printing.
 - **Eight UP** – specifies that eight logical pages are to be placed on the paper while printing.



Setting up predefined paper partitions.

- 5 To set up custom paper partitions, select the **Multi UP** option from the **Settings** tab.. The associated options are enabled.
- 6 Click  to add a partition to the custom partitions table.

7 Specify the following properties for each partition:

- **XOrig** – specifies the horizontal coordinate of the logical page relative to the bottom left corner of the physical page.
- **YOrig** – specifies the vertical coordinate of the logical page relative to the bottom left corner of the physical page.
- **XScale** – scales the logical page horizontally.
- **YScale** – scales the logical page vertically.



You can also set the unit of measurement for the logical page partitions by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting Units>*unit name* from the pop-up menu.

- **Rotate** – specifies the rotation angle for the logical page.



You can also set the rotation angle for the logical page partitions by right clicking the upward or downward arrows in the **Rotate** entry boxes, and then selecting Angle>*angle name* from the pop-up menu.

1. Select Multi UP from the NUP options.

2. Click . A corresponding row is added to the partition table to specify a page partition.

3. Set properties for each partition.

4. Pages are adjusted on the paper based on the specified settings.

Each partition is added to the end of the table, you can click  to delete the last added partition.

XOrig	YOrig	XScale	YScale	Rotate
0.00	5.50	50.00	50.00	0.00
6.00	5.00	25.00	25.00	0.00
7.00	0.50	50.00	50.00	90.00

Logical pages are placed relative to the bottom left corner.

The image shows the process of setting custom paper partitions.

Setting Default Paper Properties

You can set default values for the most commonly used paper properties that are applied each time you open a project in Vitesse.

Setting the Default Paper Size

To set the default paper size:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Layout** category under the **Defaults** node.
- 3 From the **Paper** area, select the required paper size option from the **Standard** drop-down list.

The **W** and **H** entry boxes display the width and height of the selected paper.

- 4 To set a paper size other than the predefined values, select the **Custom** option from the **Standard** drop-down list; this option enables the **W** and **H** entry boxes.



You can also set the unit of measurement for the paper size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting Units>*unit name* from the pop-up menu.

- 5 Enter the required custom paper width and height; the selected paper object is updated with the specified paper settings.
- 6 Click **OK**.

The settings take effect the next time a project is opened.

Setting Default Paper Orientation

To set the default paper orientation:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Layout** category under the **Defaults** node.
- 3 From the **Paper** area, select one of the following options from the **Orientation** drop-down list:
 - **Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - **Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
 - **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.
- 4 Click **OK**.

The settings take effect next time a project is opened.

Setting Default Media Settings

To set the default media settings:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 In the *Preferences* dialog, select the **Layout** category under the **Defaults** node.
- 3 From the **Media** area, select one of the following options from the **Type** drop-down list:
 - **Drilled** – specifies that drilled paper is to be used for printing.
 - **Plain** – specifies that non-drilled paper is to be used for printing.
- 4 Type or select the default paper weight value in the **Weight** entry box.
- 5 Select the default paper color option from the **Color** drop-down list.
- 6 Click **OK**.

The settings take effect the next time a project is opened.

User Notes:

Chapter 18: Defining the Logical Page Setup

This chapter describes the following activities and tasks for adding a logical page to a design and setting logical page properties:

- About the Logical Page
- Adding a Page
- Setting the Page Properties
- Selecting a Paper
- Selecting the Paper Side for Printing
- Selecting the Page Orientation
- Setting the Page Margins
- Adding a Page Border
- Creating a Zebra Effect
- Setting the Data Grid
- Adding the Page Numbers
- Formatting the Page Numbers
- Setting the Default Page Properties

About the Logical Page

Page properties refer to the logical page or how the design prints in relation to the physical page/paper.



Paper refers to the physical page on which the design prints. Page properties define the logical page settings for your project.

The logical page determines the area on the physical page/paper where you want to place your variable data and design objects. The logical page merges with the physical page to yield the final print output.

Within the logical page, you can set the margins that define the area in which the data is placed. You can also set the borders and page numbering option. Additional options enable you to set the print direction of the logical page and set up the number of lines and characters per line for printing on the page.

Before defining the page properties, you first need to add a page to your design.

Adding a Page

Every VIPP project contains at least one page object, identified by the default **Page** node under the **SubDoc** node in the project tree. When a new project is created, one page object is automatically added, however, you can add any number of page objects to the project.

To add a page:

- 1 Select **Insert>Page** from the menu bar; a **Page** node is created under the **SubDoc** node in the project tree.



You can also add a new page object by clicking  on the *Insert* toolbar.



You can delete a selected page object either by:

- Selecting **Edit>Delete** from the menu bar.

Or

- Clicking  on the *Standard* toolbar .

You cannot delete the first page under the **SubDoc** node.

Setting the Page Properties

A page object defines the attributes of the logical page. Vitesse enables you to select the paper on which the logical page will be mapped. You can then set the margins and orientation of the logical page in order to determine where and how you want to position the data and design objects on the physical page. This section explains the basic tasks for setting the logical page properties.

Setting the Page Name

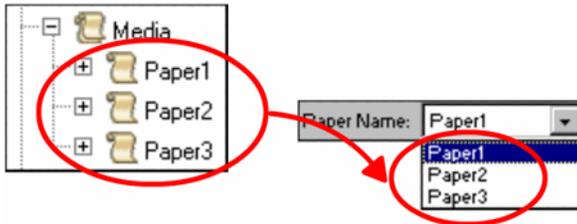
Assigning meaningful names to a page helps in referencing the page when multiple page objects are being used in a project. Vitesse assigns a default name to each newly created page. You can change the default name using the **Name** entry box in the **General** tab of the *Property View* window.

Selecting a Paper

When working with multiple paper objects, you can associate a logical page with a particular paper object. The logical page then merges with the specified paper at print time.

To select a paper:

- 1 Select the required **Page** node from the project tree.
- 2 From the **General** tab of the *Property View* window, select the required paper from the **Paper Name** drop-down list. The **Paper Name** drop-down list contains the name of all the paper objects used in the current project. If no paper option is selected from the list, the page prints with the first (default) paper object. See the image below:



Selecting the Paper Side for Printing

To select the paper side from where the logical page starts printing:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 Select the required page start printing option from the **Start on** drop-down list. Either:
 - **Default** – starts printing from the default paper side.
 - **New front** – starts printing from the new paper front.
 - **New back** – starts printing from the new paper back.
 - **New side** – starts printing from the new paper side.



Before using the **New front**, **New back**, and **New side** options, you must select the **Duplex** printing option for the associated **Paper** object.

Selecting the Page Orientation

Vitesse lets you print the logical page in one of the four print directions available from the **Orientation** drop-down list under the **Paper** area in the **General** tab of the *Property View* window. You can select either of the following options:

- **Portrait** – sets orientation to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width. This will print the data from left to right in the top to bottom direction.
- **Landscape** – sets orientation to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height. This will print the data from bottom to top in the left to right direction.
- **Inverse Portrait** – sets orientation to inverse portrait; this orientation is obtained by rotating a portrait page by 180 degrees. This will print the data from right to left in the bottom to top direction.
- **Inverse Landscape** – sets orientation to inverse landscape; this orientation is obtained by rotating a portrait page 90 degrees clockwise. This will print the data from top to bottom in the right to left direction.

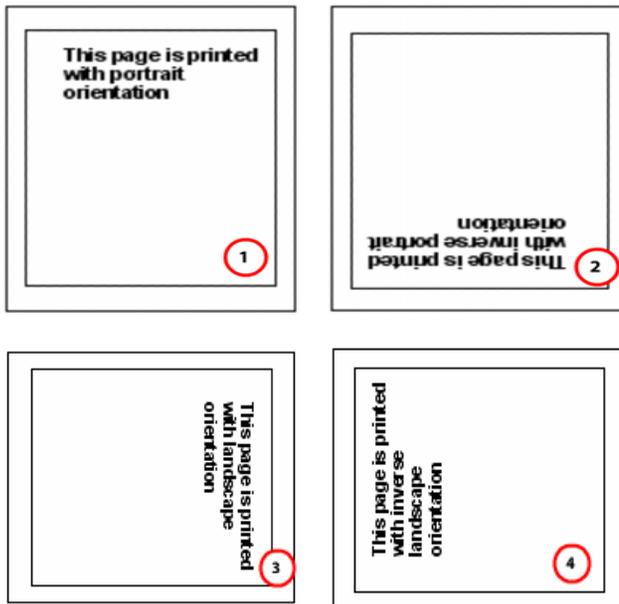


Image shows setting the logical pages to different print orientations. 1) set to portrait 2) set to inverse portrait 3) set to landscape 4) set to inverse landscape.



You can also set the logical page to print on the physical paper such that both the page and paper are set with different orientations.

Setting the Page Margins

The logical page determines the area on the physical page where you want to place your design objects and data. Within the logical page, you can set the margins that define the area in which the data is placed. You can set the top, bottom, left, and right page margins. The top and left margins are measured from the top left edge of the page, whereas the bottom and right margins are measured from the bottom right edge of the page.

You can specify the starting point for the first line of data inside the logical page using the **Margins** area in the **General** tab of the *Property View window*.

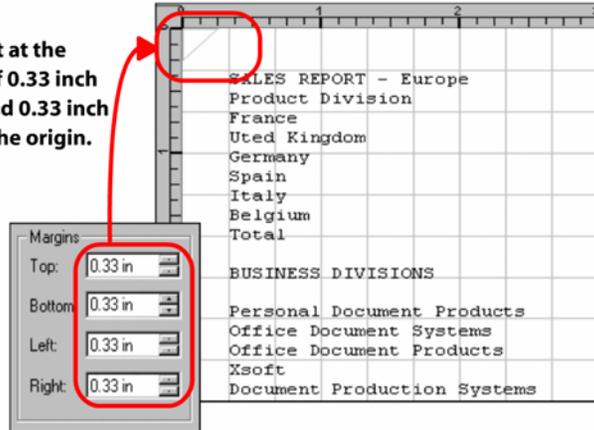
To set the logical page margins, specify the following:

- **Top, Bottom, Left, Right** – sets the starting offset values for the top, bottom, left, and right margins respectively, per specified unit of measurement.

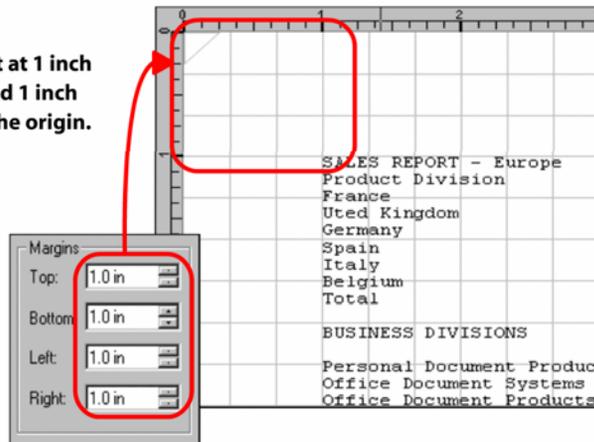


You can also set the unit of measurement for specifying margins by right clicking the upward or downward arrows in the **Top, Bottom, Left,** and **Right** entry boxes, and then selecting Units>*unit name* from the pop-up menu.

The margins set at the default value of 0.33 inch from the top and 0.33 inch to the right of the origin.



The margins set at 1 inch from the top and 1 inch to the right of the origin.



This image shows an example of setting margins.

Adding a Page Border

You can add a border as part of the logical page design, which creates a visual line along the edges of the page. You can define various border settings such as width, color, offset, and corner.

To add a border to your page:

- 1 Select the required **Page** node from the project tree.
- 2 From the Property View window, select the Settings tab.

- 3 Select the **Border** option; this enables the following border settings controls.
 - **Width** – sets the border width, per specified unit of measurement.
 - **Color** – sets the border color from the list of available colors; selecting Custom opens the *Color Selector* dialog for defining a custom color. For more information, refer to *Chapter 11: Using Colors*.
 - **Offset** – sets the space between the page margins and page border, per specified unit of measurement.
 - **Corner** – sets the border corner radius, per specified unit of measurement.



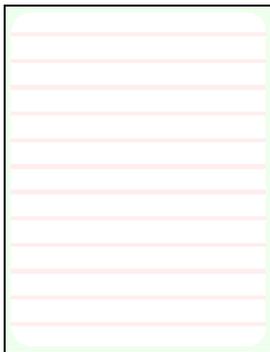
You can also set the unit of measurement while specifying border settings by right clicking the upward or downward arrows in the **Width**, **Offset**, and **Corner** entry boxes, and then selecting Units>*unit name* from the pop-up menu.

Creating a Zebra Effect

You can create a zebra effect using a variety of colors and shading patterns to visually enhance your logical page design.

To create a zebra effect:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Zebra Settings** option; this enables the following zebra settings controls.
 - **Color** – sets the color for the zebra pattern from the list of available colors; selecting **Custom** opens the *Color Selector* dialog for defining a custom color. For more information, refer to *Chapter 11: Using Colors*.
 - **Colored** – sets the number of colored lines.
 - **White** – sets the number of white lines.



Example of a logical page with green borders and zebra pattern.

Setting up a Data Grid

You can setup a data grid by specifying the number of characters per line (CPL) and the number of lines per page (LPP) that can be mapped to a logical page.



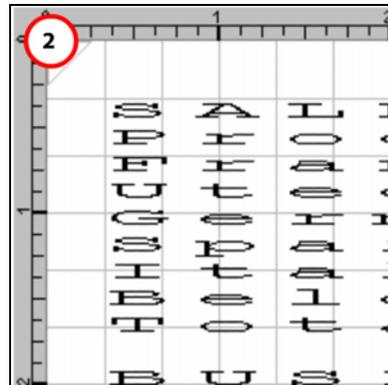
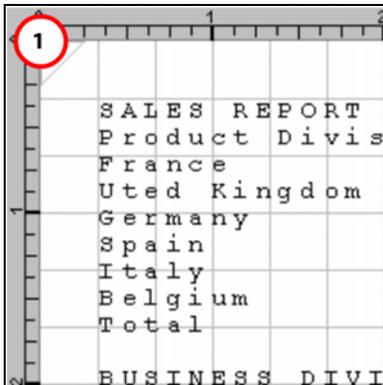
The option for setting up the data grid by specifying the **CPL** and **LPP** is only applicable for line mode project type.

To setup the data grid:

- 1 Select the required **Page** node from the project tree
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Enter or select the number of characters per line mapped from the line data file to the logical page in the **CPL** entry box.



The characters per line (CPL) settings take effect only when the font size is set to 0 (in the **Font** tab).

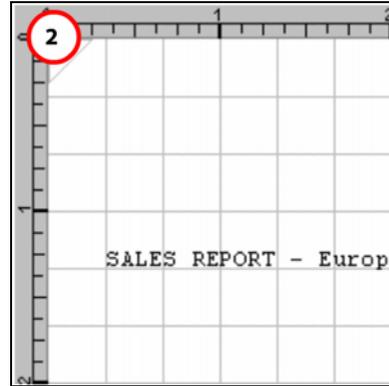


The design area reflects the characters per line value set to (1) 80 (2) 20.

- 4 Enter or select the number of lines per page (LPP) mapped from the line data file to the logical page in the **LPP** entry box.



In the Line mode, reaching the LPP value causes a page break. Page breaks occurring in the data file take higher precedence and cause a page break even if the specified LPP value has not been reached.



The design area reflects the line per page value set to (1) 66 (2) 10.

- 5 Click **OK**.



If you are working in the Line mode, the **Listing color** drop-down displays in the **Settings** tab. Refer to *Chapter 21: Understanding Line Mode Data* for more information. Similarly, in the XML mode, the **XML** drop-down list and the **Parent Tag** option displays in the **Settings** tab. See *Chapter 23: Understanding XML Mode Data* for more information.

Adding the Page Numbers

You can add page numbers to your documents. You can also customize the page number by setting the font attributes, number format, and color. The customized number can be placed anywhere on the page.

To add page numbers:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Page Number** tab.
- 3 Select the **Page Number** option; this enables the page number formatting controls, which are explained below.

Formatting the Page Numbers

You can format the page numbers by specifying the text, the number format, font, and color options.

To format the page numbers:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Page Number** tab, and then select the **Page number** option.

- 3 Enter the number format in the **Format** entry box; the # symbol represents the number and any text that you enter along with # displays with the page number on the page. You can specify the number of digits to be displayed by specifying any number of # signs in the **Format** entry box.



The mask # prints the page number. You can remove or replace the character string “Page.” For example, you can use “Pg” or select any other string appropriate for your project.

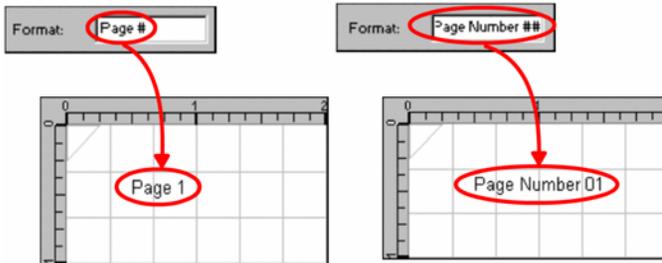


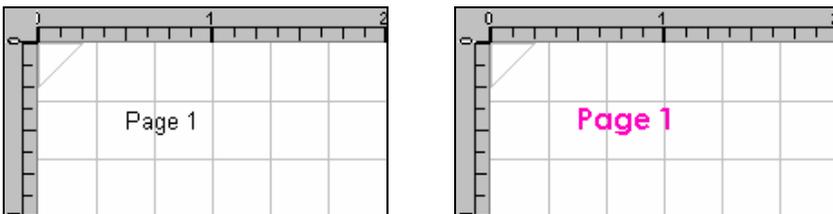
Image shows setting the page number format.

Setting the Page Number Color

To set the page number color, choose the required color from the **Color** drop-down list. Select **Custom** to set a custom color from the *Color Selector* dialog.

Setting the Page Number Font

Select the required font option for the page number from the **Font** drop-down list. All fonts defined in the **Font Styles** tree node are available in the **Font** list. To select a font other than the predefined options, click next to the **Font** list; Vitesse highlights the **Font Styles** tree node and enables you to define the new font in the **Font** tab. You can then switch to the required **Page** node and select the new font from the **Font** list.



The design area reflects (1) the page number with default font and color settings (2) the page number with modified font and color settings.

Specifying the Starting Page Number

You can choose to start the page numbering from a value other than the default value of 1.

To set the page number start value:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Page Number** tab.
- 3 Select the **Use Start from** option to specify the starting page number. The **Start from** entry box is enabled.
- 4 Specify the page number start value in the **Start from** entry box.

Positioning the Page Numbers

You can position the page number at any location on the page.

To position the page number:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Page Number** tab.
- 3 From the **Positioning** area, select the required positioning option from the **Position** drop-down list:
 - **Top Center** – positions the page number at the top center of the page.
 - **Top Right** – positions the page number at the top right of the page.
 - **Bottom Center** – positions the page number at the bottom center of the page.
 - **Bottom Right** – positions the page number at the bottom right of the page.
- 4 To position the page number at a position other than the predefined locations, select **Custom** from the **Position** drop-down list; the **X**, **Y**, and **Alignment** entry boxes are enabled.
- 5 Specify the horizontal and vertical coordinates of the page number in the **X** and **Y** boxes respectively.



You can also set the unit of measurement for the page number position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

- 6 Select either of the following alignment options from the **Alignment** drop-down list:
 - **Left** – aligns the custom page number to the left of the page.
 - **Right** – aligns the custom page number to the right of the page.
 - **Center** – aligns the custom page number to the center of the page.



The page number is aligned depending on the horizontal and vertical coordinates you specified earlier.



- For information on defining new fonts, refer to *Chapter 6: Working with Fonts*.
- For information on using the **Frame** tab, refer to *Chapter 25: Using Frames*.

Setting the Default Page Properties

Vitesse enables you to set the default page properties that are applied every time you open a project.

Setting the Default Margins

To set the default margins:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Layout** category under the **Defaults** node.
- 3 In the **Margins** area, set the default **Top**, **Bottom**, **Left**, and **Right** values in their respective entry boxes to specify the page margins.
- 4 Click **OK**. Pages are added with the specified default margins the next time you insert them.

Setting the Default Data Grid

To setup the default data grid:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the Preferences dialog, select the Layout category under the Defaults node.
- 3 In the Grid area, set a default value in the Character Per Line (CPL) entry box.
- 4 Set a default value in the Line Per Page (LPP) entry box.
- 5 Click **OK**. Pages are added with the specified default data grid the next time you insert them.

Section VI: Working in Database Mode

User Notes:

Chapter 19: Understanding Database Mode Data

This chapter provides an overview of the database mode data stream and explains the database data structure in detail. It also gives a detailed overview on printing documents using the ZSORT feature. Tasks include:

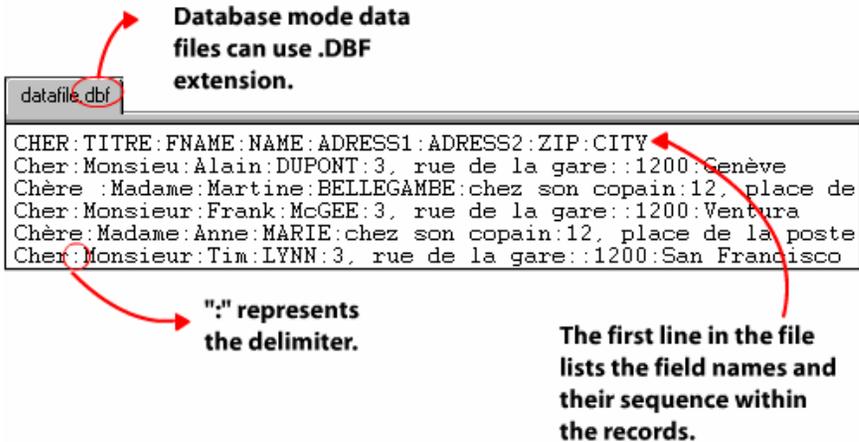
- Understanding Database Mode Data
- Inserting a Data File
- Specifying a Delimiter
- Auto-sizing Data Columns
- Database Data File Structure
- Creating a Z-sort
- Trimming Fields

Understanding Database Mode Data

Database mode projects are used for processing files containing field-delimited records. With the database mode, you can create database-enabled projects, such as mailers, promotional flyers, insurance contracts, etc. Data for these projects is extracted from a database file, which is defined in the first record. The database file specifies field names and their sequence within the record. The first line in the data file contains field names. Remaining lines contain variable data in the order of the field names. Delimiters are used to separate fields within records. Data files used in the database mode can use the .DBF extension.

The following image is an example of a data file in the Database mode:

Database mode data files can use .DBF extension.



```
datafile.dbf
CHER:TITRE:FNAME:NAME:ADRESS1:ADRESS2:ZIP:CITY
Cher:Monsieur:Alain:DUPONT:3,rue de la gare::1200:Genève
Chère:Madame:Martine:BELLEGAMBE:chez son copain:12,place de
Cher:Monsieur:Frank:McGEE:3,rue de la gare::1200:Ventura
Chère:Madame:Anne:MARIE:chez son copain:12,place de la poste
Cher:Monsieur:Tim:LYNN:3,rue de la gare::1200:San Francisco
```

":" represents the delimiter.

The first line in the file lists the field names and their sequence within the records.

Inserting a Data File

Vitesse enables you to use a data file for designing and composing variable-data documents. The project specific production data file is specified at the time of project creation. If a data file is not specified, Vitesse uses a default sample data file for the project. The sample data file normally represents the original production data file and covers all of the possible scenarios. You can change the data file any time during the design process by inserting a new data file.

To insert a new data file:

- 1 Select **Insert>Datafile** from the menu bar to open the *Insert Data File* dialog with the **Existing** tab selected.



You can also open the *Insert Data File* dialog by clicking  on the *Insert* toolbar.

- 2 Search **Look in** drop-down list for the required target folder.
- 3 Select the required data file, and then click **Open**.

The *Data View* window opens with the specified data file. The data file **Path** under the **Data File** area in the **General** tab of the **SubDoc** node property window is also updated to display the path of the newly inserted sample data file.

Specifying a Delimiter

To specify a field delimiter for the datafile used in the database mode:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Field** area, select the required delimiter from the **Delimiter** drop-down list.
- 4 To select a delimiting character other than the predefined values, select the **Other** option from the **Delimiter** drop-down list; this option enables the **Other** entry box.
- 5 Enter the required delimiting character in the **Other** entry box. The delimiter can only be one character in length.
- 6 Click **Apply**.

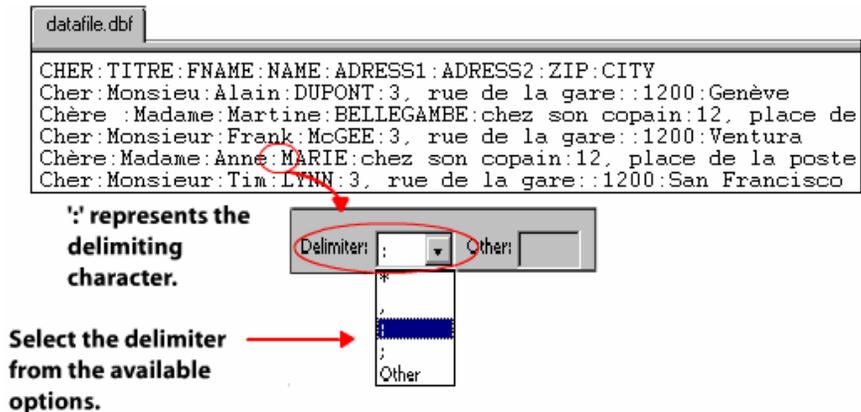


Image shows a sample database file with “:” as the specified delimiter.

Using the Data View Window for Database Mode Projects

The data files inserted in a project can be viewed in the *Data View* window. This helps you to visually drag and drop the required data directly onto the design area. The *Data View* window is docked at the right side of the workspace by default and contains two tabs:

- The **Data** tab shows the data file in a rows and columns layout for database mode projects. The data file displays the field names and their sequence within the records in rows and columns.
- The **Source** tab enables you to edit source data files, and then review the changes in the **Data** view.



For more information on editing a data file and customizing the *Data View* window, refer to the tasks: “Editing the Data File” and “Setting Data View Options” in *Chapter 21: Understanding Line Mode Data*.

The *Data View* window is not displayed by default when a project is opened. To view the *Data View* window, select Window>Data from the menu bar.



You can also view the *Data View* window either by:

- Clicking  on the status bar.
- Or
- Pressing <Ctrl> + <3> on the keyboard.

Data View window displaying the Database data as field names and their sequence within the records .

0	CHER	TITRE	FNAME	NAME
1	Cher	Monsieur	Alain	DUPONT
2	Chère	Madame	Martine	BELLECAM
3	Cher	Monsieur	Frank	McGEE
4	Chère	Madame	Anne	MARIE
5	Cher	Monsieur	Tim	LYNN
6	Chère	Madame	Dianne	MARZEL
7	Cher	Monsieur	Alain	DUPONT
8	Chère	Madame	Martine	BELLECAM
9	Cher	Monsieur	Frank	McGEE
10	Chère	Madame	Anne	MARIE
11	Cher	Monsieur	Tim	LYNN
12	Chère	Madame	Dianne	MARZEL

Data View window displaying the source code of the data file.

```
CHER: TITRE: FNAME: NAME: ADRESS1: AJ
Cher: Monsieu: Alain: DUPONT: 3, ru
Chère :Madame: Martine: BELLEGAMB
Cher: Monsieur: Frank: McGEE: 3, ru
Chère: Madame: Anne: MARIE: chez so
Cher: Monsieur: Tim: LYNN: 3, rue d
Chère: Madame: Dianne: MARZEL: chez
Cher: Monsieur: Alain: DUPONT: 3, r
Chère: Madame: Martine: BELLEGAMB
Cher: Monsieur: Frank: McGEE: 3, ru
Chère: Madame: Anne: MARIE: chez so
Cher: Monsieur: Tim: LYNN: 3, rue d
Chère: Madame: Dianne: MARZEL: chez
```

Click the source tab to view or edit the actual data file.

Data file can be viewed using the Data and Source tabs.

Auto-sizing Data Columns

Vitesse automatically adjusts the data column width in the *Data View* window depending on the maximum length of data in any column. You can also re-adjust the column width manually.

To set the column width, select the **Data** tab in the *Data View* window:

1. The data columns are displayed at the maximum width by default.



0	CHER	TITRE	FNAME
1	Cher	Monsieur	Alain
2	Chère	Madame	Martine
3	Cher	Monsieur	Frank
4	Chère	Madame	Anne
5	Cher	Monsieur	Tim
6	Chère	Madame	Dianne

2. Move cursor to the column separator in the column headers, the cursor changes to \leftrightarrow , click and drag the column width to the required width.



0	CHER	TITRE	FNAME
1	Cher	Monsieur	Alain
2	Chère	Madame	Martine
3	Cher	Monsieur	Frank
4	Chère	Madame	Anne
5	Cher	Monsieur	Tim
6	Chère	Madame	Dianne

3. Click  to display the columns at optimum width.

The image shows the process of setting data column size.



Clicking  on the toolbar available at the top of the *Data View* window (**Data** tab) displays the columns at optimum width.



You can also auto size the data view by selecting Tools>Data View>Auto size data view from the menu bar.

Understanding the Database Data File Structure

A database data file contains records and fields. A record is one row or line of data. Each record contains variable field data and each record has the same format as the previous record. A field contains one piece of information, such as a date or a name. The first line in the data file contains field names.

Within the database data file, you may encounter fields containing no data for a particular record. Such fields that have no content within the delimiters are identified as empty fields for that record.

Database mode is record oriented; it reads one record at a time. Each record's data can be mapped to a single logical page.

The following image is an example of a database data file:

A record is one row or line of data. Each record contains variable data in fields. Each record has the same format as the previous record.

The first line lists the field names and their sequence within records

No content within delimiters identifies empty field for this record.

Creating ZSort

ZSort is a newly added feature in Vitesse 2.00, which provides the ability to print simplex or duplex multi-up documents in a specified imposition order that are output in stacks. These stacks are then processed by a cutter that cuts the stacks into piles. When the piles are placed on top of each other, the resulting stack of piles is in the order of the original input data. The stack can then be sent for performing other finishing operations. A slip sheet can be created and inserted between stacks of output to identify the stack boundaries.



To create a slip sheet that identifies the stack boundaries, create a new form to use as a slip sheet. For more information on creating a new form, see the “Creating a New Form” section in *Chapter 4: Working with Forms*.



Z-sort is applicable only in Database mode that produces a single page document per record.

The maximum size of a stack is the number of sheets that can be cut by the post-processing cutter. You need to specify a small but maximum number of sheets, depending upon the amount of memory in the printer and the size of the database records.

Printing Documents Using ZSort

To print documents using the ZSort feature:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 Select the **ZSORT** option; this enables the associated controls.
- 4 Specify the following:
 - **Stack Size** – sets the maximum number of sheets in a stack. This number should be no more than the number of sheets of paper the cutter can cut.
 - **Slip Sheet** – lists forms used within the current project for selecting as a slip sheet.



The form selected as a slip sheet displays at the end of every stack to identify the stack boundaries. You can insert a text box on this form and enter any text (for example, “This is a slip sheet”) to easily identify the end of each stack.

- 5 To specify the number of times each record is to be processed, select either of the following **Process each record** options:
 - **Once** – processes each record in the data file; produces a single logical page for every record.
 - **Twice** – processes each record in the data file; produces two logical pages for each record.
- 6 To fit the number of remaining records with reference to the stack size, select either of the following **Last stack options**:
 - **Shorten last stack** – shortens last stack to fit the number of records. Some blank logical pages may be created to fill the last physical page.
 - **Insert blank pages** – adds blank logical pages to the last stack to match the specified stack size.



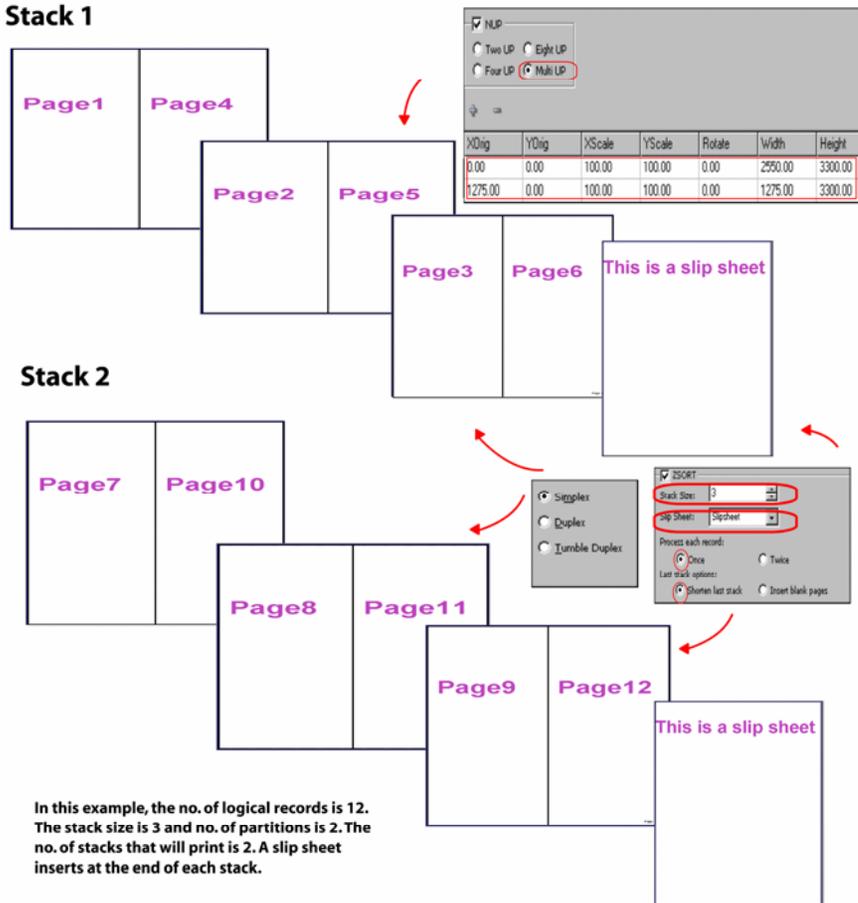
When **ZSORT** is used, the documents are printed in stacks. The number of these stacks is the ratio of the number of logical page records to the product of the stack size and the number of paper partitions.



To place multiple logical pages on a single sheet of paper, Vitesse enables you to create custom paper partitions using the **Multi UP** option from the **NUP** settings provided in the **Settings** tab of the paper property sheet. To print multi up documents using **ZSORT**, specify the sizes of your logical pages using the **Multi UP** option, or else the pages might be placed incorrectly. For more information on specifying **Multi UP** settings, see the “Printing Several Pages on One Paper (N-UP)” section in *Chapter 17: Defining Page and Print Setup*.

Example 1: Using ZSORT for Simplex Printing

The image below (example 1) illustrates the display output when the following settings are applied to create a ZSort: the number of logical page records is 12; and the stack size is set to 3. The document is imposed using the **Multi UP** option so that two logical pages are placed on the paper while printing. In this case, the number of stacks used is 2. When the paper is set to **Simplex**, and each record is processed **Once**, the papers print in the following ZSort manner:

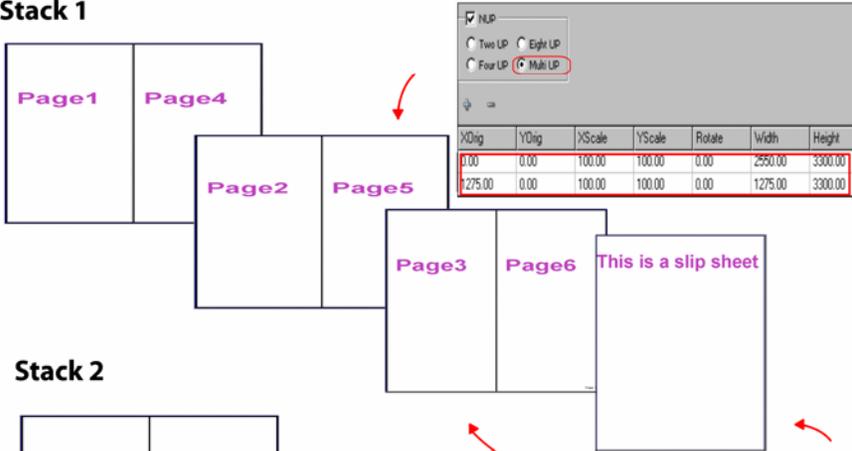


Example 1: Using ZSORT for simplex printing.

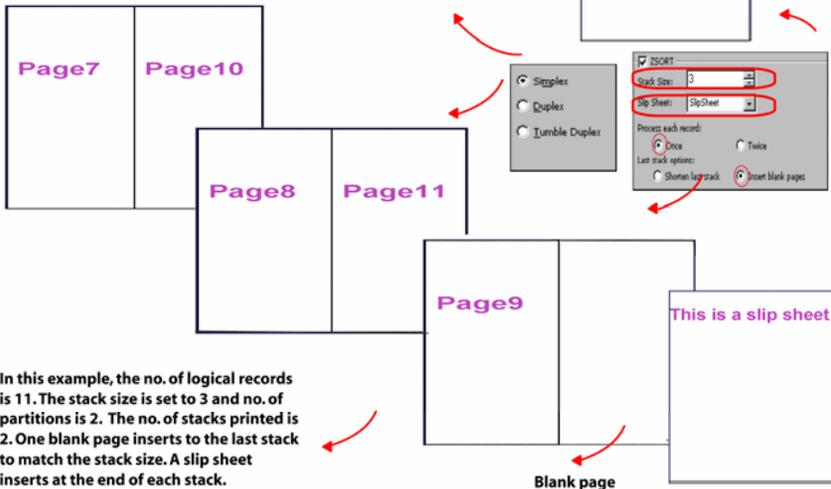
Example 2: Using ZSORT with Insert blank pages option

The image below (example 2) illustrates the display output when the following settings are applied to create a ZSort: the number of logical page records is 11; and the stack size is set to 3. The document is imposed using the **Multi UP** option so that two logical pages are placed on the paper while printing. In this case, the number of stacks is also 2. The **Insert blank pages** option is enabled here. When the paper is set to **Simplex**, and each record is processed **Once**, the papers print in the ZSort manner so that one blank page is added to the last stack to match the stack size.

Stack 1



Stack 2



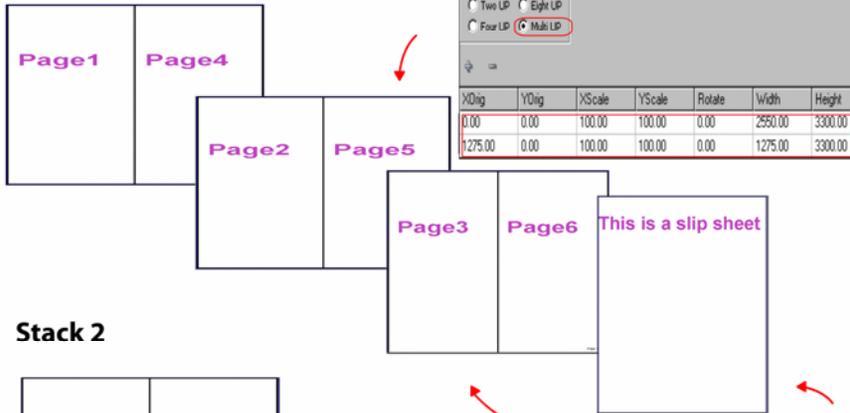
In this example, the no. of logical records is 11. The stack size is set to 3 and no. of partitions is 2. The no. of stacks printed is 2. One blank page inserts to the last stack to match the stack size. A slip sheet inserts at the end of each stack.

Example 2: Using ZSORT with Insert blank pages option.

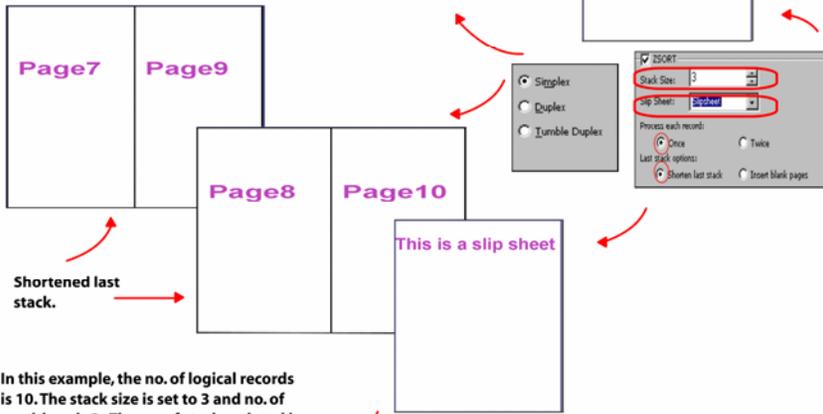
Example 3: Using ZSORT with Shorten last stack option.

The image below (example 3) illustrates the display output when the following settings are applied to create a ZSort: the number of logical page records is 10; and the stack size is set to 3. The document is imposed using the **Multi UP** option so that two logical pages are placed on the paper while printing. In this case, the number of stacks is also 2. The **Shorten last stack** option is enabled here. When the paper is set to **Simplex**, and each record is processed **Once**, the papers print in the ZSort manner so that the last stack shortens to fit the number of records.

Stack 1



Stack 2



In this example, the no. of logical records is 10. The stack size is set to 3 and no. of partitions is 2. The no. of stacks printed is 2. The pages print in Z-sort fashion such that the last stack shortens to fit the number of records. A slip sheet inserts at the end of each stack.

Example 3: Using ZSORT with Shorten last stack option.

Example 4: Using ZSORT with Record Processing set to Twice

The image below (example 4) illustrates the display output when the following settings are applied to create a ZSort: the number of logical page records is 6; and the stack size is set to 3. The document is imposed using the **Multi UP** option so that two logical pages are placed on the paper while printing. In this case, papers print in only one stack. Each record is set to process **Twice**. When the paper is set to **Simplex**, the papers print in the ZSort manner so that each logical page prints twice.

Stack 1

The diagram shows a stack of six pages (Page 1 to Page 6) arranged in a z-sort fashion. Each page is printed twice, once on the front and once on the back, resulting in a total of 12 pages. A slip sheet is placed at the end of the stack. The stack size is set to 3, and the number of partitions is 2. The pages print in only one stack. Each record is set to process twice. The pages print in z-sort fashion, such that each record prints twice. A slip sheet inserts at the end of the stack.

NUP

- NUP
- Two UP
- Eight UP
- Four UP
- Multi UP

XOrig	YOrig	XScale	YScale	Rotate	Width	Height
0.00	0.00	100.00	100.00	0.00	2550.00	3300.00
1275.00	0.00	100.00	100.00	0.00	1275.00	3300.00

Print Options

- Simplex
- Duplex
- Tumble Duplex

ZSORT

- Stack Size: 3
- Slip Sheet: SlipSheet
- Process each record:
 - Once
 - Twice
- Last stack options:
 - Shorten last stack
 - Insert blank pages

In this example, the no. of logical records is 6. The stack size is set to 3 and no. of partitions is 2. The papers print in only one stack. Each record is set to process twice. The pages print in z-sort fashion, such that each record prints twice. A slip sheet inserts at the end of the stack.

Example 4: Using ZSORT with Record Processing set to Twice.

Example 5: Using ZSORT for Duplex Printing

The image below (example 5) illustrates the display output when the following settings are applied to create a Z-sort: the number of logical page records is 16; and the stack size is set to 1. The document is imposed using the **Multi UP** option so that four logical pages are placed on the paper while printing. In this case, papers print in two stacks. Each record is set to process **Once**. When the paper is set to **Duplex**, the papers print in the Z-sort manner on the front and back sides of the paper.

Stack 1

XOrig	YOrig	XScale	YScale	Rotate	Width	Height
0.00	1650.00	100.00	100.00	0.00	1275.00	1650.00
1275.00	1650.00	100.00	100.00	0.00	1275.00	1650.00
0.00	0.00	100.00	100.00	0.00	1275.00	1650.00
1275.00	0.00	100.00	100.00	0.00	1275.00	1650.00

Stack 2

ZSORT

- Stack Size: 1
- Slip Sheet: Duplex
- Process each record: Once Twice
- Last stack options: Show last stack Insert blank pages

In this example, the no. of logical records is 16. The stack size is set to 1 and no. of partitions is 4. When the paper is set to duplex, the no. of stacks printed is 2. The pages print in this Z-sort fashion. A slip sheet inserts at the end of each stack.

Example 5: Using ZSORT for duplex printing.

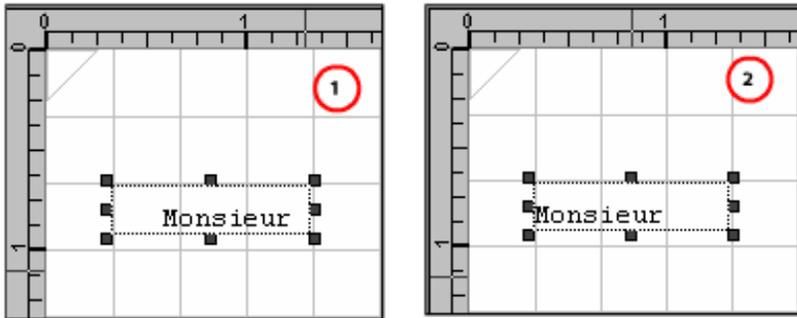
Trimming Fields

In Vitesse, you can trim any leading and trailing spaces in delimited field data in the database mode.

To trim fields:

- 1 Select the **SubDoc** node in the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Field** area, select the **Trim Field** option, and then click **Apply**.

The leading and trailing spaces in the field data are trimmed.



Example Trimming Fields: 1) Before trimming, any leading and trailing spaces are considered 2) After trimming, any leading and trailing spaces are trimmed.



For information on remaining tabs (**Cycle Copy**, **Regional**, and **Properties**), refer to *Chapter 16: "Setting Document Properties."*

User Notes:

Chapter 20: Working with Data in Database Mode

This chapter describes activities and tasks for working with data in database mode projects. Included are instructions and information for the following:

- Mapping Data
- Mapping Fields
- Inserting a Field
- Using Normal Fields
- Using Image Fields
- Using Segment Fields
- Using Static Text Fields
- Positioning a Field
- Setting Spacing between Fields
- Scaling Image and Segment Fields

Mapping Data

The data file structure in the database mode is defined in the first record, which specifies field names and their sequence within the records. The records represent a complete set of information such as the personal information of a customer. Records are composed of fields, which represent a single item of information such as the last name of a customer. The first line in the data file contains the field names for all fields contained in the file. The remaining lines contain the data in the order of the field names. Each field is of variable length separated by field delimiters.

Typically each record contains all of the data for one document. Database mode is record oriented; it reads one record at a time. This means that for any logical page, you can map data/field directly from the current record only on that page.

Data in a data file can be mapped or placed in a database mode project using any of the following three methods:

- Using the Insert menu.
- Using the tools on the *Insert* toolbar. The toolbar is located vertically between the *Project Tree View* window and the design area.
- Using drag and drop to move/map data from the *Data View* window to the design area.

You can map data directly from the *Data View* window by dragging and dropping the required data to the design area. The data is mapped to your document and the corresponding field nodes add to the project tree.



When working with data in database mode, data of multiple records cannot be mapped to one single logical page.

Understanding Database Processing Order

In database mode projects, Vitesse reads one record at a time, i.e., data for first record is processed and printed first for any number of pages, followed by data for the second record and so on. The records are read sequentially from top to bottom. The number of pages depend on the amount of data being mapped. For example, in the image below, the **FNAME** field contains 6 values; hence, 6 logical pages are printed.

0	TITRE	FNAME	NAME	ADRESS1	ADRESS2	ZIP	CITY
1	Monsieur	Alain	DUPONT	3, rue de la gare		1200	Genèv
2	Madame	Martine	BELLECAMBE	chez son copain	12, place de la poste	1034	Lausa
3	Monsieur	Frank	McGEE	3, rue de la gare		1200	Ventu
4	Madame	Anne	MARIE	chez son copain	12, place de la poste	1034	Croco
5	Monsieur	Tim	LYNN	3, rue de la gare		1200	San F
6	Madame	Dianne	MARZEL	chez son copain	12, place de la poste	1034	West

Database data file.

Mapping the first **FNAME** field value, “Alain”, onto the design area automatically maps all of the field values, “Martine”, “Frank”, “Anne”, “Tim”, and “Dianne” in this sequence creating 6 logical pages; one logical page pertaining to one value. A field node is created under the **Page** node in the project tree that identifies all of the 6 values on each logical page. Similarly, mapping the first **ZIP** field value, “1200”, automatically maps all of the remaining values in the required sequence on the logical pages. Each **ZIP** field value corresponds to the associated **FNAME** field value on the logical page, i.e., for “Alain”, “1200” will display on the first logical page, for “Martine”, “1034” is mapped on the second logical page and so on. Therefore, 6 records will be created on 6 logical pages. A second field node corresponding to the **ZIP** field is created under the first field node in the project tree.

For each logical page, you can map data directly from the current record only.

Six records are mapped on six logical pages.

Each record contains all the data for one document. You can navigate between the six documents using the status bar.

Mapping Data in a database mode project.

For a project where multiple fields within each record are mapped on different logical pages, the first record encountered in the data file is processed and printed first for each associated logical page. Once the required data mapping of the first record completes, Vitesse moves on to read the contents of the second record.

Mapping Fields

In a database mode project, a field can be directly mapped to a page.



You cannot insert a record object to a page in database mode projects because each record or line of data pertains to one document.

You can set different field properties such as the field color, font, alignment, rotation, and position using the *Property View* window. Vitesse also enables you to insert normal, image, segment, and static text fields to a project. The field data can also be used for creating charts.



For more information on using field data to create charts, see the “Using Fields as Chart Data” section in *Chapter 28: Adding Charts*.

Tasks in this section detail the steps required to insert a field, use different field types, and set associated properties.

Inserting a Field

To insert a field in a database mode project:

- 1 Select the required **Page** node from the project tree.
- 2 Select Insert>Field from the menu bar; a **Field** node is created under the selected **Page** node in the project tree.

You can also add more field objects under a selected field using the same procedure.



You can also insert a field object either by:

- Clicking  on the *Insert* toolbar.
Or
- Right-clicking the required **Page** node, and then selecting **Insert>Field** from the pop-up menu.



You can delete a selected field object either by:

- Selecting Edit>Delete from the menu bar.
Or
- Clicking  on the *Standard* toolbar.
Or
- Pressing on the keyboard.

Using Normal Fields

Normal fields are the most commonly used field type. These fields can contain static text; data from variables or from the data file.

To set the field type as normal:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Normal** from the **Field Type** drop-down list.

Specifying the Normal Field Data

The data for the normal fields can be picked up from the data file or from a variable. Normal field data can also be specified using static text. This data can then be mapped directly to a page or used as a label or value for a data driven chart. Methods for using the data file, variables, and static text as normal field data are explained below.

Using the Data File

You need to specify the field name when identifying field data from the data file.

To specify the field data using the data file:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Data**, which indicates that a data value from the data file will be associated with the current field. Selecting this option enables the associated drop-down list.
- 4 Click the drop-down arrow associated with the **Field Name** entry box. It displays all of the field names from the data file associated with the project.
- 5 Select the required **Field Name** from the drop-down list.

The field updates in the design area to display the field data from the data file.

1. Select **Data**. The **Field Name** entry box enables.

2. Click the drop-down associated with the **Field Name** entry box. Select the **FNAME** field name from the drop-down list.

3. The field updates in the design area to display the field data from the data file.

TITRE	FNAME	NAME	ADDRESS1	ADDRESS2	ZIP	CITY
Monsieur	Alain	PUPONT	3, rue de la gare		1200	Genève
Madame	Martine	BELLECAMBE	chez son copain	12, place de la poste	1034	Lausanne
Monsieur	Frank	McCREE	3, rue de la gare		1200	Ventura
Madame	Anne	MARIE	chez son copain	12, place de la poste	1034	Crocodoc
Monsieur	Tim	LYNN	3, rue de la gare		1200	San Francisco
Madame	Dianne	MAPEEL	chez son copain	12, place de la poste	1034	West Coast

The image shows the process of specifying field data using the data file.

Using Variables

To specify the field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable**, which indicates that a variable will be used to specify the field data. Selecting this option enables the associated drop-down list.
- 4 Select the required system or user-defined variable from the drop-down list.

The field updates in the design area to display the variable value.



For more information on working with variables, see *Chapter 26: Using Variables*.

You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area, as shown in the image below:

Variables

Name: VARname

Type: String Persist Variable

Value: John Brown

1. Create a variable using the MasterDoc property sheet.

2. Add a Field object under the Page object, by clicking  on the Insert toolbar.

3. In the Settings tab of the field property sheet, select the Const option.

Field Data:

Fixed Start: 0 Length: 0

Data Field Name: _____

Variable _____

Const `$$VARname.`

4. Specify the required variable name using the correct syntax.

5. The syntax displays in the design area.



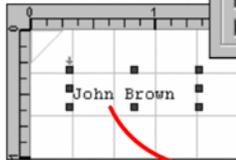
6. In the Settings tab of the field property sheet, select the Merge data option.

Data Merging

Merge data

Lookup variables

Empty if not found



7. The variable value displays in the design area.



This process is also applicable for segment and image type fields explained in later sections.

Using Constant Data

You can insert static text as field contents. To specify the field contents using static text in a database mode project:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**, which indicates that constant data will be used as field contents. Selecting this option enables the associated entry box.
- 4 Enter the required text in the **Const** entry box.

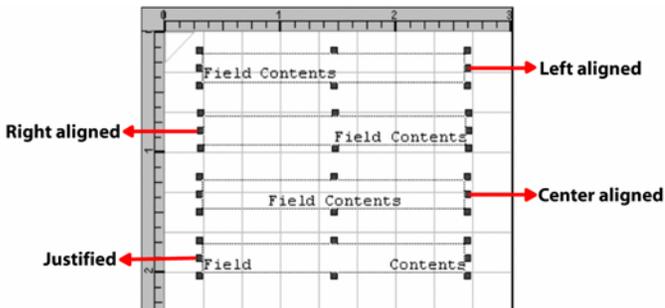
The field updates in the design area to display the text.

Aligning the Normal Field Contents

Alignment is the horizontal positioning of text within a line, relative to a point of reference. Vitesse provides various alignment options for field contents to achieve a more visually balanced design.

To align the field contents:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Left** – aligns the field contents along the left field margin.
 - **Right** – aligns the field contents along the right field margin.
 - **Center** – aligns the field contents symmetrically in the middle of the field.
 - **Justify** – aligns the field contents along both the left and right field margins by adjusting the spacing between the field data.



The design area reflects the effect of the basic alignment options.

Defining the Cell Margins

You can customize the appearance of a normal or static text field by applying borders, fill colors, and patterns using the options in the **Cell** tab of the *Property View* window of the field object. For more information on defining cell margins, see related tasks in *Chapter 22: Working with Data in Line Mode*.



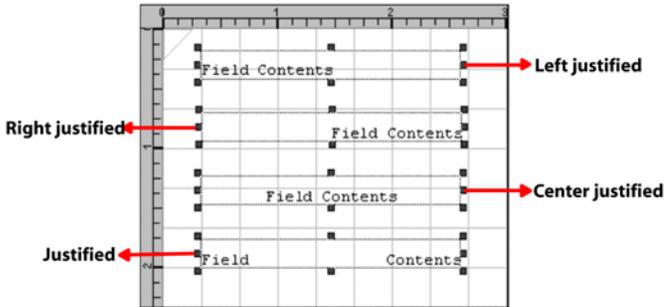
In **Line** and **XML** mode projects, applying cell options to transactional records creates a dynamic table. You cannot create a dynamic table in database mode because there are no transactional records.

Setting Field Justification

In Vitesse, you can align the field data relative to the left and right edges of the field margin. The data can be aligned to one edge (left or right) of the field margin, centered in the field object, or aligned to both edges of the field margin (full justification). Additional options to specify the vertical justification of the normal field contents to top, bottom, or center of the field object are also available.

To specify the field horizontal and vertical justification:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab.
- 3 From the **Justification** area, select one of the following horizontal justification options:
 - **Left** – click  to start data from left-most position and extend right with normal spacing.
 - **Right** – click  to start data from right-most position and extend left with normal spacing.
 - **Center** – click  to equidistance data from left and right.
 - **Justify** – click  to start data from left-most position and end in right-most position; data is aligned evenly within the field object.



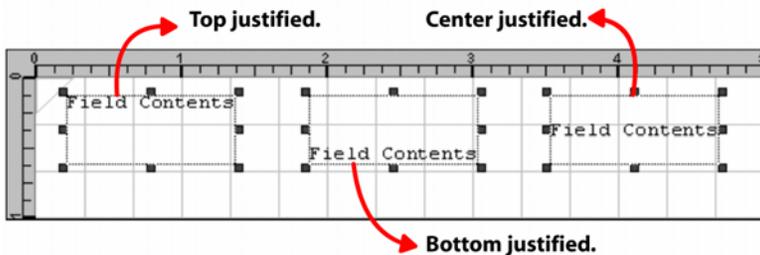
The design area reflects the effect of the horizontal justification options.



When working with **Normal** fields, specifying the horizontal justification options updates the **Alignment** drop-down list accordingly in the **Settings** tab to reflect the same settings and vice versa.

4 Select one of the following vertical justification options:

- **Top** – click  to align the field contents along the top field margin.
- **Bottom** – click  to align the field contents along the bottom field margin.
- **Center** – click  to align the field contents at the field center.
- Both the horizontal and vertical field justification settings work together to produce the final output.



The design area reflects the effect of vertical justification options. In this example, the horizontal justification is set to Left.

Using Image Fields

An image resource can be added to a database mode project using image fields. In an image field, the value of the field is the name of the image source file and its extension; this value can be picked from a data file or from a variable. The image is positioned relative to the bottom left edge of the field object. The specified image is added to the **Images** node under the **Resources** node in the project tree. You can position, align, scale, and rotate the added image field. All image formats supported by Vitesse can be inserted using image fields.



The image fields can only reference the images placed in the Vitesse global image library located at *drive:\Program Files\Elixir Technologies\Vitesse\ImgLib*. For more information on global resource folders, see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as an image:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Image** from the **Field Type** drop-down list.

Specifying an Image Resource

The data that references the image resource can be picked up from the data file or a variable (that has the image file name and extension as its value). You can also use existing image resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The image resource specified using the data is then mapped to the page. Methods for using the data file and variables as an image field data are explained below.

Using the Data File

Vitesse enables you to customize your print documents by using different images for different documents generated from a single project design. For example, you may need to display a specific picture for each of the customers in their individual profiles. Vitesse allows you to create such documents by the use of image fields, which acquire data dynamically from the data file. The name and extension of the required image is available as field data in the data file. The following image shows the steps involved in this process:

2. Select Data from the Field Data area and set Image as the Field Name in the drop-down list.

1. Select Image from the Field Type drop-down list.

The screenshot illustrates the configuration of a field in a software application. On the left, a 'Field Type' dropdown menu is open, showing options: Image, Normal, Segment, and Static Text. A red arrow points from the 'Image' option to the 'Field Data' panel. The 'Field Data' panel has three radio buttons: 'Fixed', 'Data', and 'Const'. The 'Data' radio button is selected. Below it, the 'Field Name' dropdown is set to 'IMAGE'. A second red arrow points from the 'IMAGE' dropdown to the 'Data View' window. The 'Data View' window displays a table with the following data:

0	CHER	TITRE	FNAME	IMAGE
1	Cher	Monsieur	Alain	John Brown.tif
2	Chère Madame	Martine		Mary Brown.tif
3	Cher	Monsieur	Frank	Maz Brown.tif
4	Chère Madame	Anne		Chris Brown.tif
5	Cher	Monsieur	Tim	Tim Brown.tif
6	Chère Madame	Dianne		Mark Brown.tif

Red boxes highlight the 'IMAGE' column header and the first row of data. A third red arrow points from the 'IMAGE' column header to a design area on the left. The design area shows a grid with the text 'Mr. John Brown' and a small image of a man in a suit sitting at a desk.

3. The image displays in the design area. The image is also added to the project resources.

The image shows the image field data being specified using the database data file.



Before setting the **Field Type** as an **Image**, ensure the data file that you use has the required image file names and extensions.

Using Variables

To specify the image field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable**, which indicates that a variable will be used to specify the image field data. Selecting this option enables the associated drop-down list. The variables used for referencing images store the name and extension of the image resource.
- 4 Select the required system or user-defined variable from the drop-down list. The field updates in the design area to display the referenced image.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering *\$\$VariableName*. in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area. See the “Using Variables” section above for more information.

Using an Existing Project Resource as Field Data

You can use existing image resources under the **Resources>Images** tree node for the image field.

To specify the image using the project resources:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled.
- 4 Select the required image resource from the drop-down list. The drop-down lists all of the images used in the project.

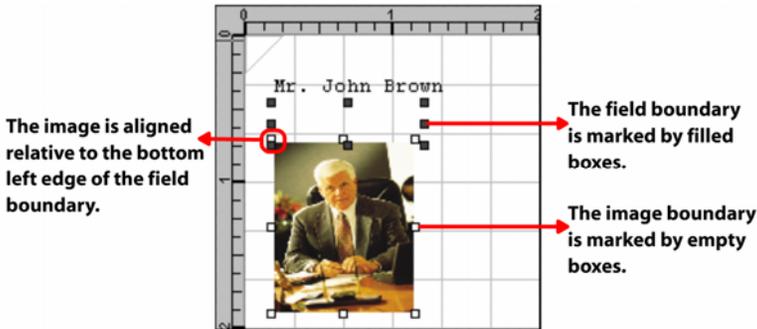
The selected image displays in the design area.



Using the **Const** option, you can also add images not present in the project resources by entering the image name and extension in the entry box.

Aligning the Image Field

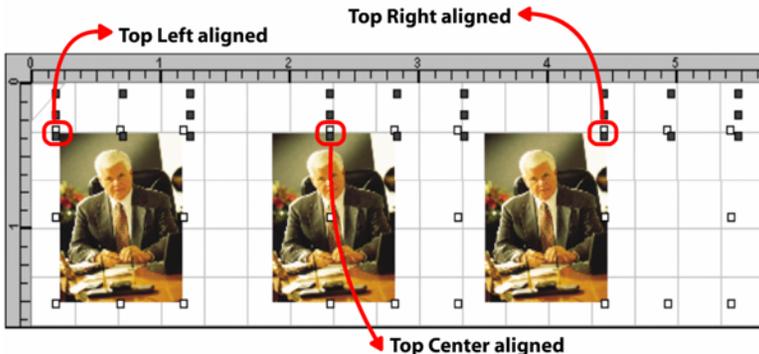
You can position an image relative to a point of reference. The referenced image is always positioned relative to the bottom left edge of the field bounding box. Vitesse provides various alignment options for image fields to achieve a more visually balanced design.



The image highlights the point of reference for various image alignment options.

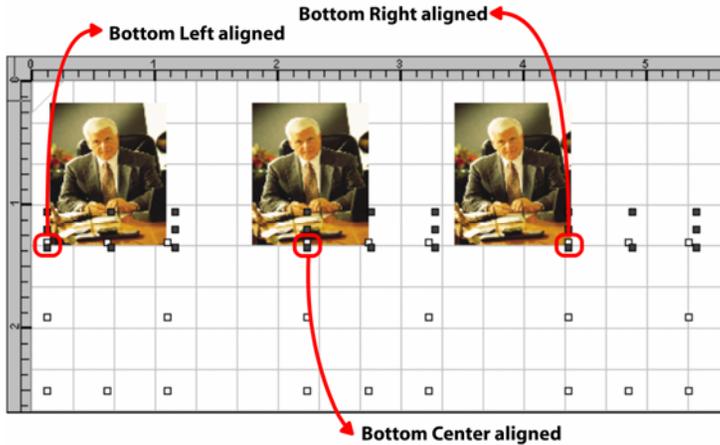
To align the image field:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Top Left** – aligns the top left edge of the image along the bottom left edge of the field.
 - **Top Right** – aligns the top right edge of the image along the bottom left edge of the field.
 - **Top Center** – aligns the top center point of the image along the bottom left edge of the field.



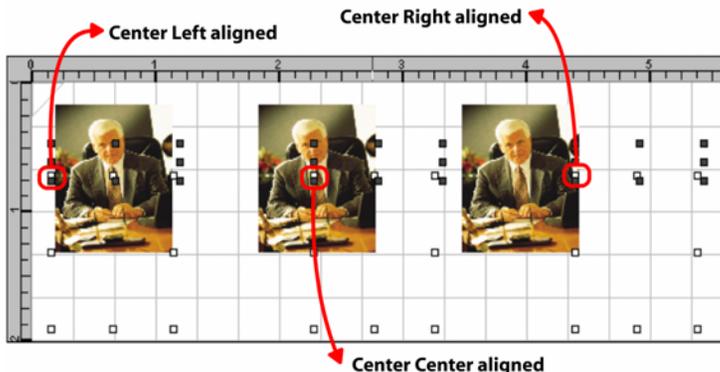
The image shows the effect of the top image alignment options.

- **Bottom Left** – aligns the bottom left edge of the image along the bottom left edge of the field.
- **Bottom Right** – aligns the bottom right edge of the image along the bottom left edge of the field.
- **Bottom Center** – aligns the bottom center point of the image along the bottom left edge of the field.



The image shows the effect of the bottom image alignment options.

- **Center Left** – aligns the center left edge of the image along the bottom left edge of the field.
- **Center Right** – aligns the center right edge of the image along the bottom left edge of the field.
- **Center Center** – aligns the center of the image along the bottom left edge of the field.



The image shows the effect of the center image alignment options.

Using Segment Fields

A segment resource can be added to a database mode project using segment fields. In a segment field, the value of the field is the name of the segment source file and its extension; this value can be picked from a data file or from a variable. The segment is positioned relative to the bottom left edge of the field object. The inserted segment is added to the **Segments** node under the **Resources** node in the project tree. You can position, scale, rotate, and dynamically size the added segment. All segment formats supported by Vitesse can be inserted using the segment fields.



The segment fields can only reference the segments placed in the Vitesse global segment library located at *drive:\Program Files\Elixir Technologies\Vitesse\SegLib*. For more information on global resource folders, see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as a segment:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Segment** from the **Field Type** drop-down list.

Specifying a Segment Resource

The data that references the segment resource can be picked up from the data file or a variable (that has the segment file name and extension as its value). You can also use existing segment resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The segment resource specified using the data is then mapped to the page. Methods for using the data file and variables as a segment field data are explained below.

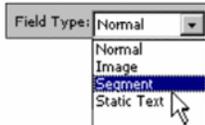
Using the Data File

Vitesse enables you to dynamically specify the segment resource used in a page design via the data file. For database mode field, you can specify the name and extension of the segment as its value. You can then associate the respective field data with the required segment field. On document composition, Vitesse acquires the segment name from the data file, references the required segment from the **SegLib** directory, and displays it in the document design. This functionality enables you to customize your documents; for example, you can create differently designed documents for different clients by the use of varying segments.

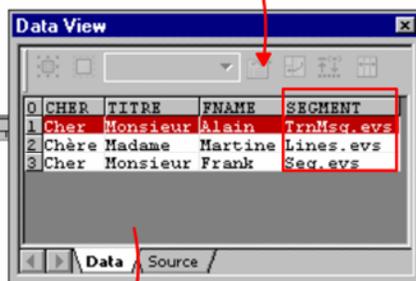


Before setting the **Field Type** as a **Segment**, ensure the data file that you use has the required segment file names and extensions.

1. Select Segment from the Field Type drop-down list.



2. Select Data from the Field Data area and set Segment as the field name in the drop-down list.



3. The segment displays in the design area. The segment is also added to the project resources.

The image shows the segment field data being specified from a database data file.

Using Variables

To specify the segment field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable**, which indicates that a variable will be used to specify the segment field data. Selecting this option enables the associated drop-down list. The variables used for referencing segments store the name and extension of the segment resource.
- 4 Select the required system or user-defined variable from the drop-down list. The field updates in the design area to display the referenced segment.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area. See the “Using Variables” section above for more information.

Using an Existing Project Resource as Field Data

You can use existing segment resources under the **Resources>Segments** tree node for the segment field.

To specify the segment using the project resources:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled.
- 4 Select the required segment resource from the drop-down list. The drop-down lists all of the segments used in the project. The selected segment displays in the design area.



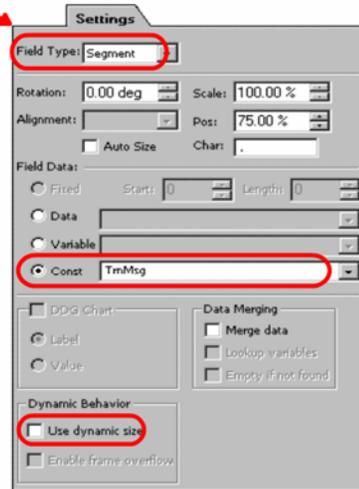
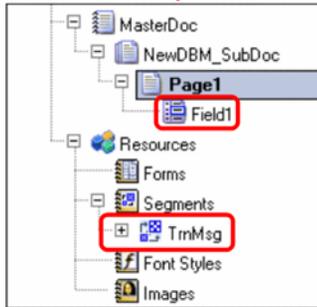
Using the **Const** option, you can also add segments not present in the project resources by entering the segment name and extension in the entry box.

Using the Dynamic Sizing Behavior in a Segment Field

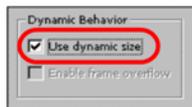
By default, the size of a segment referenced in a page or a form is the same as the size of the original segment. However, Vitesse also enables you to adjust the segment size dynamically depending on the contents. For example, the contents of a segment resource displaying in the design may vary depending on the document data or conditions applied. In such cases, the dynamic sizing option adjusts the height of the segment automatically.

The following image shows the dynamic sizing behavior for segment fields:

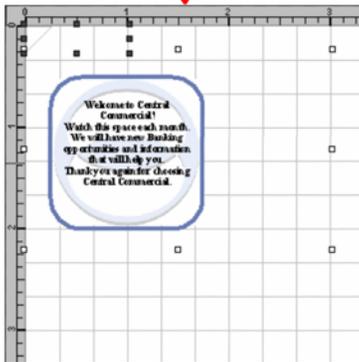
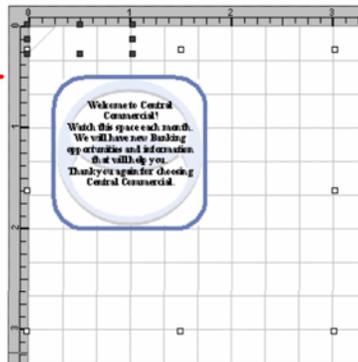
1. Segment field used to reference a segment in the project design.



3. Select Use dynamic size to adjust the segment height according to its contents.



2. Design area displays the segment referenced using a field object.



4. Design area displays the dynamically sized segment.



For more information, refer to the task, “Dynamically Sizing Segments,” in *Chapter 32: Using Segments in Projects*.

Using Static Text Fields

Static text fields extend the functionality of normal fields by providing formatting options (that are the same as available for the text and graphic text object) in addition to all of the normal field functionality. However, there are a few exceptions:

- You cannot set the alignment options for static text fields.
- Also, the static text field data cannot be used as a label and value for a data driven chart.

To set the field type as a static text:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Static Text** from the **Field Type** drop-down list.

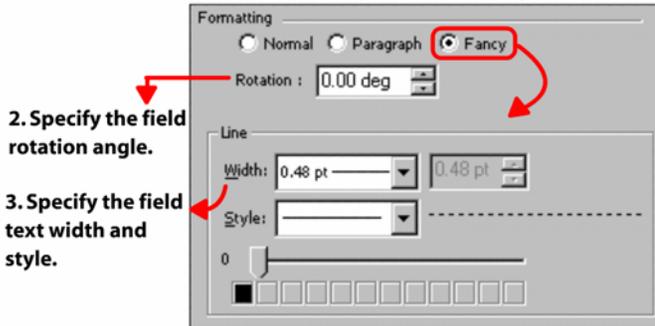
Formatting Static Text Field Data

You can choose to apply text formatting features such as text wrapping, rotation, line style, and width to static text fields.

To apply formatting options:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab.
- 3 From the **Formatting** area, select one of the following options:
 - **Normal** – applies normal formatting to the field data; any text extending the field boundary is not wrapped.
 - **Paragraph** – applies paragraph formatting to the field data; any text extending the field boundary is wrapped.
 - **Fancy** – applies enhanced formatting features to the field text; selecting this option displays the associated options for setting the line width, style, and rotation.

1. Select Fancy, the associated options are displayed.



2. Specify the field rotation angle.

3. Specify the field text width and style.

The image shows the process of applying fancy formatting.



For more information on setting rotation, line widths, and line style, see related tasks in *Chapter 22: Working with Data in Line Mode*.



The line width and style settings are not visible in the design area; however, these are applied when the project is exported.

Specifying Static Text Field Data

The data for the static fields can be picked up from the data file or from a variable. Static field data can also be specified using static text. This data cannot be used as a label or value for a data driven chart. For more information on specifying the field data using the data file, variables and constant text, refer to the “Specifying Field Data” task for normal fields earlier in this chapter.

Setting Common Field Properties

Some field properties are common to all four field types. These include the field size, position, spacing, rotation, and auto-sizing.

Positioning a Field

Field objects can be placed at an absolute position in the design area by specifying the horizontal and vertical coordinates. You can also position the field objects relative to one another.

Setting Field Absolute Position

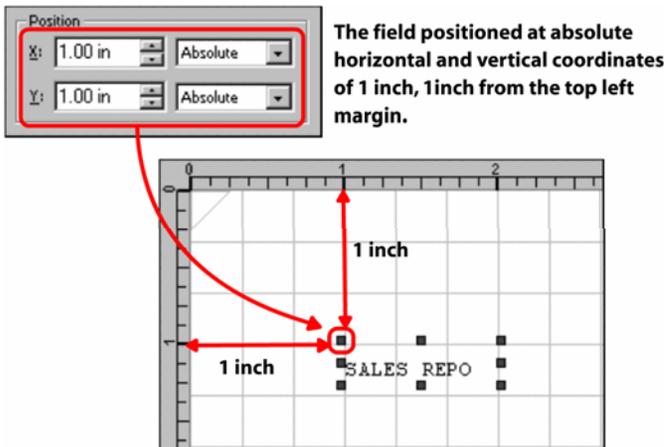
To accurately position a field object in the design area:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the field horizontal position, per specified unit of measurement in the **X** entry box. In the associated drop-down list, select **Absolute**.
- 4 Specify the field vertical position, per specified unit of measurement in the **Y** entry box. In the associated drop-down list, select **Absolute**.



You can also set the unit of measurement for the field position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

The absolute coordinates are calculated from the top left margin of the page.



The image shows the process of setting field position using absolute coordinates. In this example, the top, left margins are set at 0, 0 inches.

Setting Field Relative Position

A field object can be positioned relative to the previous fields and relatively positioned objects added to the page. In absolute positioning, the top left margin of the page acts as the point of reference, whereas in relative positioning, the previously added object is taken as the point of reference.

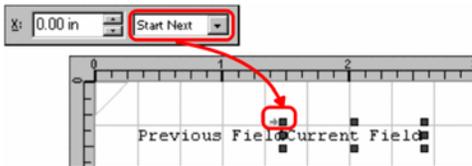
To position fields using relative positioning:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify **0** in the **X** entry box. In the associated drop-down list, select one of the following horizontal relative positioning options:



The horizontal relative positioning options are applied with reference to the field horizontal start, center, and end points.

- **Start Same** – positions the start of the current field at the start of the previous field.
- **Start Center** – positions the start of the current field at the center of the previous field.
- **Start Next** – positions the start of the current field at the end of the previous field.



In this example, the current field is positioned at the previous field end.

- **Center Before** – positions the center of the current field at the start of the previous field.
- **Center Same** – positions the center of the current field at the center of the previous field.
- **Center Next** – positions the center of the current field at the end of the previous field.
- **End Before** – positions the end of the current field at the start of the previous field.

- **End Center** – positions the end of the current field at the center of the previous field.
 - **End Same** – positions the end of the current field at the end of the previous field.
- 4 Specify **0** in the **Y** entry box. In the associated drop-down list, select one of the above-mentioned vertical relative positioning options. Vitesse provides the same set of vertical relative positioning options as the horizontal relative positioning options.



The vertical relative positioning options are applied with reference to the field vertical start, center, and end points.



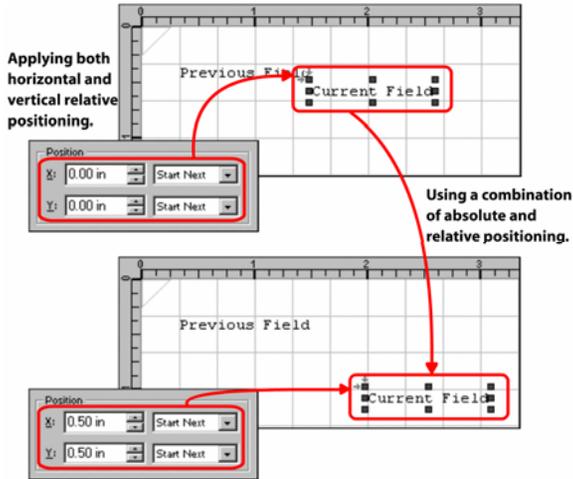
In this example, the current field is positioned at the previous field end.



Horizontal relative positioning options cannot be applied to the first field object added to a Page object.



Using a combination of absolute and relative positioning, you can place the field relative to another field or a relatively positioned object and add the spacing defined by the absolute **X** and **Y** coordinates.



In this example, the current field is positioned relatively at Same Next, Same Next and then displaced 0.5, 0.5 inches in the horizontal and vertical directions.

The following symbols appear next to a relatively positioned field node in the *Project Tree View* window:



-  - specifies that horizontal relative positioning is applied to the field.
-  - specifies that vertical relative positioning is applied to the field.
-  - specifies that both horizontal and vertical relative positioning is applied to the field.



The relative positioning options are applicable for all objects that can be relatively positioned in Vitesse.

Setting the Field Size

Field dimensions specify the width and height of the field. To set the field dimensions:

- 1 Select the required **Field** node from the project tree.

- 2 In the **General** tab of the *Property View* window, specify the following **Size** information:
- **W** – sets the field width per specified unit of measurement.
 - **H** – sets the field height per specified unit of measurement.

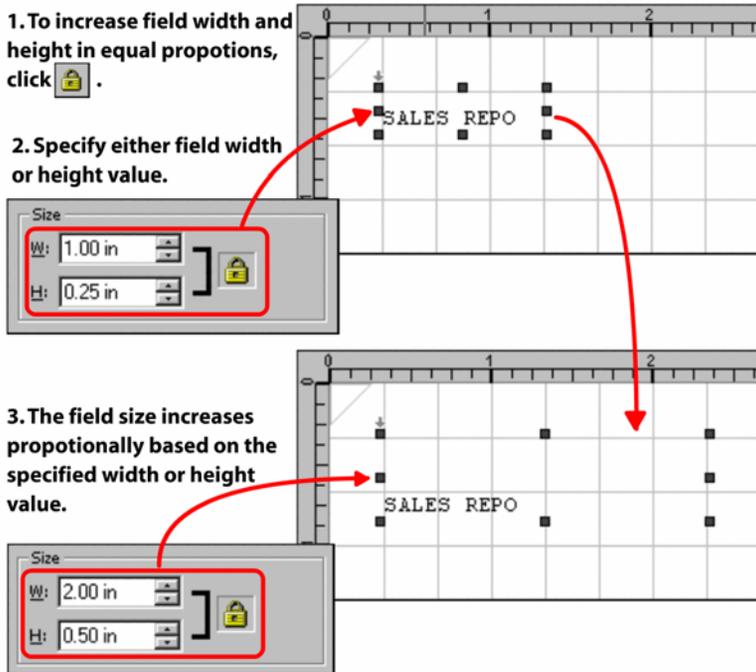


You can also set the unit of measurement for the field by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units**>*unit name* from the pop-up menu.

The  icon in the **Size** area is used for maintaining the aspect ratio of the field. Clicking this icon and then setting the width and height increases or decreases the length and width of the field in equal proportion.

1. To increase field width and height in equal proportions, click .

2. Specify either field width or height value.



3. The field size increases proportionally based on the specified width or height value.

The image shows the process of changing the field size while maintaining the aspect ratio.



You can also set the field size by dragging the field handles that define the boundary of the field object in the design area. The corresponding **Size** settings are updated in the *Property View* window.

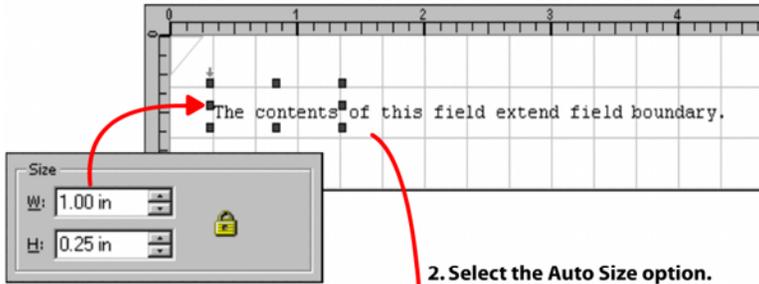
Adjusting the Field Size to its Contents

Vitesse enables you to auto-size field objects to adjust automatically to the amount of data it is displaying. This feature is especially useful when working with variable length (delimited) fields or with fields that contain merged variables. This also helps prevent the overlapping of fields that are placed relative to each other, and removes extra white space in the fields.

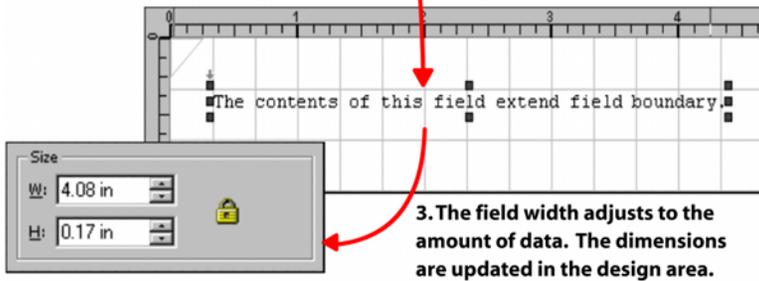
To automatically adjust the field size to the amount of data:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Auto Size** check-box. The field width automatically adjusts to fit all of the data.

1. The contents of the field extend the current field width.



2. Select the Auto Size option.



The image shows the process of adjusting the field size to its contents.



When you use the **Auto Size** option for field objects, the size of the object becomes dynamic and is subject to change depending on the contents in the field. Therefore, the **Auto Size** option affects the placement of relatively positioned objects due to its changing behavior. The VIPP converter, on the other hand, requires an exact size of the field object for positioning the relative objects. Hence, this option must be disabled before exporting projects.

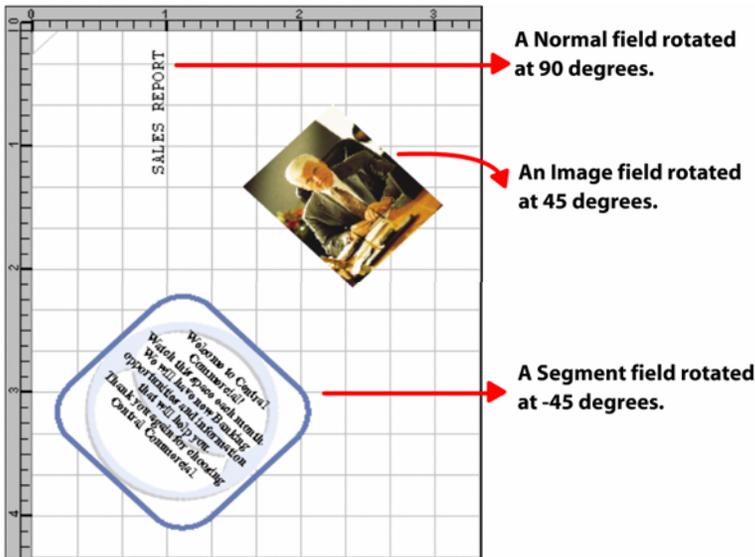
Setting Field Rotation

To set the field rotation:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required rotation angle in the **Rotation** entry box.



You can specify the angle in degrees, radians, or grads. To specify the unit of measurement for the rotation angle, right click the upward or downward arrows in the **Rotation** entry box, and then select **Angle>unit name** from the pop-up menu.



The image shows different rotations applied to various types of fields.



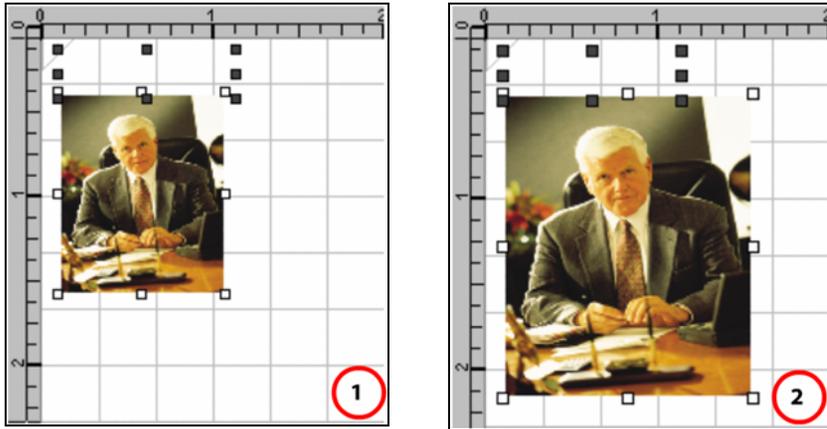
When working with normal or static text fields, you can also specify the rotation angle by selecting the **Fancy** option in the **Spacing** tab, and then specifying the rotation angle in the **Rotation** entry box.

Scaling Image and Segment Fields

The image and segment resources added to the project as fields are scaled at 100% by default. This means that the original image and segment dimensions are used. Once added to the project, you can scale the image and/or segment resource to suit your requirements.

To scale image and/or segment fields:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required scaling value as a percentage, in the **Scale** entry box.



The design area reflects an image field scaled to (1) 100% (2) 150%.



- For information on the **Color** tab, refer to *Chapter 11: Working with Colors*.
- For information on the **Font** tab, refer to *Chapter 6: Working with Fonts*.
- For information on the **Transform** tab, refer to *Chapter 10: Applying Transformations*.

User Notes:

Section VII: Working in Line Mode

User Notes:

Chapter 21: Understanding Line Mode Data

This chapter provides an overview of the line mode data stream and explains the data stream attributes in detail. The chapter also describes activities and tasks for working with data files and the *Data View* window in Vitesse. The list of topics includes:

- Understanding Line Mode Data
- Working with Page Delimiters
- Setting Document Flow Options
- Specifying Printer Control Characters in Data
- Using Advanced Vertical Control Options
- Inserting a Data File
- Specifying a Listing Color
- Using the Data View Window
- Editing the Data File
- Setting Data View Options
- Creating a Data Dictionary
- Working in Listing Mode

Understanding Line Mode Data

Line mode is used for creating and managing line mode data streams for transactional documents, such as telephone bills, banking statements, credit card statements, insurance invoices, and sales reports.

These data streams are also referred to as “line” or “print ready” data. One example is LCDS (Line Conditioned Data Streams). Line mode projects comprise line data in ASCII format that is line numbered or RPE prefixed. Moreover, the data can either have delimited or non-delimited fields within records. Line mode enables you to read, process, and format the line printer data either line-by-line (records) or column-by-column (fields). The .LM extension is recommended for Line mode data files, but is not required, as VIPP and Vitesse can use other extensions.

The page composition of the line mode data is controlled by a Job Descriptor Ticket (JDT). The JDT contains VIPP commands that define the presentation of the data. Line mode is divided into two presentation modes:

- Listing mode
- Record Processing Entry (RPE) mode

Listing Mode

The listing presentation option is used to print print-ready data that does not require any special presentation requirements. In the listing mode, no records or fields need to be defined. When this mode is used, presentation attributes for data on a page such as orientation, margins, grid, font, and form are defined globally for a page using a JDT. In addition, page numbering and automatic zebra striping can also be used.

Pages are delimited in this mode by using a maximum number of LPP (Lines Per Page) or by using an explicit page delimiter such as a form feed. See the “Working in Listing Mode” section later in the chapter for more information.

Record Processing Entry (RPE) Mode

The Record Processing Entry (RPE) presentation option allows each record to be split into fields that can be printed any number of times and at any location on the page. Records and fields can contain specific presentation attributes such as position on the page, line spacing, font, color, alignment, and rotation. These presentation options are part of the JDT. You can view the JDT VIPP commands in the design area **Source View** tab. The RPE fields are defined vertically as well as horizontally.

Defining RPE Fields Vertically

The RPE fields are defined vertically using prefixes or line numbers. Line mode data files can either be prefixed or non-prefixed. For non-prefixed data files, Vitesse automatically numbers each row. The prefixed data files contain alphanumeric prefixes at the beginning of each record to identify the record. Any line referencing a specific prefix anywhere on the data page is part of the prefixed record.

Prefixes can help identify the data that you want printed with similar presentation attributes such as font, color, alignment, and position etc. For example, in a bank statement, credit lines may be prefixed by LINEC and debit lines by LINED. You can choose to print the credit lines in a different color and the debit lines using a different font.



You need to specify a data file as either prefixed or non-prefixed at the time of project creation. These settings cannot be modified once the project is created. For more information on specifying prefixes, see *Chapter 14: Creating a Line Mode Project*

Defining RPE Fields Horizontally

The RPE fields are defined horizontally using field number or field length. A field contains one piece of information, such as a date or a name. In fixed length field data files, the fields are identified using the starting position and the length of the data field. Fields in a record can also be identified using delimiting characters. Every time a delimiter is encountered in the file, a new field starts, which allows fields to vary in length. We recommend using a delimiting character that is not part of the data.



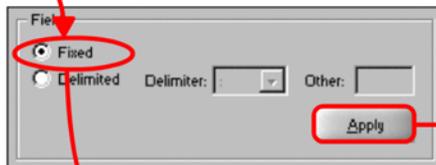
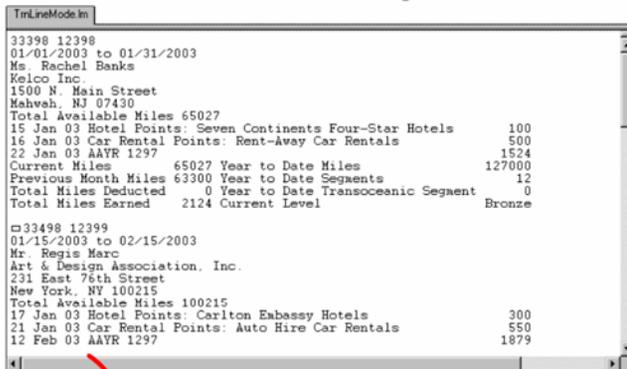
By default, all line mode projects open in the listing mode; however, you can switch to RPE mode by mapping records and fields to the project.

Specifying Fixed Length Field Data

To specify fixed length data fields in the data file:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Field** area, select **Fixed** which indicates that the data file contains fixed length fields.
- 4 Click **Apply**.

The data file uses fixed length fields.



1. Select **Fixed** to specify that the fields are of fixed length.

2. Click **Apply**.

The image shows the process of specifying fixed length field data.

Specifying a Delimiter for Delimited Data

To specify a field delimiter for delimited data:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Field** area, select **Delimited** which indicates that the data file contains variable length fields; this option enables the **Delimiter** drop-down list.
- 4 Select the required delimiter from the **Delimiter** drop-down list.
- 5 To select a delimiting character other than the predefined values, select the **Other** option from the **Delimiter** drop-down list; this option enables the **Other** edit-box.
- 6 Enter the required delimiting character in the **Other** entry box. The delimiter can only be one character in length.
- 7 Click **Apply**.



';' represents the delimiting character.

1. Select **Delimited** to specify that the file uses variable length fields.
2. Select the delimiting character.
3. Click **Apply**.

The image shows the process of specifying a data file delimiter.

Working with Page Delimiters

A variable data document can consist of multiple pages of data. Page breaks are used to indicate the end of a page of data. You can set up page breaks in a line mode project using one of the following options:

- Specifying the number of lines per page (**LPP**) mapped from the line data file to the logical page. In the line mode, reaching the LPP value causes a page break. However, page breaks occurring in the data file take higher precedence and cause a page break to occur even if the specified LPP value has not been reached.
- Using skip to channel one by specifying Printer Control Character (**PCC**) byte or **Skip** options.
- Using explicit page delimiter such as <0C> hex. The page delimiter forces the printer to eject the current page and continue printing data on another page. Page delimiters and PCC cannot be used at the same time. When working with PCC, page delimiters are processed as print data.

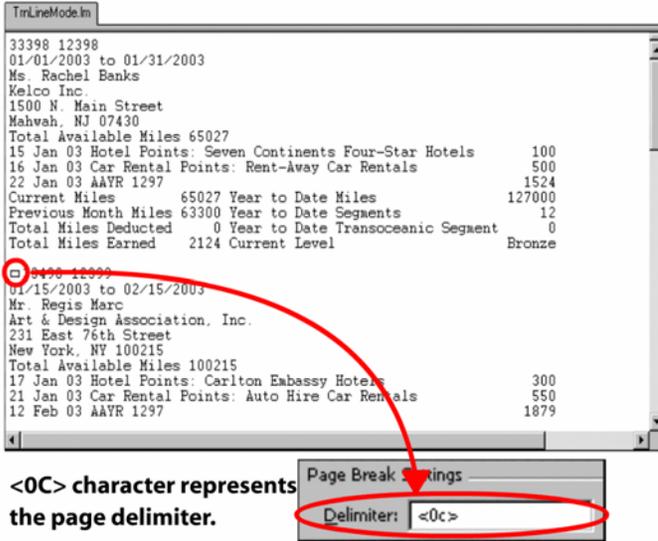
Form feed detection works with any string contained in the data file. However, you should be careful when choosing a character as a page delimiter. For instance, selecting C as a page delimiter will start a new page every time the letter C is encountered. If your customer's name is Chirac Christopher Black you will then have four pages carrying "hira", " ", "hristopherbla", and "k" respectively. We recommend that you use a character that is not part of your actual data content as a delimiter.

Specifying a Page Delimiter

To set a page delimiter:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.

- 3 In the **Page Break Settings** area, specify the required delimiter in the **Delimiter** edit-box.



Specifying the page delimiter. Any character in the file can act as the delimiting character.

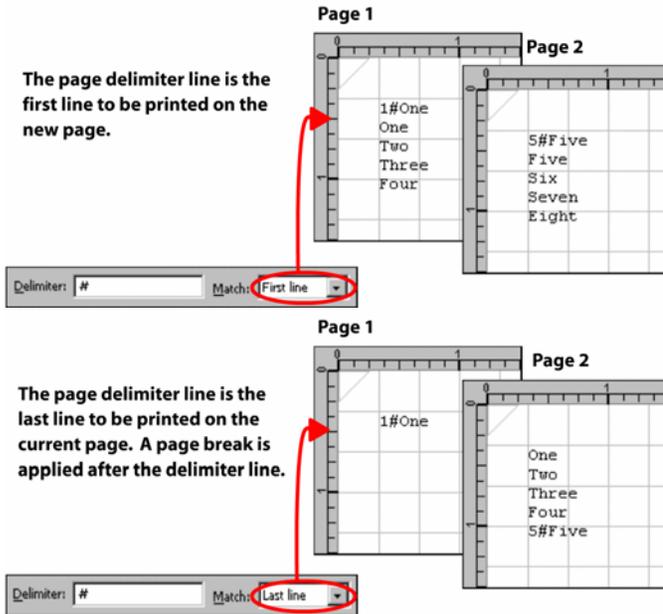
- 4 Click **Apply**.

Matching the Page Delimiter

The data line on which the page delimiter string is detected can be treated as the first line of the new page, or the last line to be printed on the current page.

To choose the page delimiter match option:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Page Break Settings** area, select one of the following options from the **Match** drop-down list:
 - **Last line** – specifies that the data string containing the page delimiter is printed as the last line on the current page.
 - **First Line** – specifies that the data string containing the page delimiter is printed as the first line on the next page.
- 4 Click **Apply**.



Printing page delimiter line as the first or last line of a page. In this example, '#' represents the page delimiter.

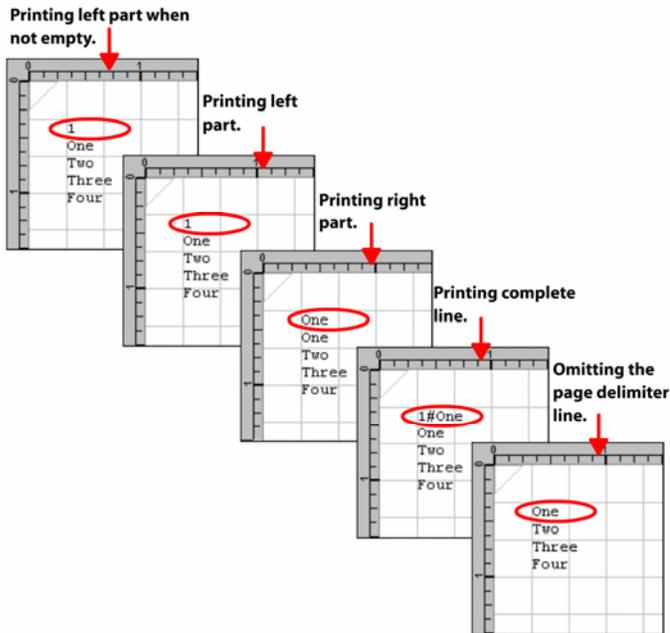
Printing the Page Delimiter Line

Vitesse provides a number of printing options for the data line containing the delimiter. You can choose to print the entire line contents including the delimiter, or print data to the right or left of the delimiting character.

To choose the page delimiter line printing option:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Page Break Settings** area, select one of the following options from the **Print** drop-down list:
 - **Left when not empty** – prints part of the line that is to the left of the page delimiter if not empty.
 - **Left part** – prints part of the line that is to the left of the page delimiter.
 - **Right part** – prints part of the line that is to the right of the page delimiter.
 - **Complete line** – prints the complete line, including the page delimiter.
 - **None** – does not print the line containing the page delimiter.

4 Click **Apply**.



The image shows various printing options for the page delimiter line. In this example, '1#One' is the page delimiter line and '#' represents the delimiter.

Looping the Page Delimiter Line

A single data line in the data file can contain multiple instances of the delimiter string. You can set options to ignore any occurrences of the delimiter string in the same line after the first occurrence. You can also set options to look for all occurrences of the delimiter in a single line, causing a page break after every occurrence.

To loop the page delimiter line:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Page Break Settings** area, select one of the following options from the **Loop** drop-down list:
 - **Right part** – processes all occurrences of the delimiter in the page delimiter line.
 - **None** – processes only the first occurrence of the delimiter in the page delimiter line.
- 4 Click **Apply**.

The page delimiter line '1#One#Two' contains two occurrences of the delimiter '#'.

Looping the page delimiter line, results in two pages, corresponding to the two occurrences of the page delimiter '#'.

Page Break Settings

Delimiter: # Match: First line

Print: Right part Loop: Right part

Page 1

Page 2

Selecting None, causes the second occurrence of the page delimiter string '#' to be ignored, resulting in a single page.

Page Break Settings

Delimiter: # Match: First line

Print: Right part Loop: None

Page 1

A data line can be checked for all possible occurrences of the delimiter string or only the first occurrence.

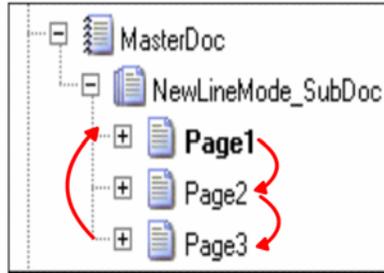
Setting Document Flow Options

When working with a document containing multiple logical pages, you need to specify the sequence in which the logical pages are merged with data.

To set the document data flow:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Document Flow** area, select one of the following document flow options:
 - **Cyclic** – maintains cyclic data flow between the logical pages. After the last logical page is used, the data merging again starts from the first page. For example, for a document containing 3 logical pages and 6 data pages, the following sequence of logical pages is used 1, 2, 3, 1, 2, 3.

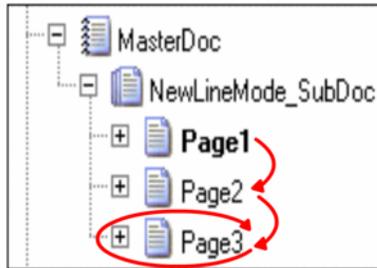
Once the last page has been reached, data again starts printing from the first page and continues the same cycle until all the data is mapped.



The image displays the data placement sequence for cyclic document flow.

- **Two Page** – specifies that only the last logical page is repeated, all the preceding logical pages are used only once. For example, for a document containing 3 logical pages and 6 data pages, the following sequence of logical pages is used 1, 2, 3, 3, 3, 3.

Initial pages are used only once, all the subsequent data pages are placed on the last logical page.



The image displays the data placement sequence for two page document flow.

- 4 Click **Apply**.

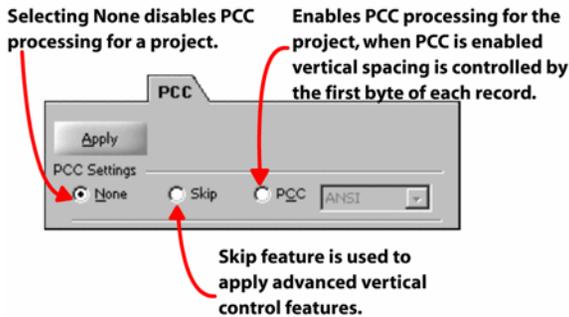
Working with Printer Control Characters

Printer Control Characters (PCC), also called control codes or printer carriage controls are data file characters that instruct the printer to perform a specific action. The first byte of a data line is interpreted as a printer control character by the print device. PCC bytes control the vertical movement or spacing of a record, and are not part of the printed data. Vitesse reads the PCC bytes and interprets them when formatting line mode data.

To enable PCC processing, a PCC table needs to be set up that lists all the PCC bytes and the associated actions. Some PCC tables are globally predefined for example the standard ANSI, and IBM1403 EBCDIC PCC definitions etc. Vitesse supports the standard ANSI PCC table. You can also create a custom PCC table, having your own definitions; for example, you can set the 'space' character to print the remainder of a record to a new line, the 'plus' character tells the printer to print the record over the old line, and a 'one' tells the printer to begin a new page and so on.

Using Vitesse, you can either:

- Disable PCC processing for a project.
- Control vertical spacing by enabling the PCC processing.
- Apply advanced vertical control using the skip feature.



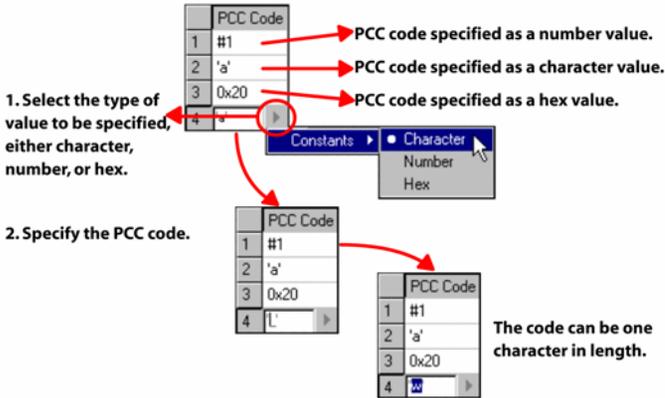
The PCC options available in a line mode project, SubDoc node.

Creating a Custom PCC Table

To create a custom PCC table:

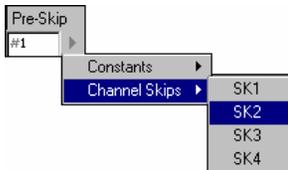
- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **PCC** tab.
- 3 From the **PCC Settings** area, select the **PCC** option to enable PCC processing for the project. When PCC is enabled, vertical spacing is controlled by the first byte of each record whose action is defined in the PCC table. Selecting this option enables the associated drop-down list and displays the associated tabs.
- 4 Select **Custom** from the drop-down list to create a custom PCC table. An empty PCC table is displayed in the **SETPCC** tab.
- 5 Click  to add an entry to the PCC table.

- 6 In the newly added entry, double-click the cell under the **PCC Code** column to specify the PCC code. The PCC code corresponds to the first byte of each record and can be specified as a character, number, or hex value. Select the type of value to be specified by clicking  in the cell and selecting the appropriate option from the sub menu. The character and number values are translated to their equivalent hex values for PCC processing.



The image shows the process of specifying a PCC Code.

- 7 Double-click the cell under the **Pre-Skip** column to specify the pre-skip value. Click  in the cell and select the appropriate option from the sub menu. The pre-skip value can be one of the following:
- **Constants** – skips the specified number of lines before the current print position, before printing the record containing the PCC code. You can only specify a positive numeric value. Specifying a pre-skip value of 0 causes the record to be printed over the last record, whereas 1 causes the record to be printed on the next line. Any other value greater than 1, for example 3 will cause 2 empty lines to be printed before the specific record.
 - **Channel Skips** – select a Vertical Format Unit (VFU) key from the available options. The VFU keys are defined in a VFU table. Each VFU key corresponds to a specific line number on which the record is placed.

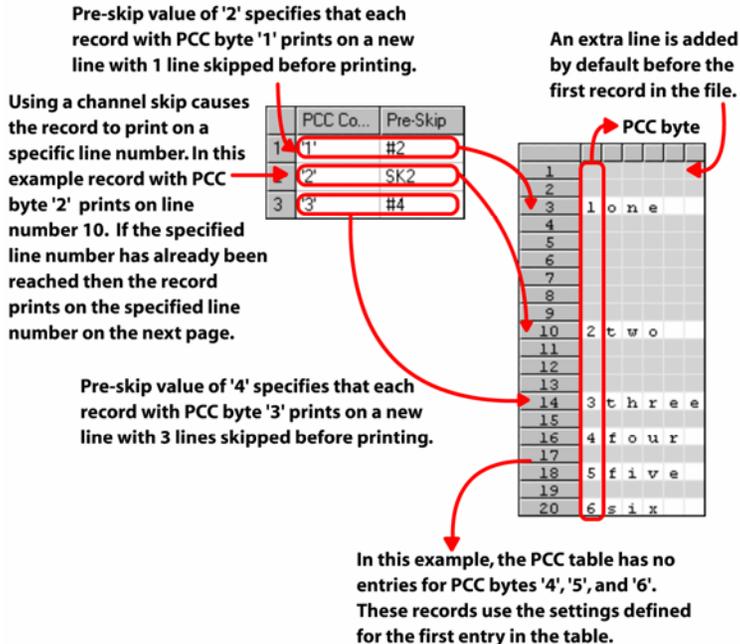


Selecting a channel skip from the available options.



For more information on defining channel skips, refer to “Setting up a VFU Key Table” later in the chapter.

If a PCC code in the project data file has no corresponding entry in the custom PCC table, then the settings defined for the first entry in the table are treated as defaults and are applied to the records.



The image shows the effect of different pre-skip values on the vertical spacing of the records.

- 8 Select the check-box in the **Print** column to print the matching record. Deselecting this option does not print the line contents, but the spacing values for the record are applied.

In this example, record with PCC byte '2' is not printed, but the corresponding spacing options are applied as displayed in the design area.

The records with the Print option disabled are highlighted in the Data View window.

PCC Co...	Pre-Skip	Print	Post-S
1 '1'	#2	<input checked="" type="checkbox"/>	#0
2 '2'	SK.2	<input type="checkbox"/>	#0
3 '3'	#4	<input checked="" type="checkbox"/>	#0

1				
2				
3	1	o	n	e
4				
5				
6				
7				
8				
9				
10	2	t	w	o
11				
12				
13				
14	3	t	h	r
15				
16	4	f	o	u
17				
18	5	f	i	v
19				
20	6	s	i	x

The image shows the effect of disabling the Print option.

- 9 Double-click the cell under the **Post-Skip** column to specify the post-skip value. Click  in the cell and select the appropriate option from the sub menu. The post-skip value can be one of the following:
 - **Constants** – skips the specified number of lines after printing the record containing the corresponding PCC code. You can only specify a positive numeric value. Specifying a post-skip value of 0 causes the next record to be printed on the next line, whereas any other value greater than 0, for example 3 will cause 3 empty lines to be printed after the specific record.

- **Channel Skips** – select a Vertical Format Unit (VFU) key from the available options. The VFU keys are defined in a VFU table. Each VFU key corresponds to a specific line number which is set as the current print position. For example, if a record has post-skip value set to key SK2 which corresponds to line number 10, the next record will be positioned relative to line number 10 depending on the record pre-skip value. If the pre-skip value is set to 1, the record is placed at line number 11. If the specified line number has already been reached, then the print position is set to the specified line number on the next page.

Post-skip value of '0' specifies that the next record will be placed on the next line.

	PCC Co...	Pre-Skip	Print	Post-S
1	'1'	#1	✓	#0
2	'2'	#1	✓	SK2
3	'3'	#1	✓	#3

An extra line is added by default before the first record in the file.

1				
2	1	o	n	e
3	2	t	w	o
4				
5				
6				
7				
8				
9				
10				
11	3	t	h	r
12				
13				
14				
15	4	f	o	u
16	5	f	i	v
17	6	s	i	x

In this example, record with PCC byte '2' has the post-skip value set to channel skip SK2 which corresponds to line 10; the next record is printed on line 11.

Post-skip value of '3' specifies that 3 lines will be skipped before printing the next record.

In this example, the PCC table has no entries for PCC bytes '4', '5', and '6'. These records use the settings defined for the first entry in the table.

The image shows the effect of different post-skip values on the vertical spacing of the records.

10 Click Apply.

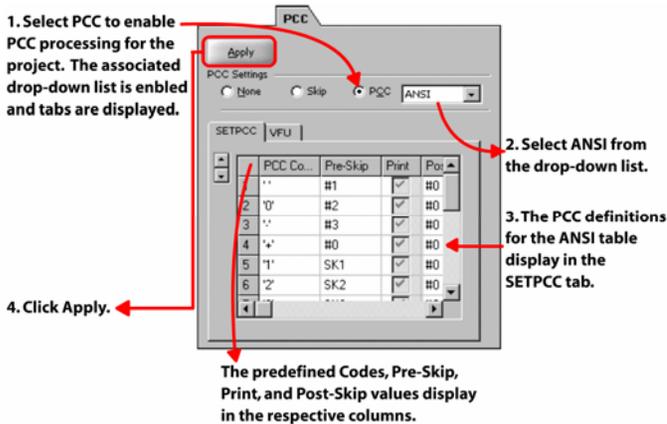


You can also delete the last added entry by clicking  in the SETPCC tab within the PCC tab, and then clicking **Apply**.

Selecting a Predefined PCC Table

To select a predefined PCC table:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **PCC** tab.
- 3 From the **PCC Settings** area, select the **PCC** option to enable PCC processing for the project. Selecting this option enables the associated drop-down list and displays the associated tabs.
- 4 Select **ANSI** from the drop-down list to use the standard ANSI (American National Standards Institute) PCC table. The ANSI PCC definitions are displayed in the **SETPCC** tab. You cannot edit a predefined PCC table.



The image shows the process of selecting a globally predefined PCC table.

- 5 Click **Apply**.

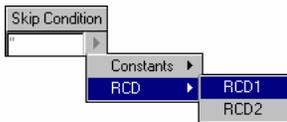
Using Advanced Vertical Control Options

Printer Control Characters (PCC) provide vertical control using the first byte of each record. You can apply advanced vertical control to your line mode project using the **Skip** feature. Skip and PCC cannot be used at the same time.

Using skip, you can specify a string of any length. If the beginning of a record matches the string, the associated skip action occurs. You can also define a condition. If the condition is true for the record, then the associated skip action occurs. Skip enables you to define a bottom and top printing threshold for the page, in addition to pre-skip, print, and post-skip options to control vertical spacing.

To set up the advanced vertical control options:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **PCC** tab.
- 3 From the **PCC Settings** area, select the **Skip** option to enable advanced vertical spacing for the project; selecting this option displays the associated tabs.
- 4 Select the **SETSKIP** tab within the **PCC** tab.
- 5 Click  to add an entry to the skip table.
- 6 In the newly added entry, double-click the cell under the **Skip Condition** column to specify the testing condition. Click  in the cell and select the appropriate option from the pop-up menu. The skip condition can be set to one of the following:
 - **Constants** – specifies a string of any length. If the beginning of the record matches the string then the associated spacing options will be applied. The matching string remains a part of the printable data. The default skip behavior can be defined by specifying an empty string. However, the empty string should be the last entry in the table, as Vitesse ignores any skip definitions after a blank (‘’) entry.
 - **RCD** – select a Record Criteria Definition (RCD) key from the available options. The conditions corresponding to the RCD keys are defined in the **RCD** tab. If the RCD condition is true for the record, the associated skip action occurs.



Selecting a RCD key from the available options.



For more information on creating RCD keys, refer to “Defining RCD Conditions” below.

- 7 Double-click the cell under the **Pre-Skip** column to specify the pre-skip value. Click  in the cell and select the appropriate option from the sub menu. The pre-skip value can be one of the following:
 - **Constants** – skips the specified number of lines before printing the record. You can only specify a positive numeric value. Specifying a pre-skip value of 0 causes the record to be printed over the last record, whereas 1 causes the record to be printed on the next line. Any other value greater than 1, for example 3 will cause 2 empty lines to be printed before the specific record.

- **Channel Skips** – select a Vertical Format Unit (VFU) key from the available options. The VFU keys are defined in a VFU table. Each VFU key corresponds to a specific line number on which the record is placed. For example, if a record has pre-skip value set to key SK2 which corresponds to line number 10, the record will be positioned at line number 10. If the specified line number has already been reached, then the record prints at the specified line number on the next page.

An extra line is added by default before the first record in the file.



For more information on defining channel skips, refer to the “Setting up a VFU Key Table” section later in the chapter.

- 8 Select the check-box in the **Print** column to print the matching record. Deselecting this option does not print the line contents, but the spacing values for the record are applied.
The records with the print option disabled are highlighted in the *Data View* window.
- 9 Double-click the cell under the **Post-Skip** column to specify the post-skip value. Click  in the cell and select the appropriate option from the sub-menu. The post-skip value can be one of the following:
 - **Constants** – skips the specified number of lines after printing the matching record. You can only specify a positive numeric value. Specifying a post-skip value of 0 causes the next record to be printed on the next line, whereas any other value greater than 0, for example 3 will cause 3 empty lines to be printed after the specific record.
 - **Channel Skips** – select a Vertical Format Unit (VFU) key from the available options. The VFU keys are defined in a VFU table. Each VFU key corresponds to a specific line number which is set as the current print position. For example, if a record has post-skip value set to key SK2 which corresponds to line number 10, the next record will be positioned relative to line number 10 depending on the record pre-skip value. If the pre-skip value is set to 1, the record is placed at line number 11. If the specified line number has already been reached, then the print position is set to the specified line number on the next page.

- 10 Double-click the cell under the **BOF** column to specify the bottom threshold value. Click  in the cell to define a positive numeric value.

A skip to the next page occurs before printing the matching record, when the number of remaining lines on the page after applying the pre-skip is less than the BOF value. The number of lines per page is defined by the LPP (Lines Per Page) value.



For more information on setting LPP (Lines Per Page) value, see the “Setting up the Data Grid” section in *Chapter 18: Projecting the Data Layout*.

	Skip Condition	Pre-Skip	Print	Post-S...	BOF	TOF
1	'one'	#60	<input checked="" type="checkbox"/>	#0	#0	#0
2	'two'	#4	<input checked="" type="checkbox"/>	#0	#3	#0
3	'three'	#1	<input checked="" type="checkbox"/>	#0	#0	#0
4	"	#1	<input checked="" type="checkbox"/>	#0	#0	#0

Data page 1.

Data page 2.

Line containing the string 'two' has a pre-skip value of '4', therefore line 62, 63, and 64 are skipped before printing the corresponding record.

Number of lines remaining on the page after applying pre-skip of '4' is 2. This value is less than the BOF value of 3 therefore, a skip to next page occurs and the record is placed at the corresponding line number on the next page.

LPP value is set to 66 lines per page.

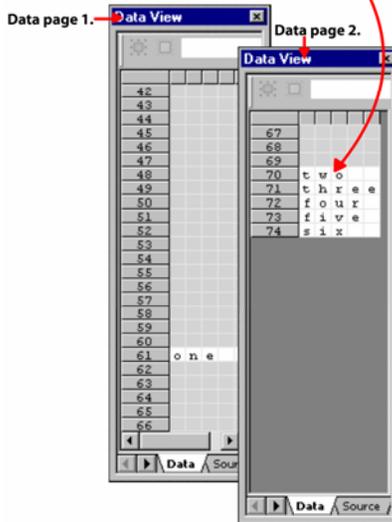
Applying page breaks using the bottom threshold value.

- 11 Double-click the cell under the **TOF** column to specify the top threshold value. Click  in the cell to define a positive numeric value.

This specified value is taken as the target line for printing when a page transition occurs during the pre-skip process. The pre-skip value is ignored in this case. A value of 0 specifies that the pre-skip value is applied, and the TOF value is ignored.

	Skip Condition	Pre-Skip	Print	Post-S...	BDF	TOF
1	'one'	#60	<input checked="" type="checkbox"/>	#0	#0	#0
2	'two'	#4	<input checked="" type="checkbox"/>	#0	#3	#4
3	'three'	#1	<input checked="" type="checkbox"/>	#0	#0	#0
4	"	#1	<input checked="" type="checkbox"/>	#0	#0	#0

TOF value of 4 causes the record with matching string 'two' to be placed at line 4 on the new page. The pre-skip value is ignored.



The image shows the effect of applying TOF value.



Use the horizontal scroll bar to view all the columns in the **SETSKIP** tab.

- 12 Click **Apply**.

If a record in the data file has no corresponding skip entry, then the defaults defined using the empty string entry are applied. If specific default settings are not set using the empty string, then the following default settings are used:

- Pre-Skip: 0
- Print: checked
- Post-Skip: 1
- BOF: 0
- TOF: 0



You can also delete the last added entry by clicking  in the **SETSKIP** tab under the **PCC** tab, and then clicking **Apply**.

Defining RCD Conditions

The advanced vertical options enable you to control vertical spacing of the records using conditional processing on the record's data. If a condition is true for a record, the corresponding spacing options are applied to the record.

To define a RCD (Record Criteria Definition) condition:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **PCC** tab.
- 3 From the **PCC Settings** area, select the **Skip** option to enable advanced vertical spacing for the project; selecting this option displays the associated tabs.
- 4 Select the **RCD** tab in the **PCC** tab.
- 5 Enter the RCD name in the **RCD** drop-down list.
- 6 Click  to add a test condition to the newly created RCD.
- 7 The test condition area contains two lines; the line at the top contains the **IF** keyword; the second line contains the test condition. To set up the test condition, click in the second line below the **IF** keyword; the associated drop-down list displays.
- 8 As appropriate, select the **Not** operator from the drop-down list; this operator reverses the logical value of the evaluated condition (i.e., true if the statement is false, and vice versa).

9 Click in the next area. This area specifies the test data. Click  in the cell and select the appropriate option from the sub menu. The test data can be one of the following:

- **Data Mapping** – sets field data as test data. Select one of the following from the data mapping sub menu:

- **Field(#)** – select this option when working with delimited data files. In the **Field(#)** syntax, the # represents the field number.
- **Field(#-#)** – select this option when working with fixed length data files. In the **Field(#-#)** syntax, the first # represents the field starting position and the second # specifies the field length from the starting position.

- **Variables** – displays a sub menu listing all the variables defined for the project. Select the required variable to be used as test data.
- **System Variables** – displays a sub menu listing all the system variables defined for the application. Select the required system variable to be used as test data.
- **Constants** – sets a constant as text data. You can specify a character, number, or hex value:

- **Number** – sets a number as the test data. Selecting this option displays a # symbol in the edit box. Enter the required numeric value next to the # symbol.
- **Text** – sets a text string as the test data. Selecting this option displays ' symbol in the edit box. Enter the required text value between the ' and ' symbol.
- **Hex** – sets a hex character as the test data. Selecting this option displays a 0x symbol in the edit box. Enter the required hexadecimal value next to the 0x symbol.

10 Click in the next area that displays the test operator (by default the operator is =); the associated drop-down list displays. Select one of the following options from the drop-down list:

- **?=** – sets the contain operator. The contain operator searches for the reference string anywhere in the selected record data.
- **<** – sets the operator to less than.
- **<=** – sets the operator to less than or equal.
- **<>** – sets the operator to not equal.
- **=** – sets the operator to equal.
- **>** – sets the operator to greater than.
- **>=** – sets the operator to greater than or equal.

- 11 Click in the area next to the operator. This area specifies the reference string for the test. Click  in the cell and select the appropriate option from the sub menu. The reference string can be one of the following:
- **Variables** – displays a sub menu listing all of the variables defined for the project. Select the required variable to be used as the reference string.
 - **System Variables** – displays a sub menu listing all of the system variables defined for the application. Select the required system variable to be used as the reference string.
 - **Constants** – sets a constant as the reference string. You can specify a character, number, or hex value:

- **Number** – performs numeric comparisons, such as less than 10, or greater than or equal to 100. Selecting this option displays a # symbol in the edit box. Enter the required numeric value next to the # symbol.
- **Text** – performs string comparisons, which can only be equal or not equal to a given string. Selecting this option displays ' ' symbol in the edit box. Enter the required text value between the ' and ' symbol.
- **Hex** – performs comparisons on a single character in a field. Selecting this option displays a 0x symbol in the edit box. Enter the required hexadecimal value next to the 0x symbol.

- 12 Click **Add**, and then click **Apply**.

You can also combine several test conditions using Boolean operators AND, OR, and NOT. Using more than one test condition enables you to define scenarios where the advanced spacing options are applied if a set of conditions are true, or one of the conditions in a set is satisfied.

To combine several RCD conditions:

- 1 Create a RCD condition, and define the first test condition in the RCD.
- 2 Click  to add another condition test to the RCD.
- 3 The condition test area contains two lines, click in the start of the first line; the associated drop-down list displays.
- 4 As appropriate, select the **Not** operator from the drop-down list; this operator reverses the logical value of the evaluated condition (i.e., true if the statement is false, and vice versa).

- 5 Click in the next area that specifies the boolean operator; the associated drop-down list is displayed. Select one of the following options from the drop-down list:
 - **AND** – specifies that both the test conditions before and after the AND keyword must be true for the condition to be met.
 - **OR** – specifies that either of the test conditions before and after the OR keyword must be true for the condition to be met.
- 6 In the second line, define the condition test using the procedure defined above.
- 7 Click **Add**, and then click **Apply**.

You can combine any number of RCD conditions.



For more information about working with conditions, see *Chapter 27: Adding Conditions*.

1. Select the required option to maintain or reverse the evaluated condition value.

2. Specify the test data; you can use either field data, variables, system variables, or constants.

3. Select the required operator.

4. Specify the reference string for the test.

5. Click to add another test condition.

6. Select the required option to maintain or reverse the evaluated condition value.

7. Specify the boolean operator that will be used to combine the RCD conditions.

The image shows the process of setting up a RCD condition.



You can delete the last added test condition by clicking in the **RCD** tab under the **PCC** tab, and then clicking **Apply**. You cannot remove the first test condition.



You can delete a selected RCD condition by clicking **Del** in the **RCD** tab under the **PCC** tab, and then clicking **Apply**. You cannot remove a RCD condition that is being used in a **Skip** definition.

The following image displays the usage of RCD conditions for applying advanced vertical control options:

SETS SKIP

	Skip Condition	Pre-Skip	Print	Pos
1	'CHER'	#1	<input type="checkbox"/>	#0
2	RCD1	#3	<input checked="" type="checkbox"/>	#0
3	"	#1	<input checked="" type="checkbox"/>	#0

RCD

RCD: RCD1 Add Del

IF
Field(0) = 'ACS3'

In this example, any record meeting the RCD condition has a pre skip value set to 3, that is 2 lines are skipped before printing.

The RCD condition is set to true if the contents of field (0) are ACS3.

	Field(0)	Field(1)	
1			
2	CHER	TITRE	
3	ACS1	Asset Allocation Summary	
4	ADRO		
5			
6			
7	ACS3:BOND, NOTE:::925,64		
8			
9			
10	ACS3:VARIOUS:::25,802.48		
11			
12	ACS3:CASH:::37,566.28:32		
13	ACS3	BOND, NOTE	
14			
15			
16	ACS3:SHARE:::1,222,538.4	ACS3	VARIOUS
17	ACS4:TOTAL:2,211,574.49:	ACS4	TOTAL
18	ADRO:Currency Allocation	ADRO	Currency Allocation Summary
19	ACS2:Market-Value:FX Rat	ACS2	Market Value
20			
21	ACS3:AUSTRALIAN DOLLAR:3	ACS3	AUSTRALIAN DOLLAR
22			
23			
24	ACS3:SWISS FRANCS:118,70	ACS3	SWISS FRANCS
25	PERO:DEUTSCHE MARK:1,297	PERO	DEUTSCHE MARK
26			
27			
28			
29	ACS3:DANISH KRONER:1,032	ACS3	DANISH KRONER
30			
31			

Setting up a VFU Key Table

A Vertical Format Unit (VFU) table defines the channel skip to line number assignments for the current PCC or Skip mapping.

To setup a VFU table:

- 1 Select the **SubDoc** node from the project tree.
- 2 From the *Property View* window, select the **PCC** tab.
- 3 From the **PCC Settings** area, select either **Skip** or **PCC** depending on the type of vertical control options to be used for the project. The associated tabs are displayed.
- 4 Select the **VFU** tab within the **PCC** tab. The VFU table displays the default channel skip to line number mappings.
- 5 Click to add an entry to the VFU table. The name of the VFU key specified under the **Skip Channel** column is automatically generated; the first key in the table is named SK1, the next SK2, and so on. This value cannot be modified.
- 6 In the newly added entry, double-click the cell under the **Skip Value** column to specify the corresponding line number. This value is used in pre-skip and post-skip to skip to a specific line number on the page. A negative line number forces a skip to the specified line on a new page. For example, a skip value of -10 would place the record on line 10 on a new page. The default skip value is 1. If the specified line number has already been reached, then the record prints at the specified line number on the next page.
- 7 Click **Apply**.

1. Select either Skip or PCC options. The associated tabs are displayed.

2. Select the VFU tab in the PCC tab.

3. The VFU table displays the default channel skip to line number mappings.

4. Click **+** to add an entry to the VFU table. The Skip Channel name cannot be modified. The default Skip Value is 1.

5. Double click the cell under the Skip Value column to change the default value.

6. Click Apply.

7. The VFU table is updated and the newly defined Skip Channel can be used for specifying the pre or post-skip values.

	Skip Channel	Skip Value
1	SK1	-1
2	SK2	10
3	SK3	24
4	SK4	32
5	SK5	45

The image shows the process of adding a VFU key to the VFU table.



You can delete the last added VFU key by clicking **▲** in the VFU tab under the PCC tab, and then clicking **Apply**. You cannot remove a VFU key that is being used in the PCC or Skip definition.

Inserting a Data File

Vitesse allows you to use a sample data file for designing the layout of a line mode project. The project specific sample data file is specified at the time of project creation. If a data file is not specified, Vitesse uses a default data file for the project. You can change the selected or default sample data file any time during the project design process by inserting a new data file.

To insert a new data file:

- 1 Select **Insert>Datafile** from the menu bar to display the *Insert Data File* dialog.



You can also open the *Insert Data File* dialog by clicking  on the *Insert* toolbar.

- 2 In the **Existing** tab, browse and select the required data file.
- 3 Click **Open**.

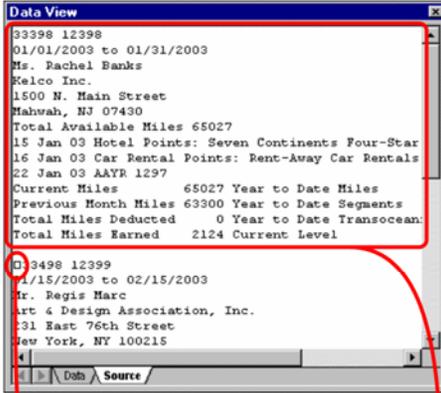
The *Data View* window and any mapped data updates to reflect the changes. The data file path in the **SubDoc** properties is also updated to the inserted sample data file path.

Using the Data View Window

Vitesse supports the use of data files for composing data-driven applications. The data files can be viewed in the *Data View* window so that you can easily drag and drop the required data directly into the design area. This window is context-sensitive and displays data files and associated options relating to specific project modes (i.e., Line, Database, or XML mode).

The *Data View* window is docked at the right side of the workspace by default and contains two tabs:

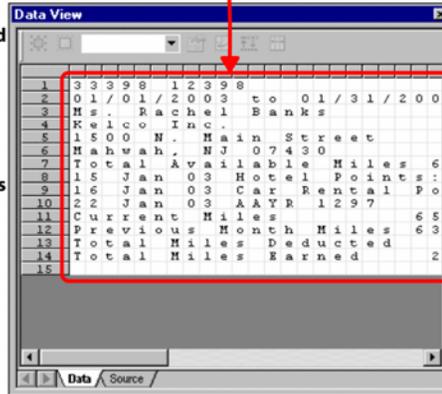
- The **Data** tab shows the data file in a tabular layout for line mode and database mode projects. For an XML mode project, the data is displayed in a tree structure. This tab displays a single data page at a time defined by the page breaks in the data file.
- The **Source** tab displays the actual data file and enables you to edit it. You can view the entire data file in this tab.



The Source tab displays the entire data file.

Data pages in the data file are defined using page breaks.

The Data tab displays one data page at a time, in grid layout.



In this example, the Data tab displays the first 15 lines of the data file that make up the first data page; the Source tab displays the entire data file.

The *Data View* window is not displayed by default when a project is opened. To view the *Data View* window, select Window>Data from the menu bar.



You can also view the *Data View* window either by:

- Clicking  on the status bar.
- Or
- Pressing <Ctrl> + <3> on the keyboard.

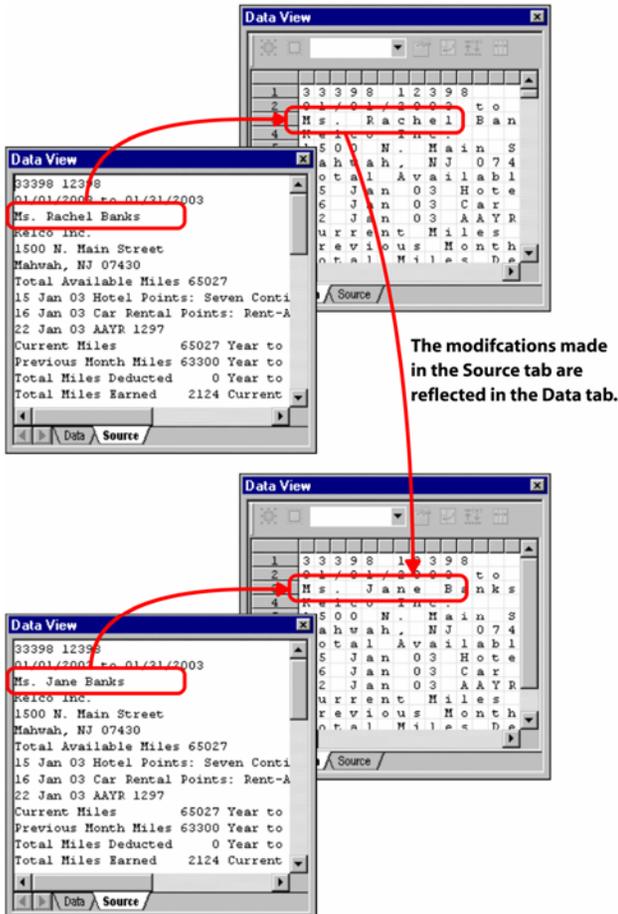
The *Data View* window provides a toolbar in the **Data** tab for performing different operations on the data.

Editing the Data File

You can make changes to the data file in the **Source** tab. Changes made here are saved to the source data file, and automatically updated for the project.

To edit the data file:

- 1 Select the **Source** tab in the *Data View* window.
- 2 Make the required modifications.
- 3 Select the **Data** tab. Vitesse asks for confirmation before applying the modifications.
- 4 Click **Yes** in the confirmation message box. The changes are reflected in the **Data** tab.



Editing the data file using the Source tab.



If you make changes to the data file externally, you will need to reload the data file. You can reload the data file by inserting the updated data file. For more information on inserting data files see “Inserting a Data File” section earlier in the chapter.



Read only data files cannot be edited in the **Source** tab.

Setting Data View Options

You can set the *Data View* window display options by specifying the listing color and the font. These options are specified in the *Preferences* dialog and are applicable globally for all applications.

Selecting Foreground and Background Color for Viewing Data

To set the *Data View* window foreground and background color:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



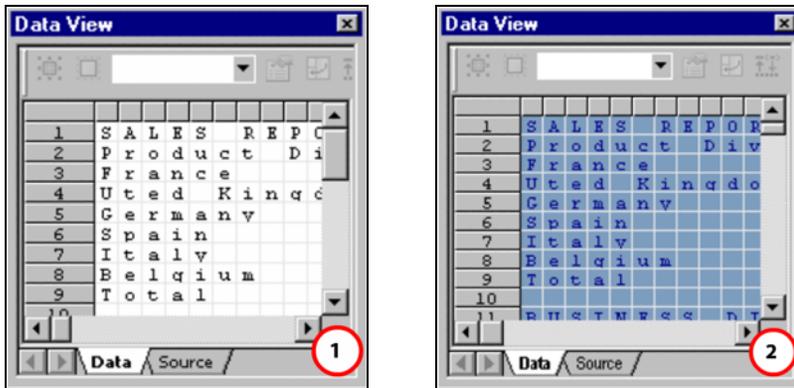
You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <K> from the keyboard.

- 2 From the *Preferences* dialog, select the **Data** category under the **Workspace** node.
- 3 Select the required background color from the **Background Color** drop-down list.
- 4 Select the required foreground color from the **Foreground Color** drop-down list.

- 5 Click **OK**.

The data file updates with the specified background and foreground color.



The *Data View* window displaying data with (1) black foreground color and white background (2) dark blue foreground color and light blue background.

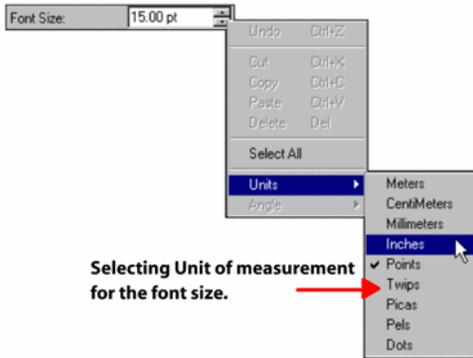
Selecting a Font for Viewing Data

To set the *Data View* window display font:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Data** category under the **Workspace** node.
- 3 Select the required font from the **Font Name** drop-down list.
- 4 Specify the required font size in the **Font Size** entry box.



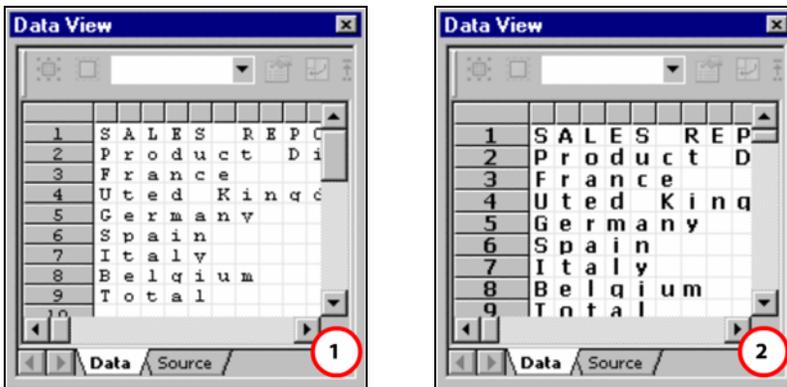
You can also set the unit of measurement for the font by right clicking the upward or downward arrows in the **Font Size** entry box, and then selecting **Units>unit name** from the pop-up menu.



The image shows the process of selecting the unit of measurement.

5 Click OK.

The data file updates with the specified font.



The Data View window displaying data with font settings of (1) Courier New, 14pt (2) Tahoma Bold, 15pt.



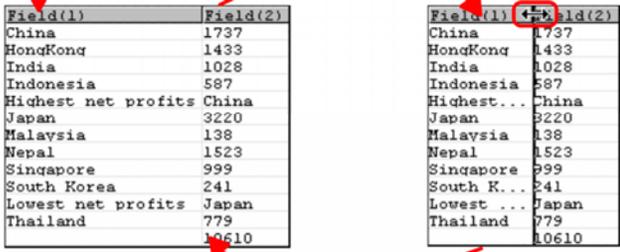
The *Data View* window display options are only applied to the **Data** tab.

Auto-sizing Data Columns

Vitesse automatically adjusts the data column width in the *Data View* window, depending on the maximum length of data in any column. You can also re-adjust the column width manually. The data column width can be adjusted only in the database, XML, and delimited line mode projects.

To set the column width, select the **Data** tab in the *Data View* window, follow the steps in the image below:

1. The data columns are displayed at maximum width by default.
2. Move cursor to the column separator in the column headers, the cursor changes to \leftrightarrow , click and drag the column to the required width.
3. Click  to display the columns at optimum width.



Field(1)	Field(2)
China	1737
HongKong	1433
India	1028
Indonesia	587
Highest net profits	China
Japan	3220
Malaysia	138
Nepal	1523
Singapore	999
South Korea	241
Lowest net profits	Japan
Thailand	779
	10610

Field(1)	Field(2)
China	1737
HongKong	1433
India	1028
Indonesia	587
Highest...	China
Japan	3220
Malaysia	138
Nepal	1523
Singapore	999
South K...	241
Lowest ...	Japan
Thailand	779
	10610

Field(1)	Field(2)
China	1737
HongKong	1433
India	1028
Indonesia	587
Highest net profits	China
Japan	3220
Malaysia	138
Nepal	1523
Singapore	999
South Korea	241
Lowest net profits	Japan
Thailand	779
	10610

The image shows the process of setting the data column size.



You can also auto size the data view by selecting Tools>Data View>Auto size data view from the menu bar.

Using Column View for Delimited Files

When working with delimited data files, you can switch delimited data file layout from column separated to delimited fields view.

To change the column view, follow the steps in the image below:

1. The data is displayed in a column separated view by default.

2. Click  to display the data in delimited fields view. In this view, all occurrences of the delimiter are highlighted.

3. Click  to switch back to the column delimited view.

Field(0)	Field(1)	Field(2)
CAS0	China	1737
CAS1	HongKong	1433
CAS0	India	1028
CAS0	Indonesia	587
CAS2	Highest net profits	China
CAS0	Japan	3220
CAS1	Malaysia	138
CAS0	Nepal	1523
CAS1	Singapore	999
CAS0	South Korea	241
CAS2	Lowest net profits	Japan
CAS0	Thailand	779
TOTO		10610

1	CAS0	China	1737
2	CAS1	HongKong	1433
3	CAS0	India	1028
4	CAS0	Indonesia	587
5	CAS2	Highest net profits	China
6	CAS0	Japan	3220
7	CAS1	Malaysia	138
8	CAS0	Nepal	1523
9	CAS1	Singapore	999
10	CAS0	South Korea	241
11	CAS2	Lowest net profits	Japan
12	CAS0	Thailand	779
13	TOTO		10610

The image shows the process of switching between delimited fields and column view for delimited data files.



You can also set the display options for delimited files by selecting Tools>Data View>Column View from the menu bar.

Changing RPE Prefix

Line mode data files can be either prefixed or non-prefixed. These attributes of the data files can be specified at the time of creation. When working with prefixed data files, you can define or modify the prefix after the data has been opened in the *Data View* window using the **Change RPE Prefix** tool. The tool displays handles for selecting a prefix.

To change a prefix, follow the steps explained in the image below:

1. Click  to display the handles for selecting a prefix.

2. Move cursor to the prefix handle, the cursor changes to , click and drag the handle to the required location.

3. Vitesse asks for confirmation before updating the prefix.

4. Click Yes in the confirmation box.

5. The updated prefix is displayed in the Data View window.

The image shows the process of updating the RPE prefix.



Changing RPE prefixes can cause any existing fields to function improperly.



You can also change the RPE prefix by selecting Tools>Data View>Change RPE Prefix from the menu bar.



You need to switch to the field delimited view to change the RPE prefix for the prefixed, delimited data files. For more information on switching to the field delimited view, see the “Using Column View for Delimited Files” section earlier in the chapter.

Creating a Data Dictionary

You can specify certain *Data View* window contents as a data area. Data areas can be saved in a list called a data dictionary. A data dictionary helps you avoid repeated *Data View* window data selections for the same operation. However, information saved in the data dictionary is not retained after the application is closed. A data dictionary can only be used for non-prefixed data files in line mode projects.

Defining a Data Area

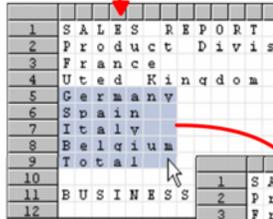
To define a data area:

- 1 Select the **Data** tab in the *Data View* window.
- 2 Enter the name of the data area in the drop-down list provided in the *Data View* window toolbar.
- 3 Click and drag the mouse pointer to define the data area.
The selected area is highlighted in the *Data View* window.
- 4 Click  adjacent to the drop-down list in the *Data View* window toolbar.

1. Enter the data area name in the drop-down list.

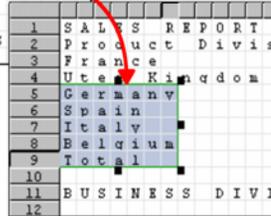


2. Click and drag the cursor to define the data area.



1	S	A	L	E	S	R	E	P	O	R	T												
2	P	r	o	d	u	c	t	D	i	v	i	s											
3	F	r	a	n	c	e																	
4	U	t	e	d	K	i	n	g	d	o	m												
5	G	e	r	m	a	n	y																
6	S	p	a	i	n																		
7	I	t	a	l	y																		
8	B	e	l	g	i	u	m																
9	T	o	t	a	l																		
10																							
11	B	U	S	I	N	E	S	S															
12																							

3. You can use the handles to modify the defined data area.



1	S	A	L	E	S	R	E	P	O	R	T												
2	P	r	o	d	u	c	t	D	i	v	i	s											
3	F	r	a	n	c	e																	
4	U	t	e	d	K	i	n	g	d	o	m												
5	G	e	r	m	a	n	y																
6	S	p	a	i	n																		
7	I	t	a	l	y																		
8	B	e	l	g	i	u	m																
9	T	o	t	a	l																		
10																							
11	B	U	S	I	N	E	S	S															
12																							

4. Click .

The image shows the process of defining a data area.



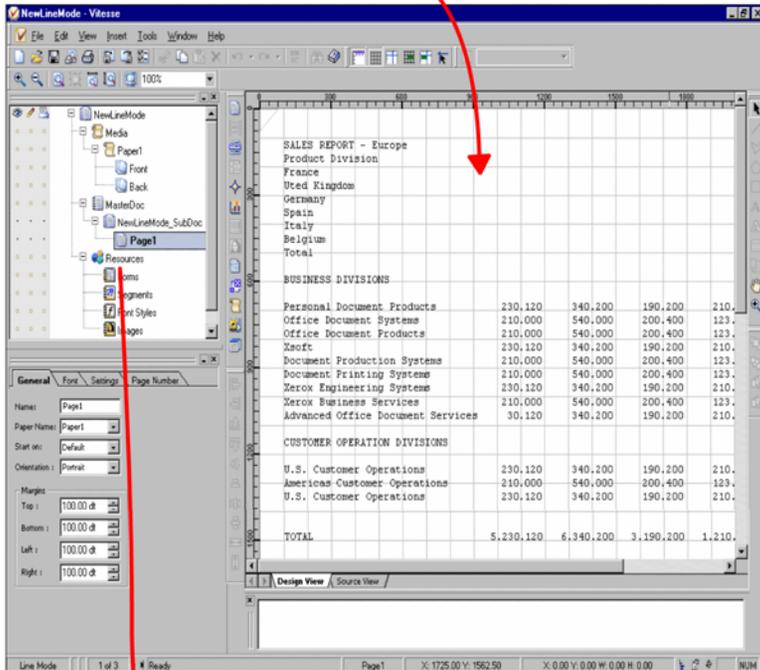
You can also define a data area by specifying the data area name, selecting the data area, and then selecting Tools>Data View>Define Area from the menu bar.



You need to switch to the column view to define a data area for delimited data files. For more information on switching to the column view, see the “Using Column View for Delimited Files” section earlier in the chapter.

In Vitesse, by default, every line mode project opens in the listing mode. In this mode, the data file displays in the design area. Once you add the first record to the project tree window, the data file disappears specifying that the project will use RPE rather than listing mode. In the listing mode, the data pages are printed exactly as they are presented in the data file.

The data file contents are displayed in the design area in the listing mode.



No records and fields are defined for listing mode. You can set global features, such as orientation, margins, grid, font, page numbering, automatic zebra stripping and frame generation using SubDoc and Page properties.

The image displays a line mode project in the listing mode.

Specifying a Listing Color

To specify a listing color:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the required color option from the **Listing color** drop-down list.
- 4 To select a listing color other than the available options, select **Custom** from the **Listing color** drop-down list. The *Color Selector* dialog is displayed. Use this dialog for creating a custom color. For more information on using color, refer to *Chapter 11: Working with Colors*.



The design area reflects the listing color set to (1) black (2) violet.

User Notes:

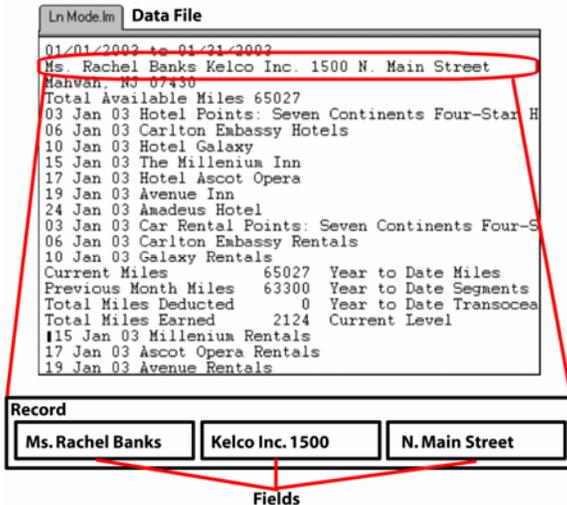
Chapter 22: Working with Data in Line Mode

This chapter describes activities and tasks for working with data in line mode projects. Included are instructions and information for the following:

- Mapping Records
- Inserting a Record in Line Mode
- Mapping Fields
- Inserting a Field
- Using Normal Fields
- Using Image Fields
- Using Segment Fields
- Using Static Text Fields
- Positioning a Field
- Setting Spacing between Fields
- Scaling Image and Segment Fields
- Creating Dynamic Tables Using Field Borders
- Inserting a Data Group in Line Mode

Mapping Data

Data in a line mode project data file is organized as records and fields. Records represent a complete set of information such as the personal information of a customer. Records are composed of fields, which represent a single item of information such as the last name of a customer. Records are defined vertically in a data file using a line number or a prefix. Fields are defined horizontally in the records using starting byte and field length or field delimiting characters. Records can consist of multiple lines of data. If a record contains more than one line of data, then the fields in the record have multiple instances.

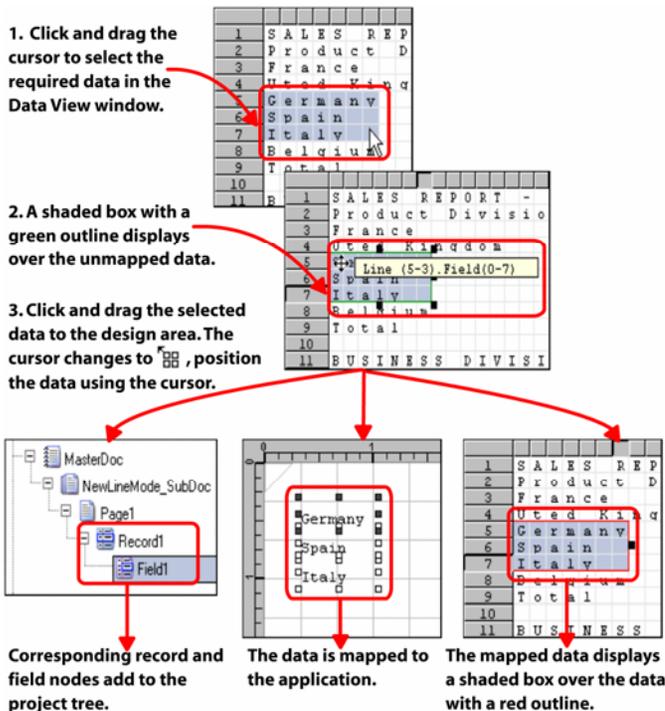


Record and fields in a data file.

Data in a data file can be mapped or placed in a line mode project using any of the following three methods:

- Using the Insert menu.
- Using the tools on the *Insert* toolbar. The toolbar is located vertically between the *Project Tree View* window and the design area.
- Using drag and drop to move/map data from the *Data View* window to the design area.

You can map data directly from the *Data View* window by selecting and dragging the required data to the design area. The data is mapped to your document and the corresponding record and field nodes add to the project tree. In the *Data View* window, you can view the data as either mapped or non-mapped. Mapped data displays with a shaded box and different outline colors depending on the objects to which it is mapped. For example, data initially selected for mapping (not yet mapped) displays with a green outline; data already mapped to records displays with a blue outline; data mapped to fields displays with a red outline; data mapped to charts displays with a gray outline; and data mapped to conditions displays with a yellow outline. Data can be subsequently resized using box handles. You can move the cursor over the selected data to view rollover help; the rollover help displays the data line and field number information.



The image shows the process of mapping data using the drag and drop technique.

When working with prefixed data, clicking any prefixed data content in the *Data View* window automatically selects all data that has the same prefix, as depicted in the image below:

Field(0)	Field(1)	Field(2)	Field(3)	Field(4)
CAS0	China	737	15.9	
CAS1	HongKong	1433	13.0	Corporate Branch
CAS0	India	1028	9.4	
CAS0	Indonesia	587	5.3	
CAS2	Highest net profits	China	525	
CAS0	Japan	220	29.5	
CAS1	Malaysia	138	1.2	Corporate Branch
CAS0	Nepal	523	13.9	
CAS1	Singapore	999	9.1	Corporate Branch
CAS0	South Korea	41	2.2	
CAS2	Lowest net profits	Japan	37	
CAS0	Thailand	79	7.1	
TOTO		10610	100	

In this example, all instances of *Field(1)* for record *CAS0* are selected.

When dragging and dropping data from a prefixed data file, fields and records are automatically added under data groups.



For more information on data groups, see the “What is a Data Group?” section later in the chapter.

Understanding Data Processing Order

In line mode projects, the processing and printing order of data is not dependent on the record placement order in the project tree, but on the order of the data in the data file. However, the objects placed under a **Record** node in the *Project Tree View* window are processed sequentially from top to bottom.

When processing a line data file, the first line encountered in the data file is processed and printed first. When working with prefixed line mode data, the data belonging to the first prefixed record in the data file is processed and printed first. Also, the fields belonging to records from the same prefix family are processed relatively.

Mapping Records

You can map a Line mode project record from a prefixed or non-prefixed data file, and set or change the record properties. For line mode projects, Vitesse accommodates the following record formats:

- Non-prefixed records, for which each record is defined vertically by an exact start and end line number.
- Prefixed records, for which each record is defined vertically using alphanumeric prefixes such as ADR0, ADR1, LFA0, LFA1 and other prefixes specified in the data file. Records having prefixes such as ADR0 and ADR1 belong to the same prefix family ADR. Clicking any prefixed record in the data file will automatically select all fields with identical prefixes.

In line mode projects, records identify the vertical limits in which data can be selected. When you drag and drop data from the data file to the design area, corresponding **Record** and **Field** nodes are added to the project tree. Selecting a required **Record** node in the project tree displays the upper and lower record limits in blue in the data file. The shaded area identifies the number of lines for the selected record that can be incremented or decremented vertically using vertical controls along the boundary. You can select single or multiple fields within the defined record limits from the data file for mapping. You can also set or change the record limits using the record property sheet (see the “Setting Record Properties in Line Mode” section for more information). Mapped records are not printable entities; only fields within the records are printed.

You can add field, condition, chart, and action objects to a record object. These objects are processed sequentially from top to bottom, according to the object placement in the project tree. Barcode and image objects cannot be directly used in a record object. These objects can be added to a record using inline segments or image fields (when the **Field Type** is set to **Image** using the field properties). Page level conditions can also be used for adding barcode and image objects to a line mode project. Form and segment resources are directly added to the page.

Inserting a Record in Line Mode

To insert a record in a line mode project:

- 1 Select the required **Page** node from the project tree.
- 2 Select Insert>Record from the menu bar; a **Record** node is created under the selected **Page** node in the project tree.

You can also add more record objects under a selected record using the same procedure.



You can also insert a record object by clicking  on the *Insert* toolbar.



You can delete a selected record object either by:

- Selecting Edit>Delete from the menu bar.

Or

- Clicking  on the *Standard* toolbar.

Or

- Pressing on the keyboard.

All of the sub nodes under the record node are also deleted.

Setting Record Properties in Line Mode

Different record properties need to be defined depending on whether a prefixed or non-prefixed data file is being used for the project. Record properties include field spacing options and record definition using RPE prefixes for prefixed data or starting line and repeating lines for non-prefixed data.

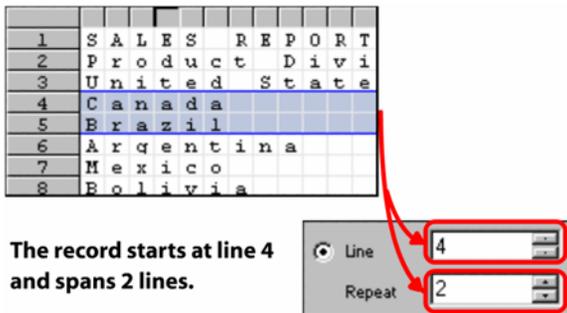
Specifying a Starting Line and the Number of Repeating Lines for Non-Prefixed Data

For non-prefixed data files, Vitesse automatically numbers each row in the data file. You can define a record by specifying the line number on which a record starts. For multi line records, you can also specify the number of repeating lines to specify the record limit.

To set the record starting line and number of repeating lines:

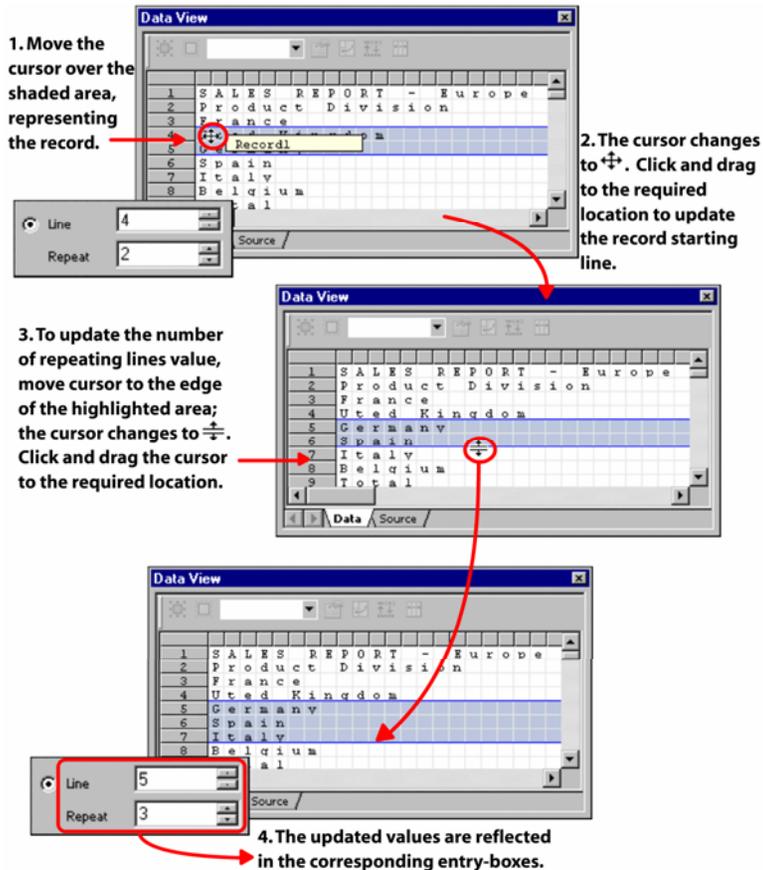
- 1 Select the required **Record** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 Select the **Line** option; this option enables the **Line** and **Repeat** entry boxes.
- 4 Specify the record starting line number in the **Line** entry box.
- 5 Specify the number of repeating lines within the record in the **Repeat** entry box. The repeat value for single line records is 1.

As you specify the start line number and number of lines within the record, the design area and the *Data View* window reflects the changes accordingly.



The image shows the process of defining a record for non-prefixed data.

You can also modify the record starting line and the number of repeating lines by moving and resizing the shaded box representing the record data in the *Data View* window. Changes that you make in the *Data View* window are reflected in the property sheet.



The image shows the process of updating record starting line and the number of repeating lines value using the Data View window.



Each record can have multiple lines. Consequently, each field within a record can have multiple lines (also known as field instances).

In Line mode projects, data (fields) are positioned relatively in either fixed or variable line spacing mode. Differences between fixed and variable spacing are explained as follows:

Fixed Line Spacing Mode

In fixed line spacing mode, you can specify the **X** and **Y** coordinate positions and **Horizontal** and **Vertical** spacing of the field object using the associated property sheet. The **X** and **Y** coordinate relative position specified for the first field inserted in the project tree and design area should be set to **Absolute**. Once **X** and **Y** coordinate positions for the first field instance are set, the spacing values will be applied to all instances of the same field. Spacing between each field instance is fixed and will not affect spacing of subsequent fields mapped for the same record. Subsequent fields inserted in the project tree and design area can be relatively positioned with respect to a previous field using the **X** and **Y** relative position options in the field property sheet. Fixed line spacing mode is recommended for non-prefixed data files.

Variable Line Spacing Mode

In variable line spacing mode, you can specify the **X** and **Y** coordinate positions and **Horizontal** and **Vertical** spacing of the field object using the associated property sheet. The **X** coordinate relative position specified for the first field inserted in the project tree and design area should be set to **Absolute**. However, as the **Y** coordinate relative position can be changed for each field instance, using the **Absolute** option for the **Y** coordinate is not recommended, as it will cause field instances to overlap. Spacing between each field instance is variable and affects the spacing of subsequent fields mapped for the same record. Fields subsequently inserted in the project tree and design area can be relatively positioned with respect to a previous field using the **X** and **Y** relative position options in the field property sheet. Variable spacing, thus, adjusts the spacing of all fields under the same record. Adding additional fields adds the spacing value of the previous field to the spacing value of the new field.

If you use data groups, you can also relatively position the first field of one record to the last field of the previous record. Variable line spacing mode is recommended for prefixed data files.



See the “Mapping Fields” section and related tasks to understand the process of adding fields using the above mentioned spacing modes.

To choose a spacing mode:

- 1 Select the required **Record** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Variable Spacing** area, select one of the following options:
 - **Fixed** – aligns the field according to the specified position and fixed horizontal and vertical spacing. Select this option when you are working with non-prefixed data.
 - **Variable** – aligns the field according to the specified position and variable vertical spacing. Select this option when you are working with prefixed data..



For more information on variable spacing, see the “Setting Spacing between Fields” section later in the chapter.

Mapping Fields

A record object is made up of fields. In a line mode project, a field cannot be directly mapped to a page; they can only be added under records. Each field can contain multiple lines or field instances. Different field properties need to be setup depending on whether prefixed or non-prefixed data is being used for the project. The field data can also be used for creating charts. You can also set the field color, font, alignment, rotation, spacing, and position using the *Property View* window. Vitesse also enables you to insert normal, image, segment, and static text fields to a project. Tasks in this section detail the steps required to insert a field, use different field types, and set associated properties.

Inserting a Field

To insert a field in a line mode project (with non-prefixed or prefixed data files):

- 1 Select the required **Record** node from the project tree.
- 2 Select Insert>Field from the menu bar; a **Field** node is created under the selected **Record** node in the project tree.

You can also add more field objects under a selected field using the same procedure.



You can also insert a field object by clicking  on the *Insert* toolbar.



You can delete a selected field object either by:

- Selecting **Edit>Delete** from the menu bar.

Or

- Clicking  on the *Standard* toolbar.

Or

- Pressing **** on the keyboard.

Using Normal Fields

Normal fields are the most commonly used field type. These fields can contain static text, data from variables, or from the data file.

To set the field type as normal:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab..
- 3 Select **Normal** from the **Field Type** drop-down list.

Specifying the Normal Field Data

The data for the normal fields can be picked up from the data file or from a variable. Normal field data can also be specified using static text. This data can then be mapped directly to a page or used as a label or value for a data driven chart. Methods for using the data file, variables, and static text as normal field data are explained below.

Using the Data File

Vitesse supports the use of both fixed length (non-delimited) and variable length (delimited) data files for a non-prefixed or prefixed line mode project. Different procedures are used for specifying the field data from fixed length and delimited files. When working with fixed length data, a field is defined by specifying the field starting position and length. With delimited data, you need to specify the field number.

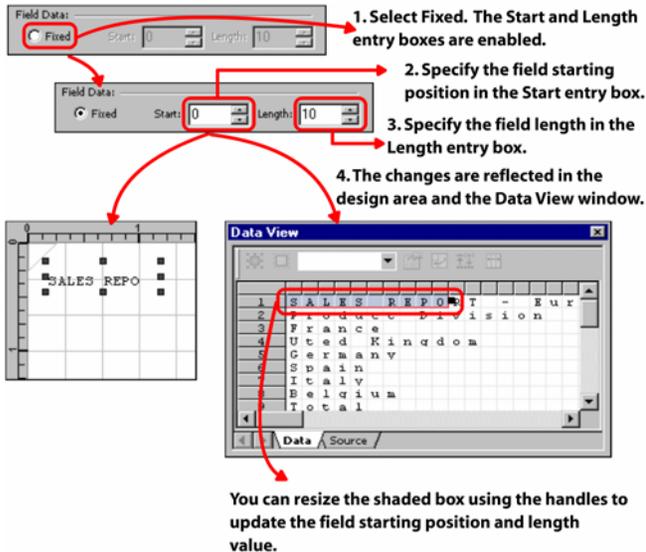
Using Fixed Field Data

To specify field data for fixed length (non-delimited) fields from a prefixed or non-prefixed data file:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Fixed**, which indicates that a fixed field data file is being used for the project. Selecting this option enables the **Start** and **Length** entry boxes:
 - **Start** – specifies the field starting position.
 - **Length** – specifies the field length from the starting position.

The field updates in the design area and data file to reflect the changes.

You can also modify the field starting position and length by resizing the shaded box representing the field data in the *Data View* window.



The image shows the process of specifying fixed field data.

Using the Field Number

To specify field data for variable length (delimited) fields from a prefixed or non-prefixed data file:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Data** which indicates that a delimited field data file is being used for the project. Selecting this option enables the **Field No** entry box.
- 4 Specify the field number in the **Field No** entry box.

The field updates in the design area and data file to reflect the changes.

Using Variables

To specify the field data using predefined variables in a prefixed or non-prefixed line mode project:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable** which indicates that a variable will be used to specify the field data. Selecting this option enables the associated drop-down list.
- 4 Select the required system or user-defined variable from the drop-down list.

The field updates in the design area to display the variable value.



For more information on working with variables, see *Chapter 26: Using Variables*.

You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge data** option under the **Data Merging** area to view the specified variable value in the design area, as shown in the image below:

1. Create a variable using the MasterDoc property sheet.
2. Add a Record object to the page, by clicking  on the Insert toolbar.
3. Add a Field object to the record, by clicking  on the Insert toolbar.

4. In the Settings tab of the field property sheet, select the Const option.
5. Specify the required variable name using the correct syntax.
6. The syntax displays in the design area.

7. In the Settings tab of the field property sheet, select the Merge data option.

8. The variable value displays in the design area.



This process is also applicable for segment and image type fields explained in later sections.

Using Constant Data

You can insert static text as field contents. To specify the field contents using static text in a prefixed or non-prefixed line mode project:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const** which indicates that constant data will be used as field contents. Selecting this option enables the associated entry box.
- 4 Enter the required text in the **Const** entry-box.

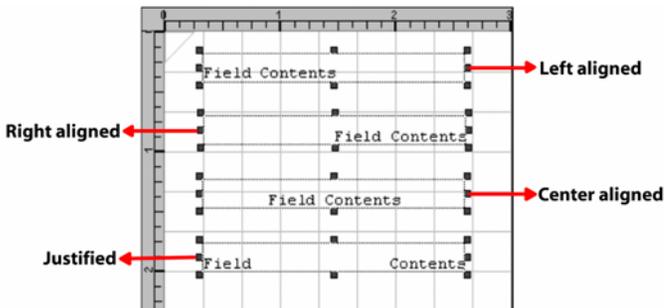
The field updates in the design area to display the text.

Aligning the Normal Field Contents

Alignment is the horizontal positioning of text within a line, relative to a point of reference. Vitesse provides various alignment options for field contents to achieve a more visually balanced design.

To align the field contents:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Left** – aligns the field contents along the left field margin.
 - **Right** – aligns the field contents along the right field margin.
 - **Center** – aligns the field contents symmetrically in the middle of the field.
 - **Justify** – aligns the field contents along both the left and right field margins by adjusting the spacing between the field data.



The design area reflects the effect of the basic alignment options.

- **String** – aligns the field contents relative to a specified character or string of characters occurring in the field; selecting this option enables the **Pos** and **Char** entry boxes.

2. The Pos and Char boxes are enabled.

1. To align field contents relative to a specific character or string, select String.

3. Specify the position of the reference point, in percentage from the left field margin.

4. Specify a single character or a string of characters that will act as the reference point.

The first occurrence of the matching string or character is placed at the specified position. In this example, the first occurrence of the character 'C' is placed at 75% of field width.

The image shows the process of aligning the field contents relative to a specific string.

If there are no occurrences of the specified string in the field data, then the last character in the data (end of the string) is placed at the specified position. The **String** alignment is case sensitive, that is, if you specify 'c' instead of 'C' in the above example, the string alignment will not be effective.

- **Left Trimmed** – aligns the field contents along the left field margin. Any leading spaces are ignored.



The alignment set to (1) Left; any leading spaces are considered (2) Left Trimmed; any leading spaces are trimmed.

- **Right Trimmed** – aligns the field contents along the right field margin. Any trailing spaces are ignored.



The alignment set to (1) Right; any trailing spaces are considered (2) Right Trimmed; any trailing spaces are trimmed.

Creating Dynamic Tables using Field Borders

You can customize the appearance of a normal or static text field by applying borders, fill colors, and patterns using the options in the **Cell** tab. When working with prefixed data, applying cell options to the transactional records creates a dynamic table (applicable for prefixed data only). A dynamic table expands or contracts with the amount of data added.

	Country	Assets (b)	Share of Total
1	China	\$1,737	15.90%
2	HongKong	\$1,433	13.00%
3	India	\$1,026	9.40%
4	Indonesia	\$587	5.30%
5	Japan	\$3,220	29.50%
6	Malaysia	\$138	1.20%
7	Nepal	\$1,523	13.90%
8	Singapore	\$999	9.10%
9	South Korea	\$241	2.20%
10	Thailand	\$779	7.10%
	Total	\$10,610	100.00%

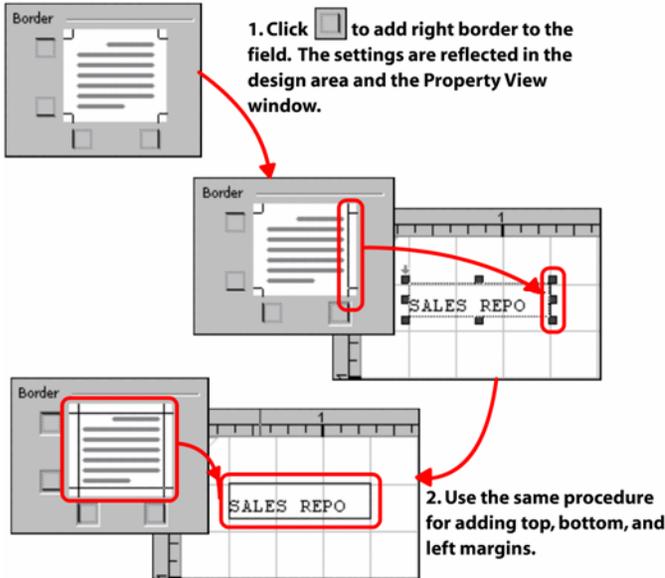
An example of a dynamic table crated using field cell properties.

To create a dynamic table and set associated properties, see the tasks below:

Specifying Cell Margins

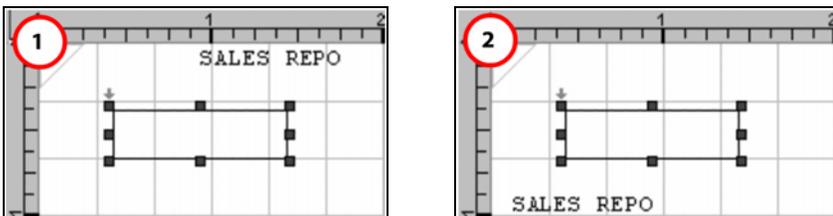
To set the cell margins for creating a dynamic table:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Cell** tab.
- 3 Select the **Cell** option to enable all of the related options.
- 4 In the **Border** area, click the respective icons to apply the top, bottom, left, and right borders.



The image shows the process of setting field margins.

- 5 Specify the field data offset from the cell margin in the **Cell Margin** entry box. You can specify both positive and negative offset values. The direction in which the offset is applied depends on the field justification and alignment.



The field data offset set to 0.5 inches with justification set to (1) right, top (2) left, bottom.



For more information on setting field justification, see the “Setting Field Justification” section later in the chapter.

Specifying Cell Margin Line Width

Vitesse provides various options to change the margin line width.

To set the margin line width:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Cell** tab.
- 3 Select the **Cell** option to enable all of the related options.
- 4 Apply borders to the field.



For more information on applying borders, see the “Specifying Cell Margins” section earlier in the chapter.

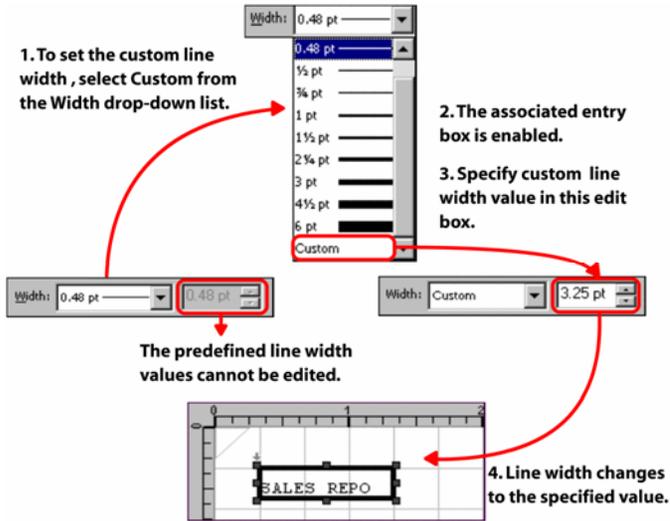
- 5 From the **Line** area, select the required width option from the **Width** drop-down list. The adjacent entry box displays the width value associated with the selected option.
- 6 To set a width other than the predefined values, select the **Custom** option from the **Width** drop-down list. This option enables the adjacent entry box.



You can also set the unit of measurement for the width by right clicking the upward or downward arrows in the entry box, and then selecting **Units**> *unit name* from the pop-up menu.

- 7 Enter the required margin line width value; width of the selected field object is modified.

The following image shows the process of setting the custom margin line width for a field object:



Specifying Cell Margin Line Style

Along with changing the margin line width, you can also apply different line styles to the field margin. For example, you can make the margin appear as dotted or dashed etc.

To set the cell margin line style:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Cell** tab.
- 3 Select the **Cell** option to enable all of the related options.
- 4 Apply borders to the field.



For more information on applying borders, see the “Specifying Cell Margins” section earlier in the chapter.

- 5 From the **Line** area, select one of the following style option from the **Style** drop-down list:

Line Style	Sample
Solid	
Dashed	

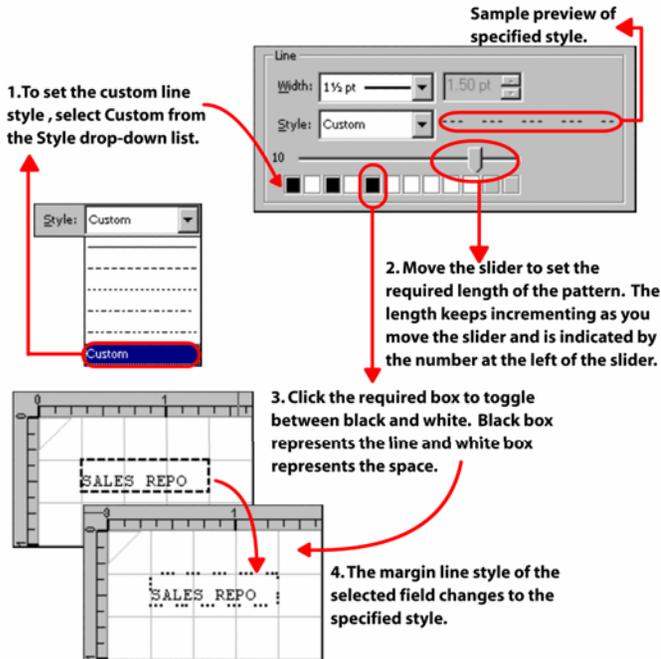
Line Style	Sample
Dotted	
Dash-Dotted	
Dash-Dot-Dotted	
Custom	Sets a customized style for a line. For example, 

The area adjacent to the **Style** drop-down list shows the sample preview of the selected style.

The black and white boxes  in the **Line** area depict the selected line style pattern. The box in black represents the line and the box in white represents the line spacing used in the line style. The pattern keeps changing, depending on the **Style** selection. For example, if you select the dash-dotted line style, the pattern changes to . The first box is always black.

- To set a style other than the predefined values, select the **Custom** option from the **Style** drop-down list.

- 7 Move the slider to set the custom line style pattern length, and then click the required box to toggle from black to white. See image below showing how to set the custom line style for the cell margin:



Specifying Outline Color and Pattern for Cell Margins

To specify the cell margin outline color and pattern options:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Cell** tab.
- 3 Select the **Cell** option to enable all of the related options.
- 4 Apply borders to the field.



For more information on applying borders, see the “Specifying Cell Margins” section earlier in the chapter.

- From the **Line** area, select the margin line color from the **Color** drop-down list.



Selecting **Custom** from the **Color** drop-down list displays the *Color Selector* dialog for creating a custom color. For more information, refer to *Chapter 11: Working with Colors*.

- Select the margin outline pattern from the **Pattern** drop-down list. The pattern is applied to the line in the selected color.



The margin line color and pattern set to (1) none, none (2) red, line pattern.

Specifying Fill Color and Pattern for Cell Margins

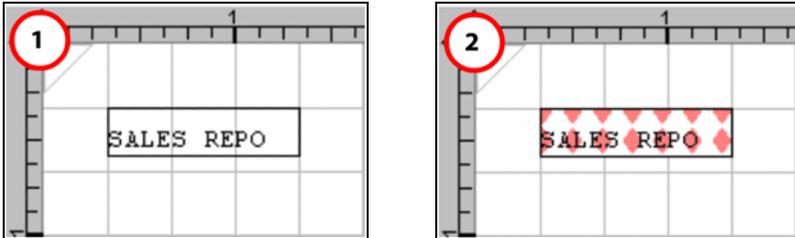
To specify the cell fill color and pattern options:

- Select the required **Field** node from the project tree.
- From the *Property View* window, select the **Cell** tab.
- Select the **Cell** option to enable all of the related options.
- Apply borders to the field.



For more information on applying borders, see the “Specifying Cell Margins” section earlier in the chapter.

- 5 From the **Fill** area, select the cell fill color from the **Color** drop-down list.
- 6 Select the cell fill pattern from the **Pattern** drop-down list. The pattern is applied to the cell in the selected color.



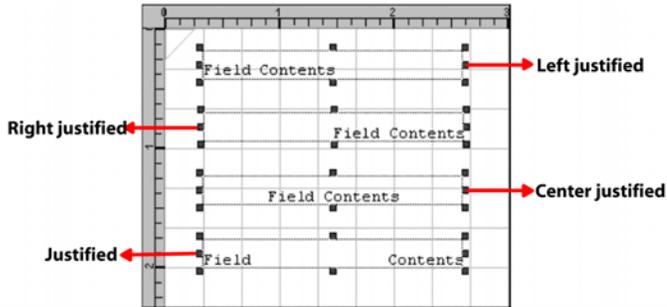
The cell fill color and pattern set to (1) none, none (2) LMRED, diamond fill pattern.

Setting Field Justification

In Vitesse, you can align the field data relative to the left and right edges of the field margin. The data can be aligned to one edge (left or right) of the field margin, centered in the field object, or aligned to both edges of the field margin (full justification). Additional options to specify the vertical justification of the normal field contents to top, bottom, or center of the field object are also available.

To specify the field horizontal and vertical justification:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab
- 3 From the **Justification** area, select one of the following horizontal justification options:
 - **Left** – click  to start data from left-most position and extend right with normal spacing.
 - **Right** – click  to start data from right-most position and extend left with normal spacing.
 - **Center** – click  to equidistance data from left and right.
 - **Justify** – click  to start data from left-most position and end in right-most position; data is aligned evenly within the field object.



The design area reflects the effect of the horizontal justification options.

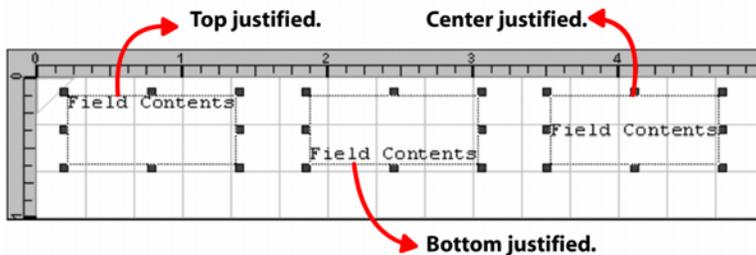


When working with **Normal** fields, specifying the horizontal justification options updates the **Alignment** drop-down list accordingly in the **Settings** tab to reflect the same settings and vice versa.

4 Select one of the following vertical justification options:

- **Top** – click  to align the field contents along the top field margin.
- **Bottom** – click  to align the field contents along the bottom field margin.
- **Center** – click  to align the field contents at the field center.

Both the horizontal and vertical field justification settings work together to produce the final output.



The design area reflects the effect of vertical justification options. In this example, the horizontal justification is set to *Left*.

Using Image Fields

An image object can be added to a line mode project using image fields. In an image field, the value of the field is the name of the image source file and its extension; this value can be picked from a data file or from a variable. The image is positioned relative to the bottom left edge of the field object. The inserted image is added to the **Images** node under the **Resources** node in the project tree. You can position, align, scale, and rotate the added image field. All image formats supported by Vitesse can be inserted using image fields.



The image fields can only reference the images placed in the Vitesse global image library located at `drive:\Program Files\Elixir Technologies\Vitesse\ImgLib`. For more information on global resource folders, see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as an image:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Image** from the **Field Type** drop-down list.

Specifying an Image Resource

The data that references the image resource can be picked from the data file or a variable (that has the image file name and extension as its value). You can also use existing image resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The image resource specified using the data is then mapped to the page. Methods for using the data file and variables as an image field data are explained below.

Using the Data File

Vitesse supports the use of both fixed length (non-delimited) and variable length (delimited) data files for a non-prefixed or prefixed line mode project. Different procedures are used for specifying the image field data from fixed length and delimited files. When working with fixed length data, a field is defined by specifying the field starting position and length. With delimited data, you need to specify the field number.



Before setting the **Field Type** as an **Image**, ensure the data file that you use has the required image file names and extensions.



For more information on specifying field data from fixed length data files, see the “Using Fixed Field Data” section earlier in the chapter.



For more information on specifying field data from delimited data files, see the “Using the Field Number” section earlier in the chapter.

1. Select Image from the Field Type drop-down list.

2. Specify the field data, select Fixed or Data depending on the type of data file being used.

Field Data:
 Fixed Start: Length:
 Data Field No:

Field Type: Image
 Normal
 Image
 Segment
 Static Text

Field(0)	Field(1)	Field(2)	Field(3)
1. Mr.	John Brown	John Brown.tif	54 High
2. Ms.	Janet Foster	Janet Foster.tif	32 Maple
3. Ms.	Kelly Michaels	Kelly Michaels.tif	54 Park
4. Mr.	Kevin White	Kevin White.tif	72 Carl

3. The image displays in the design area. The image is also added under the Resources>Images node in the project tree.

The image shows the image field data being specified for a delimited data file.

Using Variables

To specify the image field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable**, which indicates that a variable will be used to specify the image field data. Selecting this option enables the associated drop-down list. The variables used for referencing images store the name and extension of the image resource.
- 4 Select the required system or user-defined variable from the drop-down list. The field updates in the design area to display the referenced image.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge data** option under the **Data Merging** area to view the specified variable value in the design area. See the “Using Variables” section above for more information.

Using an Existing Project Resource as Field Data

You can use existing image resources under the **Resources>Images** tree node for the image field.

To specify the image using the project resources:

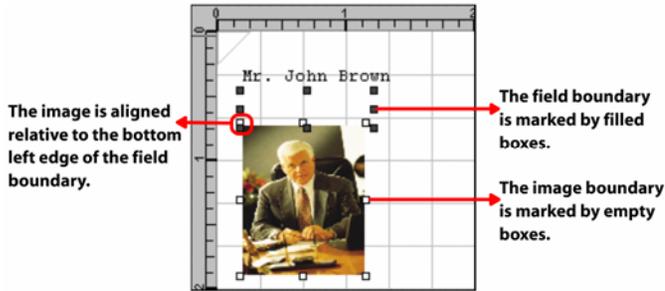
- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled.
- 4 Select the required image resource from the drop-down list. The drop-down lists all of the images used in the project. The selected image displays in the design area.



Using the **Const** option, you can also add images not present in the project resources by entering the image name and extension in the entry box.

Aligning the Image Field

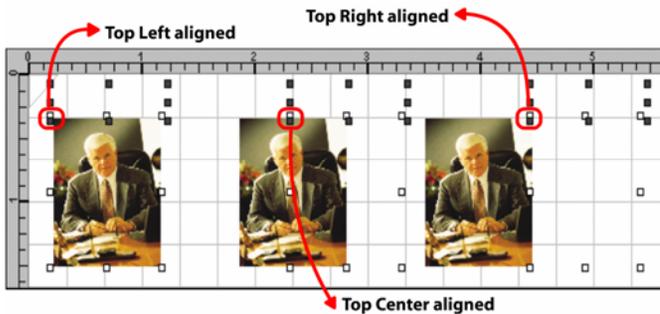
You can position an image relative to a point of reference. The referenced image is always positioned relative to the bottom left edge of the field bounding box. Vitesse provides various alignment options for image fields to achieve a more visually balanced design.



The image highlights the point of reference for various image alignment options.

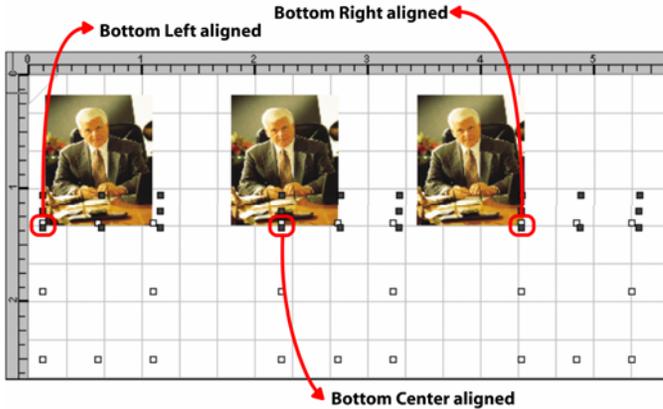
To align the image field:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Top Left** – aligns the top left edge of the image along the bottom left edge of the field.
 - **Top Right** – aligns the top right edge of the image along the bottom left edge of the field.
 - **Top Center** – aligns the top center point of the image along the bottom left edge of the field.



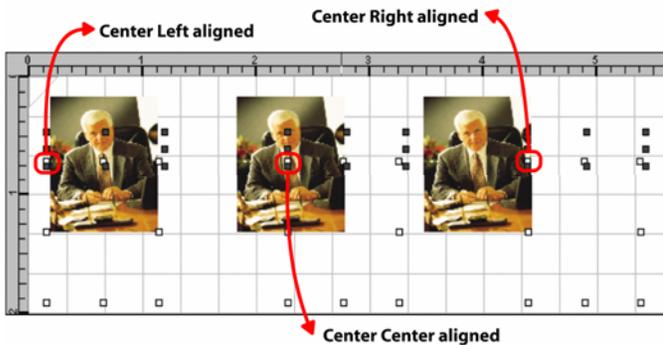
The image shows the effect of the top image alignment options.

- **Bottom Left** – aligns the bottom left edge of the image along the bottom left edge of the field.
- **Bottom Right** – aligns the bottom right edge of the image along the bottom left edge of the field.
- **Bottom Center** – aligns the bottom center point of the image along the bottom left edge of the field.



The image shows the effect of the bottom image alignment options.

- **Center Left** – aligns the center left edge of the image along the bottom left edge of the field.
- **Center Right** – aligns the center right edge of the image along the bottom left edge of the field.
- **Center Center** – aligns the center of the image along the bottom left edge of the field.



The image shows the effect of the center image alignment options.

Using Segment Fields

A segment object can be added to a line mode project using segment fields. In a segment field, the value of the field is the name of the segment source file and its extension; this value can be picked from a data file or from a variable. The segment is positioned relative to the bottom left edge of the field object. The inserted segment is added to the **Segments** node under the **Resources** node in the project tree. You can position, scale, rotate, and dynamically size the added segment. All segment formats supported by Vitesse can be inserted using the segment fields.



The segment fields can only reference the segments placed in the Vitesse global segment library located at `drive:\Program Files\Elixir Technologies\Vitesse\SegLib`. For more information on global resource folders see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as a segment:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 Select **Segment** from the **Field Type** drop-down list.

Specifying a Segment Resource

The data that references the segment resource can be picked up from the data file or a variable (that has the segment file name and extension as its value). You can also use existing segment resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The segment resource specified using the data is then mapped to the page. Methods for using the data file and variables as a segment field data are explained below.

Using the Data File

Vitesse supports the use of both fixed length and delimited data files for a line mode project. Different procedures are used for specifying the segment field data from fixed length and delimited files. When working with fixed length data, a field is defined by specifying the field starting position and length. With delimited data, you need to specify the field number.



Before setting the **Field Type** as a **Segment**, ensure the data file that you use has the required segment file names and extensions.



For more information on specifying field data from fixed length data files, see the “Using Fixed Field Data” section earlier in the chapter.



For more information on specifying field data from delimited data files, see the “Using the Field Number” section earlier in the chapter.

1. Select Segment from the Field Type drop-down list.

2. Specify the field data, select Fixed or Data depending on the type of data file being used.

3. The segment displays in the design area, the segment is also added to the project resources.

The 'Field Data' dialog box shows 'Fixed' selected with 'Start' at 0 and 'Length' at 8. The 'Field Type' dropdown is set to 'Segment'. The 'Data View' window shows a table with data, and the design area shows a segment with the text: 'Welcome to Central Commercial! With the space and service, we will have new Banking opportunities and information that will help you. Thank you again for choosing Central Commercial.'

The image shows the segment field data being specified for a fixed length data file.

Using Variables

To specify the segment field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable** which indicates that a variable will be used to specify the segment field data. Selecting this option enables the associated drop-down list. The variables used for referencing segments store the name and extension of the segment resource.
- 4 Select the required system or user-defined variable from the drop-down list.

The field updates in the design area to display the referenced segment.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName`. in the associated entry box. You can then select the **Merge data** option under the **Data Merging** area to view the specified variable value in the design area. See the “Using Variables” section above for more information.

Using an Existing Project Resource as Field Data

You can use existing segment resources under the **Resources>Segments** tree node for the segment field.

To specify the segment using the project resources:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled.
- 4 Select the required segment resource from the drop-down list. The drop-down lists all of the segments used in the project. The selected segment displays in the design area.



Using the **Const** option, you can also add segments not present in the project resources by entering the segment name and extension in the entry box.

Using Static Text Fields

Static text fields extend the functionality of normal fields by providing formatting options (that are the same as available for the text and graphic text object) in addition to all the normal field functionality. However, there are a few exceptions:

- You cannot set the alignment options for static text fields.
- Also, the static text field data cannot be used as a label and value for a data driven chart.

To set the field type as a static text:

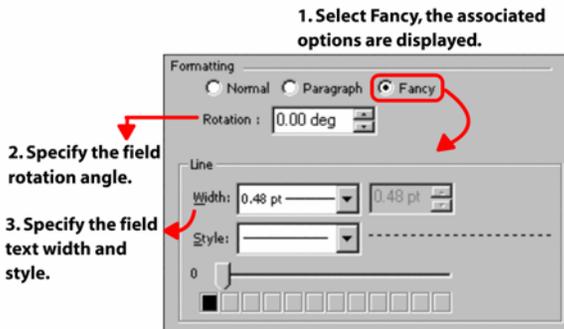
- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Static Text** from the **Field Type** drop-down list.

Formatting Static Text Field Data

You can choose to apply text formatting features such as text wrapping, rotation, line style, and width to static text fields.

To apply formatting options:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab.
- 3 From the **Formatting** area, select one of the following options:
 - **Normal** – applies normal formatting to the field data; any text extending the field boundary is not wrapped.
 - **Paragraph** – applies paragraph formatting to the field data; any text extending the field boundary is wrapped.
 - **Fancy** – applies enhanced formatting features to the field text; selecting this option displays the associated options for setting the line width, style, and rotation. See *Chapter 9: Using the Text Tools* for more information.



The image shows the process of applying fancy formatting.



For more information on setting rotation, line width, and line style, see tasks earlier in the chapter.



The line width and style settings are not visible in the design area. However, these are applied when the project is exported.

Setting Common Field Properties

Some field properties are common to all of the four field types. These properties include the field size, position, spacing, rotation, and auto-sizing.

Positioning a Field

Field objects can be placed at an absolute position in the design area by specifying the horizontal and vertical coordinates. You can also position the field objects relative to one another.

Setting Field Absolute Position

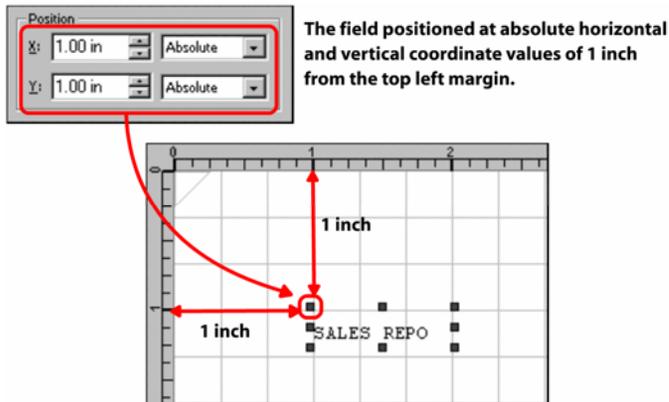
To accurately position a field object in the design area:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the field horizontal position, per specified unit of measurement in the **X** entry box. In the associated drop-down list, select **Absolute**.
- 4 Specify the field vertical position, per specified unit of measurement in the **Y** entry box. In the associated drop-down list, select **Absolute**.



You can also set the unit of measurement for the field position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The absolute coordinates are calculated from the top left margin of the page.



The image shows the process of setting field position using absolute coordinates. In this example, the top, left margins are set at 0, 0 inches.

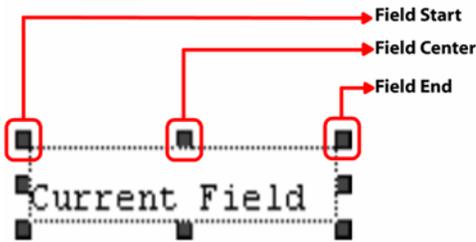
Setting Field Relative Position

A field object in a record can be positioned relative to the previous fields and relatively positioned objects added to the record. In absolute positioning, the top left margin of the page acts as the point of reference, whereas in relative positioning, the previously added object is taken as the point of reference.

Relative positioning is helpful in the placement of transactional records (prefixed data), for example, the credit card transactions for a customer. As the total number of transactions for any customer is unknown, the transaction information cannot be positioned absolutely when designing the project else, the data will overprint.

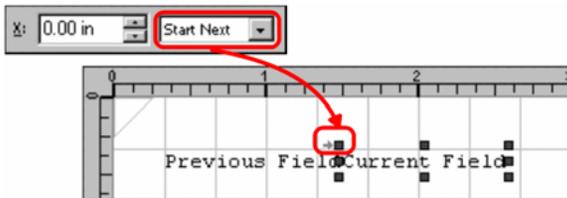
To position fields using relative positioning:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify **0** in the **X** entry box. In the associated drop-down list, select one of the following horizontal relative positioning options:



The horizontal relative positioning options are applied with reference to the field horizontal start, center, and end points.

- **Start Same** – positions the start of the current field at the start of the previous field.
- **Start Center** – positions the start of the current field at the center of the previous field.
- **Start Next** – positions the start of the current field at the end of the previous field.

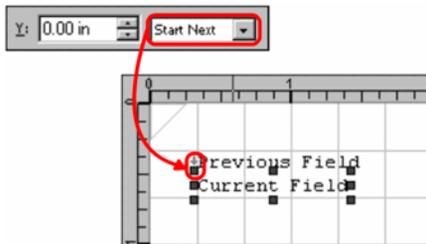


In this example, the current field is positioned at the previous field end.

- **Center Before** – positions the center of the current field at the start of the previous field.
 - **Center Same** – positions the center of the current field at the center of the previous field.
 - **Center Next** – positions the center of the current field at the end of the previous field.
 - **End Before** – positions the end of the current field at the start of the previous field.
 - **End Center** – positions the end of the current field at the center of the previous field.
 - **End Same** – positions the end of the current field at the end of the previous field.
- 4 Specify **0** in the **Y** entry box. In the associated drop-down list, select one of the above-mentioned vertical relative positioning options. Vitesse provides the same set of vertical relative positioning options as the horizontal relative positioning options.



The vertical relative positioning options are applied with reference to the field vertical start, center, and end points.



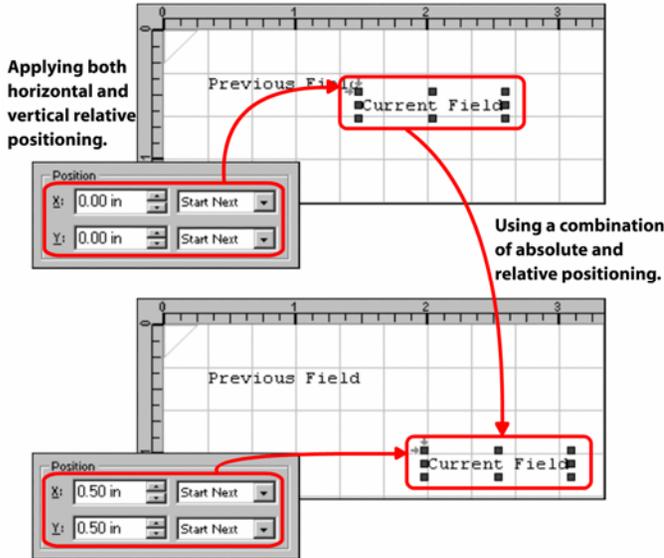
In this example, the current field is positioned at the previous field end.



Horizontal relative positioning options cannot be applied to the first field object added to a record.



Using a combination of absolute and relative positioning, you can place the field relative to another field or a relatively positioned object and add the spacing defined by the absolute X and Y coordinates.



In this example, the current field is positioned relatively at Same Next, Same Next and then displaced 0.5, 0.5 inches in the horizontal and vertical directions.

The following symbols appear next to a relatively positioned field node in the *Project Tree View* window:



-  - specifies that horizontal relative positioning is applied to the field.
-  - specifies that vertical relative positioning is applied to the field.
-  - specifies that both horizontal and vertical relative positioning is applied to the field.



The relative positioning options are applicable for all objects that can be relatively positioned in Vitesse.

Setting the Field Size

Field dimensions specify the width and height of the field. To set the field dimensions:

- 1 Select the required **Field** node from the project tree.
- 2 In the **General** tab of the *Property View* window, specify the following **Size** information:
 - **W** – sets the field width per specified unit of measurement.
 - **H** – sets the field height per specified unit of measurement.



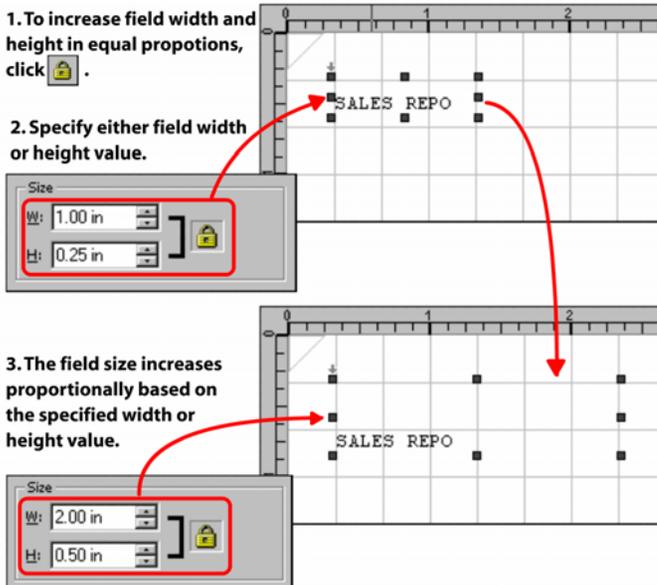
You can also set the unit of measurement for the field by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The  icon in the **Size** area is used for maintaining the aspect ratio of the field. Clicking this icon and then setting the width and height increases or decreases the length and width of the field in equal proportion.

1. To increase field width and height in equal proportions, click .

2. Specify either field width or height value.

3. The field size increases proportionally based on the specified width or height value.



The image shows the process of changing the field size while maintaining the aspect ratio.



You can also set the field size by dragging the field handles that define the boundary of the field object in the design area. The corresponding **Size** settings are updated in the *Property View* window.

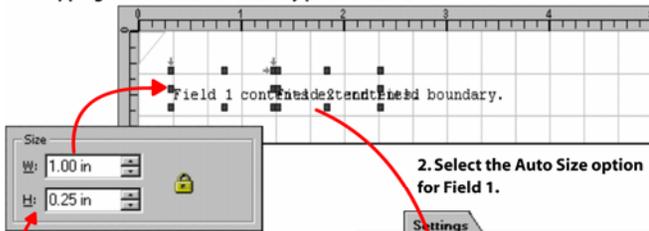
Adjusting the Field Size to its Contents

Vitesse enables you to auto-size field objects to adjust automatically to the amount of data it is displaying. This feature is especially useful when working with variable length (delimited) fields or with fields that contain merged variables. This also helps prevent overlapping of fields that are placed relative to each other.

To automatically adjust field size to the amount of data:

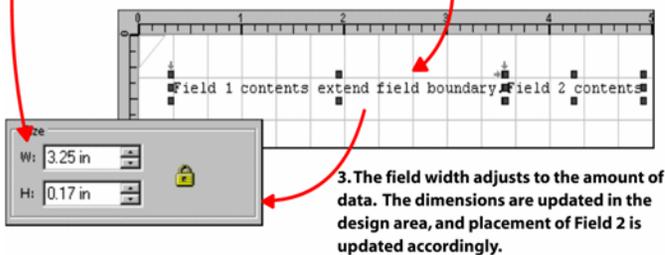
- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Auto Size** check-box. The field width automatically adjusts to fit all the data.

1. The contents of Field 1 extend the current field width, overlapping the contents of relatively placed Field 2.



Dimensions of Field 1.

2. Select the Auto Size option for Field 1.



3. The field width adjusts to the amount of data. The dimensions are updated in the design area, and placement of Field 2 is updated accordingly.

The image shows the process of adjusting field size to its contents.

Setting Spacing between Fields

You can set the spacing between the different instances of the same field object, when working with multiple line records in non-prefixed data, or records having multiple instances in prefixed data. The spacing can be set to either fixed or variable:

- Fixed spacing is used with non-prefixed data; this option adjusts the spacing of a field without affecting the spacing of other fields under the same record.
- When working with prefixed data, the variable spacing option is used. In this mode, spacing between each field instance is variable and affects spacing of the subsequent fields mapped for the same record. Variable spacing adjusts the spacing of all the fields under the same record. Adding additional fields adds the spacing value of the previous field to the spacing value of the new field.



For more information on selecting a spacing mode, see the “Using Fixed or Variable Spacing” section earlier in the chapter.

To set the field spacing options:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Spacing** area, specify the following:
 - **Horizontal** – specifies the horizontal spacing value of the field.

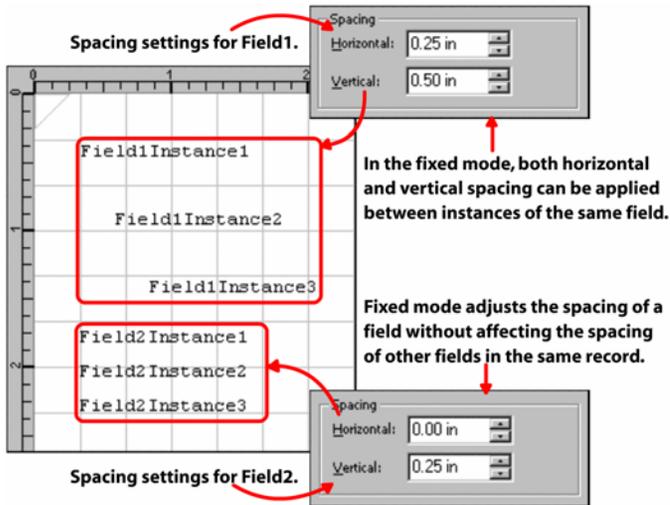


You cannot specify the horizontal spacing value when working with prefixed data. For prefixed data projects, the **Horizontal** entry box is disabled.

- **Vertical** – specifies the vertical spacing value of the field.



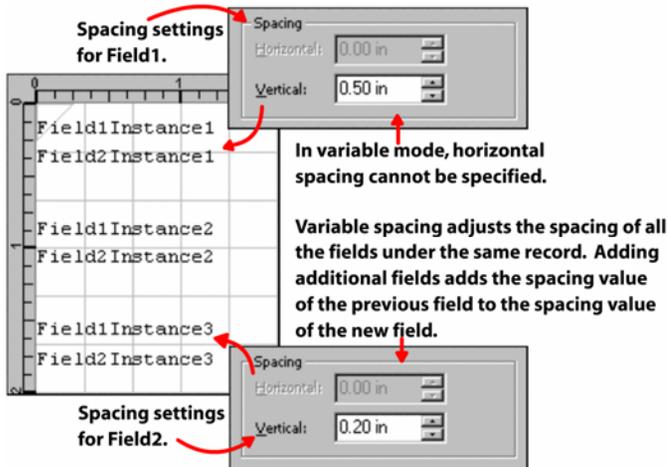
You can also set the unit of measurement for the field spacing by right clicking the upward or downward arrows in the **Horizontal** and **Vertical** entry boxes, and then selecting **Units>unit name** from the pop-up menu.



The image shows the process of specifying horizontal and vertical spacing in fixed spacing mode.



When working with prefixed data, the vertical spacing value for all fields in a record added after the first field is set to 0. This is a characteristic of variable spacing. If a spacing value is specified, it is added to the spacing value of the previous field and altogether, the spacing between the instances of both the fields will be equal to the sum of the spacing values.



The image shows the process of specifying vertical spacing in variable spacing mode.

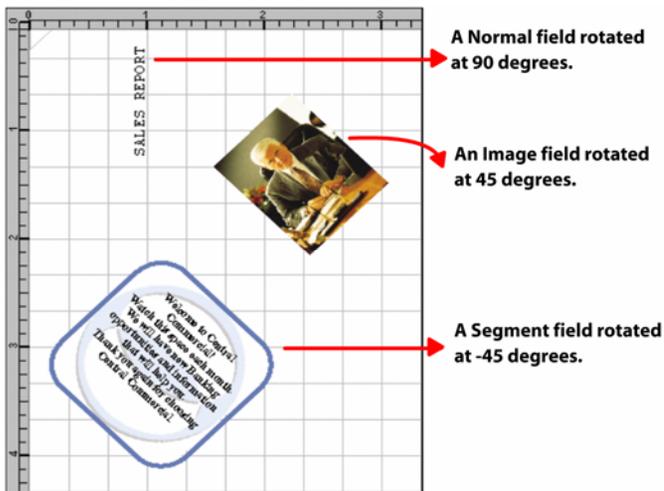
Setting Field Rotation

To set the field rotation:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required rotation angle in the **Rotation** entry box.



You can specify the angle in degrees, radians, or grads. To specify the unit of measurement for the rotation angle, right click the upward or downward arrows in the **Rotation** entry box, and then select **Angle>unit name** from the pop-up menu.



The image shows different rotations applied to various types of fields.



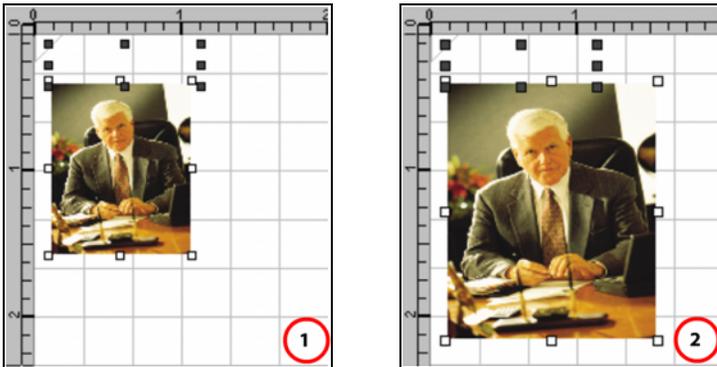
When working with normal or static text fields, you can also specify the rotation angle by selecting the **Fancy** option in the **Spacing** tab, and then specifying the rotation angle in the **Rotation** entry box.

Scaling Image and Segment Fields

The image and segment resources added to the project as fields are scaled at 100% by default. This means that the original image and segment dimensions are used. Once added to the project, you can scale the image and/or segment resource to suit your requirements.

To scale image and/or segment fields:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required scaling value as a percentage, in the **Scale** entry box.



The design area reflects an image field scaled to (1) 100% (2) 150%.

What is a Data Group?

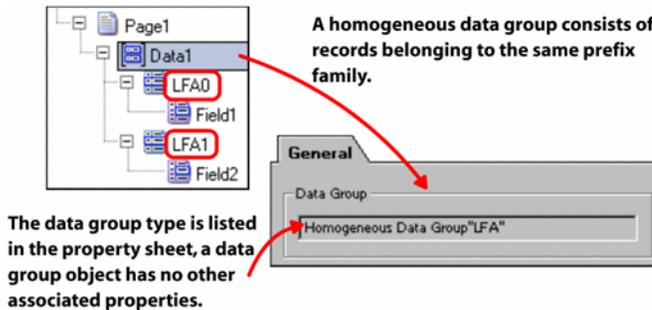
Prefixed data files consist of records that belong to the same and/or different prefix families. A prefix family is a set of related RPE keys. The initial part of the prefix is same for all the records in a prefix family. For example, the records with prefix LFA1 or LFA2 belong to the same prefix family LFA.

Line	Prefix	Field 1	Field 2	Field 3	Field 4
1	PERO	01	11	15	
2	REFO	14	15	30	
3					
4	ADRO	M.	MARCI		
5	ADRO	RESIDEN			
6	ADRO	PLACE	DI		
7	ADRO	23	23	SAC	
8					
9	DFAO	23	01	15	
10	DPYO	23	02	15	
11	DREO	7	12	15	
12	CRNO	06	8	025	
13					
14	LFAO	14	COMM		
15	LFAO	41	EAU		
16	LFAO		TAXE		
17	LFAO		LOCA		
18	LFAO	8	2	GAZ	
19	LFAO		TAXE		
20	LFAO				
21	LFA1		TOTAL	S	
22	LFAO				
23	LFAO	41	ASSA		
24					

The image displays a prefixed data file containing different prefix families.

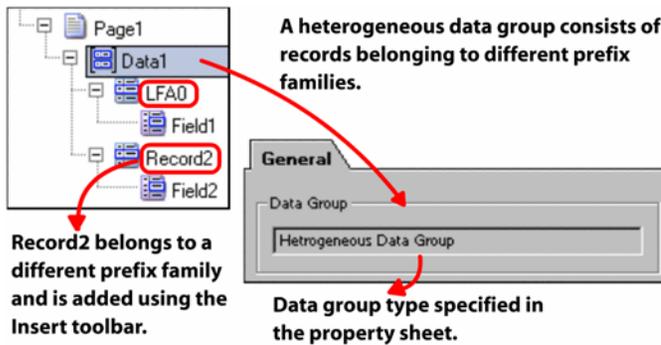
Data groups are used for processing records as a unit. Data groups containing records from a single prefix family are called homogeneous data groups; whereas data groups containing records from multiple prefix families are called heterogeneous data groups.

When dragging and dropping data from a prefixed data file, fields and records are automatically added under data groups. All records belonging to the same prefix family become part of a single data group (i.e., homogeneous data groups).



The image displays a homogeneous data group.

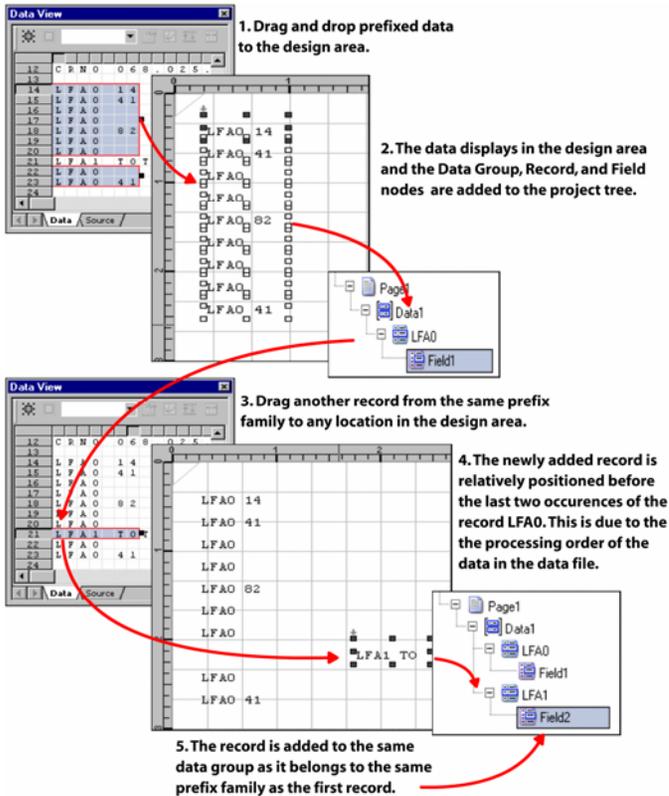
You can however, add records from different prefix families to a data group by inserting a record in the data group using the *Insert* toolbar, and selecting the prefix from the record property sheet (i.e., define heterogeneous data groups).



The image displays a heterogeneous data group.

The objects placed under a data group or record nodes in the project tree are processed sequentially from top to bottom. Data groups enable you to manipulate the processing order by adding records from the same prefix family to different data groups using the *Insert* toolbar. You can also relatively position fields that do not belong to the same prefix family by placing them under the same data group.

The spacing mode for records added in data groups is automatically set to **Variable**. Variable spacing adjusts the spacing of all fields under the same record. Adding additional fields adds the spacing value of the previous field to the spacing value of the subsequent fields mapped for the records that belong to the same data group.



The image shows the process of relatively positioning records belonging to the same prefix family. If you do not want to position record LFA1 relative to LFA0, you can add it to a separate data group.

Inserting a Data Group in Line Mode Projects

To insert a data group in a prefixed line mode project:

- 1 Select the required **Page** node from the project tree.
- 2 Select Insert>Advanced>Data Group from the menu bar; a **Data Group** node is created under the selected **Page** node in the project tree.

You can also add more data groups under a selected data group using the same procedure.



You can also insert a data group object by clicking  on the *Insert* toolbar.

You can delete a selected data group object either by:

- Selecting Edit>Delete from the menu bar.

Or

- Clicking  on the *Standard* toolbar.

Or

- Pressing on the keyboard.

All of the child nodes under the data group node are also deleted.



1. Drag and drop prefixed data to the design area.

2. The data displays in the design area and the Data Group, Record and Field nodes are added to the project tree.

3. Click , to add a new data group.

4. Click , to add a record to the data group, select LFA1 as the RPE key from the Settings tab in the record property sheet.

5. Click , to add a field to the record.

6. The fields for record LFA1 are not relatively positioned to the LFA0 fields, despite their order in the data file.

The image shows the process of manipulating data positions by placing records belonging to the same prefix family in different data groups.

User Notes:

Section VIII: Working in XML Mode

User Notes:

Chapter 23: Understanding XML Mode Data

This chapter provides an overview of the XML mode data stream and explains the XML data structure in detail. Included are the following topics:

- Understanding XML Mode Data
- Using the Data View Window
- Using the Data View Window for XML Mode Projects
- Viewing Multiple Tag Occurrences
- XML Data File Structure

Understanding XML Mode Data

XML mode projects are used for creating and managing designs that use XML data. XML data is ASCII data organized within start and end tags. Vitesse requires a BTA (Begin Tag Action) and ETA (End Tag Action) command to process XML data. Vitesse processes the data (value) whenever an End tag is encountered. The page composition of the XML mode data is controlled by an XML Job Ticket (XJT). It is similar to the Line mode JDT. It contains the document layout definitions (orientation, forms, medias, frames, fonts, colors, etc.) required for processing the XML projects. XML mode data files usually have an .XML extension.

Inserting a Data File

Vitesse enables you to use a data file for designing and composing variable-data documents. The project specific production data file is specified at the time of project creation. If a data file is not specified, Vitesse uses a default sample data file for the project. The sample data file normally represents the original production data file and covers all the possible scenarios. You can change the data file any time during the design process by inserting a new data file.

To insert a new data file:

- 1 Select **Insert>Datafile** from the menu bar to open the *Insert Data File* dialog with the **Existing** tab selected.



You can also open the *Insert Data File* dialog by clicking  on the *Insert* toolbar.

- 3 Browse and select the required XML data file (*.xml).
- 4 Click **Open**.

The *Data View* window opens with the specified data file. Any data mapped onto the design is updated to reflect the changes. The data file path in the **SubDoc** properties is also updated to display the path of the newly inserted sample data file.

Using the Data View Window

The data files inserted in a project can be viewed in the *Data View* window. This helps you to visually drag and drop the required data directly onto the design area. The *Data View* window is context-sensitive and displays data files and associated options relating to specific project modes (i.e., Line, Database, or XML mode) by using two primary views:

- **Data** view presents project data files by row numbers and columns. For XML mode projects, the data file is displayed as a tree structure.
- **Source** view enables you to edit source data files, and then review the changes in the **Data** view.



For more information on editing a data file and customizing the *Data View* window, refer to the tasks: “Editing the Data File” and “Setting Data View Options” in *Chapter 21: Understanding Line Mode Data*.

The *Data View* window is not displayed by default when a project is opened. To view the *Data View* window, select Window>Data from the menu bar.



You can also view the *Data View* window either by:

- Clicking  on the status bar.
Or
- Pressing <Ctrl> + <3> on the keyboard.

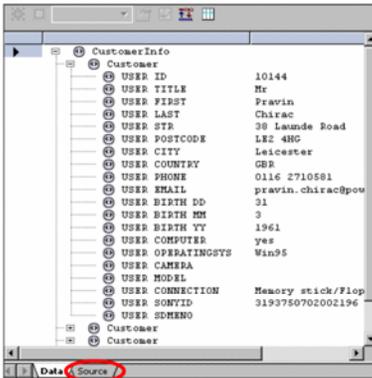
The *Data View* window is docked at the right side of the workspace by default.

Using the Data Window for XML Mode Projects

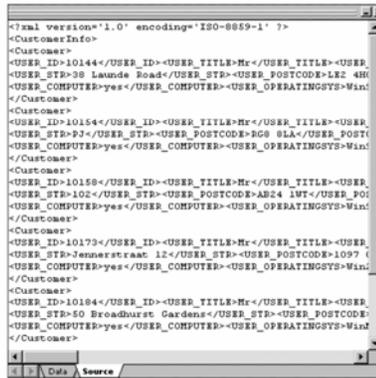
XML data files include tags, attributes, and values that can occur multiple times in the data file. The *Data* window displays XML data files in both the **Data** and **Source** views:

- The **Data** tab shows the XML data in a tree structure so that multiple tags, attributes, and values can be easily viewed.
- The **Source** tab displays the data file in the actual XML syntax as you would see in any text editor. You can view and edit the entire data file within this tab.

Data View window displaying the XML data in a tree structure.



Data View window displaying the source code of the data file.



Click the Source tab to view or edit the actual data file.

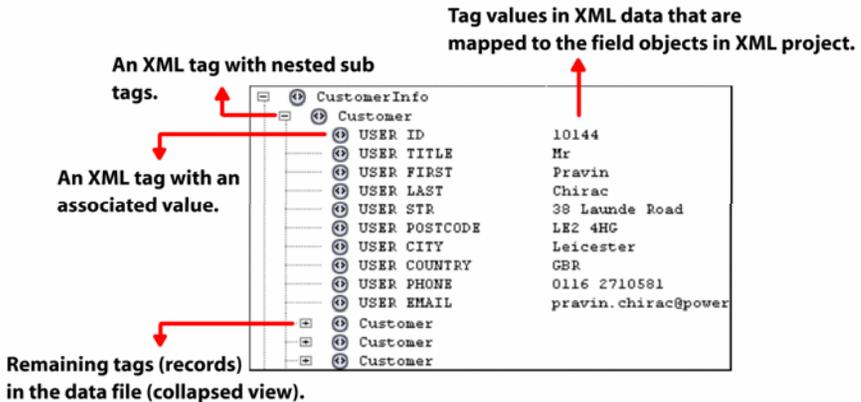
Data file can be viewed using the **Data** and **Source** tabs.

The *Data* View window also provides a toolbar for performing different operations on the data in the **Data** tab.

XML Data Tree

In the **Data** view, tag and attribute names display on the left side of the tree; values display on the right side of the tree, next to the associated tag and attribute names.

The first node in the XML data file is the root node. The root node is a tag containing sub tag nodes and their values. You can expand the root node and sub nodes to view subsequent data file tags, attributes, and their values in the **Data** view. When a data file has multiple occurrences of the same tag name or associated sub nodes, selecting the tag name or associated sub nodes will automatically select every occurrence of the same tag name.

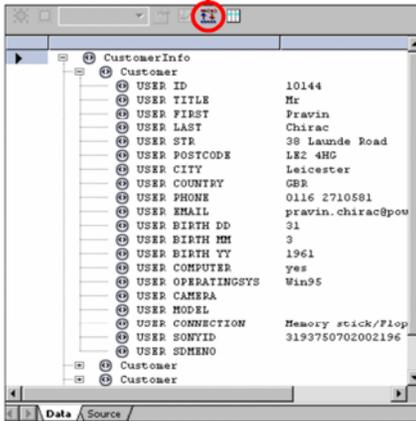


The **Data** tab further provides two views for presenting data:

- An expanded view that displays all tag name occurrences and associated attributes and values on a single page. This is the default view.
- A combined view that displays each tag name and associated attributes and values separately on each page. For example, if there are two customer's records in a data file, then, using the combined view, you can view the first record on the first page, followed by the second record of the same customer on the next page.

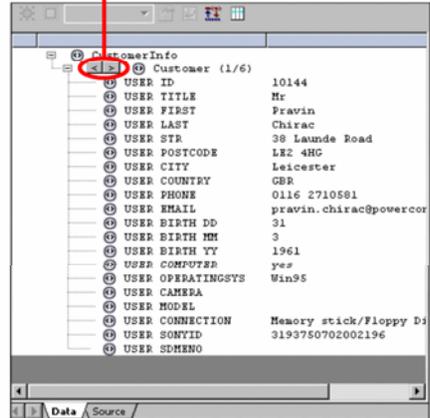
To switch between the two views, click the **XML View** button  on the toolbar available at the top of the *Data View* window (**Data** tab). This button is available for XML mode only.

1. Click this button to switch to a combined view.



Data tab - Expanded view

2. Use these arrow buttons to navigate between the records.



Data tab - Combined view

Switching from the expanded view of the data to the combined view.



To auto-size the data columns, click  on the toolbar. For more information, refer to the “Auto-Sizing Data Columns” task in *Chapter 21: Understanding Line Mode Data*.

Viewing Multiple Tag Occurrences

When a data file has multiple occurrences/instances of the same tag or associated sub nodes, selecting that tag or associated sub nodes automatically selects every instance of the same tag in the expanded view. For example, if you select any of the “Customer” nodes in the data tree (see image above), all of the other “Customer” nodes are also selected automatically.

The combined view is more useful for viewing multiple tag occurrences in a data file. This view enables you to view the multiple occurrences of a tag by displaying only one instance at a time.

- To view the next tag occurrence, click  adjacent to the tag name.
- To view the previous tag occurrence, click  adjacent to the tag name.



If a sub node is not present in one of the multiple tag occurrences, it displays gray in the combined view.

XML Data File Structure

The XML data file structure differs from data files used in other project modes. As mentioned earlier, the data files used in the XML mode projects follow the standard XML syntax. The first line is the XML declaration; an optional line stating the version of the XML file in use (normally version is 1.0), and may also contain information about character encoding and external dependencies. The remaining file consists of tag names, attribute names, and values in a structured layout. Prefixed and non-prefixed data files are not supported in XML Mode projects.

Tags

Tags organize the structure of an XML data file. In XML data files, data is enclosed between Start and End tags. These tags are viewed in the **Source** view of the XML *Data View* window.

Tags comprises of the following:

- **Tag name** – specifies the name of a tag.
- **Start tag** – specifies the beginning of a non-empty XML tag. The syntax used for a start tag is `<TagName>`.
- **End tag** – specifies the end of a non-empty XML tag. The syntax used for an end tag is `</TagName>`.
- **Content** – The content is everything that appears between the start and end tags. It can be either text or other (sub) tags.

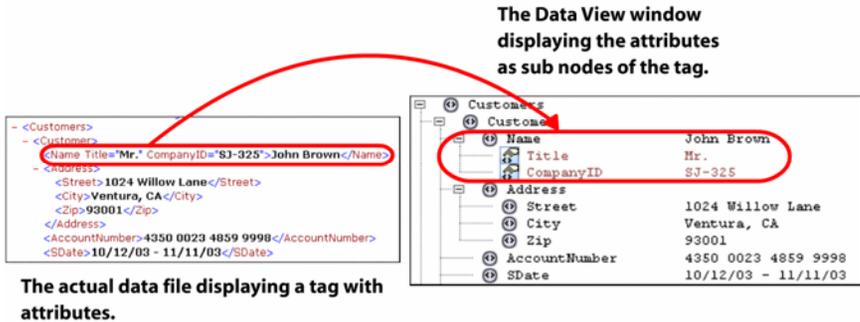


A tag with no content is an empty tag. The syntax used for an empty tag is `<TagName/>`.

Vitesse processes data only after it finds an associated End tag. If another Start tag and End tag is encountered before finding the End tag for the previous data, the data with both the Start and End tags is processed first. You can map tags to an XML project using either the *Insert* toolbar command or a drag-and-drop operation. When you drag and drop data from the data file to the design area, the tag is added as a **Record** node to the project tree. For more information, refer to *Chapter 24: Working with XML Mode Data*.

Attributes

Each tag may have a set of attributes that have a name and a value. The attributes are specified as part of the start tag. In the XML data tree, attributes of a tag appear as the sub nodes. For example, two attributes of the “Name” tag could be “Title” and “CompanyID.” You can map attributes to an XML project by performing a drag-and-drop operation. When you drag and drop data from the data file to the design area, the attribute is added as a **Field** node to the project tree. For more information, refer to *Chapter 24: Working with XML Mode Data*.



An XML data file and the tree structure illustrating the use of attributes.

Values

Values describe the tags and tag attributes. For example, a value for a “Resource” tag could be “writer;” an attribute of the “Resource” tag could be “Type;” and a value for the “Type” attribute could be “Miscellaneous.” You can map values to an XML project by using either the *Insert* toolbar command or a drag-and-drop operation. When you drag and drop data from the data file to the design area, the value is added as a **Field** node to the project tree.

An XML tag and its content tags.

```
TrnXML.xml
<?xml version="1.0" encoding="UTF-8"?>
<Customers>
  <Customer>
    <Title>Ms.</Title>
    <Name>Celine Marchbank</Name>
    <Address>
      <Street>1024 Willow Lane</Street>
      <City>Ventura, CA</City>
      <Zip>93001</Zip>
    </Address>
    <AccountNumber>4350 0023 4859 9998</AccountNumber>
    <SDate>10/12/03 - 11/11/03</SDate>
    <AccountSummary>
      <PreviousBalance>97.00</PreviousBalance>
      <Purchases>897.65</Purchases>
      <Fees>29.00</Fees>
      <Payments>600.00</Payments>
      <EndingBalance>423.65</EndingBalance>
      <MinimumPayment>21.19</MinimumPayment>
      <PaymentDueDate>11/30/03</PaymentDueDate>
      <PastDueAmount>23.32</PastDueAmount>
    </AccountSummary>
  </Customer>
</Customers>
```

Value of the "Street" tag.

```
<Address>
  <Street>1024 Willow Lane</Street>
  <City>Ventura, CA</City>
  <Zip>93001</Zip>
</Address>
```

Start and End tags identifying the "Street" tag.

An XML data file highlighting the data tags.



You can also map attributes to a page design as fields. For more information, refer to *Chapter 24: Working with XML Mode Data*.

Chapter 24: Working with Data in XML Mode

The chapter describes activities and tasks for working with data in XML mode projects. Included are instructions and information for the following:

- Mapping XML Data to the Project Design
- Understanding XML Processing Order
- Mapping Records
- Mapping Fields
- Using Normal Fields
- Using Image Fields
- Using Segment Fields
- Using Static Text Fields
- Setting Field Properties

Mapping XML Data to the Project Design

Data in XML data files is organized in a hierarchical structure using start and end tags (see *Chapter 23: Understanding XML Mode Data*). When you map data from an XML data file to a project design, the tag names and associated attributes and values display as record and field nodes respectively in the project tree.



Records in an XML data file are enclosed within start and end tags, and the tag name identifies the record name.

The data object structure of a Vitesse XML mode project has been imposed upon the XML data file structure as follows:

- A record is any start/end tag pair that can contain nested start/end tag pairs. For example, note the Address tag in the XML data file shown in the image below. The Address tag in the data file can be mapped to a record object in the project design. The nested data tags within the Address tag will be mapped as field objects. The tag values of the nested tags will be used as the field data.
- A record is any start/end tag pair with a tag value. It is mapped to the project such that the record is the tag name, and the field is the tag value. For example, the Street data tag in the data file shown below is added to the project such that the Street tag name is the record and the Street tag value is the field. If the tags contain attributes, these are also mapped in the same manner.



Tag value of a data tag mapped to a field displays as the field data in a project design.

A data tag with nested tags. You can map the "Address" tag to a record with "Street", "City" and "Zip" fields.

```
TrnXML.xml
<?xml version="1.0" encoding="UTF-8"?>
<Customers>
  <Customer>
    <Title>Ms.</Title>
    <Name>Celine Marchbank</Name>
    <Address>
      <Street>1024 Willow Lane</Street>
      <City>Ventura, CA</City>
      <Zip>93001</Zip>
    </Address>
    <AccountNumber>4350 0023 4859 9998</AccountNumber>
    <SDate>10/12/03 - 11/11/03</SDate>
    <AccountSummary>
      <PreviousBalance>97.00</PreviousBalance>
      <Purchases>897.65</Purchases>
      <Fees>29.00</Fees>
      <Payments>600.00</Payments>
      <EndingBalance>423.65</EndingBalance>
      <MinimumPayment>21.19</MinimumPayment>
      <PaymentDueDate>11/30/03</PaymentDueDate>
      <PastDueAmount>23.32</PastDueAmount>
    </AccountSummary>
  </Customer>
</Customers>
```

Value of the "Street" tag.

```
<Address>
  <Street>1024 Willow Lane</Street>
  <City>Ventura, CA</City>
  <Zip>93001</Zip>
</Address>
```

You can map the "Street" tag to a record object. The tag value of the "Street" tag will be the field.

An XML data file highlighting the data tags.

Data in a data file can be mapped or placed in an XML mode project using any of the following three methods:

- Using the Insert menu.
- Using the tools on the *Insert* toolbar. The toolbar is located vertically between the *Project Tree View* window and the design area.
- Using drag and drop to move/map data from the *Data View* window to the design area.

You can map data directly from the *Data View* window by selecting and dragging the required data to the design area. The data is mapped to your document and the corresponding record and field nodes add to the project tree.

Understanding XML Data Processing Order

The processing and printing order of data in XML mode does not depend on the object placement order in the project tree, but on the order of the data in the data file. The data file is read sequentially. Vitesse starts the processing of an XML data file by reading the first start tag in a file. For each start tag encountered, it looks if a page object in the project tree is associated with the tag. Consequently, the design objects (other than the records) inserted in the page are processed and printed. Besides the page, only the Tag Start segments inserted under records are processed whenever the associated start tag is encountered in a data file. After printing the associated page and Tag Start segments, Vitesse reads the contents of the data tag in the data file until it reaches the corresponding end tag. For each end tag encountered, it looks if there is an associated record object in the project tree. The fields and other objects inserted under this record object are then processed and printed. Tag End segments inserted in a page are also processed whenever the end tag of the associated data tag is encountered in a data file.

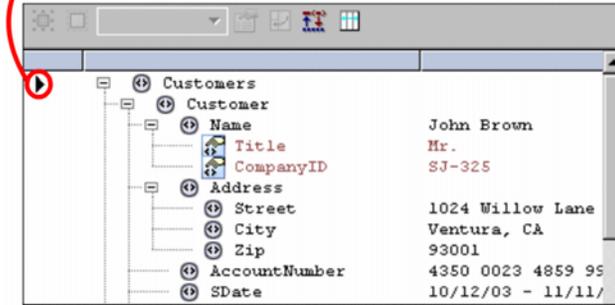
By using these segments, Vitesse enables you to control the inherent processing order of objects contained in a page and/or record. For example, if the design requirement is to process some design objects in a page after the associated data tag has been processed, you can make use of the Tag End Segment type. For more information, refer to the task “Working with Segments in an XML Mode Project” in *Chapter 32: Using Segments in Projects*.

Associating a Data Tag to a Page

In a newly created XML mode project, the root node of the data file is always associated with the first **Page** node in the project tree by default. You can choose to associate a different data tag to the page as per your design requirements.

To specify a data tag, select the **Page** object. From the **Settings** tab of the *Property View* window, select the required data tag from the **XML** drop-down list. This setting is also used as a page break. In an XML data file, a data tag may contain multiple instances. Mapping a data tag to a page displays each instance of that tag on a new logical page; hence, applying a page break for each instance. This feature is helpful when the record structure in the data file is such that each instance of a certain data tag corresponds to the data for a separate document. In the *Data View* window, page break markers are displayed with the data tag associated with the page. These markers are available only in the expanded view of the **Data** tab.

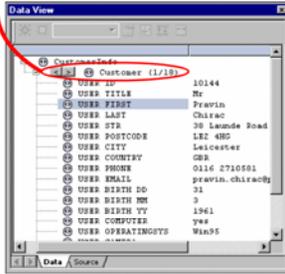
A page break marker displaying with the Customers node indicates that a new logical page starts for each instance of the Customers tag. In this data file it has one instance and therefore only one logical page will be created.



Data tab (expanded view) highlighting the page break markers.

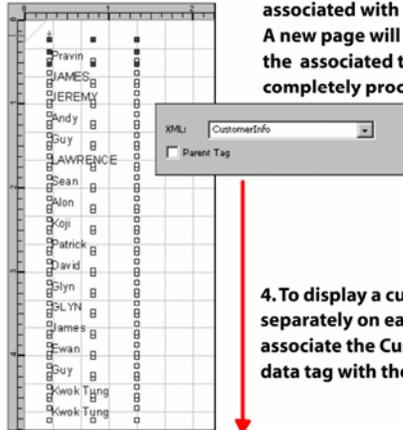
The following example shows the effect when a page break is applied on the **Customer** tag. There are eighteen Customers tags in the sample data file, which means that eighteen logical pages are created.

1. The 'Customer' tag has 18 instances each representing a different customer record.

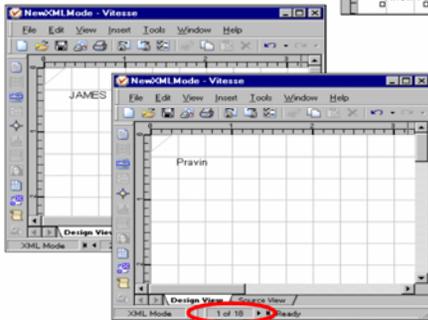


2. On mapping a field from the customer record to the page design, all instances of the record map to the same page.

3. This effect is a result of the CustomerInfo data tag associated with the page. A new page will start when the associated tag is completely processed.



4. To display a customer's data separately on each page, associate the Customer data tag with the page.



5. A new page starts for each Customer record. You can navigate between the 18 customer documents using the status bar.

You can select the **Parent Tag** option in order to associate the parent tag of the specified tag with the page. In this case, the page break is also associated with the parent tag. This setting only affects the preview in designer and must be disabled before exporting the project. If this option is not disabled, the view and print outputs of a project design will not match. Vitesse also displays a warning message in the *Message View* window regarding this setting.

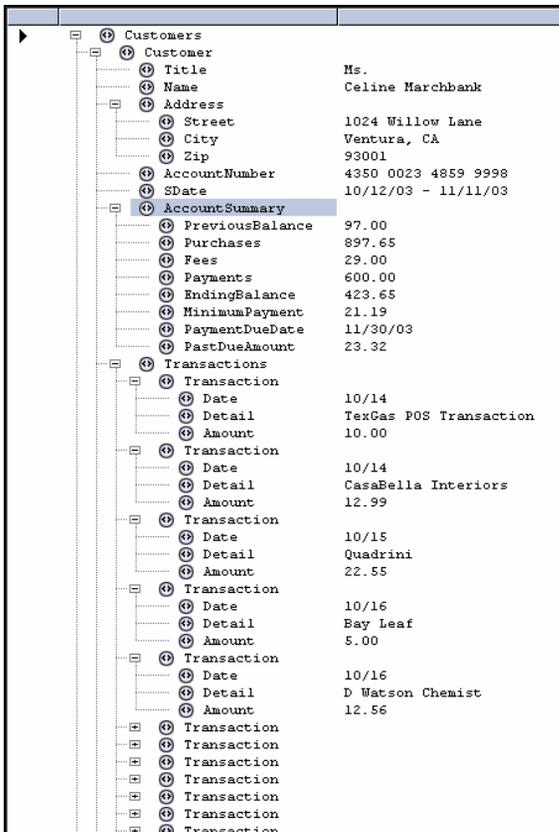
Mapping Records

You can map a data tag as a record object using either of the following ways:

- Inserting a record object and specifying the required data tag using the record properties.
- Dragging and dropping the required data tag onto the design area.

As mentioned earlier, in XML mode projects, tag names are added as **Record** nodes and tag values are added as **Field** nodes to the project tree. However, processing and print order of records is not dependent on their placement order in the project tree. It depends on the occurrence of the end tags of the data tags mapped to them. The processing order of records is explained in more detail in the tasks “Understanding XML Data Processing” and “Record Processing” in this chapter.

When you drag and drop data selecting only the tag values (not tag names) from the data file in the design area, corresponding record and field nodes are simultaneously added to the project tree, with the parent tag in the data file added as the record node. For example, in the image below, dragging and dropping “Ms” (tag value) on the design area will create “Customers_Customer” (parent tag name) as the record node, and “Title” as the field node. You can subsequently add records and associated fields using the same process. Vitesse automatically keeps track of fields that belong to a particular record, and adds them to the tree accordingly. However, Vitesse processes and prints data only when it finds the end tag of a record in the data file.



XML data file.

For example, the image above depicts the **Address** tag belonging to the **Customer** record. **Address** data will print at the end because the **Customer** record end tag is encountered at the end of the data file. The **PreviousBalance** tag, belonging to **AccountSummary** record, will print before the **Address** tag, because the end tag for the **AccountSummary** record is encountered before the end tag for the **Customer** record.

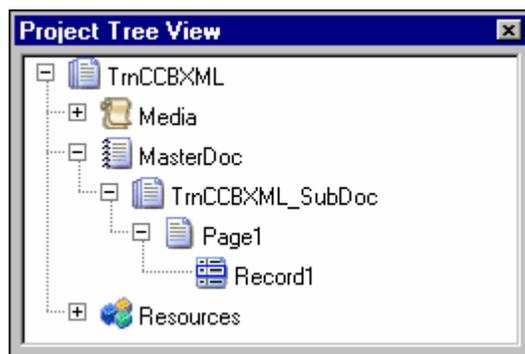
Print order can be changed by first dragging and dropping only the specific tag name from the data file, (which creates an independent record node in project tree), and then dragging and dropping the associated tag value, (which creates a field node in project tree).

Continuing from the previous example, you would first drag and drop the **Address** tag in the design area, then drag and drop the **Address** tag value. Vitesse then finds the **Address** record end tag right after the **Address** tag value, which allows for independent placement of the **Address** record and associated fields in the project tree. Vitesse then prints the **Address** tag value before the **PreviousBalance** tag value. You can repeat this process for subsequent tag names and associated values in the data file. See “Record Processing” section later in this chapter.

Inserting a Record in XML Mode

To insert a record in an XML mode project:

- 1 Select the required **Page** node from the project tree.
- 2 Select Insert>Record from the menu bar; a **Record** node is created under the selected **Page** node in the project tree.



Project tree displaying the newly inserted record.

You can also add more record objects under a selected record using the same procedure.



You can also insert a record object either by:

- Clicking  on the *Insert* toolbar.
Or
- Right-click the **Page** node, and select Insert>Record from the pop-up menu.



You can delete a selected record object either by:

- Selecting Edit>Delete from the menu bar.
Or
- Clicking  on the *Standard* toolbar.
Or
- Pressing on the keyboard.

All the child nodes under the record node are also deleted.

Mapping a Record to a Data Tag

To map the inserted record to a data tag:

Using the Record Property Sheet

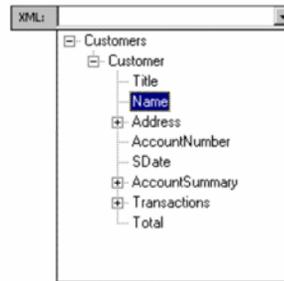
- 1 Select the required **Record** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Click the drop down arrow in the **XML** entry box. It displays all the data tags from the data file associated with the project. The tags are shown in a tree structure according to their hierarchy in the data file.

- 4 Select the required data tag from the drop down list. You can expand a collapsed node by clicking **+** adjacent to the node.
- The selected data tag is displayed in the **XML** entry box. You can also enter the required tag and its parent node in this box.

1. In the Settings tab of the record properties, click the XML drop down to display the data tags from the associated data file.



2. Select the required data tag from the drop down list.



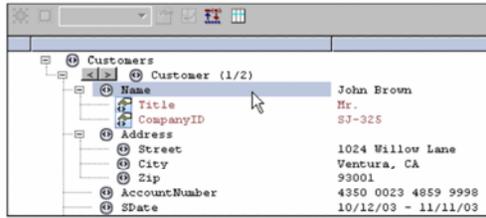
3. The selected data tag name displays. It corresponds to the hierarchy of the selected tag in the XML data tree.

Steps involved in associating a data tag to an already inserted record.

Using Drag and Drop

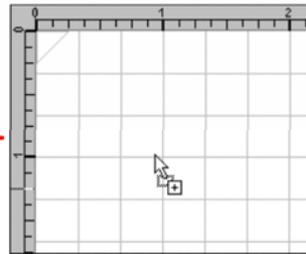
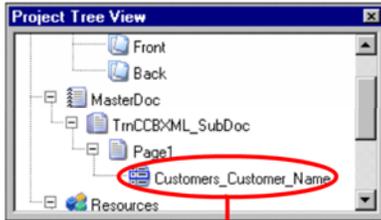
Vitesse also enables you to drag the required data tag from the associated data file and drop it to the design area. The project tree updates immediately to represent the newly added record. The record added to the project this way is named using a specific syntax (discussed below). The design area does not display the records. Records contain fields and other design objects that are actually displayed in the design area when mapped.

1. Select the 'Name' tag from the Data View window and drag it.



2. Drop the dragged tag in the design area.

3. A record node is added to the project tree.



The record name corresponds to the hierarchy of the data tag names in the XML data tree.

Steps involved in mapping a data record to the project design using drag and drop.

Syntax for Referencing XML Tags in the Project Design

The name of the newly added record using drag and drop corresponds to the hierarchy of the selected data tag in the XML tree. For example, if the **Name** tag is selected, three tags (namely **Name** (selected tag), **Customer** (parent tag), and **Customers** (parent tag of Customer)) make up the name of the record. The tag names are separated by an underscore (_) sign.

VIPP uses a similar syntax for referencing data tags in the project design. It consists of all the parent nodes and the node itself. Node names are all preceded by the (^) sign. Hence, the correct syntax for referencing this data tag will be `^Customers^Customer^Name`. This also appears as the text in the XML entry box in the record properties.

Printing a Record on a New Page

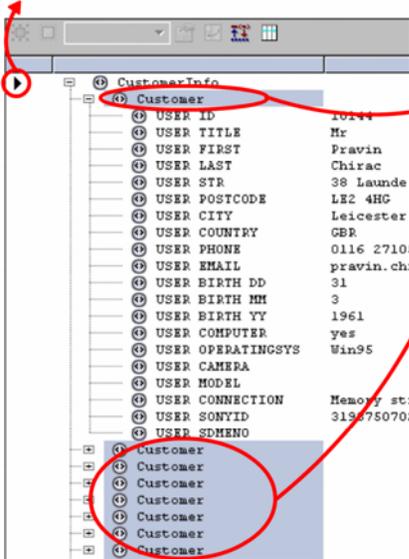
You can also choose to print a record on a new logical page using the record properties. Select the required record node from the project tree. From the **Settings** tab of the *Property View* window, select the **Page break on start** option to print the current record on a new page. This setting will have a similar effect as the page break setting applied at the **Page** level explained in the “Associating a Data Tag to a Page” task above.



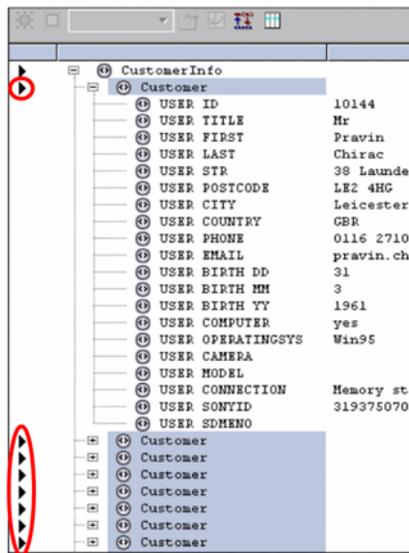
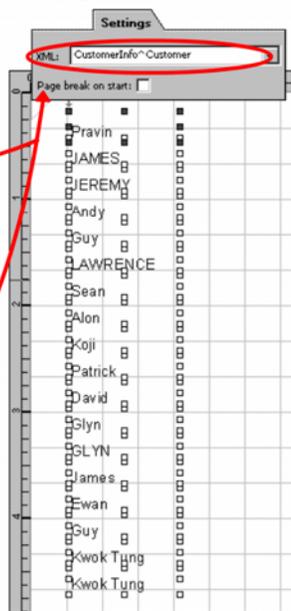
When you specify a page break at the record level, first instance of the page will not show the mapped data. This is because Vitesse will be processing two page breaks in this case; one specified at the page level and the other at the record level.

The image shows the effect of applying a page break at record level.

1. Page break marker indicating that a new page starts after the CustomerInfo tag is processed.

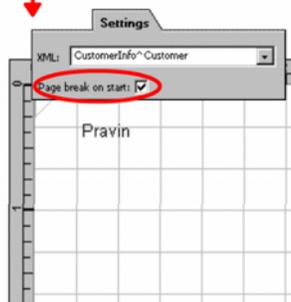


2. Map the Customer tag as a record on the page.



3. USER FIRST tag is mapped as a field to the project design. The USER FIRST tag value for all the Customer tags displays in the design area.

4. Select Page break on start option to start a new page for each instance of the Customer tag.



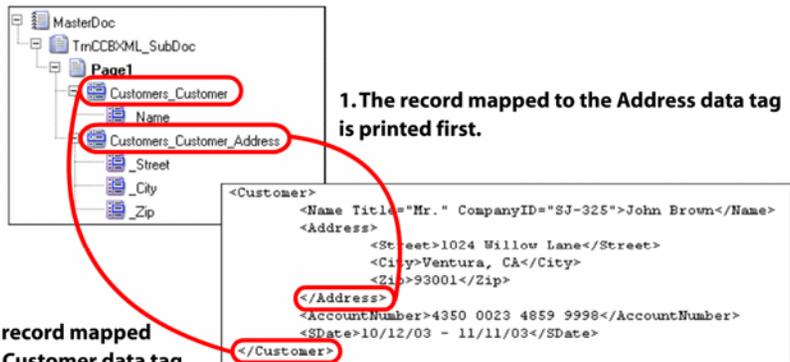
5. Each field displays on a new page in the project design.

6. Page break markers display with the Customer tag indicating that a new page starts after a Customer tag is processed.

Record Processing

Vitesse uses the end tags in the data file to determine the print order of records. When processing the XML data file, the first end tag encountered in the data file will process/print first.

The image below shows a project tree displaying the records inserted in a project design from the associated data file. In this example, Vitesse will first print the **Customers_Customer_Address** print, since its end tag is encountered first in the data file. The **Name** field will be one of the last printed fields, because it is a part of the **Customers_Customer** record. The end tag for this record comes later in the data file.



Mapping Fields

You can map a data tag value as a field object by either of the following ways:

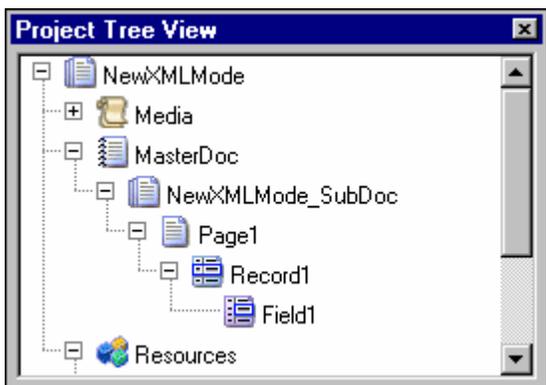
- Inserting a field object and specifying the required data tag in the record properties. The value of the specified data tag is mapped to the field.
- Dragging and dropping the required data tag value onto the design area.

Inserting a Field in XML Mode

To insert a field in an XML mode project:

- 1 Select the required **Record** node from the project tree.
- 2 Select Insert>Field from the menu bar; a **Field** node is created under the selected **Record** node in the project tree.

You can also add more field objects under a selected record using the same procedure.



Project tree displaying the newly inserted record.



You can also insert a field object either by:

- Clicking  on the *Insert* toolbar.
- Or
- Right-clicking the required record node, and then selecting *Insert>Field* from the pop-up menu.



You can delete a selected field object either by:

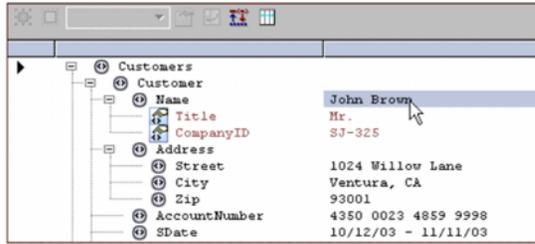
- Selecting *Edit>Delete* from the menu bar.
- Or
- Clicking  on the *Standard* toolbar.
- Or
- Pressing on the keyboard.

Mapping a Field to a Data Tag Value

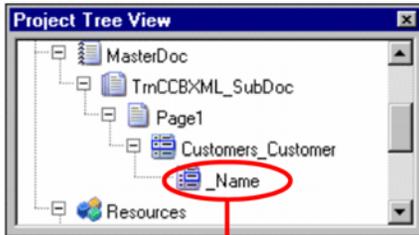
You can map a field to a data tag value by specifying the field data in the **Settings** tab of the field property sheet. The different options available for specifying field data have been explained separately for each of the field types, later in this chapter.

You can also drag and drop a tag value in the design area to create a mapped field object. Vitesse automatically keeps track of tag values (fields) that belong to a particular tag (record), and adds them to the project tree accordingly. In case, the tag value being dragged does not belong to any of the already created records, a new record will be created.

1. Select the 'Name' tag value from the Data View window and drag it.

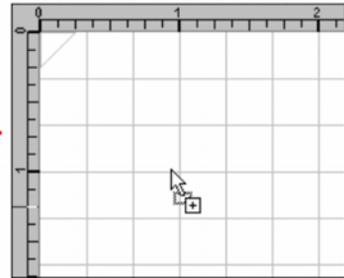


3. A field node is added to the project tree.



The field name is related to the tag name.

2. Drop the dragged tag value in the design area.



Steps involved in mapping a tag value to the project design using drag and drop.

You can also drag and drop the value of any tag attribute to the design area. Tag attributes are also mapped as fields in a project design. Tag attributes are identified by  icon in the data file.

Field Naming

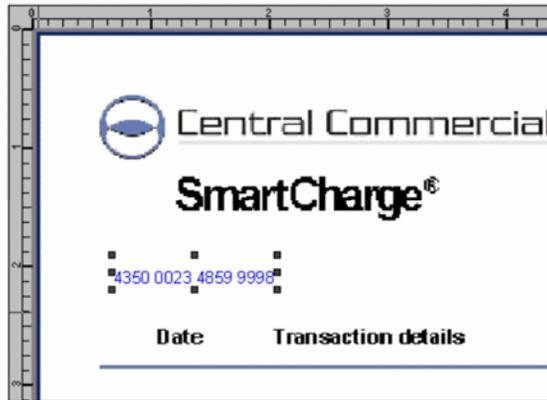
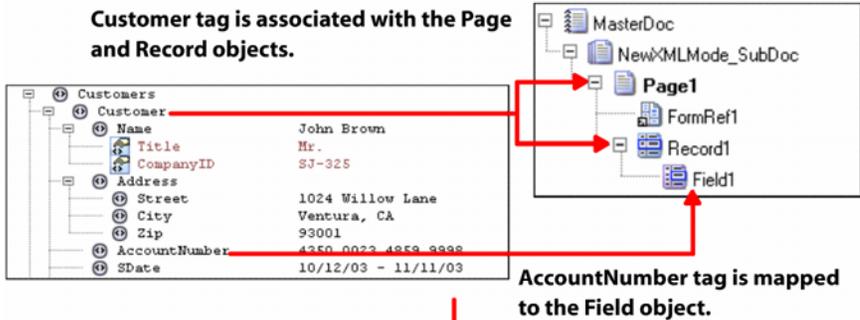
The name of the newly added field is the same as the associated tag name with an underscore (_) sign appended before it. The field name concatenated with the record name indicates the full XML path to reference the field value. VIPP uses (^) character as a separator between tag names. Hence, the full XML path used by VIPP to reference the **Name** field will be `^Customers^Customer^Name`.

When you map an attribute to the project design it is indicated by the @ character followed by the attribute name. For example, `@Title` will appear as the field name when you drag and drop the **Title** tag value to the design; and `^Customers^Customer^Name@Title` will reference the **Title** attribute of the **Name** tag.

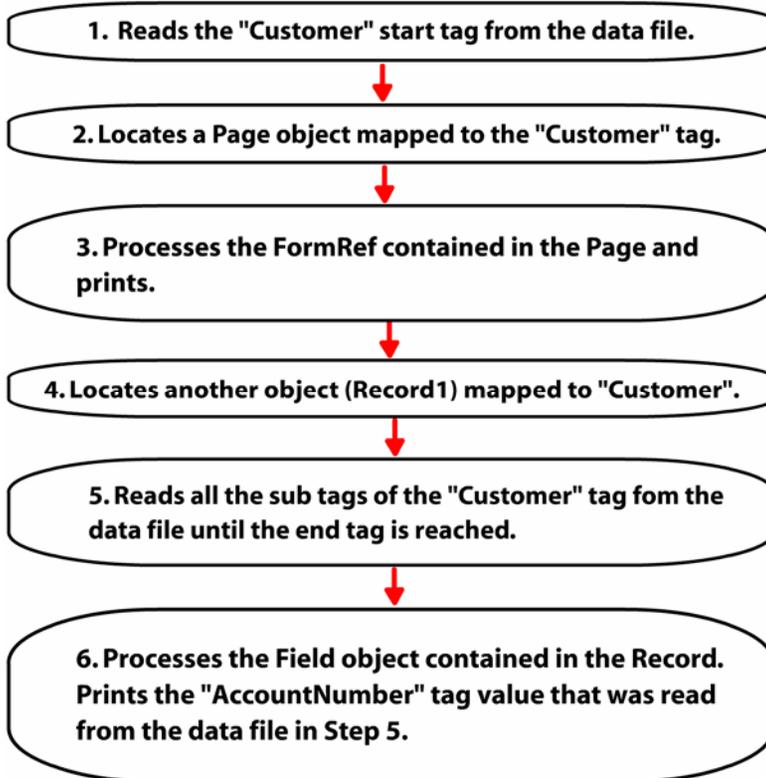
Field Processing

The objects inserted under a record are printed after the data tag mapped to this record has been processed from the data file. When a field is encountered in a record, the tag value associated with the field is printed according to the settings specified in the field property sheet (see the following tasks for more information).

The following image shows a data mapping defined for an XML mode project:



Vitesse generates the above print output following the XML processing order. The print order is also illustrated in the image below:



Using Normal Fields

Normal fields are the most commonly used field type. These fields can contain static text; data from variables or from the data file.

To set the field type as normal:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab..
- 3 Select **Normal** from the **Field Type** drop-down list.

Specifying the Normal Field Data

The data for the normal fields can be picked up from the data file or from a variable. Normal field data can also be specified using static text. This data can then be mapped directly to a page or used as a label or value for a data driven chart. Methods for using the data file, variables, and static text as normal field data are explained below:

Using the Data File

To specify the field data from the associated XML data file:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Data** which indicates that a data value from the data file will be associated with the current field. Selecting this option enables the associated drop-down list.
- 4 Click the drop down arrow in the **Data** entry box. It displays all the data tags from the data file associated with the project. The tags are shown in a tree structure according to their hierarchy in the data file.
- 5 Select the required data tag from the drop down list. You can expand a collapsed node by clicking \oplus adjacent to the node. You can also enter the required tag and its parent node in this box.

The selected data tag is displayed in the **Data** entry box. The value of the selected tag displays in the design area.



The parent record of the current field should be associated with the required data tag before specifying the field data, if you are not using the drag and drop approach for mapping data.

Using Variables

To specify the field data using variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable** which indicates that a variable will be used to specify the field data. Selecting this option enables the associated drop-down list.
- 4 Select the required system or user-defined variable from the drop-down list.

The field updates in the design area to display the variable value.



For more information on working with variables, see *Chapter 26: Using Variables*.

You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area. See “Using Variables” section in *Chapter 22: Working with Data in Line Mode*.

Using Constant Data

You can insert static text as field contents. To specify the field contents using static text in an XML mode project:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const** which indicates that constant data will be used as field contents; selecting this option enables the associated entry box.
- 4 Enter the required text in the **Const** entry-box.

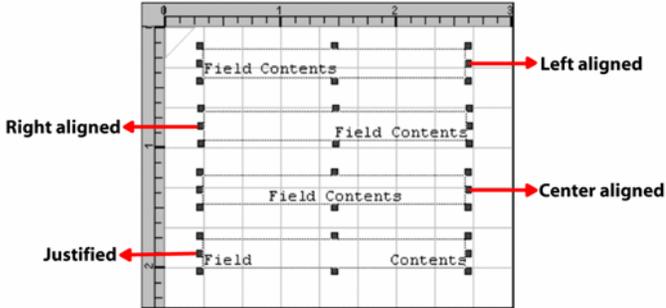
The field updates in the design area to display the text.

Aligning the Normal Field Contents

Alignment is the horizontal positioning of text within a line, relative to a point of reference. Vitesse provides various alignment options for field contents to achieve a more visually balanced design.

To align the field contents:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Left** – aligns the field contents along the left field margin.
 - **Right** – aligns the field contents along the right field margin.
 - **Center** – aligns the field contents symmetrically in the middle of the field.
 - **Justify** – aligns the field contents along both the left and right field margins by adjusting the spacing between the field data.

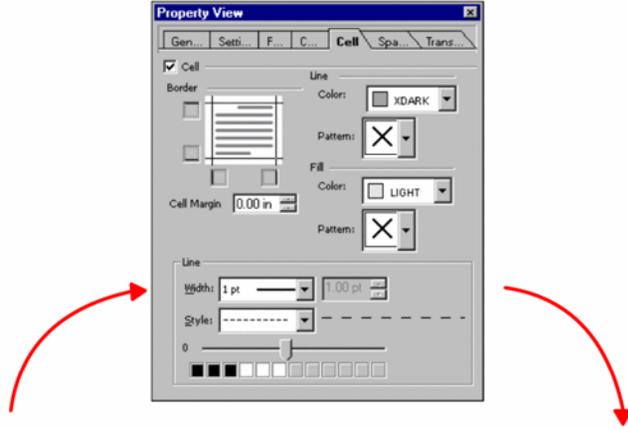


The design area reflects the effect of the basic alignment options.

Creating Dynamic Tables using Field Borders

You can customize the appearance of a normal or static text field by applying borders, fill colors, and patterns using the options in the **Cell** tab. Applying cell options to the transactional records creates a dynamic table. A dynamic table expands or contracts with the amount of data added. To create a dynamic table by setting the **Cell** properties associated with a field object, refer to the tasks in “Creating Dynamic Tables using Field Borders” in *Chapter 22: Working with Data in Line Mode*.

2. Specify the required cell options for each of the three fields to create a dynamic table.



Date	Transaction Detail	Amount
10/14	TexGas POS Transaction	10.00
10/14	CasaBella Interiors	12.99
10/15	Quadriini	22.55
10/16	Bay Leaf	5.00
10/16	D Watson Chemist	12.56
10/17	The Dollar Store	10.00
10/18	Icarus Records	12.72
10/19	Cafe Grind	10.34
10/20	Mom and Pop Hardware	13.54
10/20	The Cross Pharmacy	5.53
10/20	The Spa	30.00

1. Design area displaying a transactional record containing three fields.

Date	Transaction Detail	Amount
10/14	TexGas POS Transaction	10.00
10/14	CasaBella Interiors	12.99
10/15	Quadriini	22.55
10/16	Bay Leaf	5.00
10/16	D Watson Chemist	12.56
10/17	The Dollar Store	10.00
10/18	Icarus Records	12.72
10/19	Cafe Grind	10.34
10/20	Mom and Pop Hardware	13.54
10/20	The Cross Pharmacy	5.53
10/20	The Spa	30.00

3. Design area displaying a table created dynamically depending on the record data.

An example of a dynamic table created using field cell properties.

Setting Field Justification

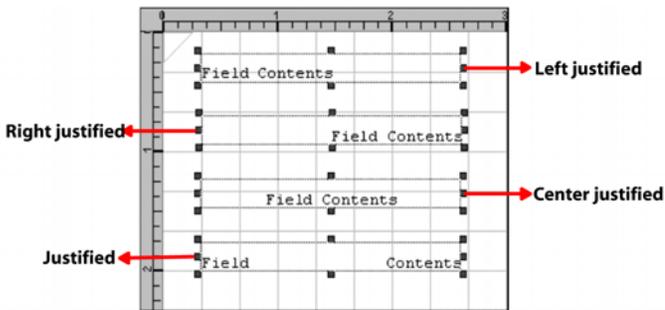
In Vitesse, you can align the field data relative to the left and right edges of the field margin. The data can be aligned to one edge (left or right) of the field margin, centered in the field object, or aligned to both edges of the field margin (full justification). Additional options to specify the vertical justification of the normal field contents to top, bottom, or center of the field object are also available.

To specify the field horizontal and vertical justification:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab

3 From the **Justification** area, select one of the following horizontal justification options:

- **Left** – click  to start data from left-most position and extend right with normal spacing.
- **Right** – click  to start data from right-most position and extend left with normal spacing.
- **Center** – click  to equidistance data from left and right.
- **Justify** – click  to start data from left-most position and end in right-most position; data is aligned evenly within the field object.



The design area reflects the effect of the horizontal justification options.

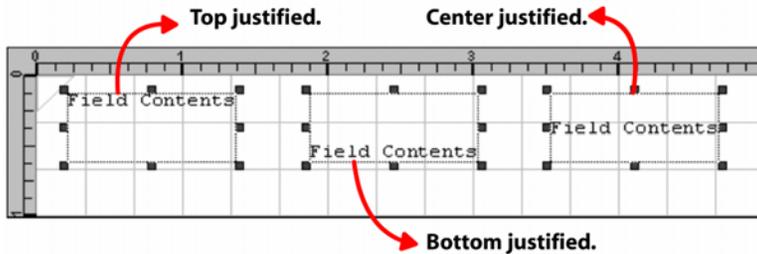


When working with **Normal** fields, specifying the horizontal justification options updates the **Alignment** drop-down list accordingly in the **Settings** tab to reflect the same settings and vice versa.

4 Select one of the following vertical justification options:

- **Top** – click  to align the field contents along the top field margin.
- **Bottom** – click  to align the field contents along the bottom field margin.
- **Center** – click  to align the field contents at the field center.

Both the horizontal and vertical field justification settings work together to produce the final output.



The design area reflects the effect of vertical justification options. In this example, the horizontal justification is set to *Left*.

Using Image Fields

An image resource can be added to an XML mode project using image fields. In an image field, the value of the field is the name of the image source file and its extension; this value can be picked from a data file or from a variable. The image is positioned relative to the bottom left edge of the field object. The specified image is added to the **Images** node under the **Resources** node in the project tree. You can position, align, scale, and rotate the added image field. All image formats supported by Vitesse can be inserted using image fields.



The image fields can only reference the images placed in the Vitesse global image library located at *drive:\Program Files\Elixir Technologies\Vitesse\ImgLib*. For more information on global resource folders see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as an image:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window,, select the **Settings** tab.,
- 3 Select **Image** from the **Field Type** drop-down list.

Specifying an Image Resource

The data that references the image resource can be picked up from the data file or a variable (that has the image file name and extension as its value). You can also use existing image resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The image resource specified using the data is then mapped to the page. Methods for using the data file and variables as an image field data are explained below:

Using the Data File

Vitesse enables you to customize your documents by using different images for different documents generated from a single project design. For example, you may require displaying a specific picture for each of the customers in their individual records. Vitesse lets you create such documents by the use of image fields that acquire data dynamically from the data file. The name and extension of the required image is available as a tag value. This tag value is then associated with the image field in the project design. The following image shows the steps involved in this process.

2. Specify the field data. Select the required data tag from the Data drop-down list.

1. Select Image from the Field Type drop-down list.

3. The image displays in the design area. It is also added to the project resources.

Customer	Title	Name	Picture	Address	Street	City	Zip
Mr. John Brown	Mr.	John Brown	John Brown.tif	1024 Willow Lane	Ventura, CA	93001	

The image shows the image field data being specified using the XML data file.



Before setting the **Field Type** as an **Image**, ensure the data file that you use must have the required image file names and extensions.

Using Variables

To specify the image field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable** which indicates that a variable will be used to specify the image field data. Selecting this option enables the associated drop-down list. The variables used for referencing images store the name and extension of the image resource.
- 4 Select the required system or user-defined variable from the drop-down list.
- 5 The field updates in the design area to display the referenced image.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area. See “Using Variables” section in *Chapter 22: Working with Data in Line Mode*.

Using an Existing Project Resource as Field Data

You can use existing image resources under the **Resources>Images** tree node for the image field.

To specify the image using the project resources:

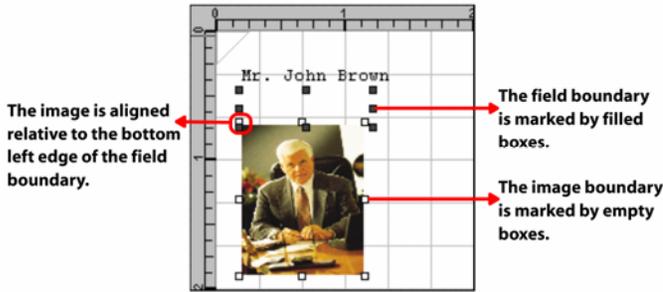
- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled
- 4 Select the required image resource from the drop-down list. The drop-down lists all the images used in the project resources. The selected image displays in the design area.



Using the **Const** option, you can also add images not present in the project resources by entering the image name and extension in the entry box.

Aligning an Image Field

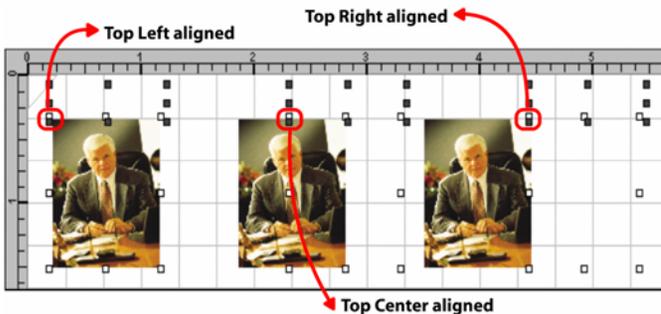
You can position an image relative to a point of reference. The referenced image is always positioned relative to the bottom left edge of the field bounding box. Vitesse provides various alignment options for image fields to achieve a more visually balanced design.



The image highlights the point of reference for various image alignment options.

To align the image field:

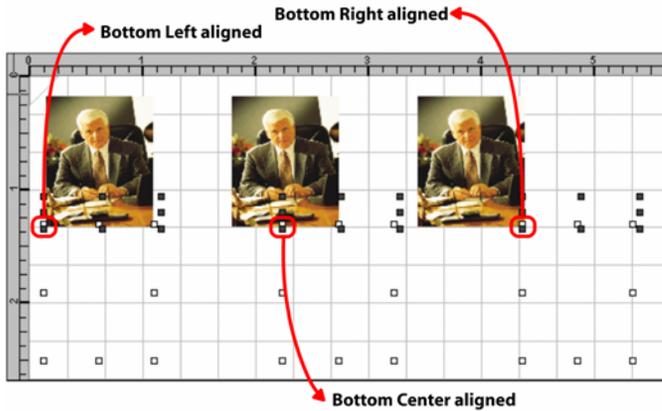
- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Top Left** – aligns the top left edge of the image along the bottom left edge of the field.
 - **Top Right** – aligns the top right edge of the image along the bottom left edge of the field.
 - **Top Center** – aligns the top center point of the image along the bottom left edge of the field.



The image shows the effect of the top image alignment options.

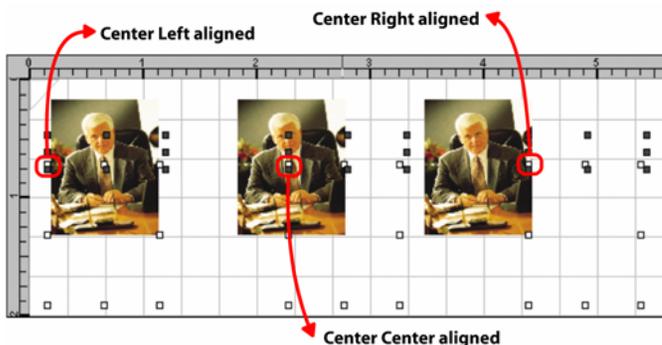
- **Bottom Left** – aligns the bottom left edge of the image along the bottom left edge of the field.

- **Bottom Right** – aligns the bottom right edge of the image along the bottom left edge of the field.
- **Bottom Center** – aligns the bottom center point of the image along the bottom left edge of the field.



The image shows the effect of the bottom image alignment options.

- **Center Left** – aligns the center left edge of the image along the bottom left edge of the field.
- **Center Right** – aligns the center right edge of the image along the bottom left edge of the field.
- **Center Center** – aligns the center of the image along the bottom left edge of the field.



The image shows the effect of the center image alignment options.

Using Segment Fields

A segment resource can be added to an XML mode project using segment fields. In a segment field, the value of the field is the name of the segment source file and its extension; this value can be picked from a data file or from a variable. The segment is positioned relative to the bottom left edge of the field object. The inserted segment is added to the **Segments** node under the **Resources** node in the project tree. You can position, scale, rotate, and dynamically size the added segment. All segment formats supported by Vitesse can be inserted using the segment fields.



The segment fields can only reference the segments placed in the Vitesse global segment library located at *drive:\Program Files\Elixir Technologies\Vitesse\SegLib*. For more information on global resource folders see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set the field type as a segment:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 Select **Segment** from the **Field Type** drop-down list.

Specifying a Segment Resource

The data that references the segment resource can be picked up from the data file or a variable (that has the segment file name and extension as its value). You can also use existing segment resources in the current project (under the **Resources>Images** tree node) for referencing as field data. The segment resource specified using the data is then mapped to the page. Methods for using the data file and variables as a segment field data are explained below:

Using the Data File

Vitesse enables you to dynamically specify the segment resource used in a page design using the data file. For an XML data tag, you can specify the name and extension of the segment as its value. You can then associate the respective data tag value with the required segment field. On document composition, Vitesse acquires the segment name from the data file, references the required segment from the **SegLib** directory, and displays it in the document design. This functionality enables you to customize your documents. For example, you can create a variety of documents for different clients using different segments.

2. Specify the field data using the Data option. Select the required data tag from the drop-down list.

1. Select Segment from the Field Type drop-down list.

The image shows a multi-step process in a software application. At the top, a 'Data' window displays a tree view of XML data. The 'Customers' folder is expanded, and the 'Seg' tag is selected. Below this, a 'Field Type' drop-down menu is shown with 'Segment' selected. To the right, a 'Data View' window displays a table of data for the selected 'Seg' tag, with the 'TrnMsg. evs' field highlighted. At the bottom, a design area shows a rounded rectangular box containing text: 'Welcome to Central Commercial! Watch this space each month. We will have new Banking opportunities and information that will help you. Thank you again for choosing Central Commercial.' A red arrow points from the 'Seg' tag in the 'Data View' to this text box. A red arrow also points from the 'Segment' option in the 'Field Type' menu to the 'Seg' tag in the 'Data' window.

Customers	TrnMsg. evs
Customer	Ms.
Title	Celine Marchbank
Name	
Address	
Street	1024 Willow Lane
City	Ventura, CA
Zip	93001

3. The segment displays in the design area. It is also added to the project resources.

The image shows the segment field data being specified from an XML data file.



Before setting the **Field Type** as a **Segment**, ensure the data file that you use must have the required segment file names and extensions.

Using Variables

To specify the segment field data using predefined variables:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Variable** which indicates that a variable will be used to specify the segment field data. Selecting this option enables the associated drop-down list. The variables used for referencing segments store the name and extension of the segment resource.
- 4 Select the required system or user defined variable from the drop-down list.
The field updates in the design area to display the referenced segment.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required field data using a variable by selecting **Const** from the **Field Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value in the design area. See “Using Variables” section in *Chapter 22: Working with Data in Line Mode*.

Using an Existing Project Resource as Field Data

You can use existing segment resources under the **Resources>Segments** tree node for the segment field.

To specify the segment using the project resources:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Field Data** area, select **Const**; associated drop-down list is enabled.
- 4 Select the required segment resource from the drop-down list. The drop-down lists all the segments used in the project. The selected segment displays in the design area.



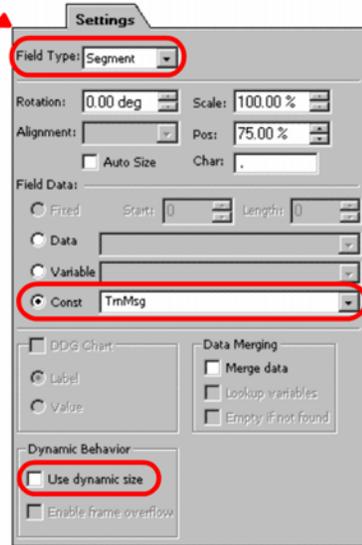
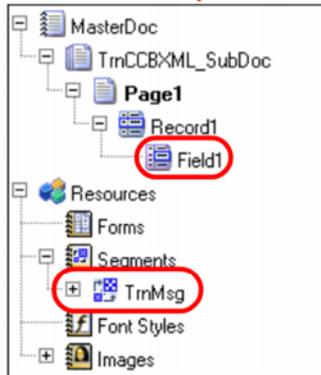
Using the **Const** option, you can also add segments not present in the project resources by entering the segment name and extension in the entry box.

Using the Dynamic Size Behavior in a Segment Field

By default, the size of segment referenced in a page or a form is the same as the size of original segment. However, Vitesse also enables you to adjust the segment size dynamically depending on its contents. For example, the contents of a segment resource displaying in the design may vary depending on the document data or conditions applied. In such cases, the dynamic sizing option adjusts the height of the segment automatically.

The following image shows the dynamic sizing behavior for segment fields:

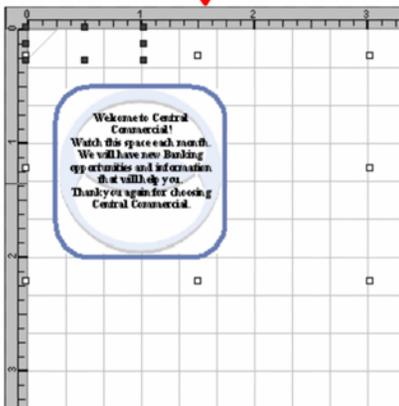
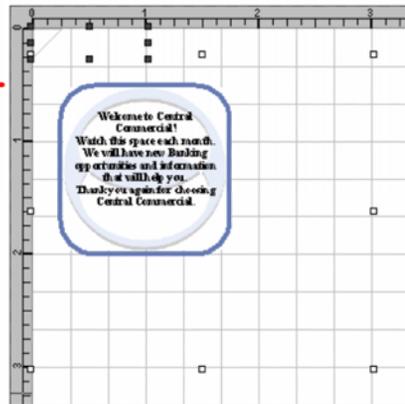
1. Segment field used to reference a segment in the project design.



3. Select Use dynamic size to adjust the segment height according to its contents.



2. Design area displays the segment referenced using a field object.



4. Design area displays the dynamically sized segment.



For more information, refer to the task “Dynamically Sizing Segments” in *Chapter 32: Using Segments in Projects*.

Using Static Text Fields

Static text fields extend the functionality of normal fields by providing formatting options (that are the same as available for the text and graphic text object) in addition to all the normal field functionality. However, there are a few exceptions:

- You cannot set the alignment options for static text fields.
- Also, the static text field data cannot be used as a label and value for a data driven chart (See *Chapter 28: Adding Charts* for more information).

To set the field type as a static text:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select **Static Text** from the **Field Type** drop-down list.

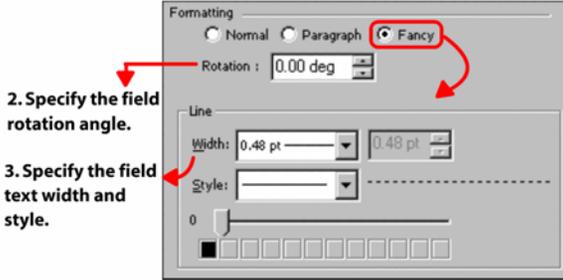
Formatting Static Text Field Data

You can choose to apply text formatting features such as text wrapping, rotation, line style, and width to static text fields.

To apply formatting options:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Spacing** tab.
- 3 From the **Formatting** area, select one of the following options:
 - **Normal** – applies normal formatting to the field data; any text extending the field boundary is not wrapped.
 - **Paragraph** – applies paragraph formatting to the field data; any text extending the field boundary is wrapped.
 - **Fancy** – applies enhanced formatting features to the field text; selecting this option displays the associated options for setting the line width, style, and rotation.

1. Select Fancy, the associated options are displayed.



The image shows the process of applying fancy formatting.



For more information on setting rotation, line width, and line style, see *Chapter 22: Working with Data in Line Mode*.



The line width and style settings are not visible in the design area; however, these are applied when the project is exported.

Specifying Field Data

The data for the **Static Text** fields can be picked up from the data file or from a variable. Static field data can also be specified using static text in the **Const** entry box. This data cannot be used as a label or value for a data driven chart. For specifying the field data using the data file, variables, and constant text, refer to the “Specifying Field Data” task for **Normal** fields earlier in this chapter.

Merging XML Data with Static Text

Vitesse enables you to merge XML data from the data file to the project design, by specifying the required data tag value in the field properties.

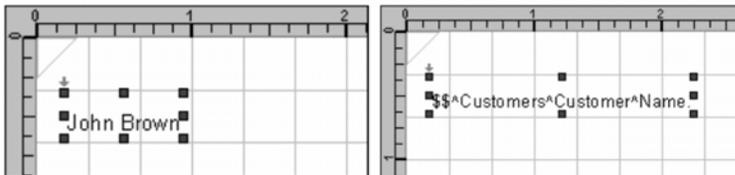
To merge XML data with static text:

- 1 Select **Const** from the **Field Data** area to enable the adjacent entry box.
- 2 Enter the correct syntax (`$$DataTagName.`) for specifying the required XML data tag.
- 3 From the **Data Merging** area, select **Merge Data** to view the specified data tag value in the design area.

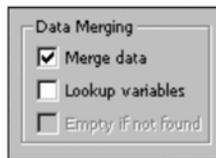
1. From the Field Data area, select the Const option.



2. Specify the required XML data tag using the correct syntax.



3. Select the Merge data option to view the specified tag value in the design area.



Setting Common Field Properties

Some field properties are common to all the four field types. These include the field size, position, spacing, rotation, and auto-sizing.

Positioning a Field

Field objects can be placed at an absolute position in the design area by specifying the horizontal and vertical coordinates. You can also position the field objects relative to one another.

You cannot position fields in the design area using record property sheet **Position** options; instead, you must use the field property sheet.

Setting Field Absolute Position

To accurately position a field object in the design area:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the field horizontal position, per specified unit of measurement in the **X** entry box. In the associated drop-down list, select **Absolute**.
- 4 Specify the field vertical position, per specified unit of measurement in the **Y** entry box. In the associated drop-down list, select **Absolute**.

Using the **Y** coordinate **Absolute** position option is recommended only if each field occurs once on each page; if a field has multiple instances, using the **Y** coordinate **Absolute** position option will cause field instances to overlap. See “Setting Field Relative Position” task for more information.

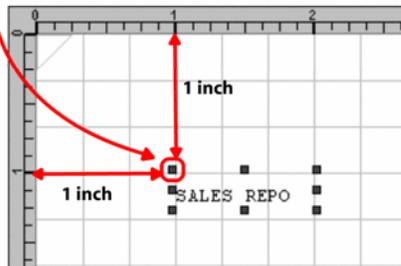


You can also set the unit of measurement for the field position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

The absolute coordinates are calculated from the top left margin of the page.



The field positioned at absolute horizontal and vertical coordinate values of 1 inch from the top left margin.



The image shows the process of setting field position using absolute coordinates. In this example, the top, left margins are set at 0, 0 inches.

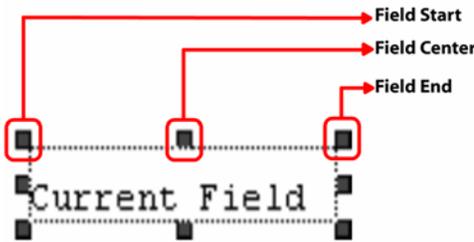
Setting Field Relative Position

A field object in a record can be positioned relative to the previous fields and relatively positioned objects added to the record. In absolute positioning, the top left margin of the page acts as the point of reference, whereas in relative positioning, the previously added object is taken as the point of reference.

Relative positioning is helpful in the placement of transactional records, for example, the credit card transactions for a customer. As the total number of transactions for any customer is unknown, the transaction information cannot be positioned absolutely when designing the project else, the data will overprint.

To position fields using relative positioning:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify **0** in the **X** entry box. In the associated drop-down list, select one of the following horizontal relative positioning options:



The horizontal relative positioning options are applied with reference to the field horizontal start, center, and end points.

- **Start Same** – positions the start of the current field at the start of the previous field.
- **Start Center** – positions the start of the current field at the center of the previous field.
- **Start Next** – positions the start of the current field at the end of the previous field.

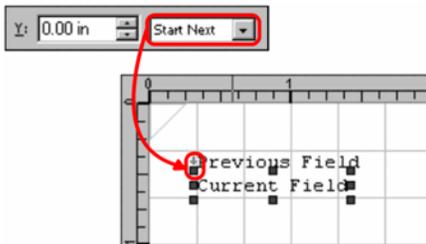


In this example, the current field is positioned at the previous field end.

- **Center Before** – positions the center of the current field at the start of the previous field.
 - **Center Same** – positions the center of the current field at the center of the previous field.
 - **Center Next** – positions the center of the current field at the end of the previous field.
 - **End Before** – positions the end of the current field at the start of the previous field.
 - **End Center** – positions the end of the current field at the center of the previous field.
 - **End Same** – positions the end of the current field at the end of the previous field.
- 4 Specify **0** in the **Y** entry box. In the associated drop-down list, select one of the above-mentioned vertical relative positioning options. Vitesse provides the same set of vertical relative positioning options as the horizontal relative positioning options.



The vertical relative positioning options are applied with reference to the field vertical start, center, and end points.



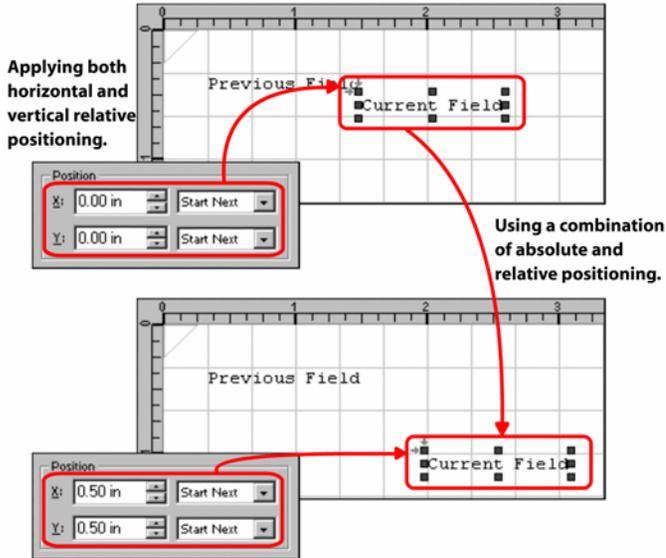
In this example, the current field is positioned at the previous field end.



Horizontal relative positioning options cannot be applied to the first field object added to a record.



Using a combination of absolute and relative positioning, you can place the field relative to another field or a relatively positioned object and add the spacing defined by the absolute X and Y coordinates.



In this example, the current field is positioned relatively at Same Next, Same Next and then displaced 0.5, 0.5 inches in the horizontal and vertical directions.

The following symbols appear next to a relatively positioned field node in the *Project Tree View* window:



-  - specifies that horizontal relative positioning is applied to the field.
-  - specifies that vertical relative positioning is applied to the field.
-  - specifies that both horizontal and vertical relative positioning is applied to the field.



The relative positioning options are applicable for all objects that can be relatively positioned in Vitesse.

Setting the Field Size

Field dimensions specify the width and height of the field. To set the field dimensions:

- 1 Select the required **Field** node from the project tree.
- 2 In the **General** tab of the *Property View* window, specify the following **Size** information:
 - **W** – sets the field width per specified unit of measurement.
 - **H** – sets the field height per specified unit of measurement.



You can also set the unit of measurement for the field by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units > unit name** from the pop-up menu.

The  icon in the **Size** area is used for maintaining the aspect ratio of the field. Clicking this icon and then setting the width and height increases or decreases the length and width of the field in equal proportion.

1. To increase field width and height in equal proportions, click .

2. Specify either field width or height value.

3. The field size increases proportionally based on the specified width or height value.

The image shows the process of changing the field size while maintaining the aspect ratio.



You can also set the field size by dragging the field handles that define the boundary of the field object in the design area. The corresponding **Size** settings are updated in the *Property View* window.

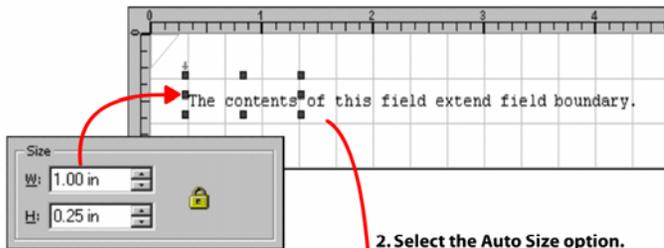
Adjusting the Field Size to its Contents

Vitesse enables you to auto-size field objects to adjust automatically to the amount of data it is displaying. This helps prevent overlapping of fields that are placed relative to each other.

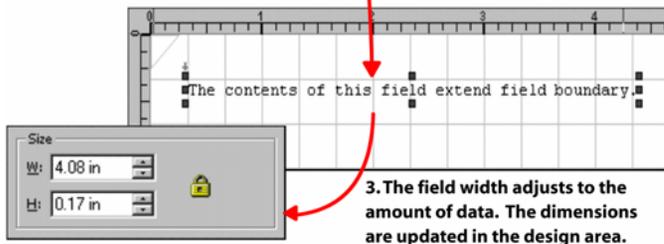
To automatically adjust field size to the amount of data:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Auto Size** check-box. The field width automatically adjusts to fit all the data.

1. The contents of the field extend the current field width.



2. Select the Auto Size option.



3. The field width adjusts to the amount of data. The dimensions are updated in the design area.

The image shows the process of adjusting field size to its contents.

Setting Spacing between Fields

You can set the vertical spacing between the different instances of the same field object. This option is applicable when working with data tags having multiple instances mapped on to the same page.

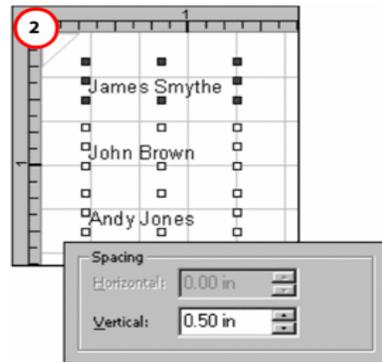
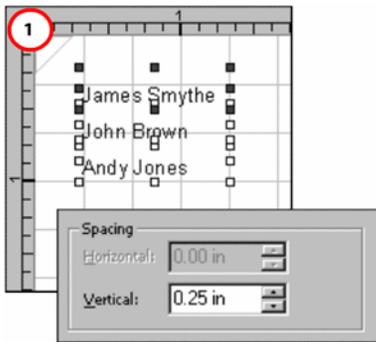
To set the vertical field spacing:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Spacing** area, specify the following:
 - **Vertical** – specifies the vertical spacing value of the field.

You cannot specify the horizontal spacing value, when working with XML data.



You can also set the unit of measurement for the field spacing by right clicking the upward or downward arrows in the **Horizontal** and **Vertical** entry boxes, and then selecting **Units>unit name** from the pop-up menu.



The design area reflects the vertical spacing set to: (1) 0.25 inches (2) 0.50 inches.

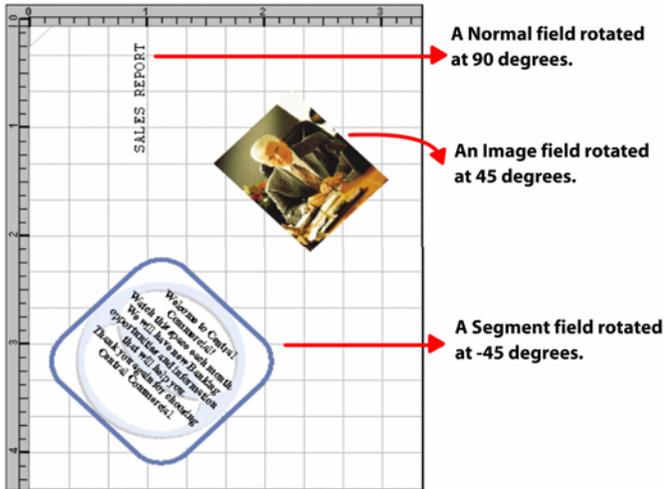
Setting Field Rotation

To set the field rotation:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required rotation angle in the **Rotation** entry box.



You can specify the angle in degrees, radians, or grads. To specify the unit of measurement for the rotation angle, right click the upward or downward arrows in the **Rotation** entry box, and then select **Angle>unit name** from the pop-up menu.



The image shows different rotations applied to various types of fields.



When working with normal or static text fields, you can also specify the rotation angle by selecting the **Fancy** option in the **Spacing** tab, and then specifying the rotation angle in the **Rotation** entry box.

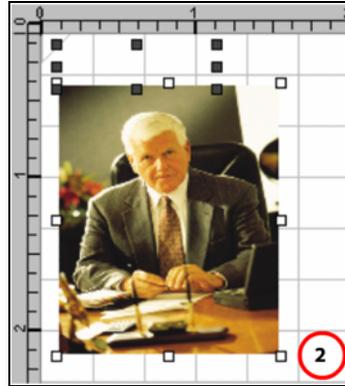
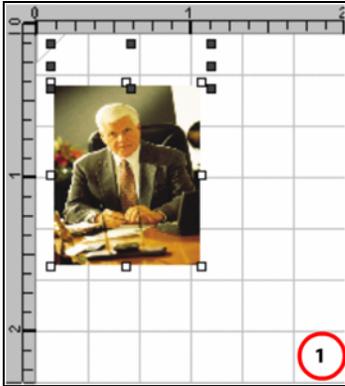
Scaling Image and Segment Fields

The image and segment resources added to the project as fields are scaled at 100% by default. This means that the original image and segment dimensions are used. Once added to the project, you can scale the image and/or segment resource to suit your requirements.

To scale image and/or segment fields:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.

- 3 Specify the required scaling value as a percentage, in the **Scale** entry box.



The design area reflects an image field scaled to (1) 100% (2) 150%.



- For information on the **Color** tab, refer to *Chapter 11: Working with Colors*.
- For information on the **Font** tab, refer to *Chapter 6: Working with Fonts*.
- For information on the **Transform** tab, refer to *Chapter 10: Applying Transformations*.

Section IX: Using Data Driven Objects in a Project

User Notes:

Chapter 25: Using Frames

This chapter describes activities and tasks for working with frames in Vitesse. Included are instructions and information for the following:

- Understanding Frames
- Understanding Segment Overflow Behavior
- Creating a Frame
- Relatively Positioning Objects in Frames
- Using Text or Field Objects in Frames
- Enabling Segment Overflow
- Using Conditions in Frames
- Enabling Segment Overflow
- Using SegRef in Frames
- Using Frame Numbers
- Working with Frames in XML Mode

Understanding Frames

Frames are virtual containers that control the page layout by defining areas on a page where text, data, or objects are to be placed before actually placing an object on the page. This gives you greater control of your page layout and makes potential modifications to the project easier.

Adding frames to a Vitesse project enables the linked frames mode. The linked frames enable you to automatically flow text and objects from one frame to the next using as many linked frames as needed to hold all of the objects.



All frames that you add in a project are automatically linked to each other, thus creating a linked frames mode.

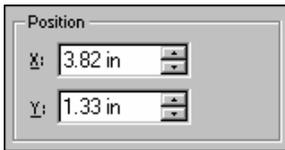
Frames are especially helpful when working with large amounts of text data. For example, if you have a long selection of text that you want to place in different locations on the same page, you can use two linked frames for this purpose. Linked frames are essential for creating any document with multiple columns of text, or several pages of related text. Frames can also be useful in newsletter or brochures design, in which an article starts on one page and continues on the next.

When working with linked frames, text and objects flow into the next frame once the current frame is full. When the last frame on a page is filled, a page break occurs automatically and the next object flows to the first frame on a new page.

Understanding Relative and Static Objects

Vitesse enables you to use two types of objects, in terms of their placement in the design area when designing a project or a resource (form and segment). These include relative and static objects:

- Static objects that can only be positioned using absolute coordinates. These include the drawing objects such as line, path, polygon, box, and table.



The property sheets for the static drawing objects only display the absolute positioning options.

- Relative objects that can be positioned using both absolute and relative positioning options. These include the text and graphic text drawing objects as well as field, chart, barcode, image, and segment objects.



The property sheets for the relative objects display both the absolute and relative positioning options.



For more information on absolute and relative positioning, see “Positioning a Field” in *Chapter 22: Working with Data in Line Mode*.

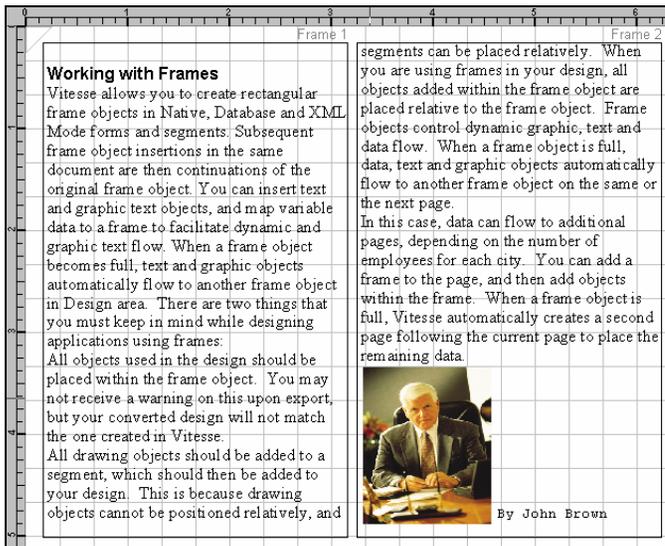
Working with Frames in Vitesse

Vitesse enables you to create rectangular frame objects for designing the page layout. We recommend that you create all of the frames before placing an object on the page. The sequence in which the frames are created determines the flow of data between the frames. The content flows from Frame1 to Frame2 and so on, depending on the amount of content and number of frames. This sequence cannot be modified. When all of the frames on the page have been used, Vitesse creates a new instance of the page, and the placement of objects again starts from Frame1.

Frames operate in relative positioning mode. Due to the overflow behavior of frames, the vertical coordinates of the objects added to a frame cannot be set to an absolute value.

In the linked frames mode, any relative object added to the page becomes a part of the frame. The first object is positioned relative to the top left edge of the frame, and all subsequent objects are positioned relative to their previous objects. These objects move with the frame if it is repositioned. Any static drawing objects added to the page are not part of the frame, as these objects cannot be relatively positioned.

It is a recommended practice to use forms for any recurrent data on the page.



The design area displays the use of frames for creating a two column layout.



Vitesse enables frame processing only in Native, Database, and XML mode projects.

The frame appears just like a box object when inserted in the design area, and you can select, move, and arrange the frame like other design objects.



Frames are properties of the page object. If you right-click a frame object and use the **Copy** and **Paste** option from the pop-up menu, you will actually create a copy of the page object, which is indicated by the new **Page** node in the project tree.

Understanding Segment Overflow Behavior

Static drawing objects can not be added to frames, as these objects cannot be relatively positioned. You can, however, add static objects to a frame using segments that can be placed relatively.

Segments are resources that can be considered as small forms containing objects and will be used repeatedly in a project design. Vitesse enables you to have segment contents overflow from one frame to another. If the contents of a segment do not fit in one frame, the rest of the contents automatically flow to the next frame. If the current frame is the last frame on the page, then the rest of the contents are placed on the first frame of a new page.

To enable the segments to overflow, the vertical size of the segment needs to be dynamic so that it changes with the amount of contents.



For more information on segment overflow, see the “Enabling Segment Overflow” section later in the chapter.

Creating a Frame

To create a frame object:

- 1 Create a new Database, Native, or XML mode project.
- 2 Select the required **Page** node from the project tree.
- 3 Click the frame tool, , on the *Drawing* toolbar. The mouse pointer changes from  to .
- 4 Click and drag the mouse anywhere on the design area to create the frame.
- 5 The frame is added to the design area and a corresponding entry is added to the frame table under the **Frame** tab in the *Property View* window of the **Page** object.

Creating a Frame Using the Property Sheet

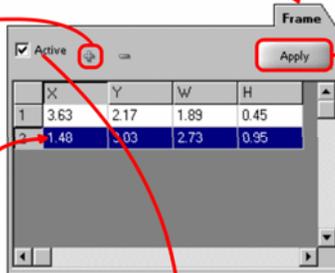
Once the first frame has been created using the frame tool, all of the frame options in the **Frame** tab are enabled. You can also use these options to create more frame objects.

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Frame** tab.
- 3 To add a new frame object, click  in the **Frame** tab. A new entry is added to the frame table.
- 4 Enter the required information by double clicking the cells in the newly added row.
- 5 Click **Apply**.
- 6 The frame with the specified settings displays in the design area.

1. Creating a frame using the frame tool enables all the options in the **Frame** tab.

2. Click  to add more frames to the page.

3. Double click the cells to modify the frame position and dimensions.



4. Click **Apply**.

The **Active** options activates or deactivates frame processing for a page.

The image shows the process of adding a frame using the frame property sheet.



You can also delete a frame entry by selecting the required row, and then clicking  in the **Frame** tab.

Using Borders to Preview Frames

Frames display in the design area within borders. You can customize the frame borders by modifying the border line width and style. The border settings are applied to all the frames used in the design area. The frame borders are not part of the final exported output.

Specifying Frame Border Width

Vitesse provides various options to change the frame border width.

To set the frame border width:

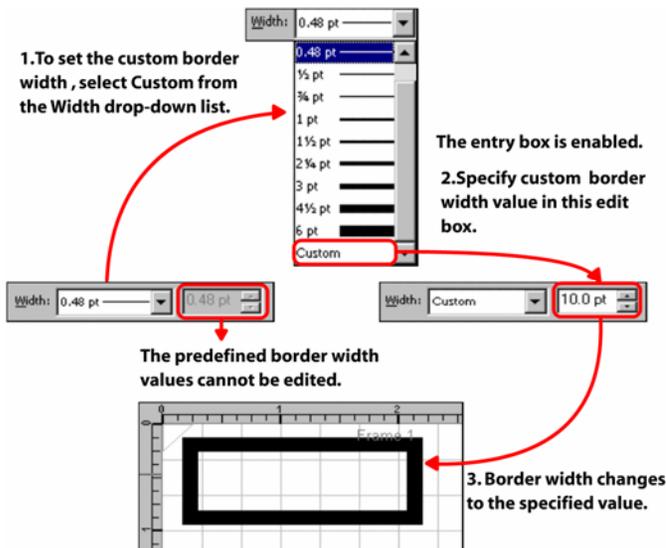
- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Frame** tab.
- 3 From the **Preview border** area, select the required width option from the **Width** drop-down list. Adjacent entry box displays the width value associated with the selected option.
- 4 To set a width other than the predefined values, select the **Custom** option from the **Width** drop-down list. This option enables the adjacent entry box.



You can also set the unit of measurement for the width by right clicking the upward or downward arrows in the entry box, and then selecting **Units>unit name** from the pop-up menu.

- 5 Enter the required frame border width value.

The following image shows the process of setting the custom frame border width.

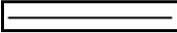


Specifying Frame Border Style

Along with changing the frame border width, you can also apply different styles to the frame border. For example, you can make the border appear as dotted or dashed, etc.

To set the frame border style:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Frame** tab.
- 3 From the **Preview border** area, select one of the following style options from the **Style** drop-down list:

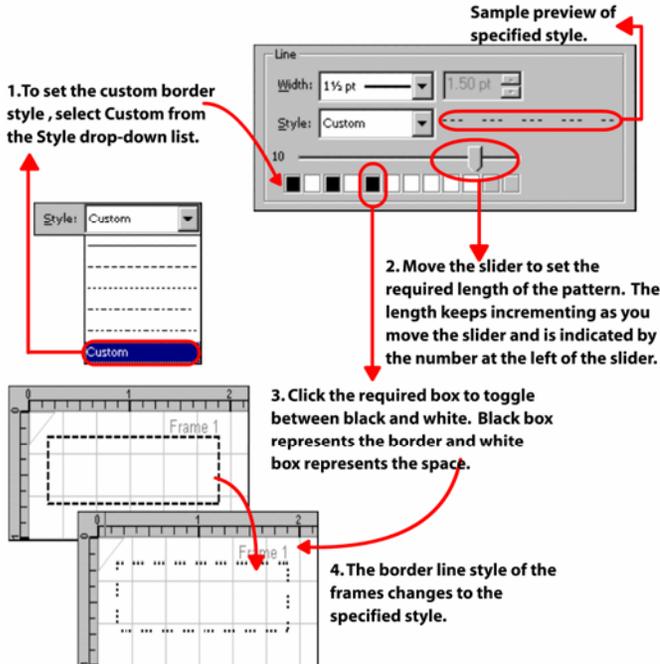
Line Style	Sample
Solid	
Dashed	
Dotted	
Dash-Dotted	
Dash-Dot-Dotted	
Custom	Sets a customized style for a line. For example  .

The area adjacent to the **Style** drop-down list shows the sample preview of the selected style.

The black and white boxes  in the **Preview border** area depict the frame border style pattern. The box in black represents the border and the box in white represents the line spacing used in the border style. The pattern keeps changing, depending on the **Style** selection. For example, if you select the dash-dotted line style, the pattern changes to . The first box is always black.

- 1 To set a style other than the predefined values, select the **Custom** option from the **Style** drop-down list.
- 2 Move the slider to set the custom border style pattern length, and then click the required box to toggle from black to white.

The image below shows how to set the custom frame border style.



Changing Frame Placement and Size

When creating frames using the frame tool, , the frame horizontal and vertical coordinates, and dimensions are displayed in a new row in the **Frame** tab.

When creating the frame using the property sheet, the frame coordinates and dimensions need to be specified. Each newly created frame inherits the settings of the previous frame, with the X coordinate incremented to 0.5 unit of measurement.



When you add a frame using the property sheet, it is added at an increment of 0.5 units along the x-axis from the previously selected frame object in the design area. You need to move the newly added frame object to clearly view it in the design area.

These values can be modified any time by double clicking the corresponding cells in the newly added row, or by moving and resizing the frame in the design area.

To change the frame placement and size options:

- 1 Select the required **Page** node from the project tree.
- 2 From the *Property View* window, select the **Frame** tab
- 3 In the frame table, select the row corresponding to the required frame object.
- 4 Specify the following position and dimension options for the selected frame by double clicking the row cells under the corresponding columns:
 - **X** – specifies the horizontal coordinate of the top left edge of the frame, per specified unit of measurement.
 - **Y** – specifies the vertical coordinate of the top left edge of the frame, per specified unit of measurement.
 - **W** – specifies the frame width, per specified unit of measurement.
 - **H** – specifies the frame height, per specified unit of measurement.



You can also set the unit of measurement for the frame position and dimensions by right clicking the upward or downward arrows in the **X**, **Y**, **W**, and **H** entry boxes, and then selecting **Units**>*unit name* from the pop-up menu.

- 5 Click **Apply**.

The frame updates in the design area to reflect the updated settings.



If you do not click **Apply**, the new frame settings will not be applied.

The image below shows the process of changing frame position and dimensions.

1. Select the frame in the design area, the cursor changes to .

2. Click and drag the cursor to the required location.

3. The frame top left corner X and Y coordinates are updated in the Frame tab. You can also modify the values by double clicking the cells under the respective columns.

4. Any relative objects in the frame, move with the frame, but the static objects are not displaced from their absolute coordinates, unless they are added using segments.

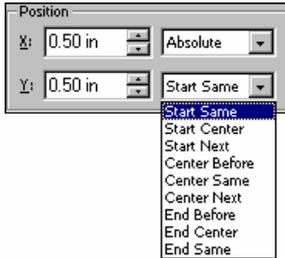
5. You can also modify the frame dimensions by moving the cursor over any of the frame handles. The cursor changes to . Click and drag the cursor to the required location.

	X	Y	W	H
1	0.31	0.32	2.02	1.34

	X	Y	W	H
1	0.67	0.67	2.02	1.34

Relatively Positioning Objects in Frames

Due to the dynamic behavior of the frame objects, the absolute Y coordinate cannot be specified for any relative objects placed in the frame. Therefore, the **Absolute** option is not provided for the relative objects in the linked frames mode.



The vertical relative positioning options for an image object placed in a frame. The vertical positioning cannot be set to absolute, however, the horizontal position can be set to an absolute value.

The same restriction also applies to objects placed under a segment, which is to be used with frames as shown in the image below:

1. A relative object can be placed using absolute vertical coordinates in a segment not used with frames.

2. To use the segment in a frame, enable the frame overflow behavior for the segment, by selecting the corresponding option from the segment property sheet.

3. The absolute vertical positioning option is removed from the drop-down list.

The diagram illustrates the process of using absolute vertical coordinates in a segment. It shows a grid with an image of a man in a suit. A red arrow points from the 'Absolute' option in the Y-axis dropdown of the Position property sheet to the image. Another red arrow points from the 'Enable frame overflow' checkbox in the Dynamic Behavior property sheet to the image. A third red arrow points from the 'Start Same' option in the Y-axis dropdown of the Position property sheet to the image.

Using Text or Field Objects in Frames

Large amount of text can be inserted in a design using text or field objects. The formatting of the text or field objects that contain a large amount of text should be set to **Paragraph**.



Refer to *Chapter 22: Working with Data in Line Mode* for details on formatting field objects. For more information on formatting text objects, see *Chapter 9: Using the Text Tools*.

For paragraph type text or paragraph type field, the vertical relative positioning options are restricted to **Start Same**, **Start Center**, or **Start Next**. Out of all the relative positioning options, these three are the only ones which are not dependent on the vertical size of the text or field object. The size of the text or field object can not be determined/set in advance when the object is expected to overflow from one frame to another.

The paragraph type text or field objects can overflow across multiple frames. Vitesse does not allow data overflow to a previous frame; therefore, vertical positioning of the first relative object that needs to be placed after a paragraph type text or field is set to **Start Next** by default so that the object is placed where the text or field object ends. For remaining objects, following the first relatively positioned object, the vertical relative positioning options are enabled (with the exception of the **Absolute** option).

The image below displays the vertical relative positioning option for the first object after a text or field object in a frame.

The image shows a software interface with a tree view on the left containing 'MasterDoc', 'NewDBM_SubDoc', 'Page1', 'Text1', and 'ImageRef1'. The main area displays a frame with two columns of text. The left column is titled 'Working with Frames' and discusses Vitesse's frame objects. The right column continues the text and includes a photo of a man in a suit with the caption 'By John Brown'. Below the frame are two 'Position' property windows. The first window shows X: 0.04 in and Y: 0.04 in, with a dropdown menu set to 'Start Next'. The second window shows X: 0.11 in and Y: 0.43 in, also with a dropdown menu set to 'Start Next'. Red arrows point from the text in the frame to the 'Start Next' options in the property windows. A separate red arrow points from the tree view to the first text column.

An image object positioned after the text.

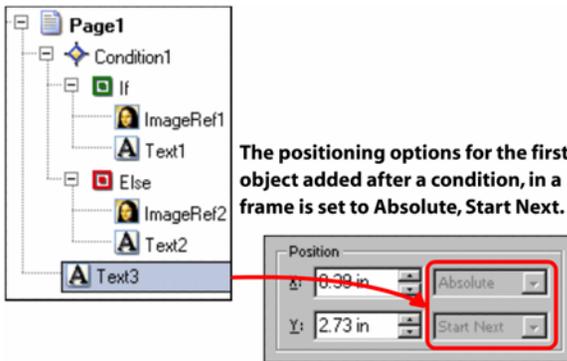
Paragraph type text object spanning two frames.

Vertical relative positioning options for a text field.

Vertical relative position can only be set to Start Next for the first object after a paragraph type text or field.

Using Conditions in Frames

You can use conditional logic to place objects in a frame. For the first relative object placed after a condition in frames, the horizontal relative position is set to an **Absolute** value and the vertical relative positioning is set to **Start Next** by default. This is due to the fact that the next object can only start from where the last object in the condition ends. The last object placed conditionally can be a text or field object split across multiple frames. As Vitesse does not allow you to go to a previous frame, **Absolute** and **Start Next** are the only positioning options available for the first object following a condition, as shown in the image below:



For more information on working with conditions, see *Chapter 27: Adding Conditions*.

Enabling Segment Overflow

A Vitesse segment is designed using static and dynamic objects such as images, composed text data, charts, barcodes, as well as lines, paths, polygons, and boxes. You can also use pre-designed PostScript segments (*.ps, *.eps, *.prn) in a Vitesse project. Vitesse also allows you to import VIPP Segments (*.seg) to the Vitesse environment.

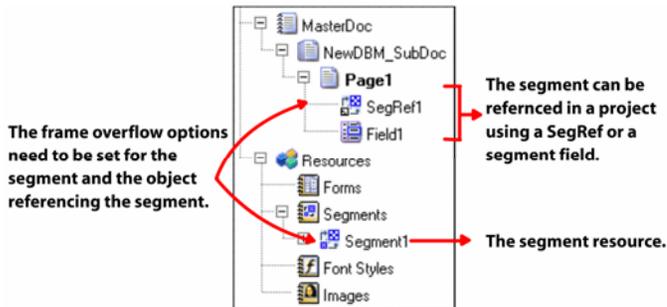
Vitesse enables the segments to flow across frames. For the segments to flow across frames, the vertical size of the segment needs to be dynamic, i.e., it should increase or decrease depending on the amount of data. All segment formats supported by Vitesse can be set to overflow in the linked frames mode except PS segments.



For more information on working with segments, see *Chapter 5: Working with Segments*.

To enable the segment overflow functionality, you need to:

- Set frame overflow options in the segment that is inserted under the project resources (**Resources**>**Segments** tree node).
- Set frame overflow options in the object referencing the segment resource in the project. The segment can be referenced using a SegRef object or a field object set to the **Segment** type.



The image highlights the objects whose properties need to be set up to enable segment overflow in frames.



Refer to *Chapter 32: Referencing Segments* for details on inserting segments in projects. For more information on using segment type fields, refer to *Chapter 20: Working with Data in Database Mode*.



Repeat, Inline, End Page, and Begin Page segments cannot be set to overflow in the linked frame mode.

Enabling Frame Overflow in Segments

To enable frame overflow in segments:

- 1 Select the required **Segment** node under the **Resources** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Dynamic Behavior** area, select the **Enable frame overflow** option; selecting this option automatically selects the **Enable dynamic sizing** option.

Enabling Frame Overflow in Objects referencing the Segment

After setting up the segment overflow options in the segment resource, the corresponding options also need to be setup in the object referencing the segment resource in the project to enable the segment to overflow across frames.

Enabling Frame Overflow in SegRef

To enable frame overflow for segments that are referenced using a SegRef object:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Dynamic Behavior** area, select the **Enable frame overflow** option; selecting this option automatically selects the **Enable dynamic sizing** option.



If a **SegRef** is created by dragging and dropping the segment from the project resources to the target node, then the newly created SegRef will inherit the **Enable frame overflow** value of the segment resource.

Enabling Frame Overflow in Segment Fields

To enable frame overflow for segments inserted using a segment field:

- 1 Select the required **Field** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Dynamic Behavior** area, select the **Enable frame overflow** option; selecting this option automatically selects the **Enable dynamic sizing** option.

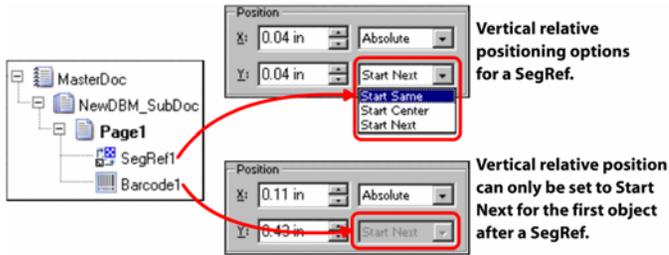


You can view the **Dynamic Behavior** options of the field object by panning the property sheet if the options are not initially visible.

Using SegRef in Frames

A SegRef is used for referencing a segment in a project. For the SegRef objects with the frame overflow feature enabled (inherited from the **Segment** resource), the vertical relative positioning options are restricted to **Start Same**, **Start Center**, or **Start Next** only. Out of all the relative positioning options, these three are the only ones which are not dependent on the vertical size of the SegRef object. The size of the SegRef object cannot be pre-determined when the segment contents overflow from one frame to another.

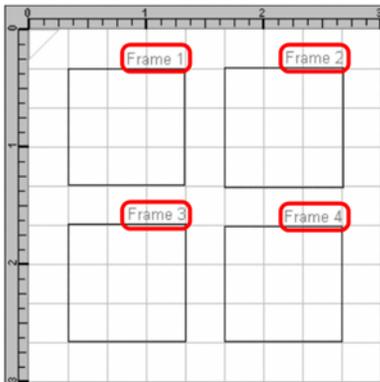
A segment object can overflow across multiple frames. Vitesse does not allow data overflow to a previous frame. Therefore vertical positioning of the first relative object that needs to be placed after a SegRef is set to **Start Next** by default so that the object is placed where the SegRef object ends. For remaining objects, following the first relatively positioned object, the vertical relative positioning options are enabled (with the exception of the **Absolute** option).



The image displays the vertical relative positioning options available for a SegRef with frame overflow enabled and the first object following a SegRef.

Using Frame Numbers

Vitesse helps the users to visualize the flow of data in frames by numbering each frame. The frame numbers display at the top right corner of the frame. These numbers are automatically generated and cannot be modified.



The frame numbers display in the design area.

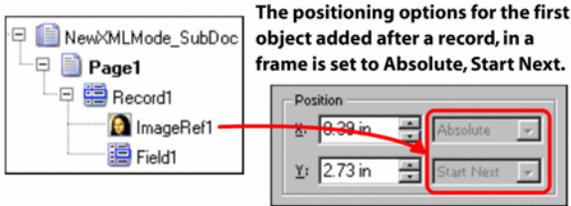
Working with Frames in XML Mode

In an XML mode project, objects can only be added to frames after a **Record** object has been added and defined for a design. Any objects added under a record node become part of the frame. Unlike the database and native mode projects, the objects cannot be directly added to a page.



For more information on defining records in an XML mode project, see *Chapter 24: Working with Data in XML Mode*.

The relative positioning option for the first object added after a record in a frame is restricted to **Absolute** and **Start Next** by default, as shown in the image below:



Chapter 26: Using Variables

This chapter describes activities and tasks for working with variables in Vitesse. Included are instructions and information for the following:

- What is a Variable?
- Understanding Variable Types
- Creating a New Variable
- Working with System Variables
- Using Persistent Variables
- Using Variables for Referencing Forms
- Using Variables for Referencing Segments
- Using Variables for Referencing Images
- Using Variables in Forms and Segments

What is a Variable?

A variable is a keyword that represents a value that is subject to change throughout the print job. A variable can be considered a container that stores information. You can either use pre-defined application variables or define your own variables using the **MasterDoc** node property sheet.

Variables are supported in all the four project modes. You can use variables to store calculation results and for numbering rows in dynamic tables. Variables can also be used in conditional logic, for creating data driven charts, and to perform simple arithmetic calculations and book marking. Images, segments, and form fields can also be referenced using variables.

Understanding Variable Types

Vitesse supports four variable types, which are used for storing different types of data. Different variables can be used to perform different actions depending on the type. The variable types include:

- **String** variables contain alphanumeric character strings. These variables can be used for setting field or barcode values, for referencing resources, and for comparisons in conditions.
- **Number** variables contain numeric values. These variables can be used for setting field or barcode values and performing calculations.
- **Array** variables contain a list of values. The values can be set to numbers, strings, system variables, and user-defined variables defined for the project. Array variables are used for defining label value pairs for a data driven chart.

- **Bool** variables can have either one of the two values: true or false. These variables are useful in conditions for making two way decisions.

Creating a New Variable

You can define a list of variables for your project using the **MasterDoc** property sheet. These variables can then be used in the project as field or barcode contents, chart data, and for referencing resources. You can also use the variables for making calculations and comparisons.



All variable names must begin with the **VAR** prefix. The variable names are case sensitive; using **Var** or **var** instead of **VAR** will produce an error when the exported project is opened in VIPPIDE or the print controller, as variables not starting with the **VAR** prefix are not recognized by VIPP.

The **Add** button is used to create variables, whereas the **Del** button is used to remove a selected variable. See tasks below for more information.

Creating a String Variable

To create a string variable:

- 1 Select the **MasterDoc** node in the project tree.
- 2 In the **Variables** tab of the *Property View* window, enter the name of the variable in the **Name** drop-down list. The variable name can contain alphanumeric characters without any spaces.
- 3 From the **Type** drop-down list, select **String**.
- 4 Enter the string variable value in the **Value** entry box. A string variable can contain alphanumeric characters as well as spaces. You can also choose not to specify a default value for the variable.
- 5 Click **Add**.

A new string variable is created, which can later be used in the project design.

Example: Using a string variable as a field value

You can set string, number, and bool variables as a field value. The following image displays the process of using a string variable as a field value:

1. Create a string variable.

2. Click , on the Insert toolbar to create a field object.

3. In the field property sheet, select the Settings tab.

4. Select Variable, and then select the string variable from the drop-down list.

5. The variable value is assigned to the field and displays in the design area.



For more information on using variables as a field value, see *Chapter 22: Working with Data in Line Mode*.

Creating a Number Variable

To create a number variable:

- 1 Select the **MasterDoc** node in the project tree.
- 2 In the **Variables** tab of the *Property View* window, enter the name of the variable in the **Name** drop-down list. The variable name can contain alphanumeric characters without any spaces.
- 3 From the **Type** drop-down list, select **Number**.
- 4 Enter the number value in the **Value** edit box. A number variable can only contain numeric characters, decimal point, and a leading '+' or '-' sign. For example, a number variable value can be set to -123.45. You can also choose not to specify a default value for the variable.
- 5 Click **Add**.

A new number variable is created, which can later be used in the project design.

Example: Using a number variable as a barcode value

You can use both number and string variables as barcode data. The following image displays a number variable being used as data for a barcode object:

The image shows a sequence of steps in a software interface. At the top, the 'Variables' tab is active, showing a variable named 'VARbarcode' of type 'Number' with a value of '100101011'. Below this, the 'Settings' tab for a barcode object is shown, with 'Bar Code Types' set to 'CODE39' and 'Data' set to 'Variable' with 'VARbarcode' selected. At the bottom, a barcode is displayed in the design area, with the value '* 1 0 0 1 0 1 0 1 1 *' printed below it. Red arrows and numbered text boxes explain each step.

1. Create a number variable.
2. Click , on the Insert toolbar to create a barcode object.
3. In the barcode property sheet, select the Settings tab.
4. Select Variable, and then select the number variable from the drop-down list.
5. The variable value acts as barcode data and displays in the design area.



For more information on working with barcodes, see *Chapter 30: Adding Barcodes*.

Creating an Array Variable

Array variables are used for creating data driven graphics (for example, charts). To create an array variable:

- 1 Select the **MasterDoc** node in the project tree.
- 2 In the **Variables** tab of the *Property View* window, enter the name of the variable in the **Name** drop-down list. The variable name can contain alphanumeric characters without any spaces.
- 3 From the **Type** drop-down list, select **Array**; selecting this option displays an empty table in the **Value** area.
- 4 Click to add an entry to the array. An array can contain any number of entries.
- 5 Enter a value in the newly added cell/entry box or click in the cell and select the appropriate option from the sub-menu. The value can be one of the following:
 - **Variables** – select a predefined variable from the list as the value. The list contains all the variables defined for the current project.
 - **System Variables** – select a system variable from the list as a value. The list contains all the system variables defined for Vitesse.
 - **Constants** – select this option to specify either a number or text value.



You can also delete the last added value by clicking in the **Variables** tab.

- 6 Click **Add**.
A new array variable is created, which can be used in the project design.

The image shows the process of defining an array variable and its values.

1. Specify the name of the array variable in the Name drop-down list.

2. Select Array from the Type drop-down list. An empty table is displayed in the Value area.

The number of values in the array is indicated by the number at the top of the table.

3. Click  to add an entry to the array.

Constant text specified as an array value.
Constant number specified as an array value.
You can also use system and user defined variables as array values.

4. Select the type of value to be specified, either variable, system variable, constant text, or constant number.

5. Specify the value.

6. Once all the array values have been specified, click Add.

Example: Using an array variable for creating DDG Charts

The following image displays the process of creating a data driven chart using an array variable:

1. Create an array variable.

2. Specify the chart label as constant text.

3. Specify the chart value as constant number.

4. Create all the key value pairs. Labels should always be placed before values, following a strict sequence of Text (Label), Number (Value); Text, Number and so on.

5. Click , on the Insert toolbar to create a chart object.

6. In the chart property sheet, select the Data tab.

7. Select Variable, and then select the array variable from the drop-down list.

8. The array values added as text display as chart labels and the values added as numbers display as chart values.

Label	Value
a	10.00
b	5.00
c	15.00



For more information on working with charts, see *Chapter 28: Adding Charts*.

Creating a Boolean Variable

To create a boolean variable:

- 1 Select the **MasterDoc** node in the project tree.
- 2 In the **Variables** tab of the *Property View* window, enter the name of the variable in the **Name** drop-down list. The variable name can contain alphanumeric characters without any spaces.
- 3 From the **Type** drop-down list, select **Bool**.
- 4 In the **Value** area, click one of the following to set the variable value:
 - **True** – sets the default variable value to true.
 - **False** – sets the default variable value to false.
- 5 Click **Add**.

A new boolean variable is created, which can later be used in the project design.



You can also delete a variable by selecting the required variable from the **Name** drop-down list, and then clicking **Del** in the **Variables** tab.



You can modify a variable value at any time. Modifying a variable type clears the previously set value. You cannot modify a variable name; making any modifications to a variable name creates a new variable. Click **Add** to apply the modifications.

Working with System Variables

Vitesse provides a list of system variables. These variables provide information about the project. The values change dynamically based on where they are inserted in a design. You can use these variables in a project design, but the value of a system variable cannot be modified. Following is a list of the available system variables in Vitesse:

Variable Name	Supported Modes	Description
PRODUCT	All modes	Identifies the PostScript interpreter on which VIPP is running (in this case, the interpreter is Vitesse).
XMLDTH	XML	Provides the depth of the current XML node in the XML tree.

Variable Name	Supported Modes	Description
XMLPATH	XML	Provides the path of the current node by listing the hierarchy of all the parent nodes. For example CustomerInfo^Customer^USER_ID is the path for the node USER_ID.
XMLPAR	XML	Provides the name of the parent node for the current node.
XMLVAL	XML	Provides the contents/value of the current node.
XMLTAG	XML	Provides the name of the current node.
XMLATV	XML	Provides the XML attribute value.
XMLATN	XML	Provides the XML attribute name.
YINIT	Line	Provides the current vertical position of the field.
XGFVER	All modes	Displays the VIPP release number.
VDISP	All modes	Provides the displacement between the current vertical print position and the secondary print position, saved by the VIPP SAVEPP command.
SVPOS	All modes	Provides the vertical print position saved by the VIPP SAVEPP command.
SHPOS	All modes	Provides the secondary horizontal print position saved by the VIPP SAVEPP command.
RPLEFT	All modes	Provides the number of remaining iterations (including the current one) of the REPEAT loop. This variable is used with repeat segments.
RPCOUNT	All modes	Provides the number (starting at 1) of the current iteration of the REPEAT loop. This variable is used with repeat segments.

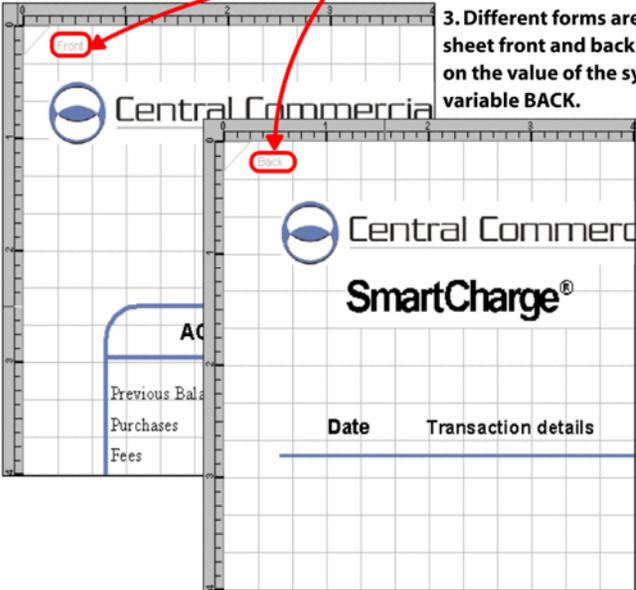
Variable Name	Supported Modes	Description
PPCOUNT	All modes	Provides the current absolute physical page number from the beginning of the job.
LSP	Line, Database	Provides the current line spacing value.
LPINDEX	All modes	Provides the index of the current logical page on the current physical page. The value ranges from the 1 to the number of partitions on the page.
LPCOUNT	All modes	Provides the current logical page number.
LNCOUNT	Line	Provides the number of data lines on the current page.
HDISP	All modes	Provides the displacement between the current secondary horizontal print position and those saved by the VIPP SAVEPP command.
HCOLOR	All modes	Selects the current highlight color on DocuPrint NPS Highlight Color systems. HCOLOR defaults to black on other printers.
FRLEFT	Native, Database, XML	Provides the amount of vertical space available from the current print position to the bottom of the current frame. The value is expressed in the specified units.
FRCOUNT	Native, Database, XML	Provides the current frame number. Frames are numbered starting from 1. A value of 0 indicates that the page contains no frame.
DJDEPAR	Line	Represents a DJDE (Dynamic Job Descriptor Entry) parameter that has been extracted from a DJDE record by the VIPP PROCESSDJDE command. PROCESSDJDE simplifies processing DJDE (LCDS data

Variable Name	Supported Modes	Description
		stream) jobs by eliminating DJDEs from the printable data and calling a user-defined procedure for each keyword/parameter pair in the DJDE line.
DJDECMD	Line	Represents a DJDE (Dynamic Job Descriptor Entry) keyword that is extracted from a DJDE record by the VIPP PROCESSDJDE command. PROCESSDJDE simplifies processing DJDE (LCDS data stream) jobs by eliminating DJDEs from the printable data and calling a user-defined procedure for each keyword/parameter pair in the DJDE line.
CURLINE	Line	Provides the line number of the current line being processed by an RPE.
CPCOUNT	All modes	Provides the current copy number. You can use this variable to perform specific actions on a particular copy pass.
COLW	All modes	Provides the current column width.
BACK	All modes	This variable is set to true when the current page is on the back side of the current sheet and false when it is on the front side of the sheet.

Example: Using system variable BACK to switch forms for a sheet front and back

The following image displays the use of a system variable BACK to switch forms for the sheet front and back, using conditional logic:

1. Click  on the Insert toolbar to add a condition to a page.
2. Set the condition to check the system variable BACK, if the value is false (sheet front), the summary form is printed. If the system variable value is true (sheet back), the details form is printed.
3. Different forms are printed on sheet front and back depending on the value of the system variable BACK.



For more information on working with conditions, see *Chapter 27: Adding Conditions*.

Using Persistent Variables

In Vitesse, the variables exhibit two types of behavior:

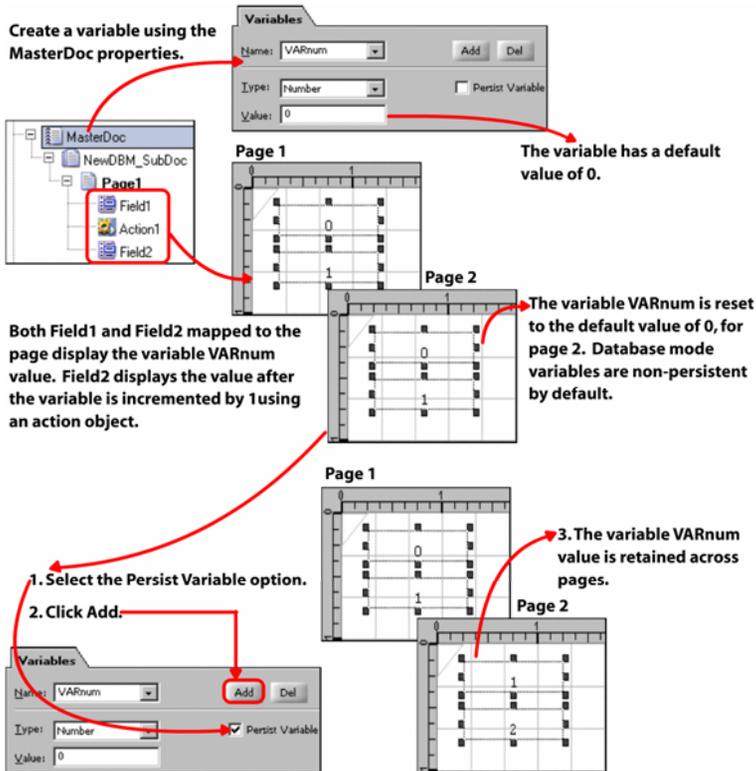
- Non-Persistent variables are reset to their default value for each newly created page.
- Persistent variables retain their value, even when a new page is created.

In the Line, Native, and XML mode projects, the variables show persistent behavior by default. On the other hand, in Database mode projects, the variables are non-persistent by default. You can, however, set a variable to show persistent behavior in a database mode.

To set a database mode project variable as persistent:

- 1 Select the **MasterDoc** node in the project tree.
- 2 From the **Variables** tab of the *Property View* window, select the required variable from the **Name** drop-down list.
- 3 Select the **Persist Variable** option.
- 4 Click **Add**.

The selected variable value is retained across pages.



The image shows the behavior of persistent and non-persistent variables in Vitesse.

Using Variables for Referencing Forms

You can use the variables defined in your project for referencing form resources. The variables used for referencing forms store the name and extension of the form resource as their value.

To use a variable for referencing a form:

- 1 Select Insert>Form from the menu bar to add a form to a page.

The *Insert Form* dialog is displayed.



You can also open the *Insert Form* dialog by clicking  on the *Insert* toolbar.

- From the *Insert Form* dialog, select either:
 - An existing form from the **Existing** tab, and click **Open**.
 - Linked**, or **Ref** form from the **New** tab, and click **New**.

A FormRef is created under the **Page** node and the corresponding existing or newly created form is added to the project resources.



Inline forms cannot be referenced using variables.

- Select the **Settings** tab from the **FormRef** property sheet.
- From the **Data** area, select **Variable**; selecting this option enables the associated drop-down list.
- Select the required variable containing the required form name and extension from the drop-down list. You can choose from a list of system and user defined variables.

The form specified using the variable displays in the design area, and is added to the project resources.

1. Create a variable that holds a form name, using the MasterDoc properties.

2. In the FormRef property sheet, select the required variable.

3. The form displays in the design area and is added to the project resources.

The image shows the process of using variables for referencing forms.



Using variables, you can only specify form resources placed in the Vitesse global form library located at *drive:\Program Files\Elixir Technologies\Vitesse\FormLib*. For more information on global resource folders see *Chapter 3: Understanding the Vitesse Directory Structure*.

Using Variables for Referencing Segments

You can use the variables defined in your project for referencing segment resources. The variables used for referencing segments store the name and extension of the segment resource as their value. You can specify variables that reference segments using either:

- Segment fields (when the field type is set to **Segment**)



For more information about using variables with segment fields, see *Chapter 22: Working with Data in Line Mode*.

- SegRef objects

To specify a variable to reference a segment in a **SegRef** object:

- 1 Select **Insert**>**Segment** from the menu bar to add a segment object to a page. The *Insert Segment* dialog is displayed.



You can also open the *Insert Segment* dialog by clicking  on the *Insert* toolbar.

- 2 From the *Insert Segment* dialog, select either:
 - An existing segment from the **Existing** tab, and click **Open**.
 - **Linked** or **Ref** from the **New** tab, and click **New**.

A SegRef is created under the **Page** node and the corresponding existing or newly created segment is added to the project resources.



Inline, Begin Page, End Page, and Repeat segments cannot be referenced using variables.

- 3 Select the **Settings** tab from the **SegRef** property sheet.
- 4 From the **Data** area, select **Variable**; selecting this option enables the associated drop-down list.
- 5 Select the required variable containing the required segment name and extension from the drop-down list. You can choose from a list of system and user defined variables.

The segment specified using the variable displays in the design area, and is added to the project resources.



Using variables, you can only specify segment resources placed in the Vitesse global segment library located at `drive:\Program Files\Elixir Technologies\Vitesse\SegLib`. For more information on global resource folders see *Chapter 3: Understanding the Vitesse Directory Structure*.

Using Variables for Referencing Images

You can use the variables defined in your project for referencing image resources. The variables used for referencing images store the name and extension of the image resource as their value. You can specify variables that reference images using either:

- Image fields (when the field type is set to **Image**)



For more information about using variables with image fields, see *Chapter 22: Working with Data in Line Mode*.

- ImageRef objects

To specify a variable to reference an image in an **ImageRef** object:

- 1 Select **Insert>Image** from the menu bar to add an image object to a page.

The *Insert Image* dialog is displayed.



You can also open the *Insert Image* dialog by clicking  on the *Insert* toolbar.

- 2 Select the required image from the **Existing** tab of the *Insert Image* dialog.

- 3 Click **Open**.

An ImageRef is created under the **Page** node and the corresponding image is added to the project resources.

- 4 To change the referenced image, select the **Settings** tab from the ImageRef property sheet.

- 5 In the **Data** area, select **Variable**; selecting this option enables the associated drop-down list.

- 6 Select the required variable containing the image name and extension from the drop-down list. You can choose from a list of system and user defined variables.

The image specified using the variable displays in the design area, and is added to the project resources.



Using variables, you can only specify image resources placed in the Vitesse global image library located at `drive:\Program Files\Elixir Technologies\Vitesse\ImgLib`. For more information on global resource folders, see *Chapter 3: Understanding the Vitesse Directory Structure*.

Using Variables in Forms and Segments

Vitesse enables you to add variables in forms and segments using a predefined variable list. As an administrator, you can create an XML file, in which you define the variables to be used in a form or segment design. The following image depicts the structure of an XML file containing the variables list:

```
<?xml version="1.0" ?>
- <VitesseVariableList Version="1.0">
  - <CustomerInfo>
    - <Variable>
      <Name>NAME</Name>
      <Type>Field</Type>
      <Mode>Variable</Mode>
    </Variable>
    - <Variable>
      <Name>ADDRESS</Name>
      <Type>Field</Type>
      <Mode>Variable</Mode>
    </Variable>
```

The `<Variable></Variable>` tag pair defines a variable; the sub tag nodes define the type and mode of the variable.

The XML file can contain any number of variables, as required in the form or segment design. Whenever a form or segment is opened in Vitesse, loading the XML file with a list of variables helps to merge data with text. These variables become available in the **Merge data** drop-down in the *Merge Data* toolbar, thus making it possible to merge these variables with the text in a form or segment.

To add predefined variables to a form or segment design:

- 1 Using any text editor, create an XML file specifying all the required variable properties such as name, type, and mode as identified by their respective tags in the XML file.



You can select the required predefined variable file using the *Preferences* dialog in the Vitesse Administration mode. For more information, see *Chapter 36: Vitesse Administrative Options*.

- 2 Open a new or existing form or segment in Vitesse.
- 3 Create a text box in the design area using the **Text** tool .



You can also merge variables with graphic text objects.

- 4 With the text object selected, select a variable from the drop-down list on the *Merge Data* toolbar.
- 5 Click the **Merge Data** tool, , on the *Merge Data* toolbar.
- 6 The variable is merged with the text object.



The *Merge Data* toolbar adds the correct syntax for merging variables with text. You can also type this syntax directly in the textbox.

- 7 From the text object property sheet, select the **Settings** tab.
- 8 In the **Data Merging** area, select the **Merge data** option.
As the value for the variable is not defined in the form or segment object, the added syntax disappears from the text box. The syntax is replaced by the variable value when the form or segment is added to a project for which the variable is defined.

The image shows the process of using predefined variables in form and segment resources.

The variables defined in the xml file display in the Merge Data drop-down list, when a form or segment is opened.

5. In the text property sheet, select the Merge 4. The selected variable data option in the Settings tab. is merged with the text object.

1. Create a text object.

2. Select the required variable from the drop-down list.

3. Click the Merge Data button.

6. The merged variable disappears in the form or segment.

7. The merged variable is replaced by the variable value when opened in a project for which the variable is defined.



For more information on working with predefined variables in forms and segments, see *Chapter 9: Using the Text Tools*.

Chapter 27: Adding Conditions

This chapter explains condition objects and conditional logic that you can apply to design and data objects inserted in a project. The list of topics is as follows:

- What is a Condition?
- Working with Page and Record Level Conditions
- Adding Conditions to the Project Tree
- Using Data in a Condition
- Using Variables in a Condition
- Using System Variables in a Condition
- Using Constants in a Condition
- Dynamically Managing Form and Segment Content using Conditions

What are Conditions?

Conditions are control objects that are added to the project tree and processed in the order of their placement. Condition objects consist of condition checks (**If**, **Else**, and **ElseIf**) as identified by their respective nodes in the project tree. These condition checks further contain test statements that are used to test the data and make processing decisions based on the results. The result of the condition is interpreted data printed on the application page.

Conditional checks/tests are listed as follows:

Tests	Description
If	Represents the first conditional logic test and evaluates a condition as true or false.
Else	Processes a condition when the preceding If (and ElseIf) statement is false .
ElseIf	Evaluates multiple If conditions; executes only when the preceding If or ElseIf statements are false.

- If a condition is found true, objects placed under the **If** node are processed.
- If a condition under the **If** node is not found true, and the condition under the **ElseIf** node is found true, objects placed under the **ElseIf** node are processed.
- If a condition under the **If** (and/or **ElseIf**) node is not found true, objects placed under the **Else** node are processed.

You can use nested **If** and **Else** conditions to perform multiple logical tests on an object. You can also perform multiple logical tests on objects using the **ElseIf** condition statement; **ElseIf** can also be used to implement different cases in conditions. The **If** test is executed every time the condition is processed. The **ElseIf** test is applied only if the entire **If** or **ElseIf** statements above the **ElseIf** test do not hold true. **Else**, on the other hand, is not a conditional check. It only forces an unconditional action when all of the **If** and **ElseIf** tests above it hold false.

Defining a Condition Statement

Defining a condition implies inserting a condition for a document at the page or record level, specifying the conditional logic statement (**If**, **ElseIf**, and **Else**), and then defining the action to be performed if the specified condition holds true. Conditional logic statements are sometimes referred to as condition tests.



You can also define a test as part of the conditional logic statement which specifies the type of operation that needs to be performed.

A conditional logic statement contains operands and operators, followed by operands again in the sequence.

Operands

An operand is a value (data; variable; system variable; or a constant that can be text, number, or a hex number) which is compared using the specified operator with another operand of the same type. Every conditional statement contains at least two operands; the number of operands, however, can exceed depending on the complexity of the conditional logic statement.

Operators

Operator is a collection of one or more symbols and/or alphabets representing an operation that needs to be performed on two operands. The type of operator varies depending upon your required conditional logic.

Conditional statements require two types of operators:

Comparison Operators

A comparison operator compares the left operand with the right operand. The following table lists the operators that Vitesse employs when creating test statements:

Comparison Operator	Description
?=	Represents the contain operator and tests if the left operand holds/contains a value as

Comparison Operator	Description
	specified in the right operand.
<	Checks if the left operand is less than the right operand.
<=	Checks if the left operand is less than or equal to the right operand.
<>	Checks for inequality between two operands.
=	Checks if the two operands in a condition statement are exactly equal to each other.
>	Checks if the left operand is greater than the right operand.
>=	Checks if the left operand is greater than or equal to the right operand.

Logical Operators

A logical operator is used to compare a number of (nested) conditional logic tests within a conditional statement. These logical operators are listed in the following table:

Logical Operator	Description
AND	Conjoins several If condition statements; used when all statements hold true. It is equivalent to the “and” connective.
OR	Disjoins multiple If condition statements; used when any statement holds true. It is equivalent to the “or” connective.
NOT	Excludes condition statements by reversing the logical value of an evaluated condition (i.e., condition holds true if and only if the condition statement is false, and vice versa).

A conditional logic statement can either return a true or a false value. A true value means that the conditional logic statement is valid and Vitesse can continue processing the condition. A false value, contrarily, is an indication to stop processing the condition.

You can apply conditions to perform a variety of operations on your data, including changing the object color, font style and text, data values, etc. Additionally, in combination with **Action** objects (discussed in *Chapter 29: Using Actions*), you can define conditions to set booklet print options, document orientation and printing modes, apply finishing and page control options, and also transform the output value by selecting a predefined format from a list of masks, and even create your own masking formats.



Vitesse uses Action objects to prompt the printer to execute specified actions while printing and to set or change application variables. An action can be applied to any object within an application. It is typically used in conjunction with conditions.

The conditional logic processing can take place either before or after a document is designed. Processing before enables Vitesse to process the condition before any other action is performed (for example, page level conditions). Processing after first formats the object, and then performs any other action (for example, record level conditions).

Conditions can be added at the page or record level. Actual evaluation of conditions takes place at the record level, but resulting actions are applied at both the page and record levels.



Only true condition results are displayed in the design area.

Working with Page Level Conditions

Page level conditions are inserted under the **Page** node in the project tree and evaluated before page printing. You can apply page level conditions to actions, charts, barcodes, images, forms, and segments in all of the project modes. Conditions added at the page level are normally used to test the variables or constants that may have been used in the page along with some other page attributes. In the Line mode, forms, segments, charts, and actions can be processed under a page level condition. You can also apply conditions to design objects in Native, Database, and XML mode projects.

Working with Record Level Conditions

Record level conditions are inserted under the **Record** node in the project tree and processed in the order of placement in the project tree. You can apply record level conditions to actions, charts, barcodes, and fields in Line and XML mode projects only. In the Line project mode, fields, charts, barcodes, and actions can be processed under a record level condition. There are no record level conditions in Database mode as there are no records.

Adding Conditions to the Project Tree

You can add a condition or multiple conditions at the page or record level in the project tree, depending on the design requirements and complexity. Conditions are added and applied only to objects under which they are added.

Inserting a Condition

To insert a condition object at the page or record level:

- 1 From the project tree, select either the **Page**, **Record**, or **Field** node to which you want to apply the condition.
- 2 Click  on the *Insert* toolbar.



You can also insert the condition object either by:

- Selecting Insert>Condition from the menu bar.
Or
- Right-clicking the required node in the project tree, and then selecting Insert>Condition from the pop-up menu.

A condition node is added to the project tree under the selected **Page**, **Record**, or **Field** node. As condition is only a logical check, therefore, only the effects are visible in the design area.

- 3 Click  next to the Condition node to view the If and Else nodes.
- 4 Select the If node; the Property View window displays the General tab associated with the condition object where you can specify your conditions.



If you want to insert nested conditions, select the **If** node and insert another condition node using the above-mentioned procedure.

Specifying a Condition Test

Once you have added the condition object under the required project tree node, you can create a condition test using the properties associated with the **If** node.



This section covers only the basics for defining a condition test. The “Using Data in Conditions” section exemplifies the conditional logic process by detailing steps that are specific to each project mode.

To add a conditional test statement:

- 1 Insert the condition object under the required node in the project tree.
- 2 Select the **If** node.
- 3 In the **General** tab of the *Property View* window, click the line below the **IF** label to activate the entry box .
- 4 Click the arrow and select operand(s) from either of the following options:
 - **Data Mapping** – enables you to perform comparisons between data mapped from the data file; provides sub options to specify the start and end bytes of the data in the data file. This option is enabled only for the Line, Database, and XML mode projects.
 - **Variables** – enables you to perform comparisons between pre-defined variables. This option is enabled only if you have defined variables in your application.
 - **System Variables** – enables you to perform comparisons between system variables.
 - **Constants** – provides options for using **Number**, **Text**, or **Hex** number:
 - The **Number** option enables number comparisons (“#”) such as greater than 10 or less than or equal to 100 and so on. This option is enabled for a record.
 - The **Text** option enables only alphanumeric string (‘) comparisons, which can be either equal to or not equal to a given string.
 - The **Hex** option performs comparisons on a single character/bit in a field. It is most commonly used to eliminate empty lines in a field carrying multiple instances.
- 5 Specify the operand value in the entry box.
- 6 Click in the middle of the line to activate the drop-down list for choosing the required comparison operator.
- 7 Click the area following the operator drop-down list to enable another entry box . The arrow in this entry box also contains the same options as listed in Step 4. Choose the required operand and specify the value in the entry box.

The **If** condition test statement completes.



To use nested conditions, you can add another condition node (see the “Inserting a Condition” section above) underneath the **If** node and repeat **Steps 2 - 7**.

Specifying Detailed Condition Tests using Logical Operators

To refine your condition test statement, depending on the complexity of your design, you can add further condition test statements by using the logical operators (see the “Logical Operators” section above). Assuming that you have followed **Steps 1 – 7** for inserting a condition and defining a condition test, the steps below enable you to continue modifying the condition by using logical operators in your condition tests.

Using the AND and OR Logical Operators

- 1 Click  in the **General** tab. Two rows add below the existing **IF** test statement; first contains the area for specifying the logical operators; the second enables you to define the condition tests.
- 2 Click the area containing the **AND** operator (used by default) to view the drop-down list options:
 - **AND** – Executes failure action when first **AND** fails; executes success action when all condition statements separated by **AND** succeed.
 - **OR** – Executes failure action when all **ORs** in a sequence fail; executes success action when the first **OR** succeeds.

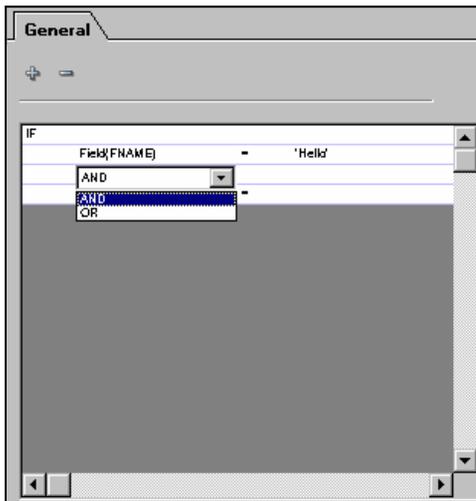


Image showing the AND and OR logical operators.

- 3 Select the required logical operator.
- 4 Click the second row to specify the operands and operators. The process for this step is the same as followed in **Steps 3 – 7**.

You can use a combination of **AND** and **OR** operators in a condition statement.

Using the NOT Logical Operator

In addition to the **AND** and/or **OR** logical operators, you can also specify the **NOT** operator to reverse the logical meaning of your specified condition.

- 1 Click the left most area either in the row containing the logical operators **AND/OR**, or the row where you specify the operands and the operators.
A drop-down list is enabled that contains the **NOT** operator option.
- 2 Select the **NOT** operator.
- 3 Repeat **Steps 3 – 7** to continue specifying the condition test.

If the **NOT** operator is used in combination with the **AND** or **OR** operator, the entire statement including **AND** or **OR** will be negated.

You can choose to repeat **Steps 8 – 14** to keep on adding condition tests within the selected **If** statement.



To delete a condition test within the **If** statement, click .

Using the ElseIf Condition

Apart from applying the basic conditional logic to an object, you can also use different cases within a condition or nested conditions by using the **ElseIf** condition option.

To insert the **ElseIf** condition node(s):

- 1 Select the **If** node and either:
 - Select Insert>Condition Else If from the menu bar.
 - Or
 - Right-click the node and select Insert>Condition Else If from the pop-up menu.
- 4 Click  in the **General** tab to add the **IF** test statement. A test is not automatically added under the **ElseIf** node.
- 5 Repeat **Steps 3 – 7** to define the test statement, and if necessary, **Steps 8 – 14** to enhance the condition tests.



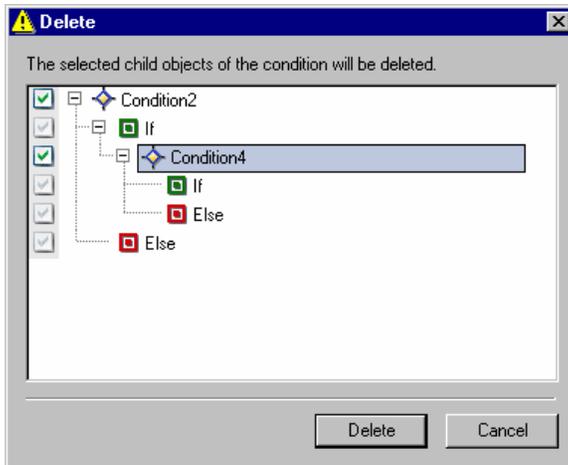
The **Else** node does not require any condition specification; objects that you insert under the **Else** node are processed only if the preceding **If** and/or **Elseif** statements are determined to be false.

Deleting a Condition

You can delete a selected condition node by either of the following methods:

- Select Edit>Delete from the menu bar.
- Click on the *Standard* toolbar.
- Right-click the selected condition node, and select **Delete** from the pop-up menu.
- Press <Delete> on the keyboard.

Using either of the methods displays the *Delete* dialog



The dialog depicts a tree structure of the condition (and nested conditions if any) used in the project tree. Select the checkbox against the required condition, and then click **Delete**. Selected condition is deleted from the project tree.

Information in subsequent sections explains how to conditionally design your page by applying conditions and/or nested conditions to objects in different project modes.

Using Data in Conditions

You can map data to conditions in Line mode (from prefixed and non-prefixed data files), Database, and XML mode projects. This section provides different exemplary tasks for mapping data to conditions in the listed modes. Basics for inserting and defining condition tests have already been covered in the “Adding Conditions to the Project Tree” section.



When mapping data for conditions, field information is always specified as field Start and End byte.

Mapping Data to a Condition in Line Mode Project

This section details tasks for mapping data to conditions in Line mode projects for non-prefixed and prefixed data files.

Mapping Data at the Page Level Condition – Non-Prefixed Data

This task provides an example of mapping data to a condition in line mode project having a non-prefixed data file (non-delimited). The condition is applied at the page level. At this stage, we are assuming that you are familiar with mapping records and fields to the design area and setting record and field properties. Therefore, the example focuses on the conditional logic that you apply on the data for printing the required results.

In the example, there are three customers, each with a separate account number. The conditional logic used here tests that if the first three digits of the account number are ‘333,’ the customer will receive a message specific to one location; if the first three digits encountered are ‘334,’ the customer will receive a message regarding the second location. To provide messages, you can use two segment resources, each containing a message specific to a location.

After inserting the condition, specify the required operands under the **IF** label. The **Data Mapping** sub-menu contains the following options:

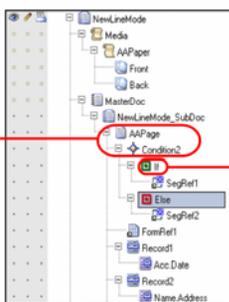
- **Field(#)** – specifies the field number from prefixed or non-prefixed delimited data file for condition test at the record level.
- **Field(#-#)** – specifies the field range from non-delimited prefixed or non-prefixed data file for condition test at the record level.
- **Line(#-#).Field(#)** – specifies the line range and field number from prefixed or non-prefixed delimited data file for a page level condition test.
- **Line(#-#).Field(#-#)** – specifies the line range and field range from non-delimited prefixed or non-prefixed data file for a page level condition test.



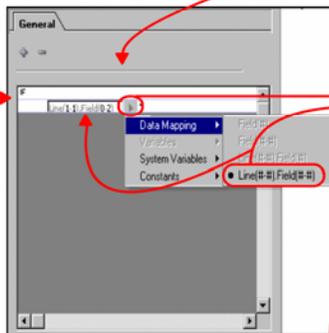
You can also specify line or field range and field number by selecting desired fields in the data file. When mapping data for conditions, field information is always specified as field Start and End byte.

See image below to understand the process of mapping data for conditions in Line mode non-prefixed (non-delimited) data file.

1. Insert the Condition node under the Page node in the project tree.



2. Expand the condition node to view the If and Else nodes. Add SegRef1 under the If node, and SegRef2 under the Else node.



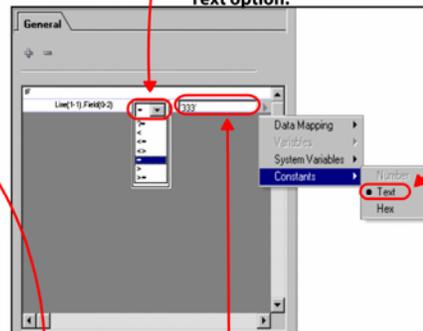
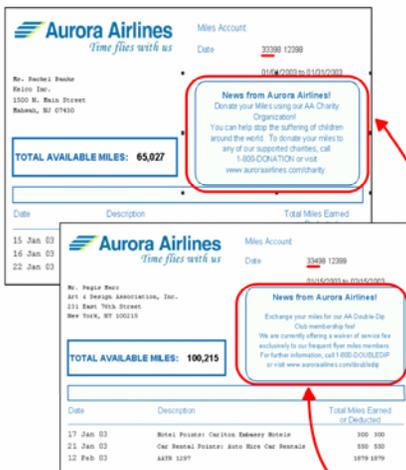
3. In the General tab, click the entry box under the IF label.

4. Click and select Data Mapping from the menu; associated sub-menu displays options that enable you to select data from the data file.

5. Select Line (#-#).Field (#-#), and then specify the start and end bytes of the record line and field length. This is the first condition operand.

6. Click the drop-down next to the first operand. Select the '=' operator.

7. Click the entry box adjacent to the operator drop-down list. Click and select Constants from the menu. From the associated sub-menu, select the Text option.



8. Specify the text as '333'. This is the second condition operand.

9. According to the condition, if '333' is found in the specified data: Line (1-1).Field (0-2), SegRef1 is printed; else if '334' is found, SegRef2 is printed. This can also be seen from the project tree order.

Mapping Data at the Record Level Condition – Non-Prefixed Data

This task provides an example of mapping data to a condition in line mode project having a non-prefixed data file (non-delimited). The condition is applied at the record level. When you insert fields at the record level, the following options are available in the **General** tab of the **If** node:

- **Record Level** – applies record level condition test at the selected field number or field range from the data file.
- **Page Level** – applies record level condition test at the selected line range and field number or field range from the data file. You can insert only one condition test statement using the Page Level option.

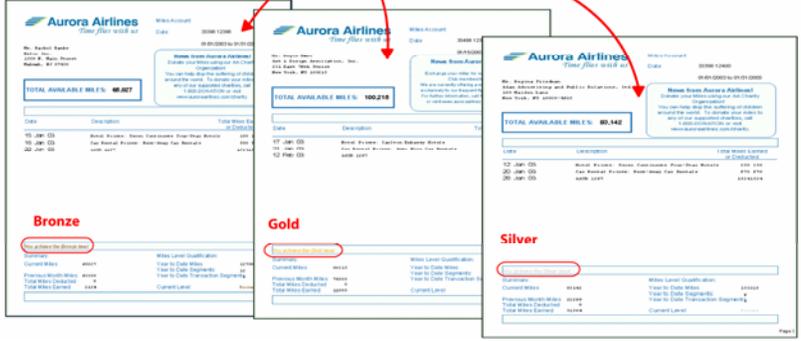
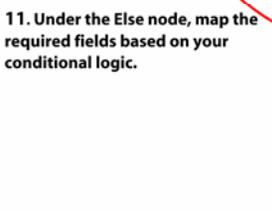
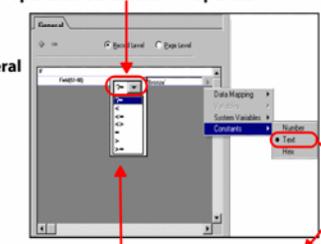
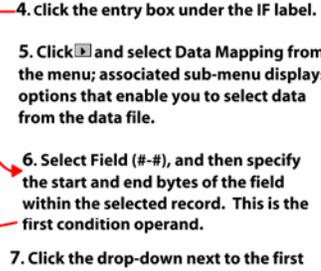
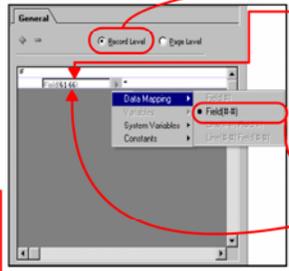
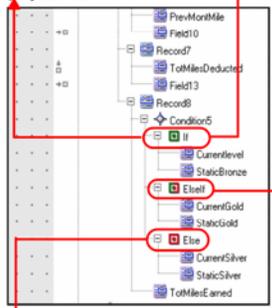
In this example, there are three customers. The customers can attain a bronze, gold, or silver level depending on miles traveled. The conditional logic used here changes the field color and prints a message in the same color, depending on the values in the mapped data.



You can use a **Static Text** field to print the required message by specifying the message text in the field properties. See *Section 6, 7, and 8* for more information.

See the image below to understand the process of mapping data for conditions in Line mode non-prefixed (non-delimited) data file at the record level.

1. Insert the Condition node under the record node in the project tree.
2. Expand the condition node to view the If and Else nodes. Select the If node and map the required fields underneath.
3. In the General tab, select the Record Level option.
4. Click the entry box under the IF label.
5. Click and select Data Mapping from the menu; associated sub-menu displays options that enable you to select data from the data file.
6. Select Field (#-#), and then specify the start and end bytes of the field within the selected record. This is the first condition operand.
7. Click the drop-down next to the first operand. Select the '?' operator.
8. Click the entry box adjacent to the operator drop-down list. Click and select Constants from the menu. From the associated sub-menu, select the Text option. Enter the text as 'Bronze'. This is the second condition operand.
9. Select the Elself node and map the required fields underneath. In the General tab, click to define the condition statement.
10. Repeat Steps 3-8. In Step 8, enter 'Gold' as the second condition operand.
11. Under the Else node, map the required fields based on your conditional logic.
12. According to the condition, if 'Bronze' is found in the specified data: Field (61-66), StaticBronze field is printed; else if 'Gold' is found, StaticGold field is printed. If none of the If and Elself conditions hold true, then the StaticSilver field is printed. This is also obvious from the project tree order.



You can use the arrow buttons underneath the *Property View* window to scroll through the documents in the data file to see where the objects placed under the **ElseIf** and **Else** nodes are processed. An image showing the arrow buttons is below:

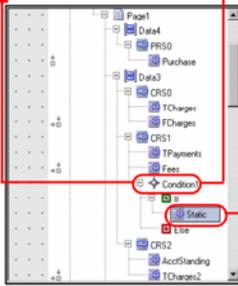


Mapping Data to a Condition in Line Mode Project – Prefixed Data

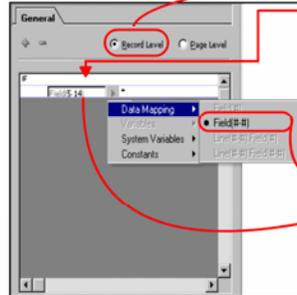
In this example, the condition tests whether a customer is eligible for an increase in credit limit. If the total payments of the customer exceed \$100,000, the credit limit will be increased by \$200,000. A text message will print on the document notifying the increase.

See image below to understand the process of mapping data for conditions in Line mode prefixed (non-delimited) data file at the record level.

1. Insert the Condition node under the required record node in the project tree.



2. Expand the condition node to view the If and Else nodes. Select the If node.



3. In the General tab, select the Record Level option.

4. Click the entry box under the IF label.

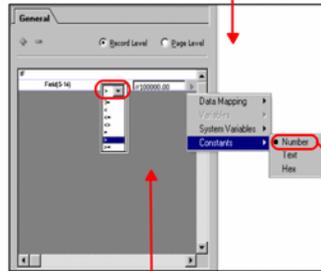
5. Click and select Data Mapping from the menu; associated sub-menu displays options that enable you to select data from the data file.

6. Select Field (#-#), and then specify the start and end bytes of the field within the selected record. This is the first condition operand.

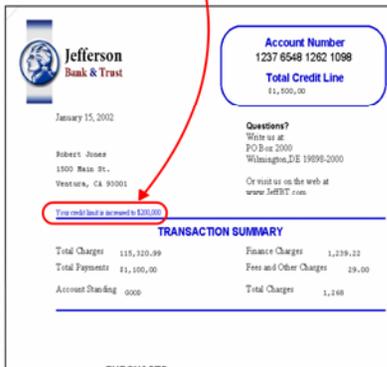
7. Click the drop-down next to the first operand. Select the '>' operator.

9. Insert a field under the If node. In the Settings tab of the field properties, set the field as Static Text and specify the text message.

10. According to the condition, if the total payments of a customer exceeds \$100,000, the credit limit will be increased by \$ 200,000. A message will print indicating the increase in the limit as shown below. This is the same message that you specified in Step 9.



8. Click the entry box adjacent to the operator drop-down list. Click and select Constants from the menu. From the associated sub-menu, select the Number option. Enter the number as '#100000.00'. This is the second condition operand.

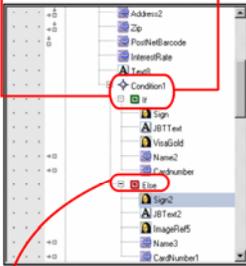


Mapping Data to a Condition in Database Mode

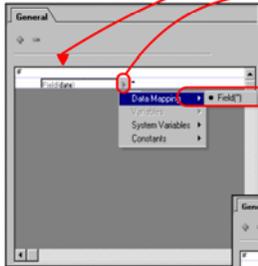
In the example, there are two customers, each processing visa cards. The conditional logic used here tests whether the customers hold a Gold Card or a Silver Card. If the field “card type” mapped from the data file contains the text “Gold Card,” the image, text, and signature associated with the Gold Card will be printed on the page. On the other hand, if the text “Silver Card” is found in the data, the image, text, and signature for the Silver Card will be printed. To provide the text and the images, you can use the required image resources and text object for each customer and place them accordingly under the **If** and **Else** nodes in the project tree.

See image on the next page to understand the process of mapping data for conditions in Database mode projects.

1. Insert the Condition node under the record node in the project tree.



2. Expand the condition node to view the If and Else nodes. Select the If node and map the required fields underneath.

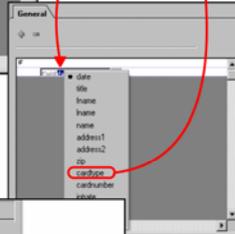


3. Click the entry box under the IF label.

4. Click **Field()** and select Data Mapping from the menu; associated sub-menu displays options that enable you to select data from the data file.

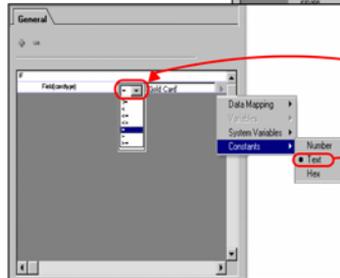
5. Select Field (' '); Field(date) appears by default in the entry box.

6. Click (date) and select cardtype from the drop-down menu. This is the first condition operand.



9. Based on your conditional logic, map the required image, text, and field objects under the If node.

10. Similarly, under the Else node, map the required image, text, and field objects.



7. Click the drop-down next to the first operand. Select the '=' operator.

8. Click the entry box adjacent to the operator drop-down list. Click **Text** and select Constants from the menu. From the associated sub-menu, select the Text option. Enter the text as 'GoldCard'. This is the second condition operand.

11. According to the condition, if 'GoldCard' is found in the specified customer data: Field ('cardtype'), the text, signature, and images pertaining to the Gold Card member will be printed on the page; else if the condition is not found true, text, signature, and images pertaining to the Silver Card will be printed. See image below:



The **Field()** option in the **Data Mapping** menu signifies the field name contained in the data file.

Mapping Data to a Condition in XML Mode

In the example, the conditional logic tests whether a customer's account balance is below or above the minimum required level i.e., \$1500. If the balance is below the required level, a warning message will print on the page. On the other hand, if the balance is above the minimum level, no message will be printed. To provide the warning message, you can use a **Static Text** field and specify the message to be printed in the **Const** option available in the **Settings** tab of the selected field object; and the message color in the **Color** tab. You can then map the field under the **If** node, as appropriate.

See image below to understand the process of mapping data for conditions in XML mode projects.

1. Insert the Condition node under the required record node in the project tree.
2. Expand the condition node to view the If and Else nodes. Select the If node.
3. Click the entry box under the IF label.
4. Click and select Data Mapping from the menu; associated sub-menu displays options that enable you to select data from the data file.
5. Select Field (' '); Field(^Customers) appears by default in the entry box.
6. Click (^Customers) and select EndingBalance from the drop-down tree. This is the first condition operand.
7. Click the drop-down next to the first operand. Select the '<' operator.
8. Click the entry box adjacent to the operator drop-down list. Click and select Constants from the menu. From the associated sub-menu, select the Number option. Enter the number as #1500. This is the second condition operand.
9. Based on your conditional logic, map the required field objects under the If node.
10. Similarly, under the Else node, map the required field object.

11. According to the condition, if the ending balance of the customer is below the minimum required balance i.e., 1500, the text message will print as shown in the image below. If, on the other hand, the balance is above the minimum level, no message will print.

Account Summary	Type	Amount	Opening Balance	Credits	Debits	Ending Balance
			4,023.00	3,003.00	3,242.00	3,784.00
ATM Withdrawal	Withdrawal	50.00			50.00	3,734.00
Paper/Plastic Payment - Crp	Payment	400.00		400.00		3,334.00
Check 102	Check	50.00			50.00	3,284.00
Check 103	Check	100.00			100.00	3,184.00
Check 104	Check	100.00			100.00	3,084.00
Check 105	Check	100.00			100.00	2,984.00
Check 106	Check	100.00			100.00	2,884.00
Check 107	Check	100.00			100.00	2,784.00
Check 108	Check	100.00			100.00	2,684.00
Check 109	Check	100.00			100.00	2,584.00
Check 110	Check	100.00			100.00	2,484.00
Check 111	Check	100.00			100.00	2,384.00
Check 112	Check	100.00			100.00	2,284.00
ATM Withdrawal	Withdrawal	50.00			50.00	2,234.00
			2,903.00	3,242.00		2,903.00



The **Field("")** option in the **Data Mapping** menu signifies the field/customer name contained in the data file.



In the above **Data Mapping** process, if a tag name is selected from the drop-down tree using <Shift> + <Click>, the selected field will display tag names and associated parent node path information.

Using Variables in a Condition

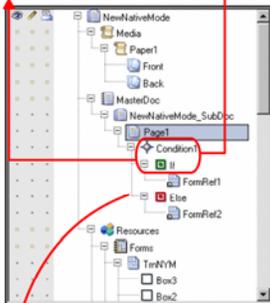
This section explains information about applying conditional logic to variables. You can use variables at both the page and record levels and in any project mode. The condition tests that if value of VARNumber1 is greater than value of VARNumber2, FormRef1 is printed on the page, else if the value is less than that of VARNumber2, FormRef2 is printed.



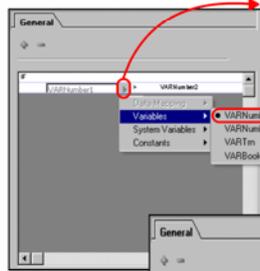
You must define variables before using them in conditions via the **Variables** tab of the **MasterDoc** project tree node. For more information, refer to *Chapter 26: Using Variables*.

See image below to understand the process of mapping variables for conditions in a Native mode project.

1. Insert the Condition node under the Page node in the project tree.



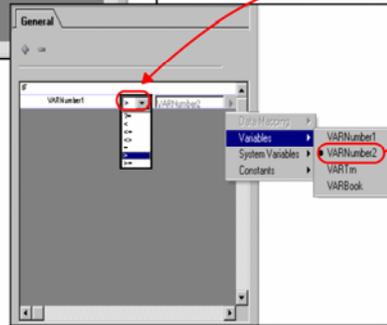
2. Expand the condition node to view the If and Else nodes. Select the If node.



3. Click the entry box under the IF label.

4. Click and select Variables from the menu, and then select VARNumber1.

5. Click the drop-down next to the first operand. Select the '>' operator.

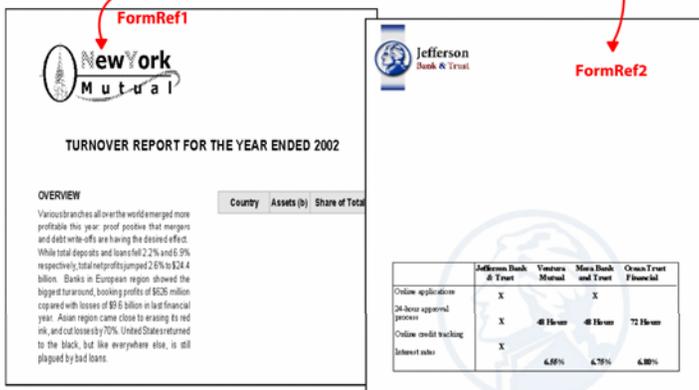


7. Based on your conditional logic, map the required form under the If node.

8. Similarly, under the Else node, map the required form.

6. Click the entry box adjacent to the operator drop-down list. Click and select Variables->VARNumber2 from the menu. This is the second condition operand.

9. According to the condition, if the variable VARNumber1 is greater than VARNumber 2, FormRef1 is printed on the page, else FormRef2 is printed. This is also visible from the project tree.

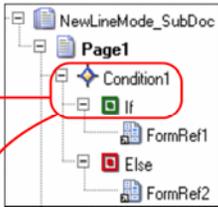


Using System Variables in a Condition

This section explains applying conditional logic to system variables. You can use system variables at both the page and record levels and in any project mode. The condition test enables you to print forms on the front and back of a sheet of paper based on the selected system variable. According to the condition, if the value of the system variable “BACK” is false, FormRef1 is printed on the page front and if the value is true, FormRef2 is printed on the page front.

The following image explains the use of a system variable BACK to switch forms between the front and back sides of the page using conditions.

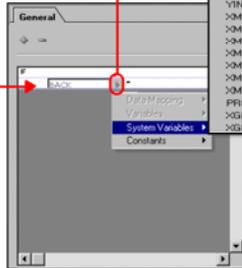
1. Insert the Condition node under the Page node in the project tree.



2. Expand the condition node to view the If and Else nodes. Select the If node.

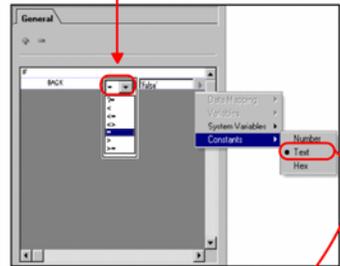
3. Click the entry box under the IF label.

4. Click and select System Variables from the menu.



5. Select the BACK system variable. This is the first condition operand.

6. Click the drop-down next to the first operand. Select the '=' operator.

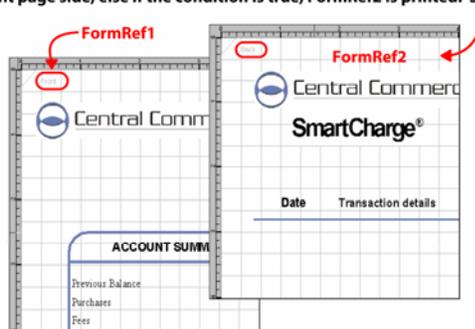


7. Click the entry box adjacent to the operator drop-down list. Click and select Constants>Text from the menu. Enter the text as 'false' in the entry box. This is the second condition operand.

8. Based on your conditional logic, map the required form under the If node.

9. Similarly, under the Else node, map the required form.

10. According to the condition, if the value of the System Variable 'Back' is false, FormRef1 is printed on front page side, else if the condition is true, FormRef2 is printed. See image below:

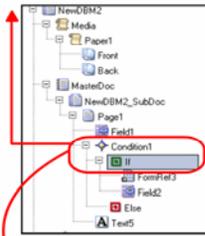


Using Constants in a Condition

This section explains how to apply conditional logic using constants. You can use constants at both the page and record levels and in any project mode. The condition tests whether a customer is eligible for a credit card. Accordingly, it tests that if the zip code of the customer is not greater than 1500, the customer stands ineligible for the credit card. A message indicating the ineligibility will print on the form based on the condition test. You can use the **Static Text** field to print the text message. **NOT** logical operator has been used for this condition text.

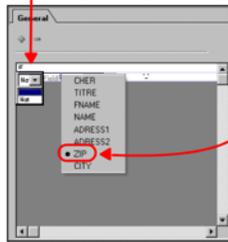
The following image explains the use of constants using conditions.

1. Insert the Condition node under the Page node in the project tree.



2. Expand the condition node to view the If and Else nodes. Select the If node.

3. Click the area before the entry box and select the NOT operator from the drop-down.

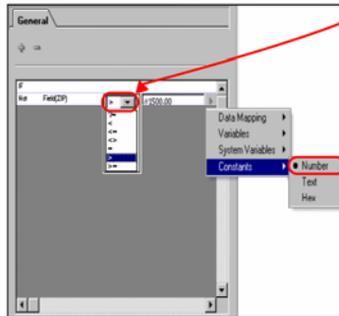


4. Click the entry box next to the NOT operator.

5. Click [F2] and select Data Mapping>Field(' ') from the menu option.

6. Click in the entry box and select ZIP from the drop-down menu. This is the first condition operand.

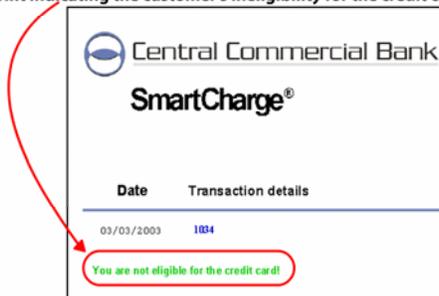
7. Click the drop-down next to the first operand. Select the '>' operator.



8. Click the entry box adjacent to the operator drop-down list. Click [F2] and select Constants>Number from the menu. Enter the number as '#1500' in the entry box. This is the second condition operand.

9. Based on your conditional logic, map the required form and field objects under the If node.

10. According to the condition, if the value in the Field ('ZIP') is NOT greater than 1500, a text message will print indicating the customer's ineligibility for the credit card. See image below:



Dynamically Managing Form and Segment Content using Conditions

Vitesse 2.00 (and higher versions) supports dynamically managing the content of segments and forms through conditional logic. Apart from adding conditions at the page and record levels in different project modes, you can also insert conditions and define a conditional logic for forms and/or segments that are either:

- Used as stand-alone projects.
Or
- Located under the **Resources>Forms** and/or **Segments** nodes respectively in the project modes.

Forms and Segments Used as Resources in Projects

There are a few limitations for conditionally manipulating segments and forms that are used as resources within different project modes:

- Conditions can be added for inline, embedded, and external segment resources with the exception of segments that contain PostScript (PS) or Encapsulated PostScript (EPS) images.
- For segments used as external and embedded resources within projects, conditions can be added only under their respective nodes in the **Resources>Segments** tree hierarchy. Such segments appear as referenced resources identified by the **SegRef** nodes and hence, condition cannot be applied for these.
- If, on the other hand, an inline segment is used in a project, you can apply a condition under its respective node under the **Page** node in the project tree.

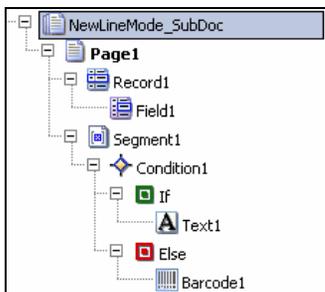


Image showing the condition inserted under the inline segment node in a Line mode project.



With the exception of inline segments, conditions (and other objects) cannot be inserted under the **SegRef** nodes.

- Conditions can be added under all types of forms used in projects except the **FormRef** node.
- Since you cannot use embedded form and segment types for forms or segments that are opened as stand-alone projects, conditions cannot be applied for such form and segment types under these circumstances only.

When you insert conditions under forms or segments that are added at the **Resources>Forms or Segments** node level in projects, you can map all drawing objects, images, barcodes, charts, and the action object under the condition **If**, **Else**, and **ElseIf** nodes and specify a condition test as required. Apart from these objects, you can choose to use predefined variables or constants as condition operands when specifying a condition statement.



You cannot use record or field nodes under the **If**, **Elseif**, and **Else** nodes at the **Resources** level in the project tree.

Data from data files can also be used as condition operands when conditions are inserted under segments and forms at the **Resources** tree level. The **Data Mapping** option, when specifying a condition test, is disabled for external forms and segments referred in Line mode projects; for remaining form and segment types, you can specify data as operands in a condition statement. Similarly, data mapping is also available in Database and XML mode projects at the **Resources** level.



The process for using conditions at the form and segment level is the same as explained earlier. Refer to tasks covered (above) in this chapter for more information.

Segments and Forms Used as Stand-Alone Projects

You can also use conditions for stand-alone forms and segments. You can map drawing objects, images, charts, barcodes, and action objects in a similar manner as mentioned in preceding tasks. Along with these, you can also use variables, constants, and data when defining your condition statements.

When forms and segments are used as stand-alone projects, you cannot use a data file to map data to conditions. To achieve data mapping, you need to use variables or fields loaded from an XML file which is defined by the administrator in the Vitesse Administration mode. See *Chapter 9: Using the Text Tools* for more information. These variables or fields inserted in the forms or segments must also be defined in the different project modes in which you intend to use the forms and/or segments.



If you use variables or fields in stand-alone forms and segments, you should be careful when using these in the different project modes.

Also, the variable or field names used in forms and segments must correlate with the field names that are available in the Database or XML mode projects that use these forms and segments.

User Notes:

Chapter 28: Adding Charts

This chapter describes activities and tasks for adding charts and setting chart properties. It also provides information on mapping data to charts using different techniques for various types of data files and project modes. Tasks include:

- Understanding Charts
- Working with Page Level Charts
- Working with Record Level Charts
- Inserting a Chart
- Setting Chart Properties
- Mapping Data to Charts

Understanding Charts

A chart is a graphical representation of variable and/or constant data. A chart is used to display data or the relationship between different sets of data in a pictorial form. Representing information in the form of charts gives an immediate grasp of business statistics and their significance.

In Vitesse, data may be represented graphically using Bar, Line, or Pie charts. You can [set](#) properties such as size, position, color, font, 3D effects, border, and legends for a chart.



The **Bar** chart displays information as a series of parallel, usually vertical or horizontal bars.
The **Pie** chart is a type of graph that presents values as percentages (slices) of a whole pie.
The **Line** chart is a type of graph in which values from one or more sets of data are connected by lines.

The data for charts is mapped as **Label** and **Value** pairs, where label identifies an item and value is the assigned numerical quantity of a particular item. In Vitesse, you can map variable data and/or enter constants for chart label and value pairs.



The label and value pairs determine the number of bars/slices /sets of data connected by lines in the chart.

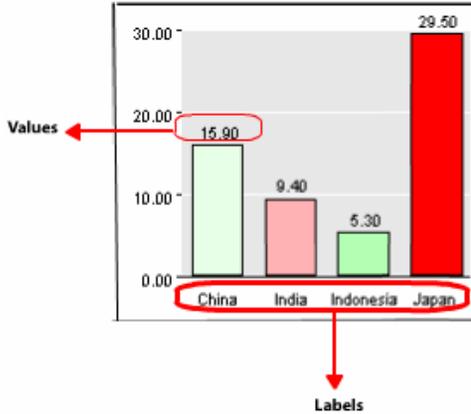


Image shows the chart with label and value pairs.

Charts can be defined at both the Page and Record levels (explained below). You can map data to charts using different techniques for various types of data files and project modes.

Working with Page Level Charts

Page level charts are inserted in the project tree under the **Page** node. You can select data for chart label and value pairs from multiple file records for a page level chart. A chart added at the page level prints at the document level.

Working with Record Level Charts

Record level charts are inserted in the project tree under the **Record** node and printed in project tree order as often as the record is repeated. Data selection is limited to a particular record (i.e., the printed chart for one record carries label and value pair data from that record only).



A chart added under a record prints each time the record is printed.

Inserting a Chart

To add a chart object at the page or record level, select **Insert > Chart** from the menu bar; a chart inserts in the design area and a new chart node adds to the project tree.



You can also add a new chart object either by:

- Clicking  on the *Insert* toolbar.
Or
- Right-clicking the required page or record node, and then selecting *Insert>Chart* from the pop-up menu.



You can delete a selected chart object either by:

- Selecting *Edit> Delete* from the menu bar.
Or
- Clicking  on the *Standard* toolbar.

You can insert any number of chart objects to the project.

Setting Chart Properties

You can insert multiple chart objects in the design area and set or change chart properties such as the chart type, size, color, font, 3D effect, position, border, and legends. Chart properties also help in displaying relationships between different sets of data used in a chart.

Naming a Chart

Assigning meaningful names to a chart helps in referencing the charts when multiple chart objects are being used. Vitesse assigns a default name to each newly created chart. You can also set a unique name for the chart using **Name** entry box in the **General** tab of the *Property View* window.

Selecting Chart Type

Vitesse offers different types of charts to illustrate quantitative relationships between data.

To select the type of chart:

- 1 Select the required chart object in the design area.



You can also select the **Chart** node in the *Project Tree View* window to set the chart object properties.

- 2 From the *Property View* window, select the **General** tab.

- 3 Select one of the following chart types from the **Type** drop-down list:
 - **Bar** – displays bar charts.
 - **Pie** – displays pie charts.
 - **Line** – displays line charts.

Positioning a Chart

Chart objects can be placed at an absolute position in the design area by specifying the horizontal and vertical coordinates. You can also position chart objects relative to one another.

Setting Chart Absolute Position

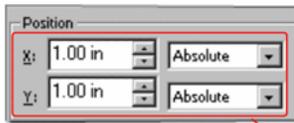
To accurately position a chart object in the design area:

- 1 Select the required **Chart** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the chart horizontal position, per specified unit of measurement in the **X** entry box. In the associated drop-down list, select **Absolute**.
- 4 Specify the chart vertical position, per specified unit of measurement in the **Y** entry box. In the associated drop-down list, select **Absolute**.

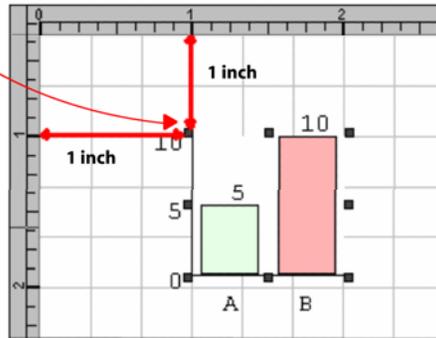


You can also set the unit of measurement for the chart position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The absolute coordinates are calculated from the top left margin of the page.



The chart positioned at absolute horizontal and vertical coordinates set to 1 inch from the top left margin.



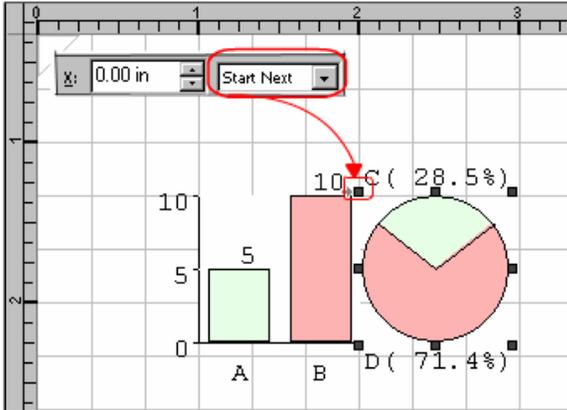
The image shows the process of setting chart object position using absolute coordinates.

Setting Chart Relative Position

A chart object can be positioned relative to the previous chart objects and other relatively positioned objects. In absolute positioning, the top left margin of the page acts as the point of reference, whereas in relative positioning the previously added object is taken as the point of reference.

To position charts using relative positioning:

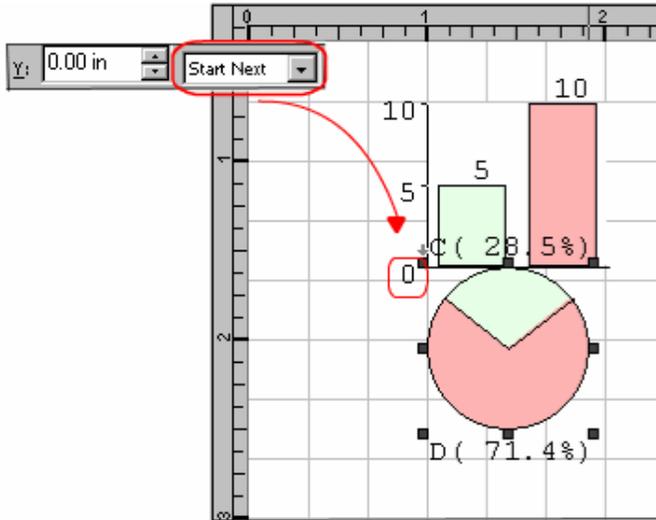
- 1 Select the required **Chart** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify **0** in the **X** entry box. In the associated drop-down list, select one of the following horizontal relative positioning options:
 - **Start Same** – positions the start of current chart object at the start of the previous chart object.
 - **Start Center** – positions the start of the current chart object at the center of the previous chart object.
 - **Start Next** – positions the start of the current chart object at the end of the previous chart object.



In this example, the current pie chart object is positioned at the previous bar chart object end.

- **Center Before** – positions the center of the current chart object at the start of the previous chart object.
- **Center Same** – positions the center of the current chart object at the center of the previous chart object.
- **Center Next** – positions the center of the current chart object at the end of the previous chart object.
- **End Before** – positions the end of the current chart object at the start of the previous chart object.
- **End Center** – positions the end of the current chart object at the center of the previous chart object.
- **End Same** – positions the end of the current chart object at the end of the previous chart object.

- 4 Specify **0** in the **Y** entry box. In the associated drop-down list, select one of the above-mentioned vertical relative positioning options. Vitesse provides the same set of vertical relative positioning options as the horizontal relative positioning options.



In this example, the current pie chart object is positioned at the end of the previous bar chart object.



Using a combination of absolute and relative positioning, you can place the chart object relative to another chart object or a relatively positioned object and add the spacing defined by the absolute X and Y coordinates.



The following symbols appear next to a relatively positioned chart node in the *Project Tree View* window:

-  - specifies that horizontal relative positioning is applied to the chart object.
-  - specifies that vertical relative positioning is applied to the chart object.
-  - specifies that both horizontal and vertical relative positioning is applied to the chart object.

Setting the Chart Size

Chart dimensions specify the width and height of the chart. To set the chart dimensions:

- 1 Select the required **Chart** node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Size** information:
 - **W** – sets the chart object width, per specified unit of measurement.
 - **H** – sets the chart object height, per specified unit of measurement.



You can also set the unit of measurement for the chart size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The  icon in the **Size** area is used for maintaining the aspect ratio of the text object. Clicking this icon and then setting the width and height increases or decreases the length and width of the object in equal proportion. For more information, see “Setting Common Object Properties” section in *Chapter 8: Using the Drawing Tools*.

Setting the Chart Spacing

Setting chart spacing is helpful when chart objects are placed at the record level. Record level charts print as often as the record is repeated. Setting chart spacing prevents multiple instances of the same chart object to over print. You can set the spacing between different instances of the same chart object; when working with multiple line records in non-prefixed data in Line mode; records having multiple instances in prefixed data in Line mode; or the vertical spacing between the different instances of the same data tag in XML mode. The spacing can be set to either fixed or variable.

- Fixed spacing is used with non-prefixed data; this option adjusts the spacing of a chart without affecting the spacing of other charts under the same record.
- Variable spacing option is used when working with prefixed data and XML data. In this mode, spacing between each chart instance is variable and affects spacing of the subsequent charts mapped for the same record. Variable spacing adjusts the spacing of all the charts under the same record. Adding additional charts adds the spacing value of the previous chart to the spacing value of the new chart.



For more information on the spacing modes, see “Using Fixed or Variable Spacing” section in *Chapter 22: Working with Data in Line Mode*.

To set the chart spacing options:

- 1 Select the required **Chart** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 From the **Spacing** area, specify the following:
 - **Horizontal** – specifies the horizontal spacing value of the chart.

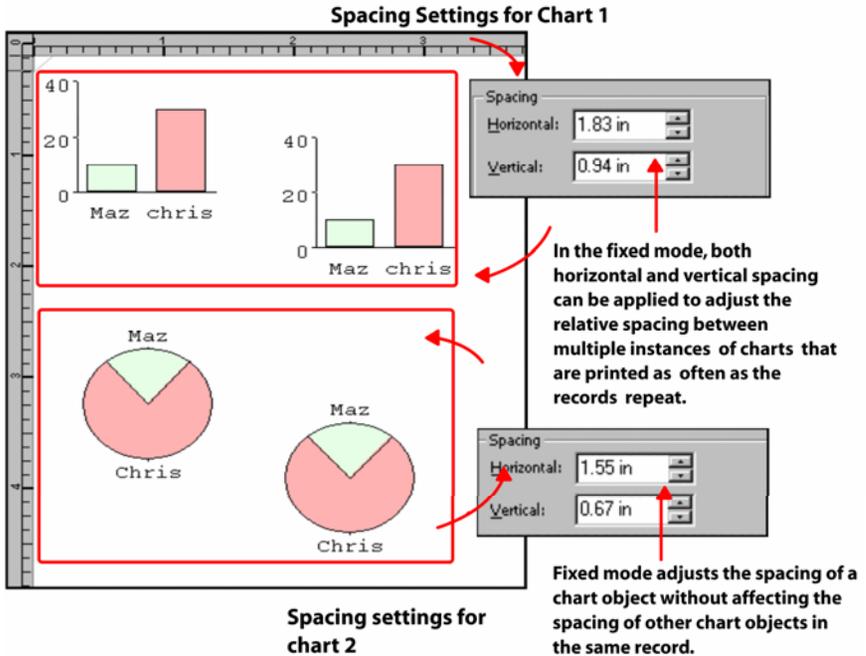


You cannot specify the horizontal spacing value, when working with prefixed data. For prefixed data projects, the **Horizontal** entry box is disabled.

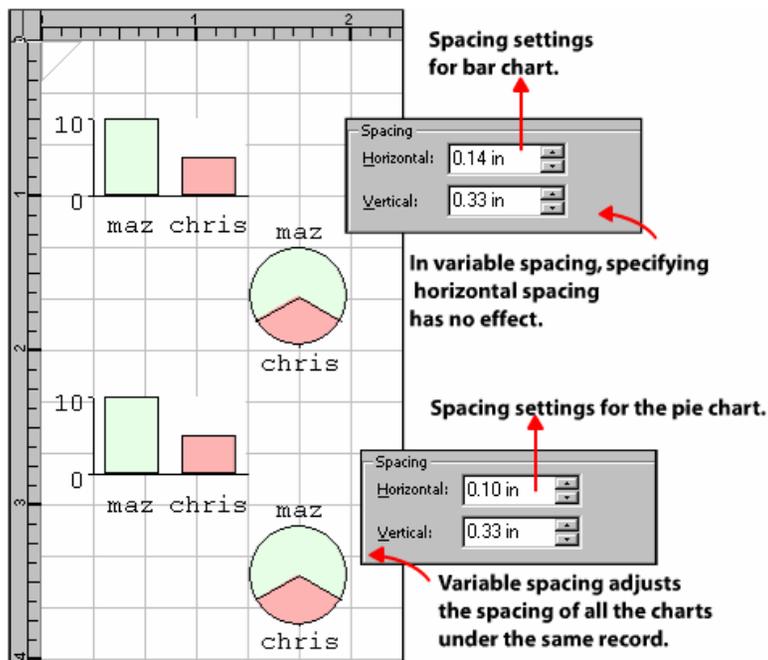
- **Vertical** – specifies the vertical spacing value of the chart.



You can also set the unit of measurement for the chart spacing by right clicking the upward or downward arrows in the **Horizontal** and **Vertical** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.



The image shows the process of specifying horizontal and vertical spacing in fixed spacing mode.



The image shows the process of specifying vertical spacing in variable spacing mode.

Applying the 3 D Effect

You can use 3D effects to enhance the chart display graphically. The 3D effect can be applied to a bar, pie, and line chart. A 3D chart depicts the three dimensions, namely: height, width, and depth.



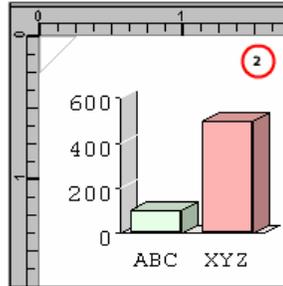
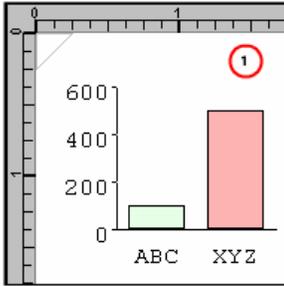
The third dimension, **depth**, is indicated by means of perspective and by techniques such as shading or gradient use of color.

To apply the 3D effect to your chart:

- 1 Select the required chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **3D effect** option; the associated options are enabled.
- 4 Specify the following:
 - **Angle** – sets the chart angle, per specified unit of measure.
 - **Thickness** – sets the chart thickness in terms of the bars/ slices/ lines width, in percentage.



You can specify the angle in degrees, radians, or grads. To specify the unit of measurement for the rotation angle, right click the upward or downward arrows in the **Angle** entry box, and then select **Angle>unit name** from the pop-up menu.



Example: Bar chart set to 1) with no 3D effect 2) with 3D effect; the angle is set to 27deg and thickness is set to 35%.

Setting the Chart Background Color

You can select a chart background color and apply it to the selected chart object.



You cannot apply a background color to the pie chart.

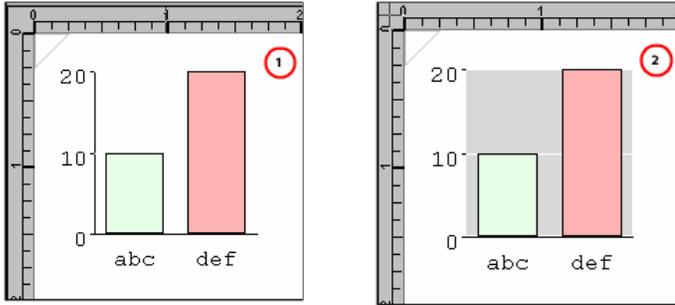
To set a chart background color:

- 1 Select a bar or line chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the required background color option from the **BG color** drop-down list.

The background color applies to the selected chart object.



We recommend not to use a background color that makes the values for the specific labels difficult to read.



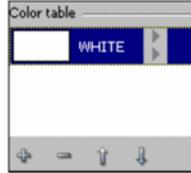
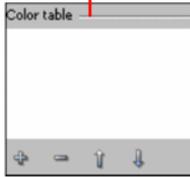
Example: Bar chart with BG color set to 1) None, 2) LMEDIUM.

Setting a Color Table for the Chart

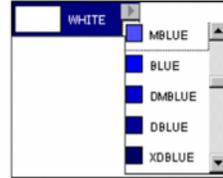
The **Color table** area in the **Settings** tab of the chart property sheet enables you to create a combination of different colors and patterns. You can then apply these colors and patterns to your charts to enhance the visual display.

Follow the steps in the following image to define a color table for a chart, and apply the colors and patterns to the chart.

1. From the Settings tab of the chart property sheet, click  to add a color cell to the color table.

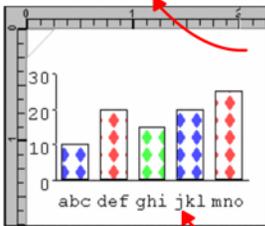
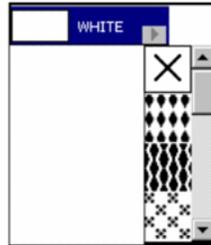


2. Click the first arrow  in the newly added color cell to select a color from the color drop-down list.



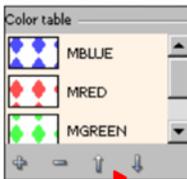
To delete a selected color from the color table click .
 To move a selected color from the color table, one row up in the color list, select it and click .
 To move a selected color from the color table, one row down in the color list, select it and click .

3. Click the second arrow  in the newly added color cell to select pattern from the pattern drop-down list.

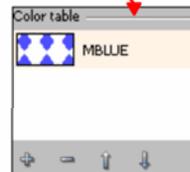
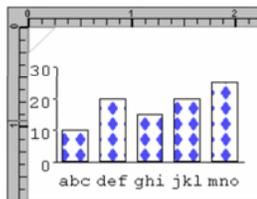


7. The colors from top to bottom in the color list apply to the bars from left to right. Once the last color in the table is applied, the colors begin to apply from the beginning of the color list.

4. The first color and pattern selected fills in the first color cell. You can see the name of the color next to the color cell.



6. Similarly, add more color cells and select colors and patterns for each of them.



5. The MBLUE color and selected pattern applies to all the bars of the chart.

Setting a Chart Legend

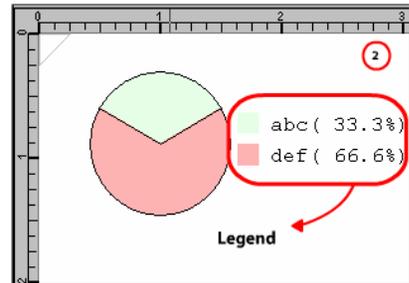
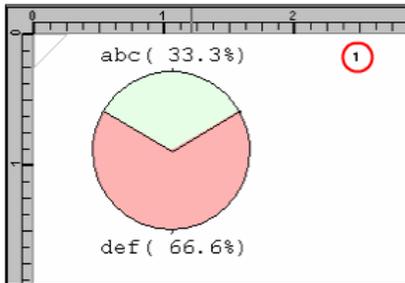
You can choose to display the chart labels as legends. Legends identify text that describes a chart and are usually printed along the side of the chart. The description involves the data based on which the chart is printed; and different colored boxes adjacent to the description that denote the corresponding different colored bars/pie slices/lines on the chart.

To define a legend for a chart:

- 1 Select the required chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the **Legend** option; the associated options are enabled.
- 4 Specify the following:
 - **Spot Offset** – adjusts the distance of the legend from the chart, in percentage.
 - **Spot Size** – adjusts the size of the legend, in percentage.



You cannot see the chart labels on the horizontal axis of the **Bar** and **Line** chart, and around the slices of the **Pie** chart when the **Legend** option is enabled.



Example: Pie chart set 1) with no Legend, 2) with legend, the Spot offset is set to 109 % and Spot size to 104 %.

Setting the Chart Border

Vitesse offers different formatting features that you can apply to the borders of a chart, including various line patterns and colors.

To set the chart border:

- 1 Select the required chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.

Setting the Chart Border Width

- 1 In the **Border** area, set the required border width in the **Width** entry box.
The border width applies to the chart edges and to the horizontal and vertical scale of the chart.

Setting the Chart Border Color

- 1 Select the required color from the **Color** drop-down list. The chart border displays with the selected color.

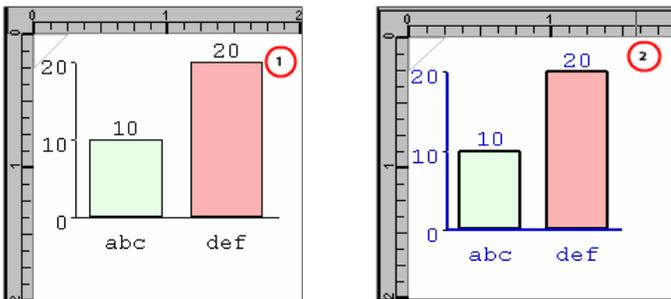
Setting the Chart Border Pattern

- 1 Select the required pattern from the **Line Pattern** drop-down list. The chart border is updated with the selected pattern.

Setting the Chart Text and Scale Color

In addition to setting the chart border, you can also set the text color for labels and values of the chart; and apply color to the scale of the bar or line chart.

To set the text color for the label and values of the chart, select the required color from the **Color** drop-down list in the **Text** area, from the **Settings** tab of the *Property View* window. The same color also applies to the horizontal and vertical chart scale.



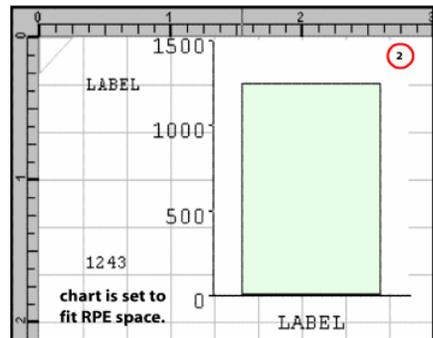
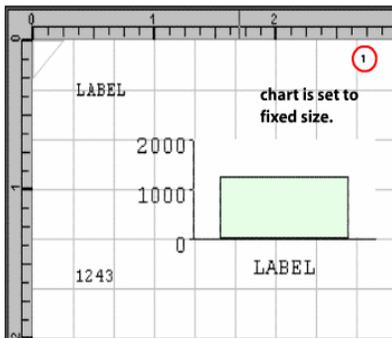
Example: Bar chart with 1) text and scale color set to black 2) text and scale color set to DM BLUE.

Setting Chart Fit Sizing Option

Vitesse enables you to set the chart fit sizing options. However, these options are applicable only when data is mapped to a chart using **DDG Index**. To map data to charts using **DDG Index**, fields are specified from the data file and assigned as label and value pairs. The field label and value pairs are then stored in the **DDG Index** and used as chart labels and values to represent relationships between different sets of data using charts. These options become available when fields specified as labels and/or values are processed prior to using them as chart labels and/or values; and also when a chart is placed at the record level. For more information, see “Using Fields as Chart Data” section in this chapter.

To set the chart fit sizing option:

- 1 Select the required bar chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 From the **Fit Space** drop-down list in the **Miscellaneous** area, select one of the following chart fit sizing options:
 - **Fixed size** – sets chart at the specified size. Adjusting the position of the fields does not affect the size of the chart mapped on these fields.
 - **Fit RPE space** – sets the chart height dynamically such that the height of the chart is set equivalent to the vertical distance between the first and the last RPE fields.



Example: fit space for bar chart 1) set to Fixed size option 2) set to Fit RPE space option.

- **Cancel 3D** – set to cancel 3D if the radius of the pie chart does not fit the vertical distance between the first and the last RPE fields (termed as RPE space). This option is valid only for pie charts and when the 3D effect option is enabled.

- **Force 3D** – set to force 3D if the radius of the pie chart does not fit the vertical distance between the first and the last RPE fields (termed as RPE space). This option is valid only for pie charts and when the 3D effect option is disabled.
- **Adjust 3D** – set to adjust 3D to fit with respect to the vertical distance between the first and the last RPE fields (termed as RPE space). This option is also valid when 3D option effect is enabled.

Setting the Space Between Bars

You can set the bar space, which determines the horizontal spacing between the parallel vertical bars when data is represented using the bar chart.

To set the space between bars:

- 1 Select the required bar chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Set the required bar spacing value, in percentage in the **Bar space** entry box.

The bar space applies to the selected chart object.

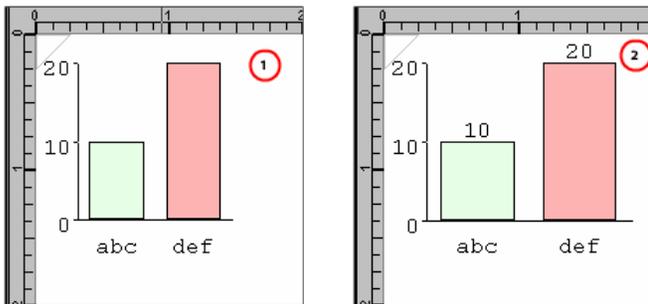


The **Bar space** option only applies to a bar type chart.

Printing Data Values

You can print the assigned numerical quantity (values) for each label on the chart.

To print the data values for each chart label, select the **Print Value** option in the **Misc** tab of the chart *Property View* window.



Example: Bar chart with 1) Print Values option not selected 2) Print Values option selected

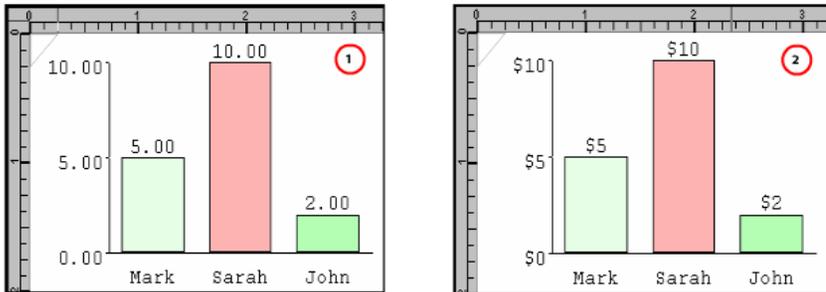
Setting the Value Format

You can set the display format for the values printed for each label on the chart using the format masks. In addition, you can also set the display format of the values defining the scale on the Y-axis in case of bar and line chart.

To set the value format:

- 1 Select the required chart object in the design area.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the required format mask from the **Format** drop-down list to apply it on the values of the selected chart. The **Format** drop-down is editable; you can also add a new format mask to the list by typing it in the entry box.

The selected format mask applies to the values of each label and also to the values defining the scale on the Y-axis. For more information on format masks, refer to *Chapter 10: Applying Transformations*.



Example: Setting Value Format: 1) format set to @, @ @.## 2) format set to \$, @ @, @#.

Merging Values in a Chart

If a chart contains same labels that represent different entities and values, you can merge the entities having same labels as a single entity. Values corresponding to each label are also merged (added) to represent one value against one label. You can merge values for all chart types.

To merge labels and values for a selected chart, select the **Merge Value** option in the **Settings** tab of the *Property View* window.

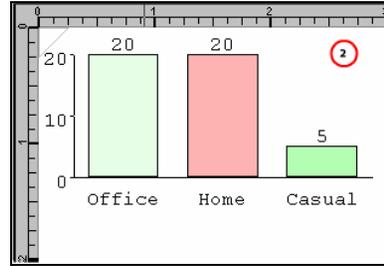
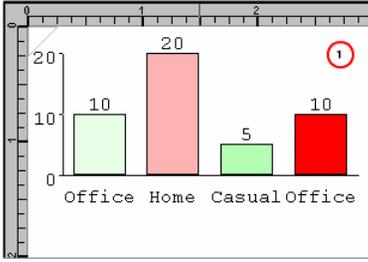


Image shows bar chart with 1) Merge Value not selected, 2) Merge Value selected.

The above example illustrates merging values in a chart. When the **Merge Value** option is selected (in the second image), the entries with the same label, in this case, vertical bars labeled as **Office**, merge into one and their values are added to show one vertical bar at **20** units of measurement on the Y-axis.

Maintaining Chart Ratio

This feature is available only when 3D effects are applied on the chart object. It is used to maintain chart ratio such that the depth is proportional to the width of the chart. To maintain the chart ratio, select the **Keep ratio** option in the **Settings** tab of the *Property View* window.

Setting Range for the Y-axis

Vitesse enables you to set the range for the Y-axis of the chart. These ranges are set dynamically depending on the data values used in the chart.

To set range for the Y-axis:

- 1 Select the required bar chart object in the design area.
- 2 From the *Property View* window, select the **Misc** tab.
- 3 Specify the following:
 - **MaxVal** – sets the maximum value of the Y-axis.
 - **MinVal** – sets the minimum value of the Y-axis.



To print the scale or the range of the Y-axis, select the **Print Scale** option in the **Misc** tab of the required chart *Property View* window.

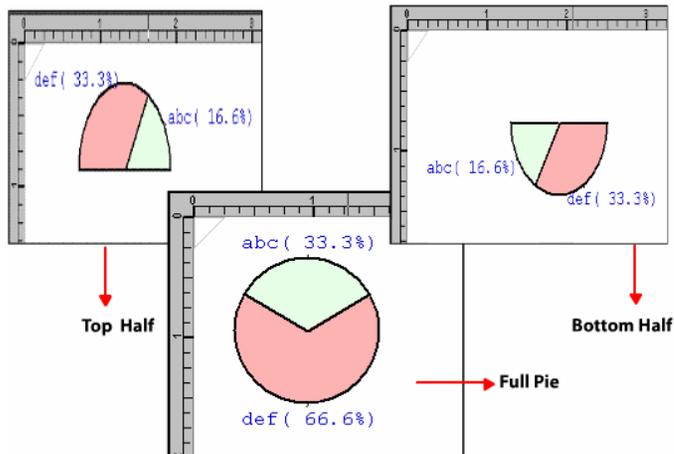
Setting the Pie Chart Properties

In Vitesse, you can present values as percentages (slices) of a whole (a pie) by using the pie chart type. A pie chart divides the circle into sectors proportional in angle and area to the relative size of the quantities represented. You can change the pie slice distance from the center and set various pie chart display options. Additional options enable you to set the chart label color, width, and display offset.

Selecting the Pie Chart Display Type

To select the pie chart display type:

- 1 Select the required pie chart object in the design area.
- 2 From the *Property View* window, select the **Misc** tab.
- 3 In the **Pie properties** area, set the required pie chart display type option from the **Half Pie** drop-down list:
 - **Full pie** – displays a full pie chart.
 - **Top half**– displays the top half of the pie chart.
 - **Bottom half** – displays the bottom half of the pie chart.



Customizing the Pie Chart Labels

You can customize the pie chart labels by setting the label dash color, label dash width, and the label offset.



The label dash is only visible when the legend option is disabled for pie chart type.

To customize the pie chart labels:

- 1 Select the required pie chart object in the design area.
- 2 From the *Property View* window, select the **Misc** tab.

Setting the Label Dash Color

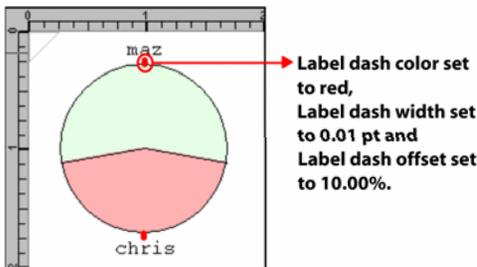
- 1 In the **Pie properties** area, select the required label dash color option from the **Label dash color** drop-down list. The label dash is set to the selected color.

Setting the Label Dash Width

- 1 In the **Pie properties** area, select the required label dash width option from the **Label dash width** drop-down list. The label dash is set to the selected width.

Setting the Label Offset

- 1 In the **Pie properties** area, select the required label offset option from the **Label offset** drop-down list. The label dash is set to the selected offset value as a percentage of the radius.



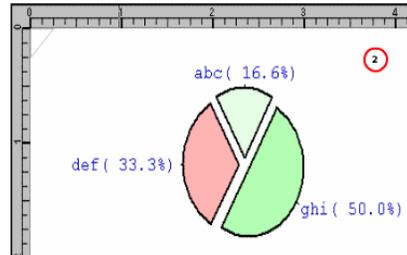
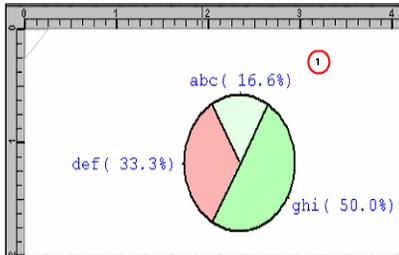
Setting the Slice Burst Value

You can set the pie slice distance from the center as a percentage of the radius. The greater the percentage of the slice burst value; the greater will be the pie slice distance from the center.

To set the slice burst value:

- 1 Select the required pie chart object in the design area.
- 2 From the *Property View* window, select the **Misc** tab.
- 3 In the **Pie properties** area, select the required pie slice distance option from the **Slice burst** drop-down list.

The pie is set to the selected percentage slice burst value.



Example: Pie chart set to 1) slice burst value 0%, 2) slice burst value 10%.

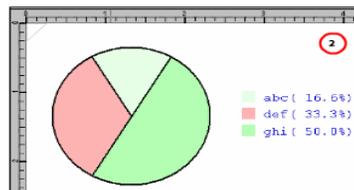
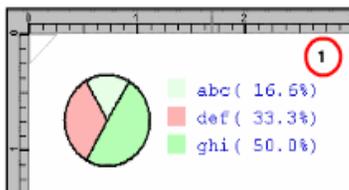
Setting the Pie Chart Radius

To set the radius of the pie chart:

- 1 Select the required pie chart node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Size** area, specify the radius of the pie chart in the **Radius** entry box.



You can set the unit of measurement for the pie chart radius by right clicking the upward or downward arrows in the radius entry box, and then selecting **Units**>*unit name* from the pop-up menu.



Example: Pie chart radius set to 1) 0.40 in, 2) 1.00 in.

Mapping Data to Charts

Vitesse enables you to map data to charts. The data for charts is mapped as label-value pairs, where label identifies an item and value is the assigned numerical quantity of a particular item. You can map variable data to and/or enter constants for chart label and value pairs. Vitesse provides the following options for mapping data to charts in the **Data** tab of the chart property sheet:

- **Using the DDG Index**

A DDG Index is an array for storing chart label and values. To map data to charts using DDG Index, fields are specified from the data file and assigned as label and value pairs. The field label and value pairs are then stored in the DDG Index and used as chart labels and values to represent relationships between different sets of data using charts.

- **Using Variables**

You can use predefined variables as chart data by defining array type variables. The array values are specified as chart labels and values. These chart label and value pairs are then mapped to the chart in order to compare quantitative relationships.

- **Using Data from the Data File (Manual Data)**

Data can be manually mapped to charts using either of the following:

- **Mapping Field Data** – you can use field data from the data file in different project modes and assign them as chart label-value pairs.
- **Mapping Variables** – you can define string and number type variables and assign them as chart label-value pairs.
- **Mapping Constant** – you can define chart label and value pairs such that label is mapped to a text constant and value to a numeric constant.

Use the following tasks for a detailed overview on mapping data to charts using the above mentioned techniques.

Using DDG Index to Map Fields as Chart Data

Vitesse enables you to map data to charts from data files by using a **Data Driven Graphics (DDG)** index for Line, Database, and XML mode projects. You can set a DDG index, which is used as an array for storing chart labels and values. Using a DDG index for charts helps you accommodate large data files and make it easier for you to align the data file data with chart legends in the design area.

Specifying Fields as Labels and Values

To map data to charts using the DDG index, you need to insert at least one field in the design area and the project tree. You can then specify the field as label or value using the field *Property View* window. Specify subsequent fields as labels and/or values in a like-wise manner.



A **Label** identifies the chart value. You can specify a field as a label. If a field contains multiple instances, each instance of a field will act as a separate label. Similarly, if you specify a field for the value that has multiple instances, each instance of that field will act as a separate value. The **Value** refers to the actual content that builds the chart.

Storing Field Label and Value Pairs

The field label and value pairs are then stored in the DDG index array and are used as chart label and value. When the chart is printed, the DDG index becomes empty in order to store labels and values for subsequent charts.



We recommend that the number of labels and values be equal, and that fields specified as labels and/or values be processed prior to using them as chart labels and/or values.

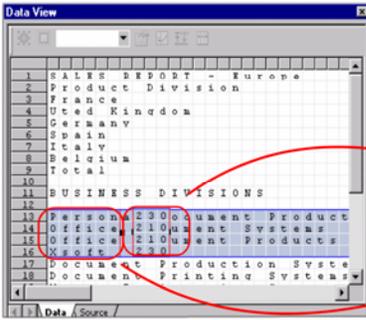


DDG chart indexing is disabled for **Native Mode** projects.

To use fields as chart data, insert a chart node at the Page or Record level and select the **DDG Index** option from the **Data** tab of the chart *Property View* window.

The task described on the next page provides an example of using fields as chart data in a line mode project having a non-prefixed data file (non-delimited). The chart is added at the page level. At this stage, we are assuming that you are familiar with mapping records and fields to the design area and setting record and field properties. Therefore, the example focuses on specifying fields as chart labels and values and using them to map data to charts.

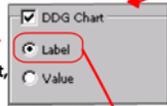
1. Insert a field object from the Insert toolbar and assign it with the name 'Label'. Specify Field Data as:



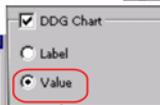
2. Insert another field object from the Insert toolbar and assign it with the name 'Value'. Specify Field Data as:



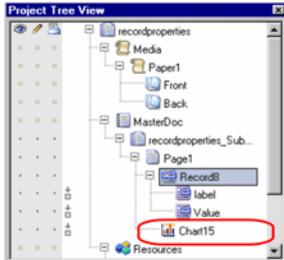
3. In the Label field property sheet, select the Settings tab. Select the DDG Chart and set field as Label. This field will be used as chart label.



4. In the Value field property sheet, select the Settings tab. Select the DDG Chart and set field as Value. This field will be used as chart value.



7. The chart adds under the Page node in the Project tree View window.



6. Select the DDG Index. This will map the fields data as label and values of the selected bar chart.



5. Click [Chart Icon] on the Insert toolbar to create a chart object under the Page node in the project tree. In the chart property sheet, select the Data tab.

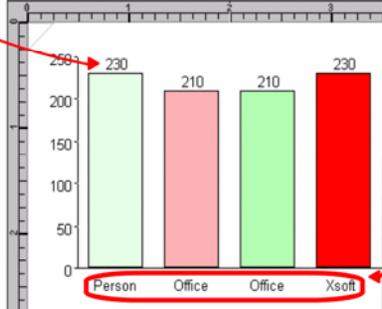


Image shows the process of using fields as chart data. Chart is added at the page level.

Using Predefined Variables as Chart Data

Array type variables can be used for defining chart data, where values of the array type variable are used as chart labels and values. These chart label-value pairs are then mapped to a chart object. To use predefined variables as chart data, insert a chart node at the Page or Record level and select the **Variable** option from the **Data** tab of the chart *Property View* window. You can use predefined variables as chart data in all the project modes. Refer to the following step-by-step illustration to use predefined variables as chart data.



You must use the MasterDoc Property sheet to define variables as an array. An array variable can contain a list of values. These values can be set to numbers, strings, system variables, and variables defined for the project. For more information on creating an array variable, see “Setting up an Array Variable” in *Chapter 26: Using Variables*.

1. Create an array variable.

2. Specify the chart label as constant text.

3. Specify the chart value as constant number.

4. Create all the key value pairs. Labels should always be placed before values, following a strict sequence of Text (Label), Number (Value); Text, Number and so on.

5. Click , on the Insert toolbar to create a chart object.

6. In the chart property sheet, select the Data tab.

7. Select Variable, and then select the array variable from the drop-down list.

8. The array values added as text display as chart labels and the values added as numbers display as chart values.

Label	Value
a	10.00
b	5.00
c	15.00

Image shows the process of using predefined variables as chart data.

Using Data from the Data File (Manual Data)

You can manually map data to charts by inserting a chart node at Page or Record level and by selecting the **Manual Data** option from the **Data** tab of the chart *Property View* window.

Data can be manually mapped to charts using either of the following techniques:

Manually Mapping Field Data

Vitesse enables you to manually map field data by selecting data from the data file. The data mapping options vary depending on the project mode and type of data file.



You cannot map field data for **Native** mode project.

You can manually map field data to charts in Line mode (from prefixed and non-prefixed data files), Database, and XML mode projects. This section provides different exemplary tasks for mapping field data to charts.

Mapping Data to a Page Level Chart – Non-Prefixed Line mode Data

This task provides an example of mapping data to a line mode project having a non-prefixed data file (delimited). At this stage, we are assuming that you are familiar with mapping records and fields to the design area and setting record and field properties. Therefore, the example focuses on manually mapping field data to a Page Level chart, from the data file.

The **Data Mapping** sub-menu, which is used to identify the label and value pair, contains the following options:

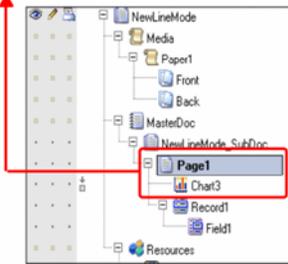


These options are enabled for selection depending on delimited or non delimited data file, and whether chart is inserted at record or page level.

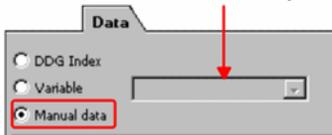
- **FIELD(#)** – maps user specified field number from non-prefixed delimited data file to chart at record level.
- **FIELD(#-#)** – maps user specified field range from non-prefixed non-delimited data file to chart at record level.
- **LINE(#-#).FIELD(#)** – maps user specified line range and field number from non-prefixed delimited data file to chart at page level and/ or maps user specified field number from non-prefixed delimited data file to chart at record level.
- **LINE(#-#).FIELD(#-#)** – maps user specified line range and field range from non-prefixed non-delimited data file to chart at page level and/ or maps user specified field range from non-prefixed non-delimited data file to chart at record level.

See the image below to understand the process of mapping field data in Line mode non-prefixed (delimited) data file to a page level chart.

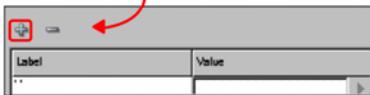
1. Insert a Chart node under the Page node in the Project tree.



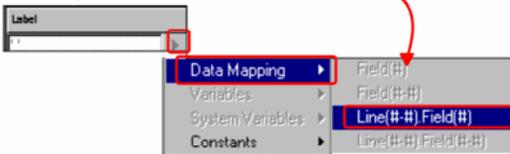
2. From the chart property sheet, select the Data tab. Select the Manual Data option.



3. Click [Add] to add a new Label/ Value entry box pair.



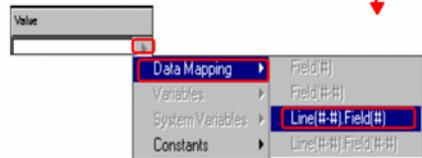
4. Click the Label entry box to display the arrow to the right of the entry box. Click the [Data Mapping] and select Data Mapping from the pop-up menu; associated sub-menu displays options that enable you to select data from the data file.



5. In the Label entry box, select Line(##).Field(#) option, and enter Line (6 - 8). Field (1). This will map the specified line range and field number from non-prefixed delimited data file to chart , thus assingng 'label' values.

	Field(0)	Field(1)
1	CHER	TITRE
2	ACS1	Asset Allocation Summary
3	ADRO	
4	ACS3	BOND, NOTE
5	ACS3	VARIOUS
6	ACS3	CASH
7	ACS3	SHARE
8	ACS4	TOTAL

6. Click the Value entry box to display the arrow to the right of the entry box. Click the [Data Mapping] and select Data Mapping from the pop-up menu; associated sub-menu displays options that enable you to select data from the data file.



7. In the Value entry box, select Line(##).Field(#) option, and enter Line (4 - 6). Field (5). This will map the specified line range and field number from non-prefixed delimited data file to chart , thus assingng 'Value' values.

	Field(5)	Field(6)
1	ADDRESS2	ZIP
2		
3		
4	41.85	
5	11.17	
6	32.70	
7	45.28	
8		

7. The Label and Value pair is mapped on to the chart.

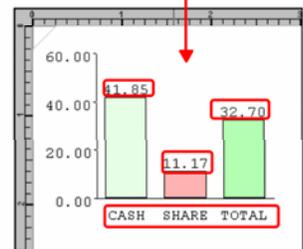


Image shows the process of manually mapping field data using Non-Prefixed Line Mode Data. Chart is added at page level.

Mapping Data to a Record Level Chart – Prefixed Line mode Data

This task provides an example of mapping data to a line mode project having a prefixed data file (non-delimited). The example focuses on manually mapping data to a chart added at Record level from the data file. In this example, the number of instances of the RPE key of the record under which the chart is placed, in the data file, is one. Therefore, the chart prints only once on the page.

The **Data Mapping** sub-menu, used to identify the label and value pair, contain the following options:



These options are enabled for selection depending on delimited or non delimited data file, and whether chart is inserted at record or page level.

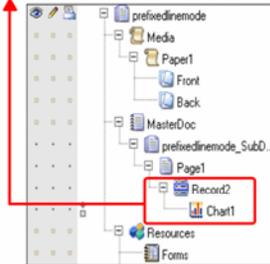
- **REC('').FIELD(#)** – Map user specified record prefix and field number from prefixed delimited data file to chart.
- **REC('').FIELD(#-#)**– Map user specified record prefix and field range from prefixed non-delimited data file to chart.



When the chart is placed at the record level, the number of instances of the RPE key of the record under which it is placed determines the number of times the chart will print on the page.

See the image below to understand the process of mapping field data in Line mode prefixed (non-delimited) data file at the record level.

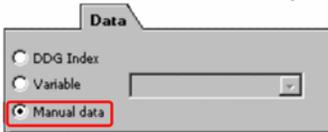
1. Insert a Chart node under the Record node in the Project tree.



2. From the record property sheet, select the Settings tab. Set the RPE key to MNT0. There is only one instance of the RPE key within the datafile. Therefore, the chart placed under this record will print only once.



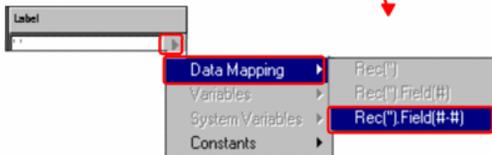
3. From the chart property sheet, select the Data tab, and then select the Manual Data option.



4. Click to add a new Label/ Value entry box pair.



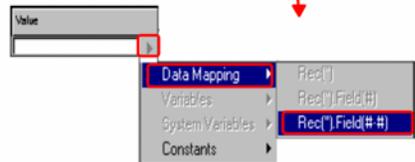
5. Click the Label entry box to display the arrow to the right of the entry box. Click and select Data Mapping from the pop-up menu; associated sub-menu displays options that enable you to select data from the data file.



6. In the Label entry box, select Rec(').Field(##) option, and enter Rec(MNT0).Field(15-17). This will map the specified record prefix and field range from prefixed non-delimited data file to chart, thus assigning the first 'label' values.

23	L	F	A	0	4	1	A	S	S	A	I	N	I	S	S	E	M	E
24																		
25	M	N	T	0	M	O	N	T	A	N	T	A	P	A	Y	R		
26																		
27	B	R	E	0	0	0	0	0	1	4	1	5	3	0	1			
28																		
29	O	L	I	0	0	1	0	0	0	0	4	6	9	6	0	9	>	0

7. Click the Value entry box to display the arrow to the right of the entry box. Click and select Data Mapping from the pop-up menu; associated sub-menu displays options that enable you to select data from the data file.



8. In the Value entry box, select Rec(').Field(##) option, and enter Rec(MNT0).Field(26-28). This will map the specified record prefix and field range from prefixed non-delimited data file to chart, thus assigning first 'Value' values.

23	L	F	A	0	4	1	A	S	S	A	I	N	I	S	S	E	M	E
24																		
25	M	N	T	0	M	O	N	T	A	N	T	A	P	A	Y	R		
26																		
27	B	R	E	0	0	0	0	0	1	4	1	5	3	0	1	7	6	4
28																9	6	0
29	O	L	I	0	0	1	0	0	0	0	4	6	9	6	0	9	>	0

9. Add a second Label/ Value entry pair. In the Label entry box, enter Rec(MNT0).Field(26-28). In the Value entry box enter Rec(MNT0).Field(26-28). This will map the specified record prefix and field range from prefixed non-delimited data file to chart, thus assigning the second Label/Value pair.

10. The Label and Value pairs are mapped on to the chart.

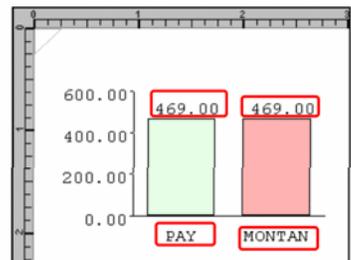


Image shows the process of manually mapping field data using Prefixed Line Mode Data. Chart is added at record level.

Mapping Data to a Chart - Database Mode Data

This task provides an example of mapping data to a database mode project. The example focuses on manually mapping data to a chart by assigning field data as label and value pairs picked from the data file. For a database mode project, you can only insert a chart at page level. When working with database mode data, multiple records data, assigned as label and value pairs cannot be mapped to one single page.

The **Data Mapping** sub-menu, used to identify the label and value pair, contain the following option only:

- **FIELD('')** – maps user specified field name from data file to chart at page level.

See the image below to understand the process of mapping field data in database mode project.

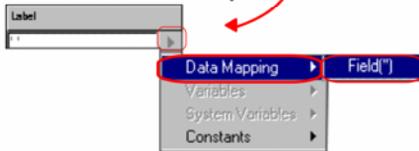
1. Click  on the Insert toolbar to create a chart object.
2. From the chart property sheet, select the Data tab, and then select the Manual Data option.



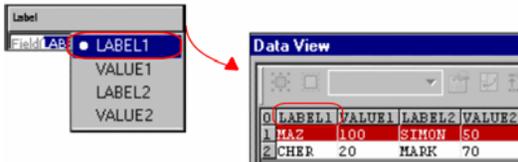
3. Click  to add a new Label/ Value entry box pair.



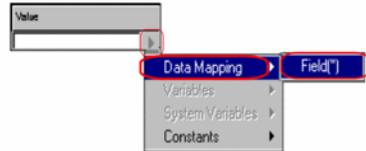
4. Click the Label entry box to display the arrow to the right of the entry box. Click  and select Data Mapping from the pop-up menu. Select the Field("") option.



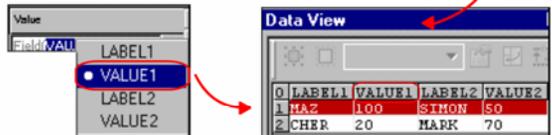
5. In the Label entry box, select LABEL1 field name to assign the first label.



6. Click the Value entry box to display the arrow to the right of the entry box. Click  and select Data Mapping from the pop-up menu. Select the Field("") option.



7. In the Value entry box, select VALUE 1 to assign the first value.



8. Repeat step 4 - 8 to assign the second label and value pair.



9. The label and value pairs are mapped on to the chart.

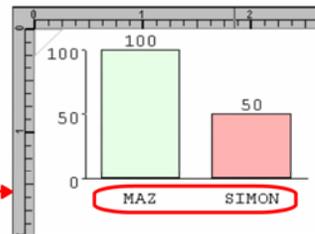


Image shows the process of manually mapping field data using Database Mode Data.



For more information about mapping data in Database Mode, refer to *Chapter 20: Working with Data in Database Mode*.

Mapping Data to a Chart – XML Mode Data

This task provides an example of mapping data to a XML mode project. For a XML mode project, you can map user specified tag names from the data file to chart by assigning them as label and value pairs.



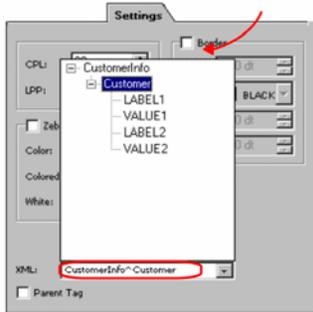
For more information about mapping data in XML Mode, refer to *Chapter 24: Working with Data in XML Mode*.

In this example, there are two customer tags in the sample data file. The data is mapped such that a new instance of the logical page is created for each instance of the data tag. Therefore, each occurrence of the data tag displays on a different page and each page corresponds to a different document. The **Data Mapping** sub-menu, used to identify the label and value pair, contain the following option only:

- **FIELD(‘)** – maps user specified tag names from XML data file to chart.

See the image on the next page to understand the process of mapping field data in XML mode project.

1. Select the Page node in the project tree. From the page property sheet, select the Settings tab and select CustomerInfo^Customer from the XML drop-down tree structure.



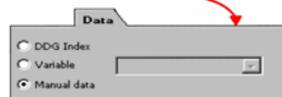
2. Click on the Insert toolbar to create a record. This enables the chart object.

3. From the record property sheet, select the Settings tab. Select the CustomerInfo^Customer option from the XML drop-down tree structure.

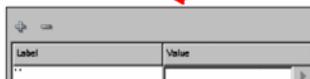


4. Click on the Insert toolbar to create a chart object.

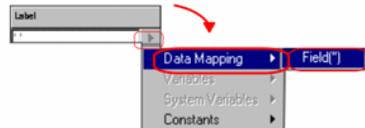
5. From the chart property sheet, select the Data tab, and then select the Manual Data option.



6. Click to add a new Label/Value entry box pair.



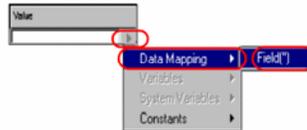
7. Click the Label entry box to display the arrow to the right of the entry box. Click and select Data Mapping from the pop-up menu. Select the Field("") option.



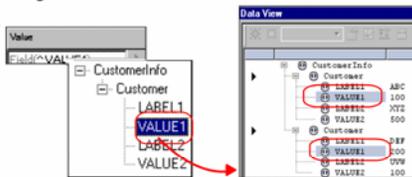
8. In the Label entry box, select LABEL 1 data tag, to assign the first label.



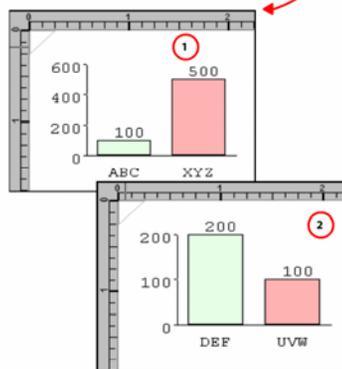
9. Click the Value entry box to display the arrow to the right of the entry box. Click and select Data Mapping from the pop-up menu. Select the Field("") option.



10. In the Value entry box, select VALUE 1 data tag to assign the first value.



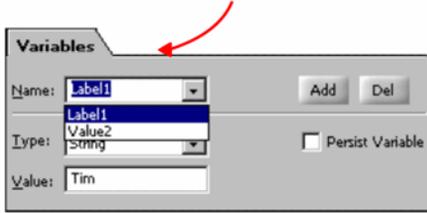
11. Repeat steps 5-10 and assign the second label value pair. The label and value pairs are mapped on to the chart. Two pages are created for two customer records.



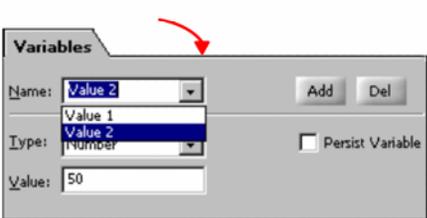
Manually Mapping Variables

You can define string and number variables and use them for defining label, value pairs for mapping variables on charts. The following image shows the process of mapping variables:

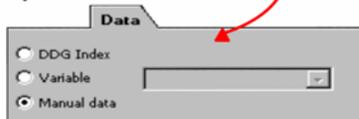
1. Create two string variables, namely Label 1 and Label 2. Assign them character strings, Tim and Max.



2. Create two number variables, namely Value 1 and Value 2. Assign them numeric values, 100 and 50.



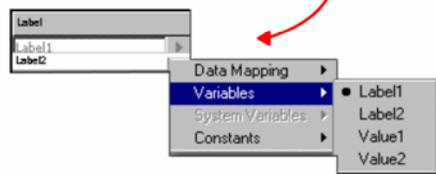
3. Click  on the Insert toolbar to create a chart object.
 4. From the chart property sheet, select the Data tab, and then select the Manual Data option.



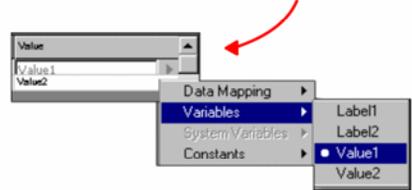
5. Click  to add a new Label/ Value entry box pair.



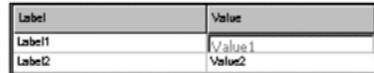
6. Click the Label entry box to display the arrow to the right of the entry box. Click  and select Variables from the pop-up menu. Select Label1.



7. Click the Value entry box to display the arrow to the right of the entry box. Click  and select Variables from the pop-up menu. Select Value1.



8. Repeat step 4 - 7 to assign the second label and value pair.



9. The label and value pairs are mapped on to the chart.

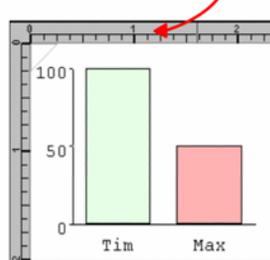
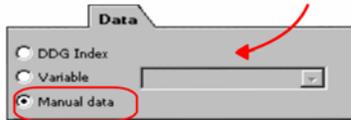


Image shows the process of manually mapping variables.

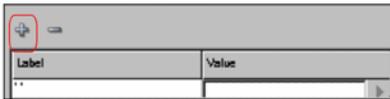
Manually Mapping Constants

You can also use text and numeric constants to define label and value pairs such that label is mapped to a text constant and value to a numeric constant. These label and value pairs can then be mapped to charts. The following image shows the process of mapping constants as data for a chart object:

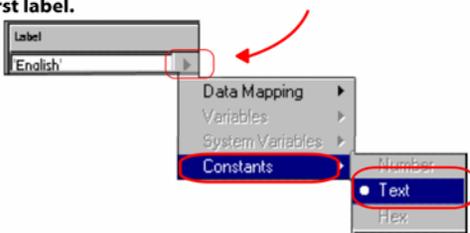
1. Click  on the Insert toolbar to create a chart object.
2. From the chart property sheet, select the Data tab, and then select the Manual Data option.



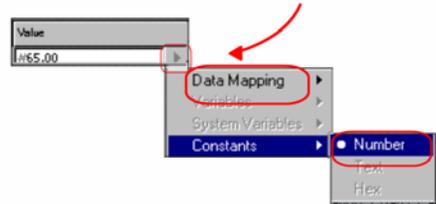
3. Click  to add a new Label/ Value entry box pair.



4. Click the Label entry box to display the arrow to the right of the entry box. Click  and select Constants from the menu. From the associated sub-menu, select the Text option. Enter English as text for the first label.



5. Click the Value entry box to display the arrow to the right of the entry box. Click  and select Constants from the menu. From the associated sub-menu, select the Number option. Enter 65 as numeric value for the first Value.



6. Follow step 3- 5 to assign the second label and value pair.

Label	Value
English	#65.00
'Maths'	#90

7. The label and value pairs are mapped on to the chart.

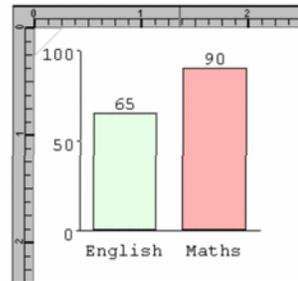


Image shows the process of manually mapping constants.



For information on the **Font** tab, refer to *Chapter 6: Working with Fonts*.

Chapter 29: Using Actions

This chapter describes activities and tasks for working with actions in Vitesse. Included are instructions and information for the following:

- Understanding Actions
- Inserting an Action
- Creating a Booklet
- Setting Document Control Printing Options
- Setting Document Set Options
- Specifying Finishing Operations
- Setting Page Control Options
- Inserting Inline VIPP Code
- Modifying Variables Using Actions
- Assigning Field Value to a Variable
- Setting Variables as PDF Bookmarks

Understanding Actions

Action object represents the various actions that a user can perform in a Vitesse project. Actions are most commonly used with conditions. You can use an action object to:

- Set booklet printing options for a project.
- Set document control options, including printing mode and orientation.
- Prompt the printer to execute a specific action while printing.
- Apply finishing operations and create document sets.
- Set page control options.
- Set or change project variables.
- Insert inline VIPP commands.

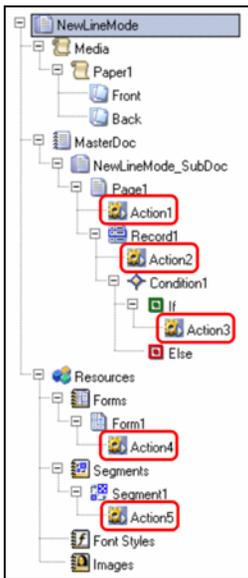
Actions display as tree nodes in the project tree, but are not visible in the design area. At print time, all nodes in the project tree are processed sequentially from top to bottom. Therefore, action object placement in the project tree is critical, as it determines when the printer executes a specific action. Action objects can be used in all of the project modes, as well as in form and segment resources. Source code for the action object source code is located within the application source code and mirrors the object placement in the project tree.

Inserting an Action

You can add action objects under the page and record objects when designing a Vitesse project. You can also use action objects in conjunction with page and record level conditions to ensure that certain actions are executed only when specified conditions hold true. When creating resources, actions can also be used in form and segment designs.



In an XML mode project, an action object cannot be directly added to a page. You need to insert a record object first in order to use the action object.



Action objects can be added to page, record, condition, form, and segment objects.

To insert an action object in a project or a resource:

- 1 Select the required node in the project tree, under which you want to add an action object.
- 2 Select **Insert**>**Advanced**>**Action** from the menu bar; an **Action** node is created under the selected node in the project tree.



You can also insert an action object by clicking  on the *Insert* toolbar.



You can delete a selected action object either by:

- Selecting Edit>Delete from the menu bar.
- Or
- Clicking  on the *Standard* toolbar.
- Or
- Pressing on the keyboard.

Setting Action Properties

You can use the action object property sheet to define the action to be performed. You can use an action object to create booklets, set document and page control, apply finishing operations, and modify variables defined for the project. You can also apply transformations to variables using the **Transform** tab in the action object property sheet.



One action object corresponds to one action that can be performed. For example, an action can either set booklet options or specify finishing operations, not both. To perform more than one type of actions, use multiple action objects.

Creating a Booklet

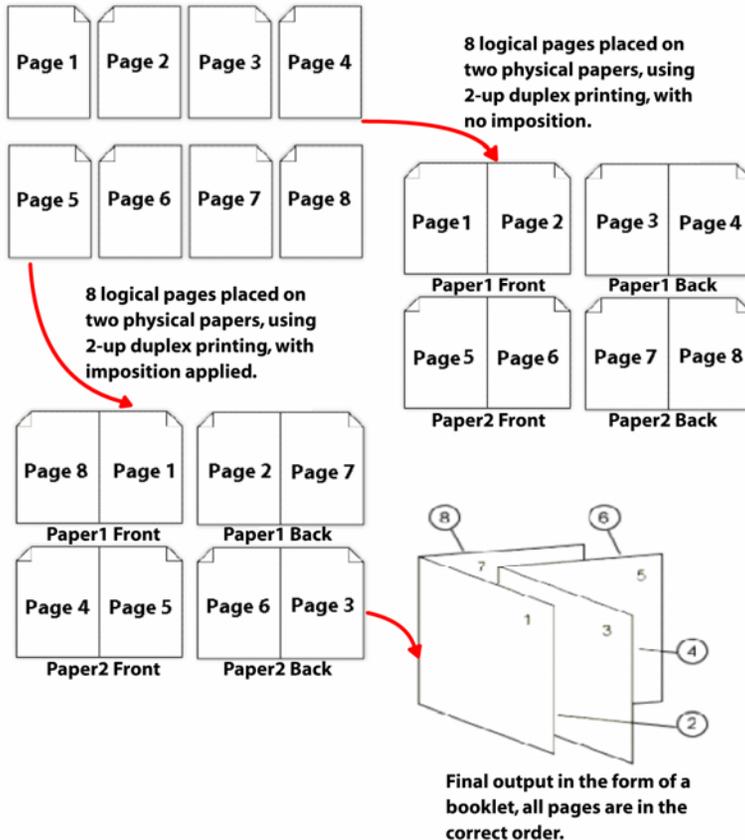
When working with paper partitions, pages are printed in a sequential order (1, 2, 3, and so on) on the partitions by default. This default process of printing pages is convenient for reading, but if the sheets are folded to form a booklet, the sequence in which the pages are printed on the partitions does not match the normal reading order. To enable printing pages in the right order, you can use the page imposition feature, in which multiple pages are printed on a piece of paper so that when folded or cut, the pages are arranged in a proper reading order.

Vitesse enables you to create booklets using the action object. Using the imposition and booklet options provided in the action object, you can generate the final output in the proper printing order so that post processing operations such as folding, cutting, and stapling, etc. can be applied.

The imposition or booklet printing is defined for a range of pages. You need to define the range of pages by marking the range start and end points using the action objects.



Booklet processing is not supported by all print drivers. Before using this option, verify support for this feature on your printer.



The image displays the partition order for booklet printing.

To create booklets using actions:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Booklet** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Booklet** option, which enables all of the related booklet and imposition options.
- 4 As appropriate, select one of the following:
 - **Begin Imposition** – turns on the imposition feature; prints a document that contains a collection of images (for example, TIFF), segments, and/or EPS files so that the printed pages form a booklet. Use this option with a multi-up setting, as well as duplex printing mode.
 - **End Imposition** – turns off the imposition feature.



We recommend you use the **Begin Imposition** and **End Imposition** actions in Native and Database mode projects.

- **Start Booklet** – starts the booklet printing. The action should be placed at the start of the first page of the booklet.
- **End Booklet** – ends the booklet printing. The action should be placed at the end of the last page of the booklet.



The **Imposition** and **Booklet** options are mutually exclusive; you cannot use both options simultaneously.

Setting Document Control Printing Options

Vitesse enables you to set document control options such as duplex printing, N-up printing, and paper orientation at run time using the action objects. The action objects are most commonly used with conditions to control the document output based on the condition outcome. The document control options specified using actions override the corresponding options set using the **Paper** object.

To set the document control options:

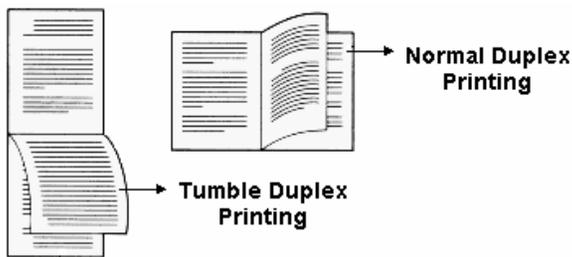
- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Document Control** from the **Category** drop-down list. The related options are displayed in the tab.

Selecting the Page Mode

- 1 Select the **Page Mode** option, which enables all of the related number up options.
- 2 As appropriate, select one of the following:
 - **One Up** – specifies that a single logical page is to be placed on the paper while printing.
 - **Two Up** – specifies that two logical pages are to be placed on the paper while printing.

Selecting the Duplex Mode

- 1 Select the **Duplex Mode** option, which enables all of the related duplex printing options.
- 2 As appropriate, select one of the following:
 - **Simplex** – prints documents so that only one side of the paper is used.
 - **Duplex** – prints documents so that both sides of the paper are used. Both paper sides have the same print orientation to enable side binding.
 - **Tumble Duplex** – prints documents so that both sides of the paper are used. The top of one paper side and the bottom of the other side print along the same sheet edge to enable top binding.



The normal duplex printing is used for side binding, whereas the tumble duplex printing is used for top binding.

Selecting the Orientation

- 1 Select the **Orientation** option, which enables all of the related orientation options.
- 2 As appropriate, select one of the following:
 - **Portrait** – sets orientation of the page to portrait; a vertical print orientation in which a page is printed across the narrower dimension of a paper, i.e., the height of the page is greater than the width.
 - **Landscape**– sets orientation of the page to landscape; a horizontal print orientation in which a page is printed sideways, i.e., the width of the page is greater than the height.
 - **Inverse Portrait** – sets orientation of the page to inverse portrait; this orientation is obtained by rotating a portrait paper by 180 degrees.
 - **Inverse Landscape** – sets orientation of the page to inverse landscape; this orientation is obtained by rotating a portrait paper 90 degrees clockwise.



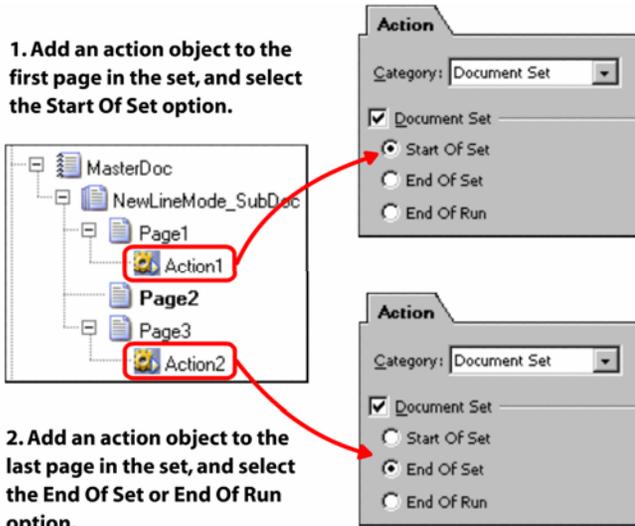
The effect of document control printing options is not visible in the Vitesse design area and is applied at production time.

Setting Document Set Options

Document set options enable you to create a subset of documents on which different finishing operations can be applied. The document set options are only available for the DocuSP print jobs. DocuSP (Document Services Platform) is a raster image processing (RIP) and print management controller for production-class printers in the Xerox DocuPrint and DocuTech families.

To create a document set:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Document Set** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Document Set** option, which enables all of the related options.
- 4 As appropriate, select one of the following:
 - **Start Of Set** – marks the starting point of a document set. You must place this action at the beginning of the first page of a set.
 - **End Of Set** – marks the ending point of a document set. You must place this action at the beginning of the last page of a set.
 - **End Of Run** – marks the ending point of a document set. You must place this action at the beginning of the last page of a set. For DocuSP printers, the **End Of Set** and **End Of Run** options are equivalent; there is no differentiation between a “set” and a “run.”



The image shows the process of creating a document set consisting of three pages.

Specifying Finishing Operations

You can use the action object to set various finishing operations for a project such as the print job staple, bind, jog, and collate options. The finishing operations are applied to a document set defined using the **Document Set** properties (see the “Setting Document Set Options” task above). If a document set has not been defined, the finishing operations are applied to the entire document. The finishing options are device dependent. Therefore, we recommend that you check your selections for compatibility with your printing device.

To set the project finishing operations using an action object:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Finishing** from the **Category** drop-down list.

The related options are displayed in the tab.

Selecting the Bind Option

- 1 Select the **Bind** option, which enables all of the related bind options. Binding refers to the process by which printed sheets are converted to a consecutive arrangement of pages, fastened together along one edge in a book.
- 2 As appropriate, select one of the following bind options:
 - **Bind On** – enables binding for DocuSP print jobs.
 - **Bind Off** – disables binding for DocuSP print jobs.

Selecting the Jog and Offset Option

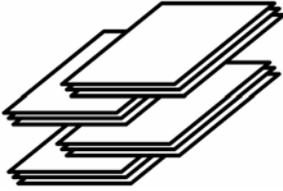
- 1 Select the **Jog/Offset** option, which enables all of the related jog and offset options. The **Offset** options stacks each set slightly shifted apart from the previous for easy retrieval, whereas the **Jog** options offsets the paper stacks on a paper-by-paper basis, rather than on a set-by-set basis in the output tray.
- 2 As appropriate, select one of the following:
 - **Jog On** – enables the page offsetting in the output tray during printing.
 - **Jog Off** – disables the page in the output tray during printing.



The image represents pages jogged in the output tray.

- **Offset On** – enables the paper set offsetting in the output tray during printing.

- **Offset Off** – disables the paper set offsetting in the output tray during printing.



The image represents sets stacked offset in the output tray.



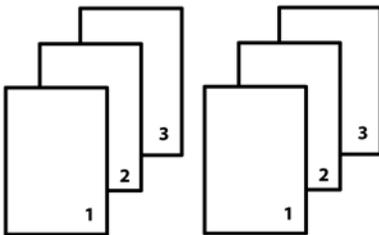
The **Offset** and **Jog** options are mutually exclusive.

Selecting the Staple Option

- 1 Select the **Staple** option, which enables all of the related staple options.
- 2 As appropriate, select one of the following:
 - **Staple On** – enables the stitching of sets in the output tray during printing.
 - **Staple Off** – disables the stitching of sets in a job in the output tray during printing.

Selecting the Collate Option

- 1 Select the **Collate** option, which enables all of the related collate options.
- 2 As appropriate, select one of the following:
 - **Collate On** – prints two or more copies of a multi-page document in normal page/binding order. For example, to print two copies of a 3-page document, pages 1, 2, and 3 are printed in this sequence as one set, and pages 1, 2, and 3 as the second set.

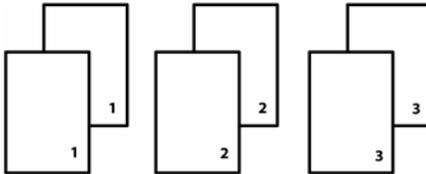


The image represents the printing sequence for two collated copies of a 3-page document.



Collate automatically sorts the printed pages and is helpful when working with a document containing multiple pages.

- **Collate Off** – prints two or more copies of a multi-page document so that all copies of the first page are printed together, then all copies of the second page are printed together, and so on.



The image represents the printing sequence for two non-collated copies of a 3-page document.

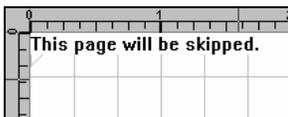
- **Collate DBM** – prints the specified number of copies immediately after each record. This option generates the same printing sequence that is generated using the **Collate Off** option and is used in the database mode projects.

Setting Page Control Options

Page control action objects provide different options for printing logical pages. You can use these actions to switch media and logical pages conditionally. You can also set actions to skip selected pages. The **Page Control** feature also provides options to move between frames and pages.

To set the page control options:

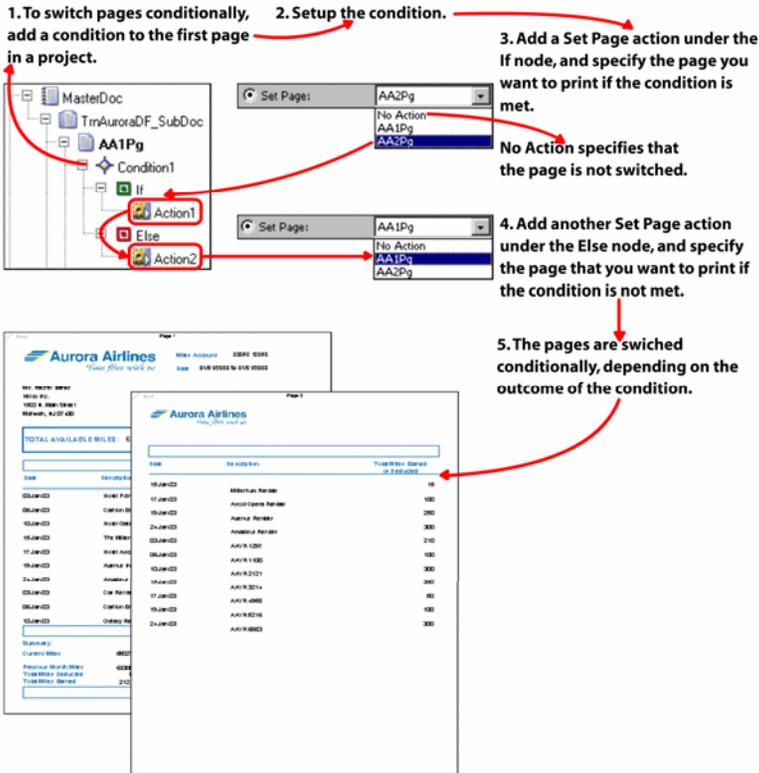
- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Page Control** from the **Category** drop-down list. The related options display in the tab.
- 3 Select the **Page Control** option, which enables all of the related page control options.
- 4 As appropriate, select one of the following:
 - **Skip Page** – skips the current page (not printed). You can use this option in conjunction with conditions to skip a page if a certain condition holds true. The following message is displayed at the top of the page in the design area which is to be skipped:



- **New Break** – causes a page break. In multi-up mode, **New Break** action causes a skip to the next logical page and the physical page only prints when the last logical page is reached.
- **New Back** – forces the current page to print on the next available back of a sheet when working in duplex mode.
- **New Front** – forces the current page to print on the front of a new sheet when working in duplex mode.
- **New Side** – forces the current logical page to print on the next physical page in multi up mode.
- **Slip Sheet** – inserts slip sheets (additional sheets are used to separate sets) in the middle of a job. Selecting this option enables the associated drop-down list. From the drop-down list, select the required paper object.
- **Set Page** – prints data on the specified page. Selecting this option enables the associated drop-down list. From the drop-down list, select the required page object. This option is only available for Line mode projects.

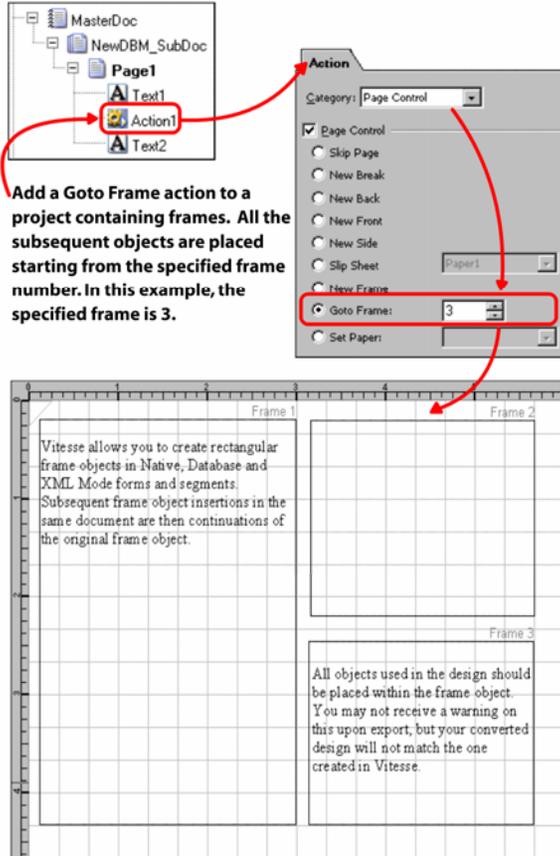


Pages can only be switched conditionally on the first page. This implies that the **Set Page** action must be used in a condition, only on the first page.



The image shows the process of switching pages conditionally using Set Page actions.

- **New Frame** – places the subsequent design objects in the next available frame. If the current frame is the last frame on the page, then a page transition occurs and the subsequent objects are placed starting from the first frame on the new page.
- **Goto Frame** – places all subsequent objects into the specified frame. Selecting this option enables the associated entry box. In the entry box, specify the required frame number. The frame number must be greater than the current frame number and less than or equal to the maximum frame number, otherwise the action is ignored.



The image shows the effect of using a Goto Frame action in a project.



The **New Frame** and **Goto Frame** actions are not available in Line mode projects.

- **Set Paper** – prints data on the specified paper. Selecting this option enables the associated drop-down list. From the drop-down list, select the required paper object.



Papers can only be switched conditionally. This implies that the **Set Paper** action can only be used in a condition.

Inserting Inline VIPP Code

Vitesse enables you to insert inline VIPP code in the generated source code using an action object. This action enables you to insert any VIPP commands to the source code that are not supported by Vitesse.

To insert inline VIPP code:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Custom** from the **Category** drop-down list.

The related options are displayed in the tab.

- 3 Specify the inline VIPP code in the **Inline VIPP Code** entry box.
- 4 Click **Apply**.

The VIPP code is added to the project source code and can be viewed using the **Source View** tab in the design area.

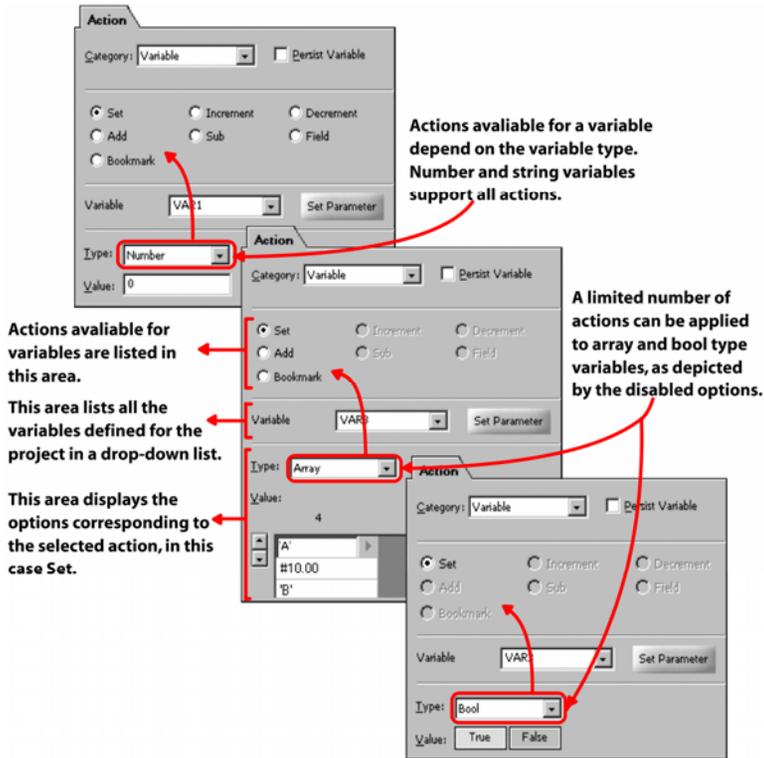
If you insert a command supported by Vitesse using inline VIPP code, the result of the code is displayed in the design area when the source code is compiled. If the inserted code corresponds to an object in Vitesse, the action object is replaced by the object corresponding to the command when the source code is compiled.

Modifying Variables Using Actions

Action objects can be used to set or change the value of variables defined for the project. Using actions, you can:

- Set a variable value.
- Create variables at runtime.
- Increment a variable value.
- Decrement a variable value.
- Add a value to a variable.
- Subtract a value from a variable.
- Set PDF bookmarks.

The actions that you can perform on a variable depend on the variable type.



The image displays the actions available for different variable types.

Setting a Variable Value

To set a variable value:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.

The related options are displayed in the tab.

- 3 Select the **Set** option, which indicates that the action object will be used to modify a variable type or value.
- 4 Select the required predefined variable from the **Variable** drop-down list.
- 5 To modify the variable type, select one of the following options from the **Type** drop-down list:
 - **String** – sets the variable type to string.
 - **Number** – sets the variable type to number.
 - **Array** – sets the variable type to array.

- **Variable** – sets the value of the variable equal to the value of another system or user-defined variable.
 - **Bool** – sets the variable type to bool.
- 6 To set the variable value, either:
- Enter the required value in the **Value** entry box for number and string type variables.
 - Update the array values in the **Value** area for array type variables.
 - Click **True** or **False** in the **Value** area for bool type variables.
 - Select the required system or predefined variable from the **Value** drop-down list for the **Variable** option.
- 7 Click **Set Parameter**.

The variable is modified. An action object can only set the value of a single variable. To set the value of more than one variable, you need to create multiple action objects.

1. Create a string variable using MasterDoc properties.

2. Field1 displays the value of the variable VAR1 in the design area.

3. Click  on the Insert toolbar, to create an action object.

4. In the Action tab of the action property sheet, select Set.

5. Select a variable from the Variable drop-down list.

6. To modify the variable type, select the required option from the Type drop-down list.

7. To modify the value, enter the new value .

8. In this example, the string variable VAR1 is reused as a number variable. Field 2 displays the variable after it has been modified.

The image shows the process of using the Set action to modify a variable.



Selecting the **Set** action for a variable enables the **Transform** tab in the action property sheet. You can use the options provided in the **Transform** tab to format the variable value. For more information on applying transformations, see *Chapter 10: Applying Transformations*.



For more information on setting variable type and value for different types of variables, see *Chapter 26: Using Variables*.

Creating a Variable at Runtime

Using Vitesse, you can define a list of variables for your project using the **MasterDoc** property sheet. These predefined variables can then be used in the project as field or barcode contents, chart data, and for referencing resources. These variables can be modified at runtime using actions. You can also create variables at run time by using the action object.

To create a variable using an action object:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
The related options are displayed in the tab.
- 3 Select the **Set** option which is used to create a new variable.
- 4 Enter the name of the variable in the **Variable** drop-down list. The variable name can contain alphanumeric characters without any spaces.



All variable names must always begin with the VAR prefix. The variable names are case sensitive; using Var or var instead of VAR will generate an error when the exported project is opened in VIPPIDE.

- 5 Select one of the following options from the **Type** drop-down list:
 - **String** – creates a string type variable.
 - **Number** – creates a number type variable.
 - **Variable** – sets the value of the variable equal to the value of another system or user-defined variable.
 - **Bool** – creates a bool type variable.



You cannot create an array variable at runtime.

- 6 To set the variable value, either:
 - Enter the required value in the **Value** edit box for number and string type variables.
 - Click **True** or **False** in the **Value** area for bool type variables.
 - Select the required system or user-defined variable from the **Value** drop-down list for the **Variable** option.
- 7 To set the variable behavior as persistent, select the **Persist Variable** option.



When working with predefined variables, selecting the **Persist Variable** option in the **Action** tab ignores any changes made to the variable by the action, and retains the original value.



For more information on creating variables, see *Chapter 26: Using Variables*.

- 8 Click **Set Parameter**.

A new variable is created. This variable can only be used in the design objects added below the action object in the project tree because the variable is created at runtime and will be available once the action has been executed.

1. In the action object property sheet, select Variable from the Category drop-down list.

2. Select Set.

3. Enter the variable name in the Variable drop-down list.

4. Select the variable type in the Type drop-down list, in this example, the type is set to Variable.

5. Specify the variable value.

7. Add a field object below the action object in the project tree.

8. In the Settings tab of the field property sheet, specify the runtime variable as field content by using the syntax \$\$VariableName. in the Const edit box. Variables created at runtime are not listed in the Variable drop-down list, and therefore, have to be specified using the Const entry box, as below:

6. Click Set Parameter.

9. Select Merge data in the field property sheet.

10. The runtime variable displays as field contents when the action object is encountered in the project tree.

The image displays the process of creating a variable using an action, and then using the runtime variable in the design as field content.



An action object can only be used to create a single variable. To create more than one variable, you need to create multiple action objects.

Incrementing a Variable

Vitesse enables you to increment number and numeric string variables. To increment a variable value:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Increment** option. Using this option, the variable value is increased by 1. For example, a numeric string variable with a value of 123 will be incremented to 124 after the action is executed.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Click **Set Parameter**. The variable value is incremented.

Decrementing a Variable

Vitesse enables you to decrement number and numeric string variables. To decrement a variable value:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
The related options are displayed in the tab.
- 3 Select the **Decrement** option. Using this option, the variable value is decreased by 1. For example, a number variable with a value of 123 will be decremented to 122 after the action is executed.
- 4 Select the required number or string variable from the **Variable** drop-down list.
- 5 Click **Set Parameter**. The variable value is decremented.

Adding a Variable

Vitesse enables you to add number, numeric string, and array variables. You can either define the value to be added to the variable in the **Action** tab, or choose an existing variable to add to the specified variable. You should ensure type compatibility when adding variables, i.e., a number type variable should only be added with a number value or another number type variable.

To add a variable:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Add** option, which enables you to add a value to a selected variable.
- 4 Select the required predefined number, string, or array variable from the **Variable** drop-down list.

Adding a Value to the Variable

- 1 To add a value to the variable, select the corresponding type; either **String**, **Number**, or **Array** from the **Type** drop-down list.
- 2 To set the value, either:
 - Enter the required value in the **Value** entry box for number and string type variables.
 - Update the array values in the **Value** area for array type variables.

Adding an Existing Variable to the Variable

- 1 To add an existing variable to the selected variable, select **Variable** from the **Type** drop-down, and then select the required predefined variable from the **Value** drop-down list.
- 2 Click **Set Parameter**. The specified value is added to the variable.

1. Define two array variables using MasterDoc properties.

2. Add an action to the Page, and select Variable from the Category drop-down list.

3. Select Add.

4. Select the first array variable from the Variable drop-down list.

5. Select Variable from the Type drop-down list.

6. Select the second array variable from the Value drop-down list.

7. Click Set Parameter.

8. Add a chart to the Page, and in the Data tab of the chart property sheet, select the first array variable.

9. The design area displays the chart which contains values defined in both the array variables.

10. The action executes for each page, updating the chart each time.

Page 1

Category	Value
A	10.00
B	5.00
C	20.00
D	15.00

Page 2

Category	Value
A	10.00
B	5.00
C	20.00
D	15.00
C	20.00
D	15.00

The image displays the process of adding two array variables using an action object.

Subtracting a Variable

Vitesse enables you to subtract number and numeric strings. You can either define the value to be subtracted from the variable in the **Action** tab, or subtract an existing variable from the selected variable. You should ensure type compatibility when subtracting variables, i.e., a numeric variable or value should only be subtracted from a numeric type variable.

To subtract a variable:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Sub** option, which enables you to subtract a value from a selected variable.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.

Subtracting a Value from the Variable

- 1 To subtract a value from the variable, select the corresponding type; either **String** or **Number** from the **Type** drop-down list, and then enter the required value in the **Value** entry box.

Subtracting an Existing Variable from the Variable

- 1 To subtract an existing variable from the selected variable, select **Variable** from the **Type** drop-down, and then select the required predefined variable from the **Value** drop-down list.
- 2 Click **Set Parameter**. The specified value is subtracted from the variable.

1. Create a Number variable using MasterDoc properties.
2. Create an action object, and select Variable as the Category.
3. Select Sub.
4. Select the required variable from the Variable drop-down list.
5. From the Type drop-down list, select Number.
6. Specify the value to be subtracted from the variable in the Value edit box.
7. Click Set Parameter.
8. Create a field object below the action object in the project tree. In the Settings tab of the field property sheet, specify the variable as the field data.
9. The field displays the updated value, in this example, $100 - 20 = 80$.

The example in the image displays the process of subtracting a number value from a number variable.



Press <Shift> + <F5> on the keyboard to refresh the application if the result of the action is not immediately visible.

Assigning Field Value to a Variable

Vitesse enables you to specify data file contents as variable values when working with Line, Database, or XML mode projects. You can use the **Field** action defined for variables to assign data file contents to a variable. Selecting this action enables the **Transform** tab in the action object property sheet. You can use the options provided in the **Transform** tab to format the variable value. For more information on applying transformations, see *Chapter 10: Applying Transformations*.

Assigning Value in Line Mode

Line mode data consists of fixed or variable length records, and also includes prefixed and/or line numbered data streams. You can use line mode to read, process, and format line mode data on a line-by-line (record) or column-by-column (field) basis.

When working with a line mode project, you can set data from the data file as the value for a variable. The data in the line mode data file can be specified by:

- Identifying the data line using either line number or a test condition.
- Identifying the required field on the specified line.

The value corresponding to the field is set as the variable value.

Choosing a Data Line – Using Conditions

You can specify the line in the data file which contains the required field using the line number or by setting up a condition.

To identify the data line using a test condition:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.

The related options are displayed in the tab.

- 3 Select the **Field** option to set the variable value using the data file contents.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Select the **PCD** option; selecting this option displays the test condition area and the **Line Offset** entry box.



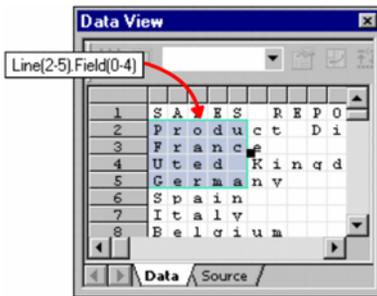
PCD option is used to define a Page Criteria Definition. The PCD conditions are evaluated at the page level.

- 6 The test condition area contains two lines, the line at the top contains the **IF** keyword, the second line contains the test condition. To set up the test condition, click in the second line below the **IF** keyword, the associated drop-down list displays.
- 7 As appropriate, select the **Not** operator from the drop-down list; this operator reverses the logical value of the evaluated condition (i.e., true if the statement is false, and vice versa).



You cannot use the **Not** operator in PCD conditions for actions that are defined at the record level.

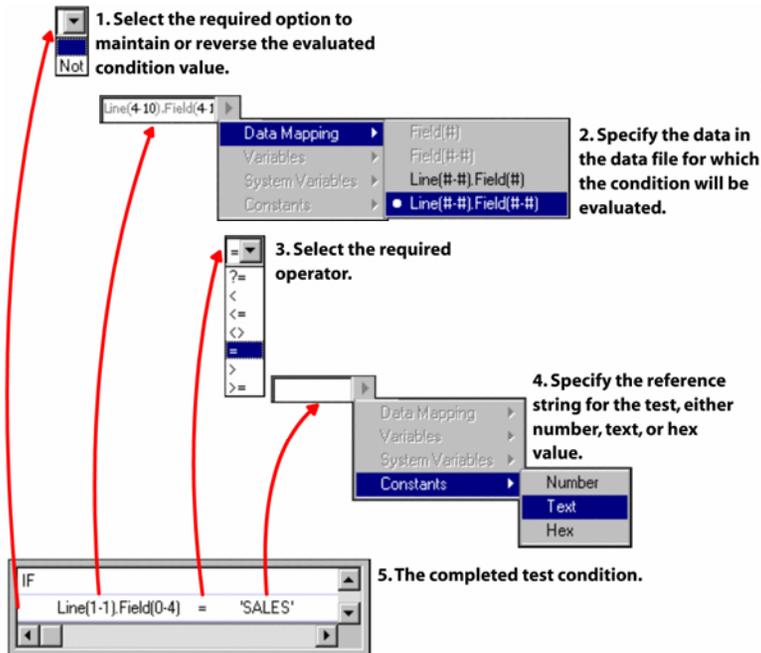
- 8 Click in the next area, displaying the line and field information. This area specifies the data in the data file for which the condition will be evaluated. Click  in the cell. Select **Data Mapping** from the pop-up menu, and then select one of the following options from the data mapping sub menu.
- **Line(#-#).Field(#)** – select this option when working with data files that have records with delimited fields. In the **Line(#-#)** syntax, the first # specifies the starting line and the second # specifies the ending line. For example Line(2-5) will specify a data area starting from line 2 and ending at line 5. In the **Field(#)** syntax, the # represents the field number.
 - **Line(#-#).Field(#-#)** – select this option when working with data files that have records with fixed length fields. In the **Line(#-#)** syntax, the first # specifies the starting line and the second # specifies the ending line. For example Line(2-5) will specify a data area starting from line 2 and ending at line 5. In the **Field(#-#)** syntax, the first # represents the field starting position and the second # specifies the field length from the starting position.



The image displays an example of using the Line(#-#).Field(#-#) syntax for specifying data.

- 9 Click in the next area that displays the test operator (by default the operator is =); the associated drop-down displays. Select one of the following options from the drop-down list:
- **?=** – sets the contain operator. The contain operator searches for the reference string anywhere in the selected record data.
 - **<** – sets the operator to less than.
 - **<=** – sets the operator to less than or equal.
 - **<>** – sets the operator to not equal.
 - **=** – sets the operator to equal.
 - **>** – sets the operator to greater than.
 - **>=** – sets the operator to greater than or equal.

- 10 Click in the area next to the operator. This area specifies the reference string for the test. Click  in the cell. Select **Constants** from the pop-up menu, and then select one of the following options from the constants sub menu:
- **Number** – performs numeric comparisons, such as less than 10, or greater than or equal to 100. Selecting this option displays a # symbol in the entry box. Enter the required numeric value next to the # symbol.
 - **Text** – performs string comparisons, which can only be equal or not equal to for a given string. Selecting this option displays the ‘ ’ symbol in the entry box. Enter the required text value between the ‘and’ symbol.
 - **Hex** – performs comparisons on a single character in a field. Selecting this option displays an 0x symbol in the entry box. Enter the required hexadecimal value next to the 0x symbol.



The diagram shows a sequence of five steps for setting up a test condition:

- 1. Select the required option to maintain or reverse the evaluated condition value.** A dropdown menu is shown with 'Not' selected.
- 2. Specify the data in the data file for which the condition will be evaluated.** A menu path is shown: 'Line(4-10).Field(4-1)' > 'Data Mapping' > 'Constants' > 'Line(##).Field(##)'. The 'Line(##).Field(##)' option is highlighted.
- 3. Select the required operator.** A dropdown menu is shown with '=' selected.
- 4. Specify the reference string for the test, either number, text, or hex value.** A menu path is shown: 'Data Mapping' > 'Constants' > 'Text'. The 'Text' option is highlighted.
- 5. The completed test condition.** The final result is shown in a text box: 'IF Line(1-1).Field(0-4) = 'SALES''.

The image displays the process of setting up a test condition.



For more information about working with conditions, see *Chapter 27: Using Conditions*.

11 In the **Line Offset** entry box, specify the required data line offset from the data line that matches the test condition. If the matching line is the required data line, the offset is set to 0. If the required data line is above or below the matching line, the offset is specified as a positive or a negative value. For example, if the required line is the second line after the matching line, the offset is set to 2.

12 Click **Set Parameter**.

The value corresponding to the specified field is assigned to the variable. The value of the variable is modified for each occurrence of the specified field in the data file.

Choosing a Data Line – Using Line Number

To select the required line from the data file using the line number:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
The related options are displayed in the tab.
- 3 Select the **Field** option to set the variable value using the data file contents.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Type or select the required line number in the **Line Number** entry box.

Choosing a Data Field – Using Fixed Field Data

In fixed field data files, the fields are identified using the starting position and the length of the data field.

To specify a data field as a variable value for fixed field data files:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list. The related options are displayed in the tab.
- 3 Select the **Field** option to set the variable value using the data file contents.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Select the required data line using either a line number or a PCD test condition.
- 6 Select **Fixed**, which indicates that a fixed field data file is being used for the project; selecting this option enables the **Start** and **Length** entry boxes:
 - **Start** – specifies the field starting position.
 - **Length** – specifies the field length from the starting position.
- 7 Click **Set Parameter**.

The value corresponding to the specified field is assigned to the variable. The value of the variable is modified for each occurrence of the specified field in the data file.

1. To use data file contents as variable value, select the Field action in the action object property sheet.

2. Select the required number or string variable from the Variable drop-down list.

3. Specify the data line number in the Line Number entry box.

4. Select the Fixed option.

5. Specify the field starting position in the Start entry box.

6. Specify the field length in the Length entry box.

7. Click Set Parameter. The specified field is highlighted in the Data View window.

8. You can view the variable value in the design area by using a field object.

The image displays the process of specifying line mode fixed field data as the value for a variable.

Choosing a Data Field – Using Delimited Data

In delimited data files, the fields are identified using delimiting characters. Every time a delimiter is encountered in the file, a new field starts, which allows fields to vary in length.

To specify a data field as a variable value for delimited data files:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
The related options are displayed in the tab.
- 3 Select the **Field** option to set the variable value using the data file contents.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Select the required data line using either a line number or a PCD test condition.
- 6 Select **Data**, which indicates that a delimited field data file is being used for the project; selecting this option enables the **Field No** entry box.
- 7 Specify the field number in the **Field No** entry box.

8 Click Set Parameter.

The value corresponding to the specified field is assigned to the variable. The value of the variable is modified for each occurrence of the specified field in the data file.

1. To use data file contents as a variable value, select the Field action in the action object property sheet.

2. Select the required number or string variable from the Variable drop-down list.

3. Select the PCD option to display the test condition area and the Line Offset entry box.

4. Setup the PCD condition. In this example, the condition is set to true if field(0) of line 65 is 'PPA1'.

5. Specify the line offset value.

6. Select the Data option.

7. Specify the field number.

8. Click Set Parameter.

9. You can view the variable value in the design area by using a field object.

Field(0)	Field(1)	Field(2)	Field(3)
TITO	TIME WEIGHTED GROSS P...		
PERD	Period from 1 January...		
PPAO	TOTAL ASSET VALUE END...	2,211,574.49	SGI
PPAO	TOTAL ASSET VALUE STA...	2,208,526.77	SGI
PPAO	DEPOSITS	0.00	SGI
PPAO	WITHDRAWALS	0.00	SGI
PPAO	CUSTOMER MANAGEMENT FEE...	0.00	SGI
PPAO	WITHHOLDING TAX	0.00	SGI
PPA1	VALUATION CHANGE	3,047.72	SGI
PPAO			

The image displays the process of specifying line mode delimited data as the value for a variable.



For more information on working with Line mode data, see *Chapter 21: Understanding Line Mode Data*.

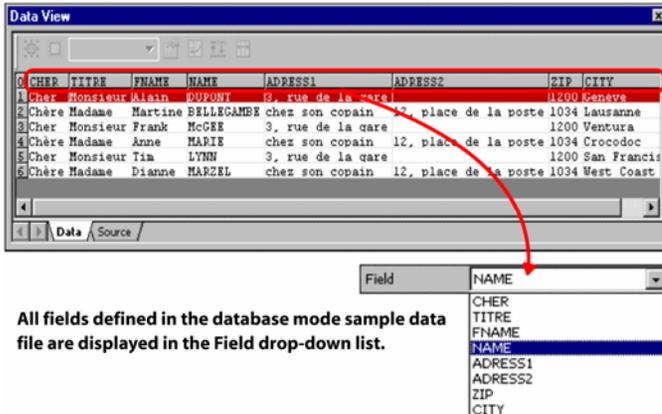
Assigning Value in Database Mode

Database mode is used to process data files containing delimited records. The delimited data file is typically extracted from a relational database. Records in a data file may be of fixed or variable length, and must have the same sequence and number of fields. The first line in the data file contains field names; remaining lines contain data in order of field names. Delimiters are used to separate fields within records.

When working with a database mode project, you can specify the fields defined in the first line of the sample data file as string or number type variable data. The value corresponding to the field is set as the variable value.

To assign field contents as variable value:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
- 3 The related options are displayed in the tab.
- 4 Select the **Field** option to set the variable value using the data file contents.
- 5 Select the required predefined number or string variable from the **Variable** drop-down list.
- 6 Select the required field from the **Field** drop-down list. The list contains all of the fields defined in the first line of a database mode sample data file.



All fields defined in the database mode sample data file are displayed in the Field drop-down list.

The image displays the database mode data fields displayed in the Field drop-down list.

- 7 Click **Set Parameter**.

The value corresponding to the specified field is assigned to the variable. The value of the variable is modified for each occurrence of the specified field in the data file.



For more information on working with Database mode data, see *Chapter 19: Understanding Database Mode Data*.

Assigning Value in XML Mode

XML data files include tags, attributes, and values. The first node in the XML data file is the root node. The root node is a tag containing sub tag nodes and their values. Tags organize the structure of an XML data file. Each tag may have a set of attributes that have a name and a value.

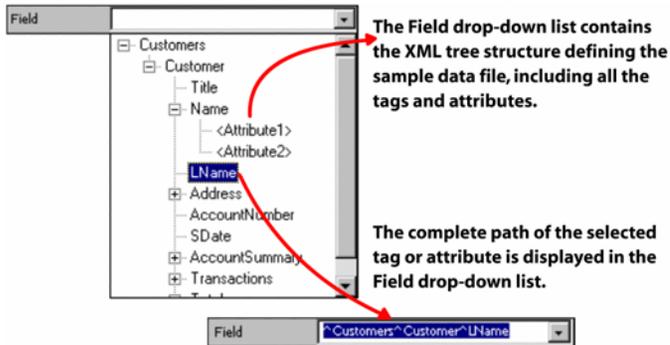
When working with an XML mode project, you can specify the tags and attributes defined in the project data file as a string or number type variable data. The value corresponding to the tag or attribute is set as the variable value.

To assign data file contents as variable data:

- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.

The related options are displayed in the tab.

- 3 Select the **Field** option to set the variable value using the data file contents.
- 4 Select the required predefined number or string variable from the **Variable** drop-down list.
- 5 Click  to view the contents of the **Field** drop-down list.
- 6 Click  to expand the XML tree root node in the **Field** drop-down list. The XML tree represents the structure of the sample XML data file.
- 7 Select the required tag or attribute from the XML tree structure.



The image displays the XML tree structure displayed in the field drop-down list.

- 8 Click **Set Parameter**.

The value corresponding to the specified tag or attribute is assigned to the variable. The value of the variable is modified for each occurrence of the specified tag or attribute in the data file.



For more information on working with XML mode data, see *Chapter 23: Understanding XML Mode Data*.

Setting Variables as PDF Bookmarks

Bookmarks are navigational tools for PDF documents. In a PDF document, the bookmarks are displayed in a side-panel on the left of the document. When you click on a bookmark, you instantly move to the bookmarked page.

Vitesse enables you to create bookmarked PDF documents. Xerox VIPP Thin Printer is used to convert Vitesse VPC (VIPP Project Container) to a PDF document. When converting a Vitesse VPC (VIPP Project Container) to PDF, the Xerox VIPP Thin printer uses a variable specified as a bookmark to generate bookmarks.

The text or title of a bookmark is the variable value for which the bookmark is set. You can set static text as a bookmark by initializing a string variable with the required text value. You can also specify the value of a field as a bookmark by using the field action object.

When encountered in the application source code, a PDF bookmark points to the page currently being processed. In turn, action object placement in the project tree determines the page to which the bookmark points.

You can create a bookmark hierarchy by specifying one or more sub-bookmarks within a bookmark; subsequent bookmarks equal to the specified sub-bookmark number will be considered sub-bookmarks of that bookmark.



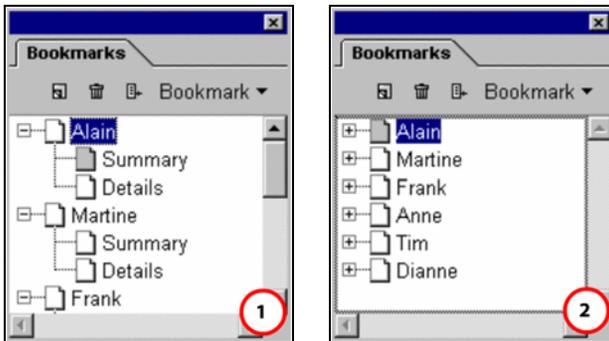
If you want bookmarks and sub-bookmarks to be at different levels for different pages, you can control them by using a boolean variable in conjunction with conditions.

Setting Bookmark Properties

To bookmark the project:

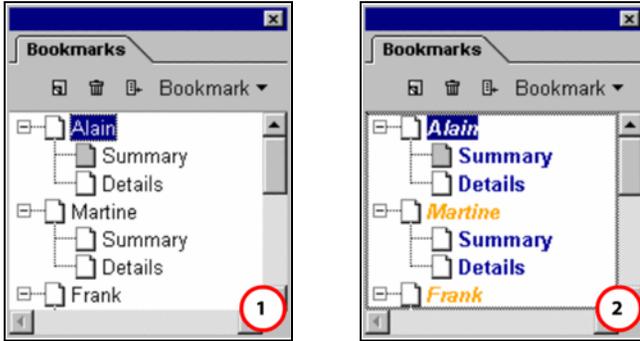
- 1 Select the required **Action** node from the project tree.
- 2 From the **Action** tab of the *Property View* window, select **Variable** from the **Category** drop-down list.
The related options are displayed in the tab.
- 3 Select the **Bookmark** option to set the selected variable as a bookmark for the generated PDF.

- 4 Select the required predefined string variable from the **Variable** drop-down list. The value of the variable represents the string used to create the bookmark.
- 5 Specify the number of sub-bookmarks for the current bookmark in the **Number of Sub Bookmarks** entry box. By default, the number is zero, which means all bookmarks will be at the same level. You can also specify a positive or a negative value. If a positive value is specified, the PDF file opens with the bookmark tree expanded. If a negative value is specified, the bookmark tree remains collapsed in the PDF file.



The generated bookmarks with (1) number of sub-bookmarks set to 2 (2) number of sub-bookmarks set to -2.

- 6 Select the color used to display the bookmark text from the **Bookmark Color** drop-down list. The **Custom** option in the list opens the *Color Selector* dialog. For more information, refer to *Chapter 11: Working with Colors*.
- 7 Select one of the following options from the **Bookmark Style** drop-down list:
 - **Plain** – applies no style to bookmark text.
 - **Italic** – applies italic style to bookmark text.
 - **Bold** – applies bold style to bookmark text.
 - **Bold Italic** – applies bold and italic style to bookmark text.



The bookmark text displayed with (1) Black, Plain style (2) Orange, Bold Italic style for main-bookmarks, and Blue, Bold style for the sub-bookmarks.

8 Click Set Parameter.

The variable value is set as the bookmark text in the generated PDF file.

The following image displays the process of using the value of a field as a bookmark by using the field action object in a database mode project:

1. Add an action object to the page object. Use this action to populate a string variable with a field value.
2. Add another action to the page. Use this action to set the variable as the bookmark text.
3. Click Set Parameter.
4. Export the project and use VIPP Thin Printer to create the PDF file with bookmarks.
5. Each bookmark displays different text depending on the field value.



You can create bookmarks for documents in all project modes.

Chapter 30: Adding Barcodes

This chapter describes the following activities and tasks for adding a barcode to a design and setting barcode properties. It also gives a detailed overview on specifying the barcode data using barcode, field, text, and graphic text objects.

- Understanding Barcodes
- Selecting the Barcode Fonts
- Adding a Barcode
- Setting Barcode Properties
- Selecting a Barcode Type
- Specifying Data for the Barcode
- Applying Barcodes Using the Barcode Object
- Applying Barcodes Using Field, Text, and Graphic text Objects

Understanding Barcodes

A barcode is a special identification code representing binary information read by an optical scanner. The code may include vertical bars of differing widths, numbers, and letters, or a combination of both. Bars and spaces are arranged in predetermined patterns following pre-defined rules.

Barcodes are used for identification in almost all types of business. When barcodes are implemented in business processes, procedures can be automated to reduce human error and increase productivity. Bar coding is an effective method used to track, label, and automatically scan inventory items. Barcodes are also used by the postal service, which helps speed the delivery of mail.

All barcodes require a start and stop character and these must also be included in the barcode. Without the start and stop character in the barcode, the scanner does not know where the barcode starts or ends.

Vitesse enables you to insert a barcode within your designs in order to encode information that is readable only by a scanner. The barcodes can be added to your design once you have employed custom barcode fonts. The fonts used to implement the barcodes are a third party component and are acquired and installed in Windows PCs by copying them into the `\Windows\Fonts` directory during the setup or installation process.

Vitesse supports a variety of different barcode types, which you can add to your design. Each bar code type has a pre-defined symbology. When you type a character that is not according to the defined symbology of that barcode type, Vitesse displays an error indication as “**ERROR!!!**” in the design area. These bar codes will not scan.



You must install the appropriate barcode fonts in the `!Windows!Fonts` directory and associate them to the required barcode types prior to using the barcodes in the design. For information on associating the barcode fonts to a barcode type, see the “Associating the Barcode Fonts with the Barcode Types” section later in the chapter.

The following are the details on each of the barcode types supported in Vitesse, along with their representations:

- **2 OF 5** – also known as Code 25 Industrial. This code is not often used because of the amount of space required. The symbology consists of the numbers 0-9 and the start character and the stop character (parentheses). For example, to scan the numbers 1234, you need to type and print (1234).



2 OF 5 Barcode.

- **Code 39** – This is an alpha-numeric code. The symbology includes 43 characters: A-Z, 0-9, and characters for a dash (-), period (.), dollar sign (\$), slash (/), plus sign (+), percent (%), and a space. Each character in Code 39 is made up of five bars and four spaces. Of these nine bars and spaces, six are always narrow and three are always wide. There is also a start and stop character to initiate and to stop the reading. The bar code shown is a typical Code 39. Code 39 is a bi-directional bar code implying that a stop can be read first and the number read backwards, and it can still be recognized in its proper sequence.



Code 39 Barcode.

- **Code 128** – represents non-retail 4 line width barcode, used for the full ASCII character set.



Code 128 Barcode.

- **EAN 128** – provides a worldwide standard for exchanging data between different companies. EAN-128 not only encodes the data, but provides a method of defining the meaning of the data by defining a list of "Application Identifiers" (AIs). The data is structured according to the Application Identifier (AI) format. An AI is a two, three, or four-digit number that defines the meaning of the data that follows. EAN 128 barcodes are capable of supplying much more detail about the product, including dates, batch numbers, weight, quantity and dimensions.



EAN 128 Barcode.

- **EAN 13** – represents the most commonly used 4 line width barcode, composed of 13 digits and used for retail world-wide, except for the US.



EAN 13 Barcode.

- **EAN 8** – represents a 4 line width barcode, used for retail world-wide except for the US. It is composed of only 8 digits to fit on smaller packaging.



EAN 8 Barcode.

- **PostNet** – stands for *Postal Numeric Encoding Technique*, which represents a 4 line width barcode, specific to mail delivery in the US. It is composed of a start and end character (asterisk) and 12 other digits: 5-digit zip code, 4-digit zip+4 code, 2 delivery point digits, and 1 check digit.



PostNet Barcode.

- **UPC version A (Universal Product Code – A)** – UPC is the most popular and most frequently used of all barcodes. It is mostly used in the grocery industry and price product. UPC is designed to match the number of the product with information, such as the manufacturer, product, and price. It provides a high read rate even as the product moves rapidly past the scanner. Some other considerations for the scan ability of UPC lie in the fact that the codes must be on products that are frozen, wet, or shrink wrapped, etc. A 4 line width barcode; used for retail in the US; composed of 12 digits. UPC is a fixed-length, numeric-only code.



UPCA Barcode.

Associating the Barcode Fonts with the Barcode Types

To select the barcode fonts associated with the barcode type you want to use in your design:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Barcode** category.



After the fonts are installed by copying them into the *Windows\Fonts* directory, they become available in the drop-down list for each barcode type. You can then select the required font as explained in Step 3.

- 3 For the list of available barcodes in this category, specify a **True Type** or **Post Script** font from the drop-down list for associating with the selected barcode type.
- 4 Click **OK**. The selected barcode fonts become available for use with any of the barcode types.

Adding a Barcode

In Vitesse, you can add a barcode to your designs to encode information, and later on print them using the barcode fonts.

To add a barcode, select **Insert>Barcode** from the menu bar; a **Barcode** node adds to the project tree under the selected node.



You can also add a new barcode object either by:

- Clicking  on the *Insert* toolbar.
- Or
- Right-clicking the selected tree node, and selecting **Insert>Barcode** from the pop-up menu.



You can delete a selected barcode object either by:

- Selecting **Edit>Delete** from the menu bar.
- Or
- Clicking  on the *Standard* toolbar.
- Or
- Pressing on the keyboard.

Setting Barcode Properties

In Vitesse, you can insert multiple barcode objects in the design area and set or change barcode properties, such as size, position, spacing, type, and color, etc.

Naming a Barcode

Assigning meaningful names to a barcode helps in referencing the resource when multiple barcode objects are being used. Vitesse assigns a default name to each newly created barcode object. You can change the default values to set a unique name for the barcode object using the **Name** entry box in the **General** tab of the *Property View* window.

Positioning a Barcode

Barcode objects can be placed at an absolute position in the design area by specifying the horizontal and vertical coordinates. You can also position the barcode objects relative to one another.

Setting the Barcode Absolute Position

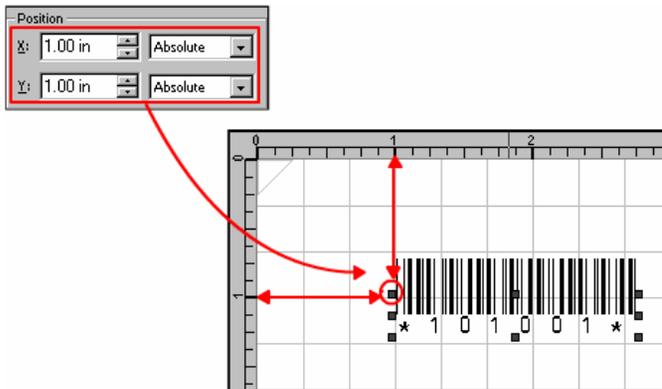
To accurately position a barcode object in the design area:

- 1 Select the required **Barcode** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the barcode horizontal position, per specified unit of measurement in the **X** entry box. In the associated drop-down list, select **Absolute**.
- 4 Specify the barcode vertical position, per specified unit of measurement in the **Y** entry box. In the associated drop-down list, select **Absolute**.



You can also set the unit of measurement for the barcode position by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**> *unit name* from the pop-up menu.

The absolute coordinates are calculated from the top left margin of the page.

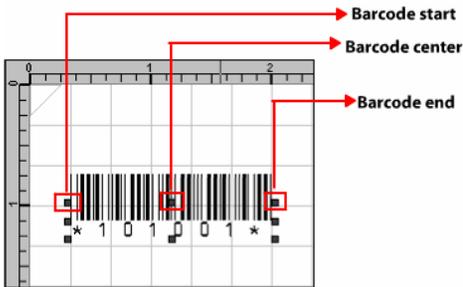


The image shows the process of setting the barcode position using absolute coordinates. In this example, the top, left margins are set at 0, 0 inches.

Setting the Barcode Relative Position

A barcode object can be positioned relative to another barcode object or another relatively positioned object. In absolute positioning, the top left margin of the page acts as the point of reference, whereas in relative positioning, the previously added object is taken as the point of reference.

- 1 To position barcodes using relative positioning:
- 2 Select the required **Barcode** node from the project tree.
- 3 From the *Property View* window, select the **General** tab.
- 4 In the **Position** area, specify 0.20 in the **X** entry box. In the associated drop-down list, select one of the following horizontal relative positioning options:



The horizontal relative positioning options are applied with reference to the barcode horizontal start, center, and end points.

- **Start Same** – positions the start of the current barcode at the start of the previous barcode.
- **Start Center** – positions the start of the current barcode at the center of the previous barcode.
- **Start Next** – positions the start of the current barcode at the end of the previous barcode.



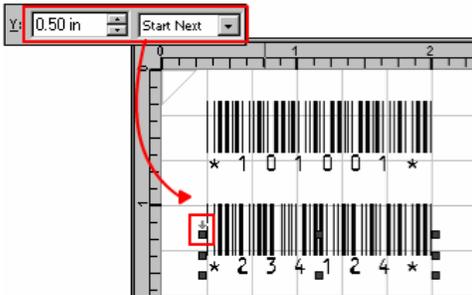
In this example, the current barcode(101234) is positioned at 0.20 inches farther from the previous barcode end.

- **Center Before** – positions the center of the current barcode at the start of the previous barcode.

- **Center Same** – positions the center of the current barcode at the center of the previous barcode.
 - **Center Next** – positions the center of the current barcode at the end of the previous barcode.
 - **End Before** – positions the end of the current barcode at the start of the previous barcode.
 - **End Center** – positions the end of the current barcode at the center of the previous barcode.
 - **End Same** – positions the end of the current field at the end of the previous field.
- 5 Specify 0.50 in the **Y** entry box. In the associated drop-down list, select one of the above-mentioned vertical relative positioning options. Vitesse provides the same set of vertical relative positioning options as the horizontal relative positioning options.



The vertical relative positioning options are applied with reference to the barcode vertical start, center, and end points.



In this example, the selected barcode (234124) is positioned at 0.50 inches farther from the previous barcode end.



The following symbols appear next to a relatively positioned barcode node in the *Project Tree View* window:

-  - specifies that horizontal relative positioning is applied to the barcode object.
-  - specifies that vertical relative positioning is applied to the barcode object.
-  - specifies that both horizontal and vertical relative positioning is applied to the barcode object.



Using a combination of absolute and relative positioning, you can place the barcode object relative to another barcode or a relatively positioned object and add the spacing defined by the absolute **X** and **Y** coordinates.

Setting the Barcode Size

Barcode dimensions specify the width and height of the barcode object.

To set the barcode size:

- 1 Select the required **Barcode** node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Size** information:
 - **W** – sets the barcode object width, per specified unit of measurement.
 - **H** – sets the barcode object height, per specified unit of measurement.



You can also set the unit of measurement when specifying the barcode size by right clicking the upward or downward arrows in the **W** and **H** entry boxes, and then selecting **Units>unit name** from the pop-up menu.

The  icon in the **Size** area is used for maintaining the aspect ratio of the barcode object. Clicking this icon and then setting the width and height increases or decreases the length and width of the object in equal proportion. In the unlock mode, you can set any values for the width and height. For more information, see the “Setting Common Object Properties” section in *Chapter 8: Using the Drawing Tools*.



You can also set the barcode size by dragging the barcode handles that define the boundary of the barcode object in the design area. The corresponding **Size** settings are updated in the *Property View* window.

Selecting the Barcode Type

Vitesse offers different types of barcodes to add to your designs.

To select the type of a barcode:

- 1 Select the required barcode object in the design area.



You can also select the **Barcode** node in the *Project Tree View* window to set the barcode object properties.

- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select the required barcode type option from the **Bar Code Types** drop-down list. Either:
 - **2 OF 5** – sets to 2 OF 5 barcode type.
 - **CODE 39** – sets to Code 39 barcode type.
 - **CODE 128** – sets to Code 128 barcode type.
 - **EAN 128** – sets to EAN 128 barcode type.
 - **EAN 13** – sets to EAN 13 barcode type.
 - **EAN 8** – sets to EAN 8 barcode type.
 - **POSTNET** – sets to PostNet barcode type.
 - **UPCA** – sets to UPCA barcode type.

Specifying the Barcode Data

In Vitesse, you can specify barcode data by either of the following methods:

- Specify barcode data using the barcode object.
- Specify barcode data using field, text, and graphic text objects.

In either case, to generate the correct output of the barcode, you should install the required fonts for the selected barcode type and also specify the characters according to the defined symbology of that barcode type. Once correct data is specified for the barcode, you can use these barcodes to track, label, and automatically scan items to automate your business needs.

The two methods for specifying data for barcodes are explained in the following sections.

Applying Barcodes Using the Barcode Object

When using the barcode object, you can specify the barcode data using constants and predefined variables. You can also specify barcode data using the merge data functionality provided in Vitesse.



To view text\data in a barcode format, ensure that a barcode font type has also been applied to it. Barcode fonts used for creating barcodes must be separately purchased.

Using Variables as Barcode Data

To specify the barcode data using predefined variables:

- 1 Select the required **Barcode** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Data** area, select **Variable**, which indicates that a variable will be used to specify the barcode data; selecting this option enables the associated drop-down list.
- 4 Select the required system or user-defined variable from the drop-down list. See *Chapter 26: Using Variables* for information on creating variables.

The barcode updates in the design area to display the variable value in the selected barcode type.

1. Create a number type variable Varbarcode, and set its value.

2. Click  on the Insert toolbar to add a barcode object.

3. In the Settings tab of the barcode property sheet, select CODE 39 as the barcode type.

4. Select Variable and then choose the required number type variable (created in Step 1) from the drop-down list.

5. The variable value acts as barcode data and displays in the design area as CODE 39 barcode type.

Using Constants as Barcode Data

You can insert static text as barcode content.

To specify the barcode data using constants (static text):

- 1 Select the required **Barcode** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Data** area, select **Constant**, which indicates that constant data will be used as the barcode content; selecting this option enables the associated entry box.
- 4 Enter the required text or number in the **Constant** entry box.

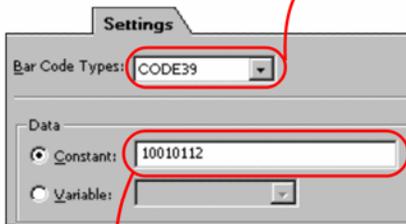
The barcode updates in the design area to display the constant text or number in the selected bar code type.

1. Click  on the Insert toolbar to create a barcode object.

2. In the Settings tab of the barcode property sheet, select CODE 39 as the barcode type.

3. Select Constant and then enter the required number in the associated entry box.

4. The number entered acts as barcode data and displays in the design area as CODE 39 bar code type.



Using Merge Data

You can also specify the barcode data using a variable by selecting **Constant** from the **Data** area, and entering `$$VariableName`. in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified variable value as barcode in the design area.

See the image below explaining the use of data merging with a number variable to display barcode data:

The image illustrates the process of creating a barcode from a variable through several steps:

- 1. Create a number type variable Varbarcode, and set its value.** The 'Variables' dialog shows a variable named 'VARbarcode' of type 'Number' with a value of '1001100110'.
- 2. Click [Barcode Icon] on the Insert toolbar to add a barcode object.**
- 3. In the Settings tab of the barcode property sheet, select CODE 39 as the barcode type.** The 'Settings' dialog shows 'Bar Code Types' set to 'CODE39'.
- 4. Select Constant and enter \$\$VARbarcode. in the associated entry box.** The 'Data' section of the 'Settings' dialog has 'Constant' selected with the value '\$\$VARbarcode.'.
- 5. Select the Merge data option in the Data Merging area.** The 'Data Merging' section of the 'Settings' dialog has 'Merge data' checked.
- 6. The specified value contained in the Varbarcode variable acts as barcode data and displays in the design area in CODE 39 barcode type.** The design area shows a barcode with the value '* 1 0 0 1 1 0 0 1 1 0 *' displayed below it.

Applying Barcodes Using Field, Text, and Graphic Text Objects

In addition to applying barcodes using barcode objects, you can also apply barcodes using field, text, and graphic text objects. This is done by transforming field, text, and graphic text data as barcode data by using the **Bar Code Types** option in the **Transform** tab of the *Property View* window of each object.

The examples for formatting data as barcodes using field, text, and graphic text objects are explained in subsequent tasks.

Example: Using Field Data as Barcode Value

You can format field data as barcode value using the **Transform** tab in the *Property View* window.

The following image shows the example of using field data as barcode value when data for a field is picked from the data file:

1. Insert a field object in the design area, and select the field node from the project tree.

2. From the Settings tab of the field property sheet, select Data.

3. Select BARCODE from the Field Name drop-down list. This field is picked from the specified data file.

0	CHER	TITLE	FNAME	BARCODE
1	Cher	Monsieur	Alain	10111011001
2	Chère	Madame	Martine	10230405621

Settings
Field Data:
 Fixed Start: 0 Length: 0
 Data Field Name: BARCODE
 Variable
 Const

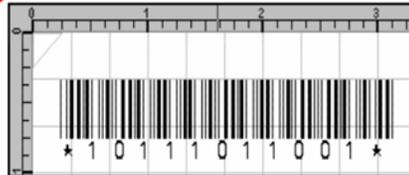
4. From the Transform tab of the field property sheet, select the Bar Code Types option and select CODE 39 from the drop-down list.

Transform
 Bar Code Types: CODE39

5. From the Font tab of the field property sheet, select the required barcode font for CODE 39 and set font size to 20.

Font
Style: Style9
Font
Type: True Type
Name: [Barcode Font]
Size: 20.00 pt

6. The BARCODE field data picked from the data file is transformed as barcode data and displays in the design area as CODE 39 barcode type.



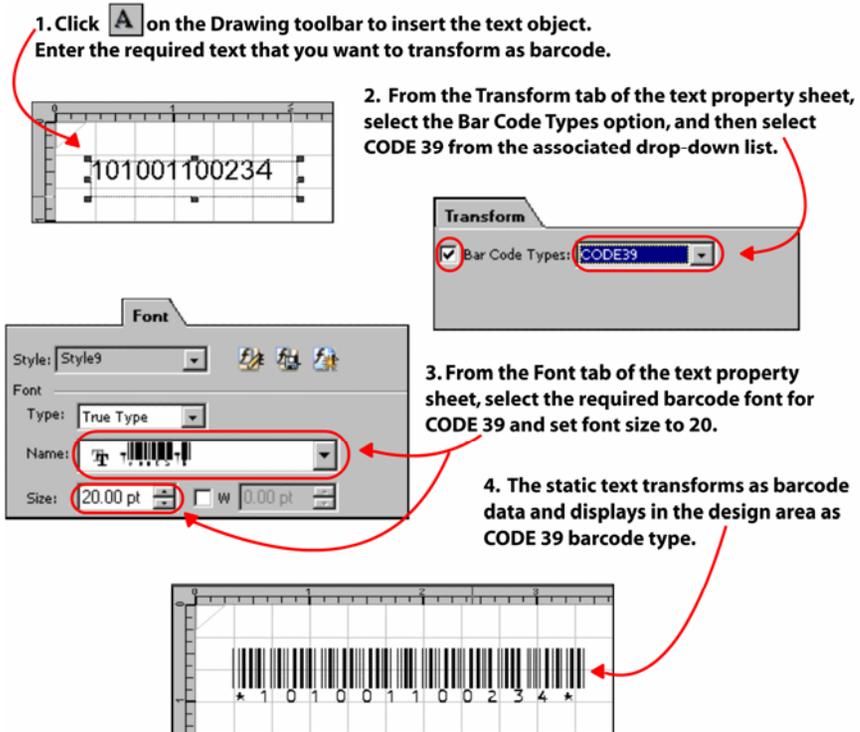
Bar Code Types is the last option on the **Transform** tab. If you cannot see this option, resize the *Property View* window.

You can also use the **Variable** and **Const** options to specify field data and use it as a barcode value.

Example: Using Static Text as Barcode Value

You can format static text as barcode value using the **Transform** tab in the *Property View* window. The following image shows the example of using static text as barcode value:

1. Click **A** on the Drawing toolbar to insert the text object.
Enter the required text that you want to transform as barcode.
2. From the Transform tab of the text property sheet, select the Bar Code Types option, and then select CODE 39 from the associated drop-down list.
3. From the Font tab of the text property sheet, select the required barcode font for CODE 39 and set font size to 20.
4. The static text transforms as barcode data and displays in the design area as CODE 39 barcode type.



You can also use graphic text and format it as a barcode value using the **Transform** tab of the *Property View* window.

User Notes:

Section X: Referencing Resources

User Notes:

Chapter 31: Using Forms in Projects

This chapter describes activities and tasks for inserting and referencing forms in Vitesse. The following information is included in the chapter:

- Using Forms in Projects
- Inserting Forms
- Setting Properties of a Form Reference

Using Forms in Projects

Forms are typically created as stand-alone resources in Vitesse. You can reference these resources and use them as background overlays in a project design. You can also insert a form within a project design as part of the project source. Forms predominantly compose the static content of a design. For variable data documents, these are merged with the variable data to generate the final output.

A form can be associated with a project as:

Embedded Forms

When you insert an embedded form in the project, it is included in the project source code embedded as an individual block that can be referenced multiple times within the same project. Note that an embedded form is not saved as a separate resource file and therefore, is not available for use in other design projects.

Inline Forms

When you insert an inline form, it is integrated in the project source code and cannot be re-used within the same or other projects. Properties associated with an inline form are the same as those for a stand-alone form resource. See the “Setting Form Properties” section in *Chapter 4: Working with Forms* for more information on form properties.

Linked Forms

When a linked form is inserted on a page or paper node (back or front side), two tree nodes are created:

- A stand-alone form resource represented by a form node under the **Resources>Forms** node in the project tree. Selecting this node displays the newly created form in the design area, where you can design the form to meet your requirements.

- A **FormRef** (Form Reference) object represented by a **FormRef** node in the project tree. The **FormRef** node is created under the node on which you inserted the form. It links the newly created form resource to the project design.

The new form is created as a separate resource file and can be referenced repeatedly within the same project and in other projects.

Referenced Forms

The referenced form type is a reference to a stand-alone form that is included in the project. Initially, when you first insert this form, a **FormRef** node is created and it does not contain any references. You can, later on, associate another form resource used in the project design as a reference to this form.



FormRef objects created using the Linked and Referenced form types possess the same properties.

Inserting Forms

Vitesse enables you to insert or reference forms to the **Front** and **Back** sides of a **Paper** or to the **Page** objects in a project design. You can also insert new forms as project resources (**Resources>Forms** tree node) and reference them in a project design.

Inserting a Form as a Project Resource

When you insert a form as a project resource, a new resource is created in the project. To insert a form as a project resource:

- 1 From the *Project Tree View* window, select the **Resources>Forms** node.
- 2 Select **Insert>Form** from the menu bar. The *Insert Form* dialog is displayed with the **New** tab selected.

You can also open the *Insert Form* dialog either by:



- Selecting the **Resources>Forms** node and clicking  on the *Insert* toolbar.
- Or
- Right-clicking the **Resources>Forms** node and selecting **Insert>Form** from the pop up menu.



To use already designed forms in a project, select the **Existing** tab from the *Insert Form* dialog and open the required form. The selected form is added to the project as a linked/referenced resource.

3 As appropriate, select the required form type available in the **New** tab:



- - creates a new form resource that is saved as a separate file and can be re-used in the current and any other Vitesse projects.



- - creates an embedded form resource.

4 Click **New** to create the form.

A new node identifying the newly created form is created under the **Resources>Forms** node.



For information on setting the properties of form resources, refer to the “Setting Form Properties” section in *Chapter 4: Working with Forms*.

Inserting a Form in the Project Design

Vitesse enables you to use forms that are created and saved as separate files by referencing them in a project. You can insert forms to the **Front** and **Back** sides of a **Paper** or to **Page** objects, depending upon the design requirements.

Both the referenced and linked types of forms (described above) are used as form references. Inline forms are actually inserted in a project or a form design and become a part of the source code.

To insert a form in a project design:

- 1 From the *Project Tree View* window, select the required project node under which a form is to be added.
- 2 Select **Insert>Form** from the menu bar. The *Insert Form* dialog is displayed with the **New** tab selected.



You can also open the *Insert Form* dialog either by:

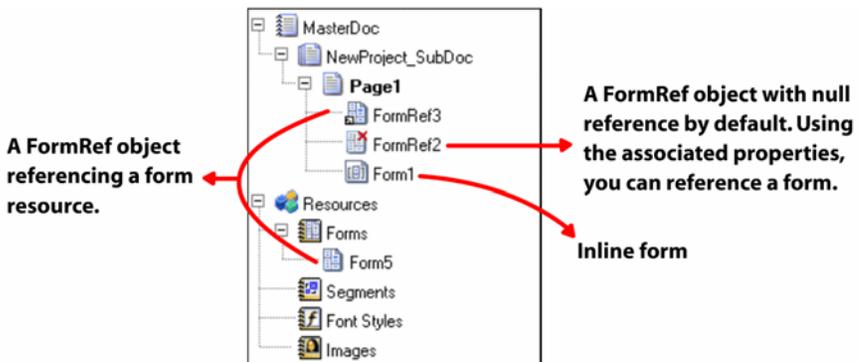
- Selecting the required node and clicking  on the *Insert* toolbar.
- Or
- Right-clicking the required node and selecting **Insert>Form** from the pop up menu.

- 3 Select the required form type from the following to reference a new form in the project design:

-  – creates a new form resource under the **Resources>Forms** node in the project tree and references the newly created form as a linked resource in the project design as a **FormRef** tree node.
-  – creates an inline form that is integrated into the project source code and cannot be re-used.
-  – inserts an empty reference object (FormRef) with no form specified to be used in the project design. You can choose to reference a form resource via this object by using the procedure in the “Changing the Referenced Form” task later in this chapter.

- 4 Click **New** to insert the required form in the project design.

A new node identifying the form inserted in the design is created under the node you selected in **Step 1**. In case of linked and referenced forms, the new node is identified by the **FormRef** node. **FormRef** is the reference to the actual form resource added under the **Resources>Forms** node. In case of a linked form, a resource node is also created under the **Resources>Forms** node identifying the newly added form.



Project tree highlighting different forms inserted in a page.

Setting Properties of a Form Reference

A form may exist independently as a stand-alone resource. Vitesse enables you to reference these form resources in a project. When a resource is referenced in a design, it does not actually become part of the project source code; rather, it is accessed from the actual file whenever it occurs in a design. The reference to a form resource is represented by a **FormRef** (Form Reference) node in a project tree. You can specify various settings for a **FormRef** object that affects the appearance of a form in the project design. Vitesse enables you to reference forms by using constant data or variables.

Changing the Referenced Form

You can associate a form resource already used in a project (and available under the **Resources>Forms** node) to the **FormRef** node that is inserted using the **Ref** icon



You can also reference the form resources that are located in the global form library.

To select the form for referencing in a design:

- 1 Select the required **FormRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Data** area, select **Constant** and choose the required form from the adjacent drop-down list. This list displays all the forms that are already used in the project under the **Resources>Forms** tree node.

The selected form displays in the design area.



By using the **Constant** option, you can also add forms not present in the project resources by entering the form name and extension in the entry box. The specified form should be available in the **FormLib** directory.

Using Variables to Reference a Form

To reference a form resource in the design using variables:

- 1 Select the required **FormRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 In the **Data** area, select **Variable** and choose the required variable from the adjacent drop-down list. The specified variable should contain a reference to the appropriate form. The variables used for referencing forms store the name and extension of the form resource. The drop-down list contains both the system and user-defined variables.

The project design updates to display the newly referenced form.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required form resource using a variable by selecting **Constant** from the **Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified form in the design area.

Setting Properties of an Inline Form

The properties associated with an inline form are the same as for a form resource. You can set the form name, size, and orientation using the options available in the **General** tab of the *Property View* window. Inline forms cannot be cached. See the “Setting Form Properties” section in *Chapter 4: Working with Forms* for more information on setting these properties.

Chapter 32: Using Segments in Projects

This chapter describes activities and tasks for inserting and referencing different segment types in Vitesse. The following information is included in this chapter:

- Understanding Segment Types
- Inserting Segments
- Setting Properties of SegRef
- Working with Segments in XML Mode

Understanding Segment Types

A segment resource mainly contains design objects and is designed to be used repeatedly in a project. Segments are of different types. Some segment types, such as linked, embedded, and inline segments are visible both in the design area and in the project tree. You can add different design objects to these segments as you would when designing a form. Some segment types, on the other hand, are visible only in the project tree. Such segment types are normally used to define the processing order of objects within a design. Examples of such segments are begin-page, end-page, repeat, tag start, tag end, and tag orphan segments.

An explanation for the different types of segments used in Vitesse project designs follows.

Embedded Segments

When you insert an embedded segment in the project, it is included in the project source code as an embedded block of code that can be re-used multiple times within the same project. Note that an embedded segment is not saved as a separate resource file and therefore, is not available for use in other design projects.

Inline Segments

When you insert an inline segment, it is integrated in the project source code and cannot be re-used within the same or other projects. Properties associated with an inline segment are the same as those for a stand-alone segment resource. For more information on segment properties, refer to the sections “Setting Segment Properties” and “Dynamic Behavior of Segments” in *Chapter 5: Working with Segments*.

Linked Segments

A linked segment can be re-used within the same project and among other projects. It is created as a separate resource file under the **Resources>Segments** node in a project tree and is linked as a reference within the same project. The referenced segment displays as a **SegRef** node in the project tree.

Referenced Segments

The referenced segment type is a reference to a stand-alone segment that is included in the project. Initially, when you first insert this segment, it does not contain any references. You can, later on, associate another segment used in the project design as a reference to this segment.

Begin Page Segments

A begin page segment logically groups the objects in order to process them at the beginning of an application page or before an application page is loaded. It is merely a container of objects that determines the processing order of these objects. A begin page segment is inserted as the first **BeginPage** node under the **Page** node in the project tree and processed before any other application code. You can insert form, condition, and action objects in a begin page segment. Begin page segments cannot be used as external resources. You cannot drag and drop a **BeginPage** node in the project tree. No properties are associated with this segment type.

End Page Segments

An end page segment logically contains the objects in order to process them at the end of a page (i.e., after all the other objects in an application page have been processed). An end page segment is inserted as the last **EndPage** node under the **Page** node in the project tree. You can insert conditions, actions, images, charts, and barcodes in an end page segment. End page segments cannot be used as external resources. You cannot drag and drop an **EndPage** node in the project tree. No properties are associated with this segment type.

Repeat Segments

A repeat segment is used to execute a sequence of actions multiple times. For example, you can use repeat segments for placing a collection of objects multiple times in a design. Repeat segments are normally used for objects that can be relatively positioned, such as text, graphic text, charts, and images. This segment type cannot be used as an external resource.

You can use segments in a page and/or in a form design. In an XML mode project, the use of segments is slightly different and is therefore, explained separately in the “Working with Segments in XML Mode Projects” section later in this chapter.

Inserting Segments

Vitesse enables you to insert segments (such as begin page, end page, repeat, and inline segments) directly in a project design or you may insert them as external resources under the **Resources>Segments** node in the required project tree first, and then reference them. Segments can also be used in forms.

Inserting a Segment as a Project Resource

When you insert a segment as a project resource, a new resource is created in the project.

To insert a segment as a project resource:

- 1 From the *Project Tree View* window, select the **Resources>Segments** node.
- 2 Select **Insert>Segment** from the menu bar. The *Insert Segment* dialog is displayed with the **New** tab selected.



You can also open the *Insert Segment* dialog either by:

- Selecting the **Resources>Segments** node and clicking  on the *Insert* toolbar.
- Or
- Right-clicking the **Resources>Segments** node and selecting **Insert>Segment** from the pop up menu.



To use already designed segments in a project, select the **Existing** tab in the *Insert Segment* dialog and open the required segment.

- 3 As appropriate, select the required segment type available in the **New** tab:

-  – creates a new segment resource that is saved as a separate external file and can be re-used in the current and any other Vitesse projects.
-  – creates an embedded segment resource.

- 4 Click **New** to create the segment.

A new node identifying the newly created segment is created under the **Resources>Segments** node.



For information on properties of segment resources, refer to the sections “Setting Segment Properties” and “Dynamic Behavior of Segments” in *Chapter 5: Working with Segments*.

Inserting a Segment in the Project Design

Vitesse enables you to use segments that are created and saved as separate files by referencing them in a project or a form. Both the referenced and linked segment types (described above) are used as segment references. Inline, begin page, end page, and repeat segments are actually inserted in a project and become part of the source code.

To insert a segment in a project design:

- 1 From the *Project Tree View* window, select the required node under which a segment is to be added.



You can insert segments under the **Page** node in Native, Line, and Data Base mode projects. You can also insert segments under forms, conditions, and action objects. In an XML mode project, working with segments is slightly different and is explained in the “Working with Segments in an XML Mode Project” section later in this chapter. Segments can also be inserted in forms.

- 2 Select Insert>Segment from the menu bar. The *Insert Segment* dialog is displayed with the **New** tab selected.



You can also open the *Insert Segment* dialog either by:

- Selecting the required project tree node and clicking  on the *Insert* toolbar.
- Or
- Right-clicking the required node and selecting **Insert>Segment** from the pop up menu.

- 3 Select the required segment type from the following to reference a new segment in the project design:

-  – creates a new segment resource under the **Resources>Segments** node in the project tree and references the newly created segment as a linked resource in the project design.
-  – inserts an empty reference object with no segment specified to be used in the project design. You can choose to reference a segment resource via this object by using the procedure explained in the “Changing the Referenced Segment” task later in this chapter.



Linked and Referenced segments have the same behavior.

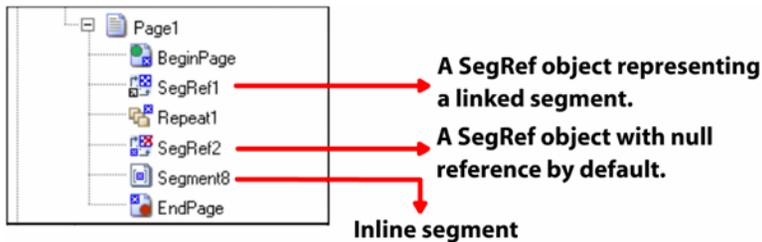
-  – creates an inline segment that is integrated into the project source code and cannot be re-used.
-  – inserts a begin page segment in the project.
-  – inserts an end page segment in the project.
-  – inserts a repeat segment in the project.



Begin page segments can be added to a page or a stand-alone form only. End page segments cannot be inserted in conditions.

4 Click **New** to insert the required segment in the project design.

A new node identifying the inserted segment is created under the node you selected in **Step 1**. In case of linked and referenced segments, the new node is identified by the **SegRef** node. **SegRef** is the reference to the actual segment resource added under the **Resources>Segments** node in the project tree. In case of a linked segment resource, a resource node is also created under **Resources>Segments** node identifying the newly added segment.



Project Tree displaying different segments inserted in a page object.

Setting Properties of a Segment Reference

A segment may exist independently as a stand-alone resource. Vitesse enables you to reference these segment resources in a page or a form. When a resource is referenced in a design, it does not actually become a part of the project source code; rather, it is accessed from the actual file whenever it occurs in a design. The reference to a segment resource is represented by a **SegRef** (Segment Reference) node in a project tree. You can specify various settings for a **SegRef** object that affects the appearance of a segment in the project design.

Positioning a Referenced Segment

Placement of a segment in a project design is determined by three settings:

- The point of reference in the design area with respect to which the position will be calculated.
- The absolute spacing from the point of reference.
- The edge of the segment that will be placed along the position specified.

You can specify the first two settings under the **Position** area in the **General** tab as explained in this task. The edge of the segment is determined by the **Alignment** option in the **Settings** tab, discussed in the following task, “Aligning a Segment Reference.” By default, it is set to the top left corner.

To specify the position of a **SegRef** object:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Position** area, specify the following:
 - **X** – determines the position of the segment along the horizontal axis in the design area. In the associated entry box, specify the absolute spacing value, and select the required relative/absolute positioning option from the adjacent drop-down list.
 - **Y** – determines the position of the segment along the vertical axis in the design area. In the associated entry box, specify the absolute spacing value, and select the required relative/absolute positioning option from the adjacent drop-down list.



For more information on absolute/relative positioning options, refer to the task “Positioning a Field” in *Chapter 22: Working with Data in Line Mode*.



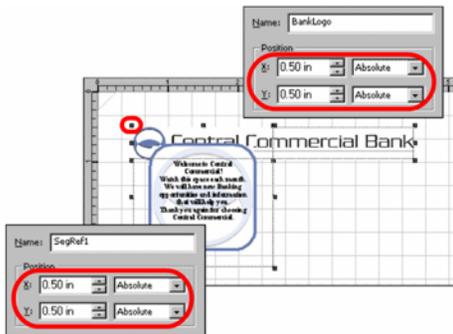
You can also resize a **SegRef** by specifying the required dimensions in the **General** tab of the *Property View* window. The steps involved in resizing a **SegRef** are the same as resizing any other object in the design area. For more information, refer to the “Setting the Object Size” section in *Chapter 8: Using the Drawing Tools*.

Aligning a Segment Reference

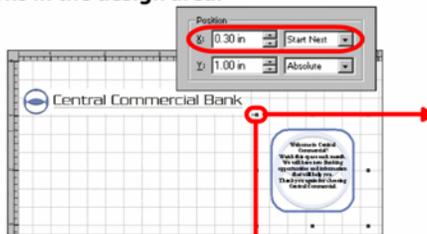
When a **SegRef** object is inserted on a page, it is aligned at the **Top Left** corner by default. You can change the segment alignment using the **Alignment** options available in the **Settings** tab of the *Property View* window, as explained below.

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Select one of the following options from the **Alignment** drop-down list:
 - **Top Left** – aligns the top left edge of the segment along the specified horizontal and vertical position.
 - **Top Right** – aligns the top right edge of the segment along the specified horizontal and vertical position.
 - **Top Center** – aligns the top center edge of the segment along the specified horizontal and vertical position.
 - **Bottom Left** – aligns the bottom left edge of the segment along the specified horizontal and vertical position.
 - **Bottom Right** – aligns the bottom right edge of the segment along the specified horizontal and vertical position.
 - **Bottom Center** – aligns the bottom center edge of the segment along the specified horizontal and vertical position.
 - **Center Left** – aligns the center left edge of the segment along the specified horizontal and vertical position.
 - **Center Right** – aligns the center right edge of the segment along the specified horizontal and vertical position.
 - **Center Center** – aligns the center of the segment along the specified horizontal and vertical position.

The following example shows the steps involved in placing a segment (SegRef) next to an image in the design area. It also shows the affect of the **Alignment** setting on the positioning of an object.



1. A SegRef and an image placed at absolute positions in the design area.



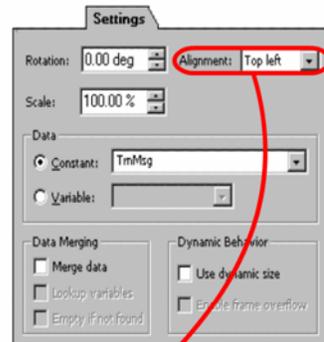
The top left corner of the SegRef placed at the specified position.



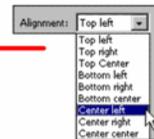
The center left corner of the SegRef placed at the specified position.

2. Select Start Next in the X drop-down to place the SegRef next to the image along the horizontal axis.

3. Specify 0.3 in the X entry box to position the SegRef at a spacing of 0.3 inches from the image.



4. In the Alignment drop-down, select Center left to position the SegRef along the center left edge.



In this example, the current field is positioned relatively at Same Next, Same Next and then displaced 0.5, 0.5 inches in the horizontal and vertical directions.

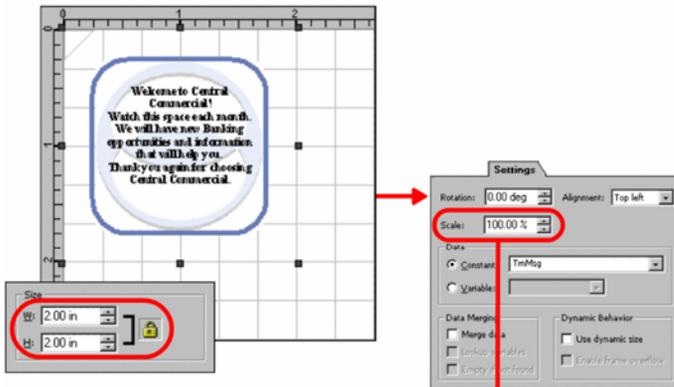
Scaling a Referenced Segment

Scaling is the process of enlarging or reducing an image size. You can scale a segment by dragging the object handles in the design area or by specifying the scaling value in the **SegRef** properties.

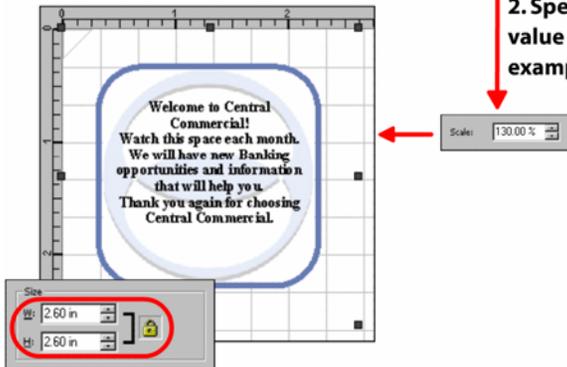
To scale a segment referenced in a design:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 Type or select the required percentage value in the **Scale** entry box. All segments are scaled to 100% by default.

1. A referenced segment scaled to 100 % by default.



2. Specify a percentage value for scaling (For example 130%).



3. Segment is enlarged based on the specified scale value. The segment size is also adjusted accordingly.

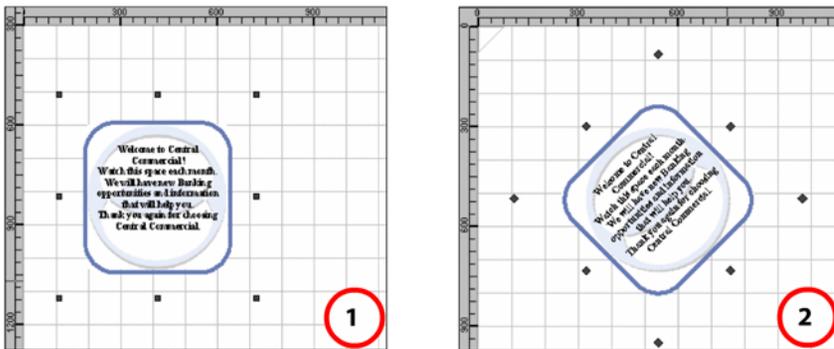
Steps involved in scaling a SegRef object.

Rotating a Referenced Segment

Vitesse enables you to use a segment resource in a project design with different orientations, depending on the design requirements. You can only rotate the linked, referenced, and inline segments.

To rotate a segment inserted in a project design:

- 1 Select the required **SegRef** or **Inline** segment node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Specify the required angle in the **Rotation** entry box. All segments are positioned at an angle of zero degrees with the x-axis by default.



Design area displaying a SegRef rotated to 1) 0 degrees by default, 2) 45%. The position of the top left corner remains the same, only the orientation is changed.

Changing the Referenced Segment

You can associate a segment resource already used in a project (and available under the **Resources>Segments** node) to the **SegRef** node that is inserted using the **Ref**

icon ).

To select the segment for referencing in a design:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 In the **Data** area, select **Constant** and choose the required segment from the adjacent drop-down list. This list displays all of the segments already used in the project under the **Resources>Segments** tree node.

The selected segment displays in the design area.



By using the **Constant** option, you can also add segments not present in the project resources by entering the segment name and extension in the edit-box. The segment resource specified using this option should be located in the current project folder, **fShared** folder or the global **SegLib** directory. You can also place the segment in any of the directories listed in the **Folder** preferences (**Segment** category). The specified segment is automatically added under the **Resources>Segments** node in the project tree. For more information, refer to *Chapter 3: Understanding the Vitesse Directory Structure*.

Using Variables to Reference a Segment

To reference a segment resource in the design using variables:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab
- 3 In the **Data** area, select **Variable** and choose the required variable from the adjacent drop-down list. The specified variable should contain a reference to the appropriate segment. The variables used for referencing segments store the name and extension of the segment resource. The drop-down list contains both the system and user-defined variables.

The project design updates to display the newly referenced segment.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required segment resource using a variable by selecting **Constant** from the **Data** area, and entering `$$VariableName` in the associated entry box. You can then select **Merge Data** under the **Data Merging** area to view the specified segment in the design area.

Dynamically Sizing Segments

By default, the size of a segment reference used in a page or a form is the same as the size of the original segment. However, Vitesse also enables you to determine the size of a **SegRef** (Segment Reference) object dynamically depending on the contents. For example, the contents of a segment resource displaying in the **SegRef** may vary depending on the document data or conditions applied. In such cases, the dynamic sizing option adjusts the height of the **SegRef** object automatically.

To dynamically size the segments, select the **Enable dynamic sizing** option under the **Dynamic Behavior** area in the properties of the segment. After enabling the feature, you then need to select the **Use dynamic size** option in the properties of the **SegRef** object for which you want to create a reference. In case you are using a field object as a segment, the same option is available in the properties of the field object. For inline segments, you only need to enable the **Enable dynamic sizing** option to dynamically resize the segment.

You can also reference a segment resource by dragging a segment resource from the project tree and dropping it onto the target node under which you want to reference the segment. If the **Enable dynamic sizing** option is already selected for the segment resource, then the reference that you just created will also have the **Use dynamic size** option selected. This helps you not to explicitly specify the **Use dynamic size** option in the properties of the referenced segment (**SegRef**).

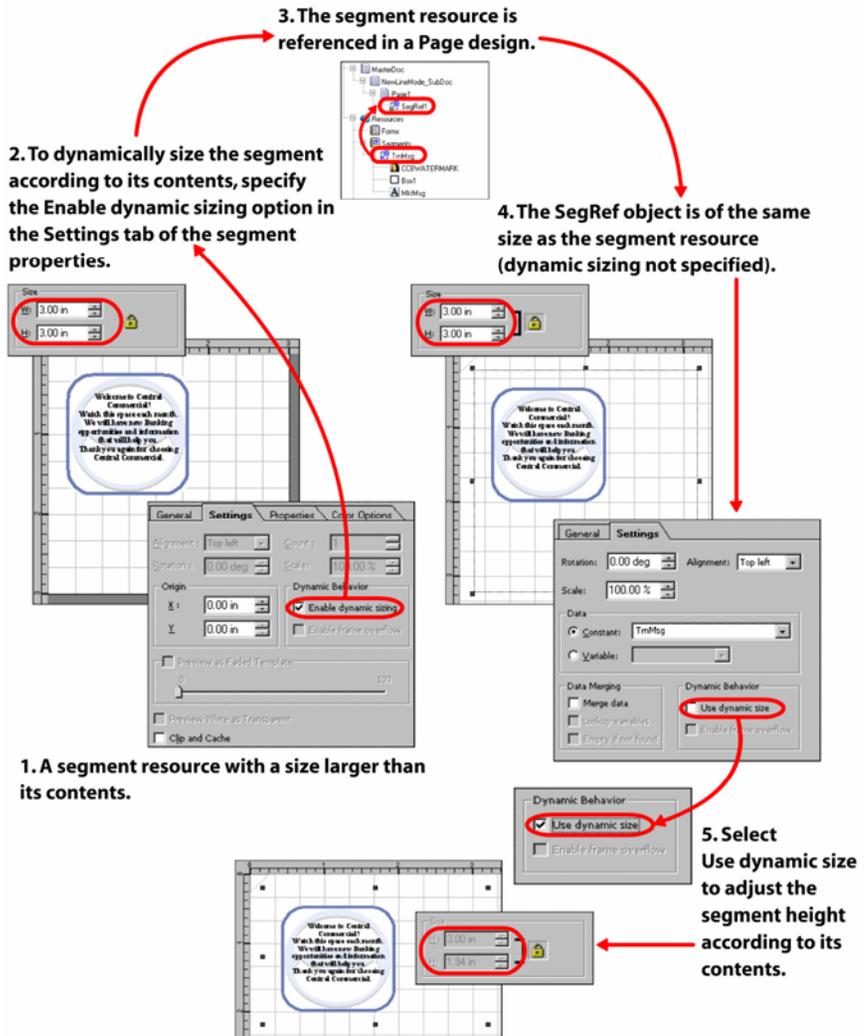
The size of a segment is calculated dynamically using the bottom corner of the last relative object in the **Segment** tree. In order to make dynamic resizing operational, the last sub-node under the required segment node must be a relative object (for example, chart, barcode, and text). The dynamically calculated size of the segment is then used to place relative objects next to it.



The dynamic size feature will not work if the last relative object under the segment node is an image.

If a condition is used as the last sub-node of a dynamically sizeable segment, then all the **If**, **Elseif**, and **Else** nodes should have a relative object mapped to them such as chart, barcode, and text.

The image below shows the steps involved in setting the dynamic sizing feature.

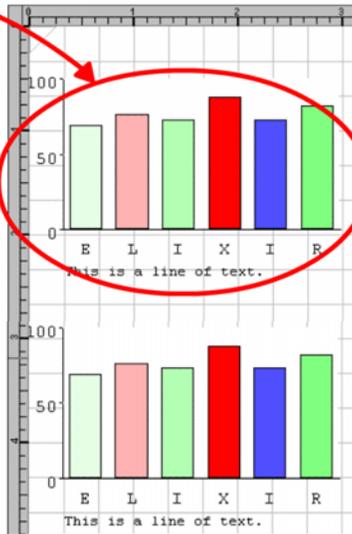
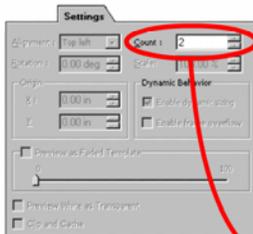
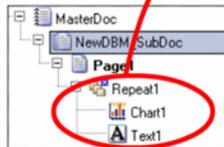


Using Repeat Segments

In a repeat segment, you can insert objects that you want to display multiple times in a project design. Insert the **Repeat** segment in the project tree. Select the **Repeat** segment node in the project tree and insert the required objects using the *Insert* toolbar and/or the *Drawing* toolbar. Field, chart, barcode, image, action, text, and graphic objects can be added in repeat segments. The objects contained in the repeat segments should be relatively placed. It is the relative positioning values of objects that determines the spacing between the repeating instances of a segment in the design. Static drawing objects can be added to a repeat segment, but they will appear only once.

You can specify the number of times you want to repeat the segment in the **Count** entry box in the **Settings** tab of the *Property View* window. No other properties are associated with a repeat segment.

A repeat segment logically grouping a chart and a text object.



The objects contained in the repeat segment appear twice in the design area, as set in the Settings tab of the Property View window.

Image showing a repeat segment containing two objects repeated twice in the design area.

Working with Segments in an XML Mode Project

In an XML mode project, Vitesse enables you to use the segments under the record, action, and condition objects as linked or referenced resources. You can also add inline and repeat segments under these three objects only. These segments can be used in the same manner in XML mode as in other project modes. Begin page and end page segments are not available in an XML mode project.

Apart from the above-mentioned segment types, Vitesse also provides the tag start, tag orphan, and tag end segments that are only specific to the XML mode projects. Tag start and tag orphan segments are used under the record objects; whereas a tag end segment is the only type that can be inserted at the page level.

To insert any of these segment types, select the required node under which you want to add a segment and click  on the *Insert* toolbar. The *Insert Segment* dialog is opened. You can select the required segment type and click **New** to insert the segment.

Using Tag Start Segments

A tag start segment is an internal document object and is inserted as the first sub node under an XML record node in the project tree. When Vitesse encounters the start tag of a record in the data file, it starts processing the tag start segment mapped under the associated record node in the project tree. The objects (such as segments, images, drawing object, etc) contained in the tag start segment are processed first (i.e. before the data contained in that record is processed).

All other objects added under this record node are processed after the tag start segment is completely processed. Objects inserted in the tag start segment are positioned relative to the previously processed objects. Each record object can contain only one tag start segment. No properties are associated with this segment type.

To insert a tag start segment, select the required record node in the project tree and

 from the *Insert Segment* dialog, select **Start**. A tag start segment is represented by a **Start** node in the project tree. You cannot drag and drop the **Start** node anywhere in the project tree.

You can select the tag start segment and using the *Insert* toolbar, insert chart, barcode, condition, and action objects in it. We recommend adding objects that you want to be processed first before the associated record has been processed.

Using a Tag Orphan Segment

A tag orphan segment is inserted under the XML record object and is processed when an orphan value is encountered in that record. Each record object can contain only one tag orphan segment.



An XML tag can contain more than one sub tag; it can also contain values between the sub tag elements. In such cases, the first value is processed normally; the remaining values are processed as orphan values. Orphan values are shown in the image below.

To insert a tag orphan segment, select the required record node in the project tree and from the *Insert Segment* dialog, select . A tag orphan segment is represented by an **Orphan** node in the project tree. You cannot drag and drop the **Orphan** node anywhere in the project tree.

You can select the tag orphan segment and using the *Insert* toolbar, insert chart, barcode, condition, and action objects in it. Add objects that you want to be processed whenever an orphan value is encountered in the associated XML data record. Tag orphan segments cannot be used as external resources. No properties are associated with this segment type.

Using a Tag End Segment

A tag end segment logically contains the objects to be processed after all the other objects in a page have been processed. It is the only type of segment that can be inserted at the page level (under the **Page** node) in an XML mode project.

To insert a tag end segment, select the required **Page** node in the project tree and from the *Insert Segment* dialog, select . A tag end segment is inserted as the last sub node of the page object. It is represented by the **End** node in the project tree. You cannot drag and drop the **End** node anywhere in the project tree.

You can select the tag end segment and using the *Insert* toolbar, insert condition and action objects in it. Add objects that you want to be processed after all other objects in the page have been processed.

Tag end segments cannot be used as external resources. There are no properties associated with a tag end segment. It only helps you to specify a required processing order of objects in a document.

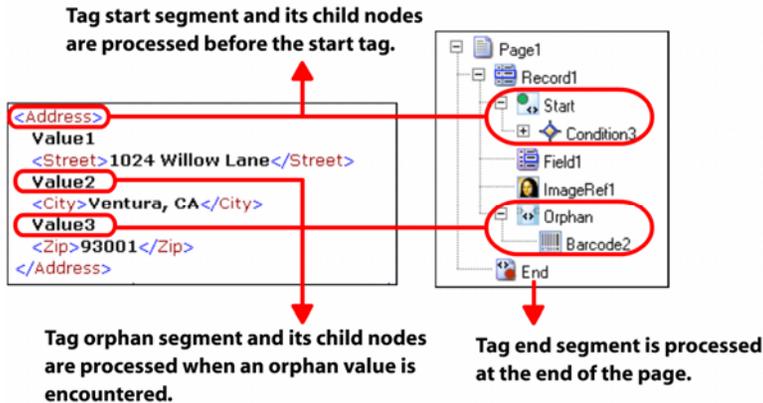


Image shows the segments used in an XML mode project and their association with the data in the data file. The Address tag is associated with the Record1 (data object in design).

Specifying Vertical Spacing between Segments

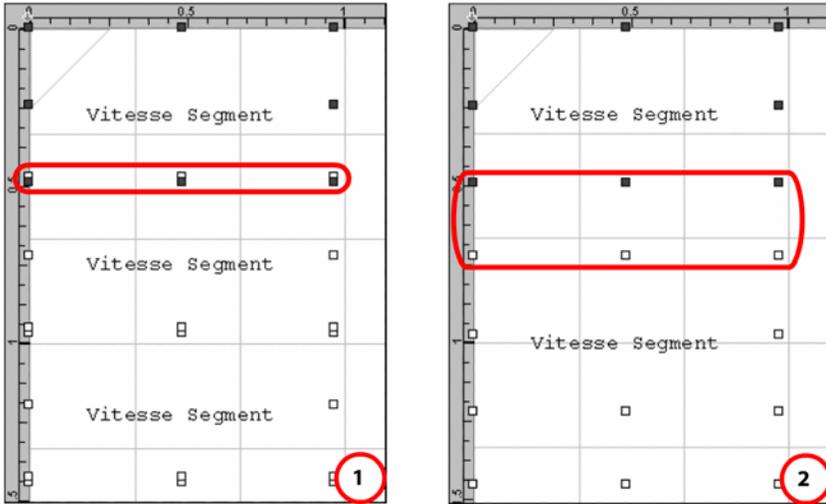
In an XML mode project, **SegRef** objects are associated with records. If the associated record or tag has multiple occurrences in the data file, the **SegRef** will be repeated for each occurrence. Each **SegRef** will be placed adjacent to the previous occurrence in the vertical direction in the design area. For such instances, Vitesse enables you to specify a vertical spacing between the segments.



The remaining properties associated with the SegRef objects in an XML mode project are the same as explained above in the section, “Setting Properties of a Segment Reference.”

To specify the vertical spacing between SegRefs:

- 1 Select the required **SegRef** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.
- 3 In the **Spacing** area, specify the required spacing value in the **Vertical** entry box. By default, all segments are placed vertically in the design area with no spacing assigned to each.



Design area displaying a SegRef object associated with an XML record with multiple occurrences: 1) Segments placed vertically with no spacing, 2) Vertical spacing set to 0.5 inches.

Chapter 33: Referencing Images

This chapter describes how to reference image resources in a project design and set image properties of the image references. The chapter covers the following tasks:

- Referencing Images
- Inserting an Image in a Project Design
- Positioning an Image References
- Resizing an Image Reference
- Rotating an Image Reference
- Scaling an Image Reference
- Aligning an Image Reference
- Changing the Referenced Image
- Using Variables to Reference an Image

Referencing Images

Vitesse enables you to use images in project designs as linked resources. In other words, image resources are not part of the actual project source and are referenced as external resources in the design. References to images in a Vitesse project design are represented by the **ImageRef** nodes in the project tree. When you insert an image in a project design, an **ImageRef** node is created under the selected design object node in the project tree. You can use images in different formats such as: *.bmp, *.jpg, *.jpeg, *.tif, *.pcx, *.png, etc.



Referencing an image in a design is different from adding an image to the project resources. An image is actually used in a project design only when it is referenced.

Inserting an Image in a Project Design

To insert an image in a project design:

- 1 From the *Project Tree View* window, select the required project node under which an image is to be added.



You can reference images at the page level in Native and Data Base Master mode projects. In a Line mode project, you can insert image references in conditions, records, and inline segments. In an XML mode project, we recommend using images at the record level. However, you can also use them at the page level under conditions and inline segments. Images can also be referenced in forms and segments.

- 2 Select Insert>Image from the menu bar. The *Insert Image* dialog is displayed with the **Existing** tab selected.



You can also open the *Insert Image* dialog either by:

- Clicking  on the *Insert* toolbar.
- Or
- Right-clicking the **Images** node and selecting **Insert>Image** from the pop up menu.

- 3 Browse to the required location and select the required image.
- 4 Click **Open** to insert the selected image as a resource in the project.

An **ImageRef** node is created under the node you selected in **Step 1**. Also, an image node is created under the **Resources>Images** node identifying the newly added image resource. **ImageRef** is the reference to the actual image resource added under the **Resources** tree node.

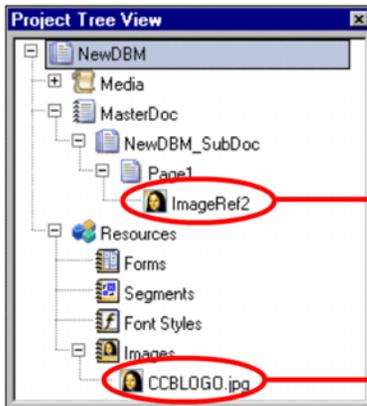


Image referenced in a Page design in DataBase Master mode project.

The actual image resource file.

Project Tree View window displaying an image resource referenced in a page.

Setting Properties of an Image Reference

ImageRef objects have certain properties associated with them that affect the appearance of the referenced image in the project design. These property settings do not affect the actual image resource.

Positioning an Image Reference

Placement of an image in a project design is determined by three settings:

- The point of reference in the design area, relative to which the position will be calculated. Vitesse provides a list of relative positioning options from which you can choose.
- The absolute spacing from the point of reference. You can indicate the absolute position of the object from the point of reference, as per the specified unit of measurement.
- The edge of the image that will be placed along the position specified.

You can specify the first two settings under the **Position** area in the **General** tab as explained in this task. The edge of the image is determined by the **Alignment** option in the **Settings** tab, discussed in the following task, “Aligning an Image Reference.” By default, it is set to the top left corner.

To specify the position of the **ImageRef** object:

- 1 Select the required **ImageRef** node from the project tree.
- 2 From the *Property View* window, select the **General** tab.



To specify the position of an object in the design area, we recommend selecting the required relative positioning option before specifying the absolute spacing in the **X** and **Y** entry boxes (if required).

- 3 From the **Position** area, select one of the following relative positioning options for setting the horizontal and vertical positions from the drop-down list:
 - **Absolute** – positions the image with reference to the top left margin of the design area.
 - **Start Same** – positions the start of the current image at the start of the previous relative object in the project tree.
 - **Start Center** – positions the start of the current image at the center of the previous relative object in the project tree.
 - **Start Next** – positions the start of the current image at the end of the previous relative object in the project tree.
 - **Center Before** – positions the center of the current image at the start of the previous relative object in the project tree.
 - **Center Same** – positions the center of the current image at the center of the previous relative object in the project tree.
 - **Center Next** – positions the center of the current image at the end of the previous relative object in the project tree.

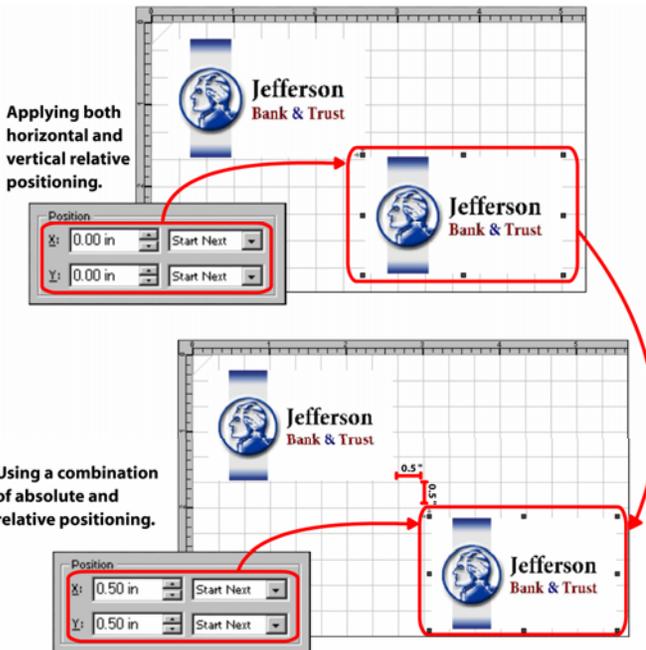
- **End Before** – positions the end of the current image at the start of the previous relative object in the project tree.
- **End Center** – positions the end of the current image at the center of the previous relative object in the project tree.
- **End Same** – positions the end of the current image at the end of the previous relative object in the project tree.



Vitesse provides the same set of vertical positioning options as the horizontal positioning options. Horizontal relative positioning options cannot be applied to an **ImageRef** object if it is the first relative object within a record object.

- 4 In the **X** and **Y** entry boxes, specify the absolute spacing of the current image from the reference point selected in **Step 3**.

Using a combination of absolute spacing and relative positioning, you can place the image relative to another image (or any other relative object) and add the spacing defined by the absolute coordinates.



In this example, the current image is positioned relatively at Same Next, Same Next and then displaced 0.5, 0.5 inches in the horizontal and vertical directions.



For more detailed information on absolute and relative positioning of objects, refer to the “Positioning a Field” section in *Chapter 22: Working with Data in Line Mode*.

Resizing an Image Reference

Vitesse enables you to resize an image maintaining its aspect ratio. By keeping the aspect ratio of an image constant, the width and height of an image increases or decreases the length and width of the object in equal proportion.

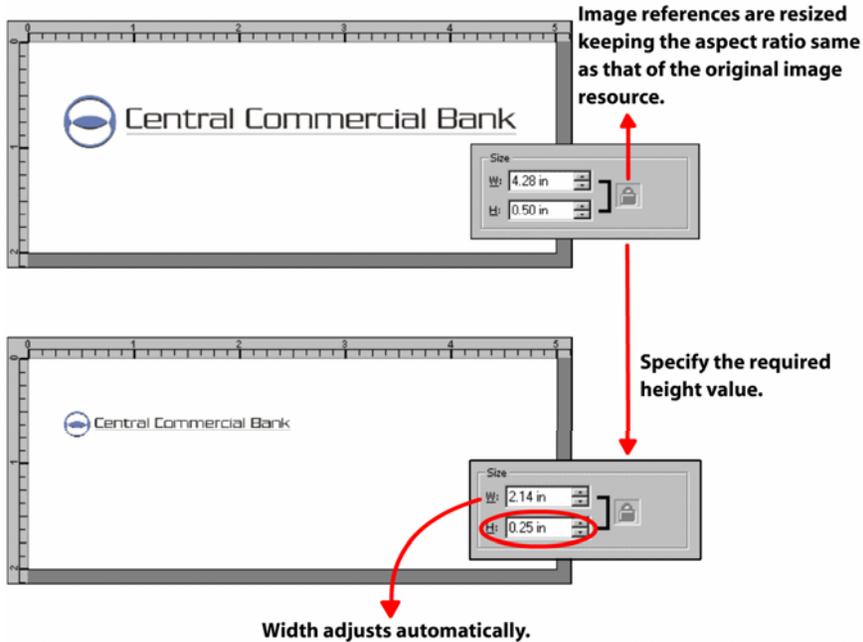
To set the image size:

- 1 Select the required **ImageRef** node from the project tree.
- 2 In the **General** tab of the *Property View* window, set the following **Size** information:
 - **W** – resizes the image with the specified horizontal value.
 - **H** – resizes the image with the specified vertical value.



You can also set the unit of measurement for the size by right clicking the upward or downward arrows in the **X** and **Y** entry boxes, and then selecting **Units**>*unit name* from the pop-up menu.

The image below shows the effects of decreasing the image height. The width decreases proportionally.



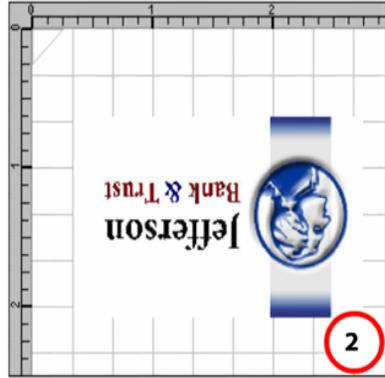
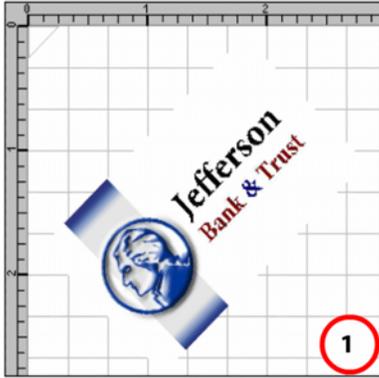
Rotating an Image Reference

To set the image rotation:

- 1 Select the required **ImageRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Type or select the required rotation angle value in the **Rotation** entry box.



For more detailed information on absolute and relative positioning of objects, refer to the “Positioning a Field” section in *Chapter 22: Working with Data in Line Mode*.



The image shows different rotations applied to an ImageRef: 1) Rotated at 45 degrees, 2) Rotated at 180 degrees.

Scaling an Image Reference

Scaling is the process of enlarging or reducing an image in size. You can scale an image by dragging in the design area or by specifying the scaling value in the **ImageRef** properties.

To scale an ImageRef:

- 1 Select the required **ImageRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab.
- 3 Type or select the required percentage value in the **Scale** entry box. All images are scaled to 100% by default.

The scaling of an image reference affects the image size, as displayed in the images below:



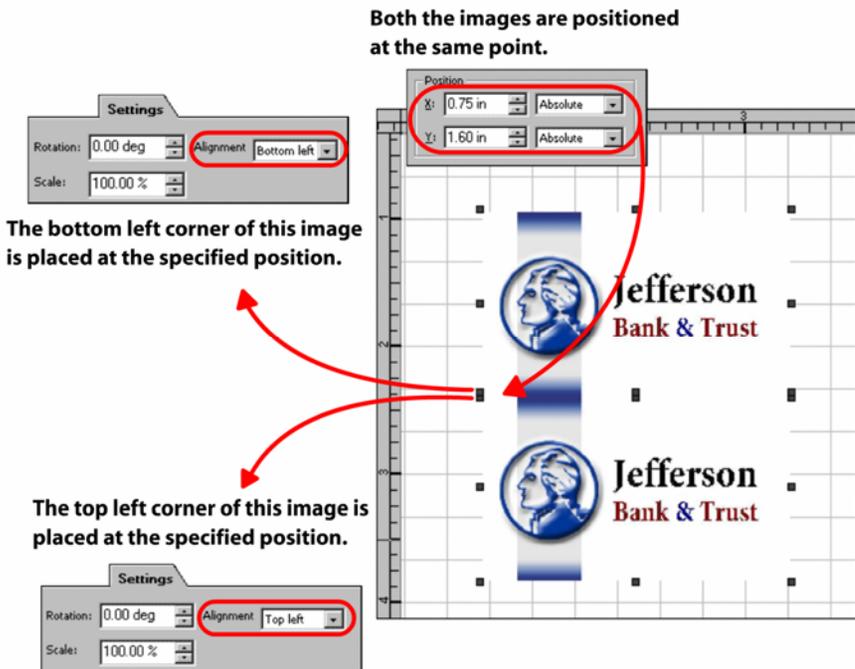
Design area displaying an ImageRef: 1) Scaled to 100%, 2) Scaled to 50%.

Aligning an Image Reference

When you specify a position (horizontal and vertical) for an ImageRef, the top left corner of the image is placed onto that position by default. The **Alignment** options enable you to specify a different edge of the image to be placed onto that position.

To align an **ImageRef** object:

- 1 Select the required **ImageRef** node from the project tree.
- 4 From the *Property View* window, select the **Settings** tab..
- 5 Select one of the following options from the **Alignment** drop-down list:
 - **Top Left** – aligns the top left edge of the image along the specified position.
 - **Top Right** – aligns the top right edge of the image along the specified position.
 - **Top Center** – aligns the top center edge of the image along the specified position.
 - **Bottom Left** – aligns the bottom left edge of the image along the specified position.



The image shows the effect of the bottom left and top left image alignment options.

- **Bottom Right** – aligns the bottom right edge of the image along the specified position.
- **Bottom Center** – aligns the bottom center edge of the image along the specified position.
- **Center Left** – aligns the center left edge of the image along the specified position.
- **Center Right** – aligns the center right edge of the image along the specified position.
- **Center Center** – aligns the center of the image along the specified position.

Changing the Referenced Image

The image resource referenced by an **ImageRef** object can be changed to any image already added to the resources of the current project. These are the resources displayed under the **Resources>Images** node in the project tree.

To change the referenced image:

- 1 Select the required **ImageRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab..
- 3 In the **Data** area, select **Constant** to select an image from the project resources; selecting this option enables the associated drop-down list.
- 4 Select the required image resource from the drop-down list. The drop-down lists all of the images in the project resources. The selected image displays in the design area.



Using the **Constant** option, you can also add images not present in the project resources by entering the image name and extension in the edit-box.

Using Variables to Reference an Image

To specify the image resource using variables:

- 1 Select the required **ImageRef** node from the project tree.
- 2 From the *Property View* window, select the **Settings** tab..
- 3 In the **Data** area, select **Variable** to specify that a variable will be used to indicate the required image resource; selecting this option enables the associated drop-down list.

The variables used for referencing images store the name and extension of the image resource.

- 4 Select the required variable from the drop-down list. You can choose from a list of system and user defined variables.

The image updates in the design area to display the newly referenced image. The image specified using the variable should be located in the current project folder, **fShared** folder or the global **ImgLib** directory. You can also place the image in any of the directories listed in the **Folder** preferences (**Image** category). For more information, refer to *Chapter 3: Understanding the Vitesse Directory Structure*.



For more information on working with variables, see *Chapter 26: Using Variables*.



You can also specify the required image resource using a variable by selecting **Constant** from the **Data** area, and entering `$$VariableName` in the associated entry box. You can then select the **Merge Data** option under the **Data Merging** area to view the specified image in the design area.

Section XI: Preparing the VIPP Project for Production

User Notes:

Chapter 34: Previewing and Testing the Project

This chapter describes activities and tasks for previewing and testing projects in Vitesse. The chapter also provides information about performing standard operations in Vitesse, including opening and saving resources. The list of topics includes:

- Opening Existing Resources
- Opening Recently Used Resources
- Previewing a Project
- Setting Memory Size for Optimizing of View
- Working with Source Code
- Compiling VIPP Code
- Exporting Source Code
- Saving Resources

Opening Existing Resources

Vitesse enables you to open previously created resources such as documents (*.EVD), segments (*.EVS), forms (*.EVF), and Vitesse zip files (*.EVZ).

To open an existing resource in Vitesse:

- 1 Select File>Open from the menu bar to open the *New Project* dialog.



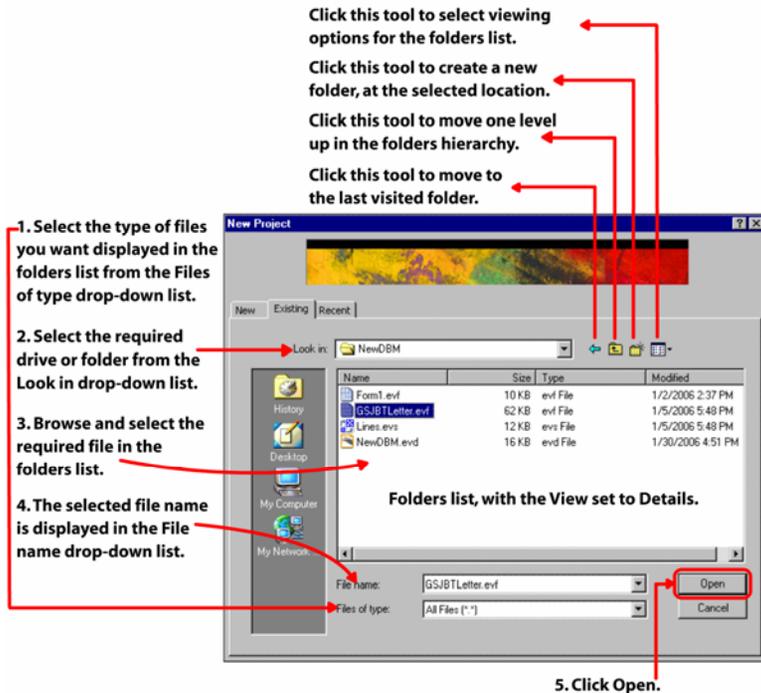
You can also open the *New Project* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <O> on the keyboard.

- 2 From the *New Project* dialog, select the **Existing** tab.
- 3 Select one of the following options from the **Files of type** drop-down list:
 - **All Vitesse Files (*.evd, *.evf, *.evs, *.evz)** – displays all Vitesse files in the folder list available in the tab.
 - **Document Files (*.evd)** – displays only Vitesse document files in the folder list available in the tab.
 - **Segment Files (*.evs)** – displays only Vitesse segment files in the folder list available in the tab.
 - **Form Files (*.evf)** – displays only Vitesse form files in the folder list available in the tab.

- **Vitesse Zip Files (*.evz)** – displays only Vitesse zip files in the folder list available in the tab.
 - **All Files (*.*)** – displays all file formats in the folder list available in the tab.
- 4 Search **Look in** drop-down list for the required target folder.
 - 5 Select the required file, and then click **Open**.

The selected resource opens in Vitesse.



The image displays the process of opening a resource in Vitesse.



- You can also close a resource either by:
- Selecting File>Close from the menu bar.
 - Or
 - Pressing <Ctrl> + <F4> on the keyboard.

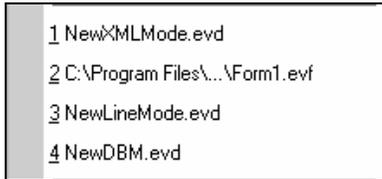


Vitesse converts any resources created using previous versions of the application to the new format before opening them.

Opening Recently Used Resources

Vitesse maintains a list of recently used resources to help you keep track of the frequently used files. To open a recently used resource file, you can either:

- Select the required resource from the File menu.



The most recently used file displays at the top of the list with sequence number 1.

- Select the required resource using the *New Project* dialog.

To open a recently used resource file using the *New Project* dialog:

- 1 Select File>Open from the menu bar to open the *New Project* dialog.



You can also open the *New Project* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <O> on the keyboard.

- 2 From the *New Project* dialog, select the **Recent** tab.
- 3 Select the required resource file from the list, and then click **Open**.

The selected resource opens in Vitesse.



You can also close a resource either by:

- Selecting File>Close from the menu bar.
- Or
- Pressing <Ctrl> + <F4> on the keyboard.

Previewing a Project

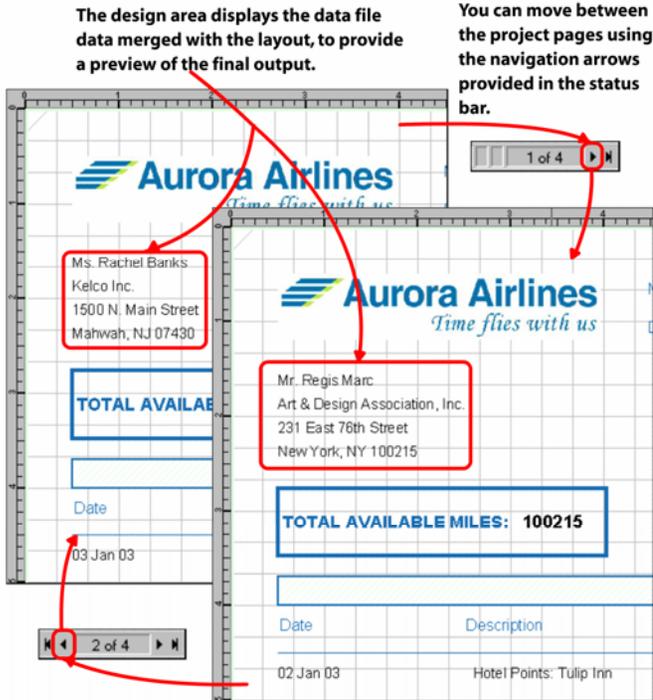
Vitesse is a real time visual designer that provides a preview of the final output by merging the design with the variable data. The Vitesse print preview feature accurately reproduces the output that would be produced using a VIPP enabled printer.

Vitesse uses a project specific sample data file to display the composed design in the design area during the application development process. You can also open a separate preview window to print preview your design. It is a recommended practice to preview your document before exporting the completed document as a .VPC file.

Proof-Printing a Project

You can print proof your project using either:

- Design area
- Separate preview window



Status bar highlights the total number of pages generated by the project with the current data file.

The design area displays the project with merged data from the sample data file.



You can also preview your project in a separate preview window either by:

- Selecting File>Print Preview from the menu bar.

Or

- Clicking  on the *Standard* toolbar.

Or

- Pressing <Ctrl> + <Shift> + <P> on the keyboard.

The **Print Preview** tool is enabled for page, form, segment objects, and any sub-nodes of these objects. See the “Print Previewing Multi-Up Projects” task later in this chapter.

Print Previewing Multi-Up Projects

The N-up settings applied to a project are not visible in the design area. To view the effect of these settings, you need to use the print preview feature provided by Vitesse.

To print preview a project using the preview window:

- 1 Create an N-up enabled project in any mode.
- 2 Select File>Print Preview from the menu bar to display the preview window.

The following tools are available in the preview window on the preview toolbar:

Tool	Purpose
	Print tool – opens the <i>Print</i> dialog. You can print the composed design using the <i>Print</i> dialog options.
	Back tool – moves to the previous page in the design. The tool is disabled for the first page in the design.
	Next tool – moves to the next page in the design. The tool is disabled for the last page in the design.
	Zoom In tool –magnifies the print view.
	Zoom Out tool – reduces the size of items in the view.
	Full Screen tool – scales the preview window to fit the entire viewing area of the screen.
	Show/Hide Properties tool – opens the <i>Properties</i> dialog. This dialog reflects the N-up options specified in the project. You can modify the N-up settings using this dialog; the corresponding settings are also updated in the project.
	Close tool – closes the preview window and returns to the Vitesse workspace.

- 3 Click  to return to the Vitesse workspace.

Setting Memory Size for Optimization for View

When designing graphic-intensive projects in Vitesse, the application can slow down due to the large image and PostScript file sizes. You can optimize the performance of Vitesse by setting optimization memory size and using low resolution images for the preview process.

To optimize designer preview:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **General** category.
- 3 Select the **Use low resolution 96 dpi images for optimized preview in Designer** option. Selecting this option minimizes memory usage by creating 96 dpi versions of the images for display in Vitesse.



Selecting this option does not modify the actual image file. When you export your project as a VPC file, the original image is packaged. This option only enables you to create a low-resolution version of large image files so that the Vitesse design process is not slowed down while processing the images.

- 4 Enter the amount of memory you want to allocate for the design process in MB in the **Maximum Optimization Memory** entry box. 30 MB is the recommended memory size for an image intensive design.
- 5 Click **OK**.

The settings take effect the next time a project is opened.

Working with Source Code

Vitesse enables you to view and edit the generated VIPP source code using the **Source View** tab in the design area. You can view the source code at any time when designing. The source code contains variables, constants, comments, integers, key words, and real numbers displayed using different colors in the **Source View** tab. Vitesse also provides you with an inbuilt VIPP compiler to compile your source file after making any modifications. You can customize the appearance of the **Source View** tab using options provided in the *Preferences* dialog.

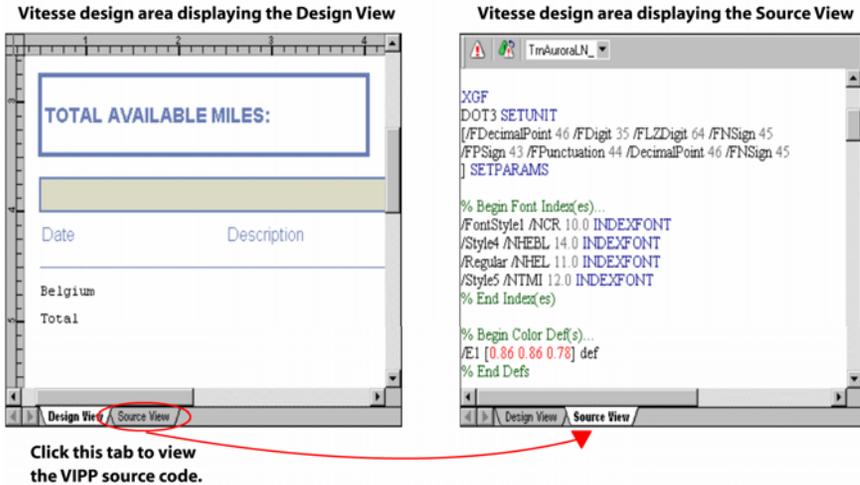
The **Source View** tab displays the following source files depending on the type of resource:

- **Line Mode Project** – *.JDT (Job Descriptor Ticket) file containing VIPP commands that define the required presentation of line printer data.
- **Native Mode Project** – *.NM (Native Mode Job) file that contains only VIPP commands. These commands are used to place text, images, graphics, logos, forms, and segments directly on a page.
- **Database Mode Project** – *.DBM (Data Base Master) file containing VIPP native mode commands and variable references used to control the page composition of a job.
- **XML Mode Project** – *.XJT (XML Job Ticket) file containing instructions on how to process and arrange XML data in a document.
- **Form** – *.FRM containing VIPP native mode commands.
- **Segment** – *.SEG containing VIPP native mode commands.

Viewing the Source Code

To view the source code for an entire project or external resources used in a project either:

- Select the resource or project node in the project tree, and select the **Source View** tab in the design area. Selecting the **Source View** tab generates the VIPP source code for the selected node. To view the project source code, select the **SubDoc** or any of the sub nodes.



You can view both the design and the generated source code using the design area tabs.

- Select the **Source View** tab in the design area, and select the required source file from the drop-down list provided on the **Source View** tab toolbar.



The drop-down in the Source View toolbar displays the source file names for the project, and any resources added to the project.



Source code is not generated for all the nodes in the project tree. You cannot view source code for **Media** and **MasterDoc** nodes.

Searching in the Source Code

Vitesse helps you to quickly search for every occurrence of a required keyword in the source code using the **Find** tool.

Finding Text in the Source Code

To search for a particular word or phrase:

- 1 Select the **Source View** tab in the design area.
- 2 Select Edit>Find from the menu bar to open the *Find* dialog .



You can also open the *Find* dialog either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <F> on the keyboard.

- 3 Enter the text you want to search for in the **Find what** entry box; entering text in this entry box enables the **Find Next** button.
- 4 Select the **Match case** option to make the search case sensitive.
- 5 Click **Find Next**. The first occurrence of the specified text is highlighted in the source view. You can click the button again to view the next occurrence and so on. Vitesse displays a message when all of the source code has been searched for the specified text.

Replacing Text in the Source Code

You can also search for and replace occurrences of a particular word or phrase in the source code with a specified string:

- 1 Select the **Source View** tab in the design area.
- 2 Select Edit>Replace from the menu bar to open the *Replace* dialog.



You can also open the *Replace* dialog by pressing <Ctrl> + <H> on the keyboard.

- 3 Enter the text you want to search for in the **Find what** edit box; entering text in this edit box enables the **Find Next**, **Replace**, and **Replace All** buttons.
- 4 Enter the replacement text in the **Replace with** edit box.
- 5 Select the **Match case** option to make the search case sensitive.
- 6 Do either of the following:
 - Click **Find Next**. The first occurrence of the specified text is highlighted in the source view. Click **Replace** to replace the selected text with the replacement string. You can click the **Find Next** button again to view the next occurrence and so on.

- Click **Replace All** to automatically replace all occurrences of the specified text with the replacement text.

Vitesse displays a message when all of the source code has been searched for the specified text.



You can also use the find and replace functionality for text and graphic text objects. See *Chapter 9: Using Text Tools* for more information.

Editing the Source Code

Vitesse enables you to edit VIPP commands in the **Source View** tab. You can modify the Vitesse generated source code, add your own code, or add comments to the code. The **Source View** tab supports Cut, Copy, Paste, Delete, and Select All operations to help you in editing the source code.

Compiling VIPP Code

Whenever you edit the source code in the **Source View** tab, the code has to be compiled. To compile the source code:

- 1 Select the **Source View** tab in the design area.
- 2 Make the required modifications.
- 3 Do one of the following:
 - Switch from the **Source View** to the **Design View** tab. The source code is automatically compiled.
 - Click  on the **Source View** tab toolbar.
 - Select Tools>Source View>Compile from the menu bar.

When compiling the source code, error, warning, and/or informational messages may display in the *Message View* window. You should not switch from the **Source View** tab to the **Design View** tab until all errors are corrected, or else you may lose your changes.



For more information on source code error handling, see *Appendix A: Error and Warning Message Handling*.

Reverting to the Last Compiled State

If errors cause the edited source code compilation to fail, you can restore the source code to the last valid compiled state as follows:

- Select Tools>Source View>Return to last valid state from the menu bar.
Or
- Click  on the **Source View** tab toolbar.
Or
- Right-click at any location in the **Source View** tab, then select **Revert to Last** from the pop-up menu.

Exporting Source Code

Vitesse enables you to save (export) the source code generated for an entire project or external resources.

To export the source code:

- 1 Select the **Source View** tab in the design area.
- 2 Select the required source file from the drop-down list provided on the **Source View** toolbar. The selected file displays in the **Source View** tab.
- 3 Right-click at any location in the **Source View** tab and select the **Source Export** option from the pop-up menu.
The *Source Export* dialog is displayed.
- 4 In the *Source Export* dialog, browse and select the required location using the **Save in** list.
- 5 Specify the name of the source file in the **File name** entry box.
- 6 Click **Save**.



You do not need to specify the file type when exporting the source file. Vitesse automatically saves the file with the correct extension.

- Form source code saves as *.FRM.
- Segment source code saves as *.SEG.
- Line, native, database, and XML project source code save as *.JDT.

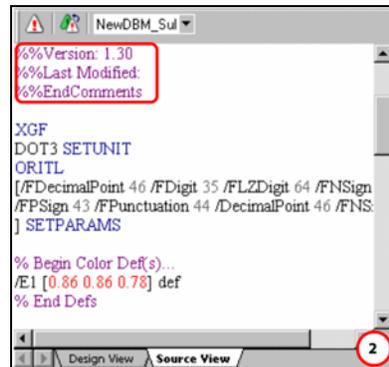
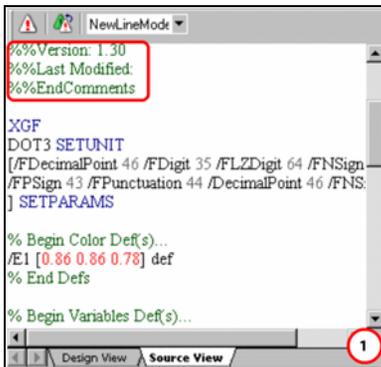
Setting Default Source Code Options

Within the **Source View** tab, you can use different colors for displaying the source code comments, integers, keywords, and real numbers, etc. Using different colors for different object codes increases the readability of the source code. You can customize the color scheme using the options provided in the *Preferences* dialog. You can also set the font used for displaying the source code. The scope of these settings is for the entire application, not only for the current project.

Setting Colors for Source Code Objects

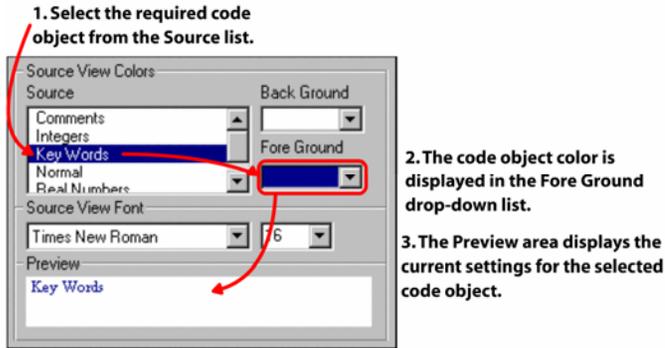
To set the source code objects color options:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Source** category under the **Workspace** node.
- 3 From the **Source View Colors** area, select the code object (for example, **Comments**) from the **Source** list whose color you want to modify.
- 4 Select the required color from the **Fore Ground** drop-down list. The **Preview** area displays the updated settings.
- 5 Click **OK**.



The *Source View* tab displaying comments with (1) dark green color (2) violet color.

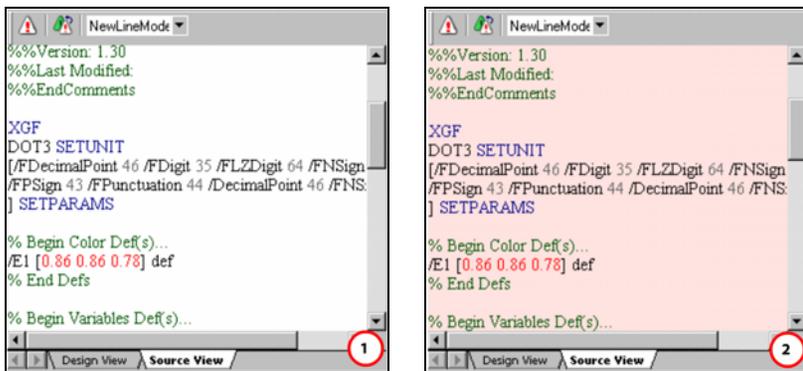
You can also use the **Source** category in the *Preferences* dialog to view the colors assigned to different code objects, as shown in the image below:



Setting Source View Background Color

To set a background color within the **Source View** tab:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Source** category under the **Workspace** node.
- 3 From the **Source View Colors** area, select the required background color from the **Back Ground** drop-down list.
The **Preview** area displays the updated settings.
- 4 Click **OK**.

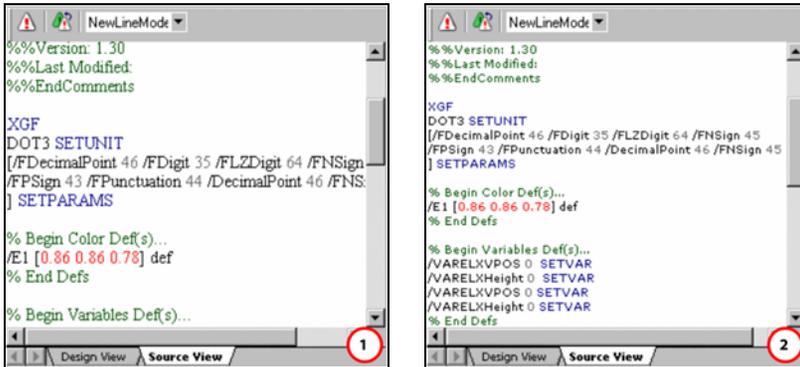


The *Source View* tab displaying source code with (1) white background (2) rose background.

Setting a Default Font for the Source Code

To set the display font within the **Source View** tab:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Source** category under the **Workspace** node.
- 3 From the **Source View Font** area, select the required font from the drop-down list.
- 4 Select the required font size from the drop-down list next to the font drop-down list in the **Source View Font** area.
The **Preview** area displays the updated settings.
- 5 Click **OK**.



The Source View tab displaying source code with font settings of (1) Times New Roman, 16pt (2) Verdana, 12pt.

Saving Resources

Vitesse saves resources and projects using the name and location specified at the time of creation. You can also save projects under different client folders, using a different name or at a location other than the default location at the time of saving the project.

Vitesse also enables you to save projects or resources in Vitesse zip format. With the Vitesse zip format, you can package all the resources used in a project, and any external resources specified in the **Other Files** tab associated with the root project node property sheet as a single zip file. This helps in moving your project from one PC to another. You can open this file on another PC using File>Open from the menu bar. Once this file opens, Vitesse prompts you to save this project in the current PC working folder. All resources are automatically placed in their respective folders while maintaining their local, global, and shared scope.

Saving a Project

To save a native, line, database, or XML mode project:

- 1 Select File>Save from the menu bar.

The project saves as a separate folder at the location specified when creating the project. The name of the project folder is the same as the project file.



You can also save a project either by:

- Clicking  on the *Standard* toolbar.
Or
- Pressing <Ctrl> + <S> on the keyboard.

- 2 To save the project using a different name, under a different client, or project folder, select File>Save As.

The *Save As* dialog displays.

- 3 Specify the project name in the **File name** entry box.
- 4 Specify the project folder name from the **Project Name** drop-down list.
- 5 Specify the client name from the **Client Name** drop-down list.
- 6 Click **OK**.



All projects created in Vitesse are saved as an Elixir Vitesse Document (*.EVD).

Saving a Resource

To save a form or segment resource:

- 1 Select File>Save from the menu bar.

The resource saves at the location specified when creating the resource.



You can also save a resource either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <S> on the keyboard.

- 2 To save the resource using a different name or at a location, select File>Save As from the menu bar.

The *Save As* dialog displays.

- 3 Browse and select the desired location using the **Save in** list.
- 4 Specify the name of the resource in the **File name** entry box.
- 5 Click **Save**.



All segments created in Vitesse are saved as an Elixir Vitesse Segment (*.EVS).
All forms created in Vitesse are saved as an Elixir Vitesse Form (*.EVF).

Saving Projects in Vitesse Zip Format

To save a resource or project in zip format:

- 1 Select File>Package Vitesse Zip from the menu bar.

The *Save As Zip* dialog displays.

- 2 Browse and select the desired location using the **Save in** list.
- 3 Specify the name of the zip file in the **File name** entry box.
- 4 Click **Save**.



All projects and resources packaged as Vitesse zip are saved as an Elixir Vitesse Zip (*.EVZ).

Chapter 35: Importing and Exporting Projects

This chapter provides information regarding the import and export operations of the Vitesse design projects and describes activities and tasks related to them. Information includes the following:

- Setting Import Preferences
- Setting Text Merge Preferences
- Using Comments in the Imported Project
- Using Preferences from a DEF File
- Importing a Project to Vitesse
- Setting Export Preferences
- Setting the Default Measurement Unit
- Setting a Comment Level for the Exported Project
- Converting TIFF Files
- Setting the Buffer Size
- Exporting a Vitesse Project

Importing Projects

Vitesse allows you to import native VIPP projects and resources to your design using the import functionality. All of the native VIPP formats, namely: Job Descriptor Ticket (*.jdt), VIPP form (*.frm), VIPP segment (*.seg), Native mode projects (*.nm), DBM mode projects (*.dbm), VIPP container files (*.vpc), and VIPP project files (*.vpf) can be imported to their corresponding Vitesse formats. Additionally, you can also import the Legacy Elixir Format (*.elx) and Xerox formats (*.frm, *.fsl) for migration from legacy streams (for example, Line Condition Data Stream (LCDS) and Metacode). Once imported, Vitesse enables you to edit these projects using the visual interface or by editing the source code, and you can save them.

Setting Import Preferences

Accurate import settings that match the requirements of the source project are important for a successful import result. Before importing projects, we recommend specifying certain options (explained in subsequent tasks) for accurate importing.

Setting Text Merge Preferences

When you import a project in Vitesse, any text used in the imported project is represented within the text objects in the Vitesse design area. You can then specify the required layout for the imported text in the project using different settings. For example, a native design source may contain more than one line of text with the same properties. By applying the text merge feature, you can choose to keep the lines as different text objects or merge the lines into a single text object in the Vitesse design area.

To specify the text merging preferences for imported projects:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Import** category to view the associated options.
- 3 From the **Text Grouping** area, select any one of the following options:
 - **None** – disables text merging.
 - **Group** – merges text that belongs to the same run. Run is a sequence of characters that have common properties such as color, font, style, etc. and are placed or positioned close to each other in the design.
 - **Maximum** – merges text that may not belong to the same run but are positioned close together.
- 4 Click **OK**.

The settings take effect the next time a project is imported in Vitesse.

Using Comments in the Imported Project

The comments specified in a native VIPP or legacy project can also be made available for viewing after importing the project into the Vitesse environment.

To use comments in the imported project:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Import** category to view the associated options.
- 3 Select the following options as required:
 - **Process Elixir Comments** – processes comments inserted by Vitesse (non VIPP comments) when importing a previously exported Vitesse project.
 - **Process Legacy Comments** – processes the inherent comments in a legacy project. For example, VIPP comments in a native VIPP project.
- 4 Click **OK** to make the settings effective for the imported projects.

Using Preferences from a DEF File

Using Vitesse, you can either create a project and export it to VIPP for printing, or create a project and generate a Def file for it before exporting to VIPP for printing. A Def file stores default settings specified for VIPP applications.

Before exporting, appropriate settings are important in order to perform a successful export. Defaults for a project exported to VIPP for printing are generated within the application source code; whereas, defaults for an application for which a Def file is generated are stored in the Def file.

This feature is intended for use by advanced users or VIPP coders only.

The **xgf.def** file is a setup file specifying the general settings used by the Vitesse application whenever it is launched. It is located in the **Config** folder within the **Vitesse** directory. You can view the default values by opening the **xgf.def** file in any text editor.

To ensure that Vitesse uses the correct default preferences for importing a project into Vitesse, you can reload these settings. In the **Import** category of the *Preferences* dialog, click the **Reload values from def file** button; the default settings are reloaded from the local Def file (**xgf.def**) at *drive:\Program Files\Elixir Technologies\Vitesse\Config*, and a message box displays, informing that the default Def file has been loaded.

After specifying the default preferences, you can now import the required project into Vitesse.

Importing a Project

To import a project or a resource:

- 1 Select File>Import from the menu bar to open the *Import* dialog with the **Existing** tab selected.



You can also open the *Import* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <M> on the keyboard.

- 2 Select one of the following options from the **Files of type** drop-down list:
 - All Vitesse Imports (*.jdt, *.dbm, *.frm, *.seg, *.nm, *.vpc, *.vpf, *.elx, *.fsl) – displays all Vitesse files in the folder list available in the tab.
 - JDT Project (*.jdt) – displays only JDT resource files in the folder list available in the tab.
 - VIPP Form (*.frm) – displays only VIPP form files in the folder list available in the tab.

- VIPP Segment (*.seg) – displays only VIPP segment files in the folder list available in the tab.
- Native Mode Project (*.nm) – displays only VIPP Native Mode project files in the folder list available in the tab.
- DBM Mode Project (*.dbm) – displays only VIPP DBM project files in the folder list available in the tab.
- VIPP Container Files (*.vpc) – displays only VIPP Project Container files in the folder list available in the tab.
- VIPP Project Files (*.vpf) – displays only VIPP project files in the folder list available in the tab.
- Legacy Elixir Format (*.elx) – displays only Legacy Elixir Format files in the folder list available in the tab.
- Xerox FSL Format (*.fsl) – displays only Xerox FSL Format files in the folder list available in the tab.
- All Files (*.*) – displays all of the file formats in the folder list available in the tab.

3 Search the **Look in** drop down list for the required target folder.

4 Select the required file.

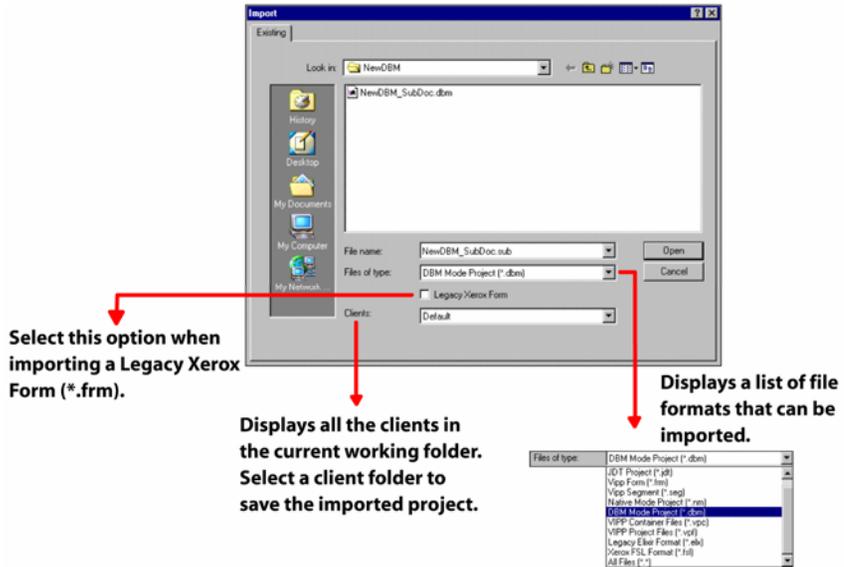


If the file being imported is a Xerox legacy form file that has the FRM extension, select the **Legacy Xerox Form** check box. Both VIPP Form and Legacy Xerox Form types use the FRM extension. Therefore, Vitesse requires this checkbox to be enabled to determine whether the FRM file being imported is a VIPP Form or a Xerox Legacy Form.

5 If you are importing a Native, Database, Line, or XML mode project (*.nm, *.dbm, *.jdt, *.xjt), select the required client folder from the **Clients** drop down list. The **Clients** drop down displays all of the clients in the current working folder. When you save the imported project, it will be saved in the selected client folder.



The imported form and segment resources will be saved in the source directory. Source directory is the folder where the resource files being imported reside. When saving the imported VPC and VPF files, Vitesse prompts you to specify the required location using the *Save As* dialog. For more information, refer to the “Saving a Project” task in *Chapter 35: Previewing and Testing the Project*.



The Import dialog highlighting various options.

6 Click **Open**.

The selected resource imports in Vitesse.

Exporting Projects

All resources and projects created in Vitesse are saved in their respective Vitesse directories, using Vitesse file formats. However, once a document is composed and finalized, you are required to export the project in the VIPP format for printing at the destination VIPP printer. Before exporting, you can choose to specify various preferences as explained in the “Setting the Export Preferences” section below.

Setting the Export Preferences

Before exporting your project, we recommend specifying certain preferences for a successful export operation. For this, you can use the **Export** preferences that define how a resource will be exported. These are application settings, and they only need to be set up once.



We also recommend reviewing the font map settings before exporting a project or a resource. For information on font maps, refer to *Chapter 6: Working with Fonts*.

Setting the Default Measurement Unit

Vitesse enables you to specify a unit of measurement that will be used for the accurate placement of document objects in the exported resources. This unit is used in the VIPP source code and must match your printer preferences.



Generally, VIPP uses the **Dot3** unit of measurement.

To specify a unit of measurement:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Export** category to view the associated options.
- 3 Specify the required unit of measurement from the **Default unit** drop-down list.
- 4 Click **OK**.

Setting a Comment Level for the Exported Project

When you are designing a project or a resource in Vitesse, the corresponding VIPP source code is generated in parallel and can be viewed using the **Source View** tab in the design area. Vitesse inserts useful comments in the source code automatically. When the project is exported, Vitesse enables you to export the comments as well, along with the VIPP source code. You can set a comment level as per your design requirements. When exporting complicated designs, we recommend making use of extensive comment levels in order to make the source code easily understandable.

To specify a comment level for the exported projects:

- 1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Export** category to view the associated options.
- 3 From the **Comments** area, select one of the following options as required:
 - **No Comments** – does not include any Elixir (or Vitesse) comments.
 - **Lightly Commented** – includes Elixir comments for selective objects only.
 - **Heavily Commented** – includes Elixir comments for all objects used in the document design. This also helps find the source code for each design object.
- 4 Click **OK**. The settings are applied for subsequent project exports.

Using Elixir Comments

When exporting a project to the VIPP format, some comments related to the export process can be automatically generated. These are helpful notes regarding the conversion of various Vitesse design objects (for example, dynamic tables, polygons, etc.) to the VIPP format. These comments are later used by Vitesse when the project is imported so that the design objects are reconstructed from the VIPP code as they actually were in the original Vitesse project.

You can use this feature by selecting the **Generate Elixir Comments** option in the **Export** category of the *Preferences* dialog.

Converting TIFF Files

Vitesse enables you to use images in various formats in a document design. VIPP however, supports images in JPEG and TIFF formats when generating the final print output. TIFF is the preferred format for using images in VIPP. To comply with this requirement, Vitesse automatically converts all of the images used in a document design to the TIFF format when exporting a document. It also enables you to optionally skip the TIFF conversion of images by selecting the **Ignore TIFF files conversion** option in the **Export** category of the *Preferences* dialog.

Setting the Buffer Size

When a document is composed (using Line, Database, or XML mode) to generate the final print output, it also requires variable data from a data file to be merged into the static layout. During this phase, data from the data file is processed line by line for all of the three types of data file formats (*.dbm, *.jdt, *.xjt). Vitesse enables you to specify a buffer size, enough to handle this variable data when actually composing the document.

To set the buffer size:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Export** category to view the associated options.
- 3 Select **Set Buffer Size** to enable the associated entry box.
- 4 Specify the maximum number of characters available in a line of data in the specified data file in the **Buffer Size** entry box.
- 5 Click **OK**. The settings take affect for subsequent project exports.

Setting Maximum Number of Forms

For generating the final print output, the resources referenced in a document design are transported to the print controller (as a VPC package), from where they are used for the document composition. These resources must reside on the printer controller during the process of document composition and therefore, make use of its memory space. Vitesse enables you to optimize the memory usage by specifying a maximum number of forms in a document. Depending on the specified number of forms, a memory is allocated for form resources on the print controller accordingly.

To specify the maximum number of forms in a document:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Export** category to view the associated options.
- 3 Specify the maximum number of forms in the **Maximum Forms** entry box.
- 4 Click **OK**. The settings take effect for subsequent project exports

After specifying the default preferences, you can now export the required project to the required VIPP format.



If the number of forms in your document exceeds the specified **Maximum Forms** size, a warning is generated at the printer controller indicating the exceeding size.

Exporting a Vitesse Project

After a project or a resource design is complete and is ready for printing, you can export the completed projects or stand-alone Vitesse resources (forms and segments) to the required VIPP formats.

When you export a complete project, a set of VIPP files is generated. These include:

- VIPP project job file (*.nm, *.jdt, *.dbm, *.xjt), depending on the project type.
- VIPP Project File (*.vpf).
- Submission file (*.sub).
- Production data file (*.lm, *.dbf, *.xml), depending on the project type.



In keeping with Dynamic Document Composition (DDC), VIPP resources are typically transported over the network to the PostScript print controller, and then merged with the production data file by the VIPP compiler.

- Resources contained in a project and converted to their respective VIPP formats. Un-mapped fonts used in a design are also exported as PostScript fonts.

All of these files are saved in a project folder that is created within the **Client** folder in a user-specified location. Vitesse also enables you to package these in a VIPP Project Container (VPC) file.

VIPP Project File

A VIPP Project File (*.vpf) contains all of the information needed for a single project. It is located in the project directory along with the other project resources. The VPF contains four kinds of information about each project:

- Bibliographic information.
- VIPP resources.
- Non-VIPP resources.
- Historical information.

VIPP Project Container

A VIPP Project Container (*.vpc) is a compressed archive file containing the physical grouping of all the VIPP project resources, including the VPF. They simplify transferring projects between applications, devices, and locations. It is a zipped file and can be extracted using any un-zip utility to view the included resources. VPCs can also be printed directly by submitting them to the print controller.

The stand-alone resources, when exported individually, are created as a single VIPP file within the specified location.

Packaging Other Files with a Project

The resources inserted in a Vitesse project are all packaged into the VIPP Project Container (*.vpc) on export. Besides the resources already included in a project, Vitesse also enables you to specify any other files to be exported as separate resources with the composed project.



The stand-alone resources are exported as a single VIPP file. Therefore, you cannot package other files with the exported segments and forms.

To specify the required resources:

- 1 Select the root project node from the *Project Tree View* window.
- 2 From the *Property View* window, select the **Other Files** tab.
- 3 Click  to add a new row for specifying the required resource file.
- 4 Click  in the **File Path** cell to view the *Open* dialog with the **Existing** tab selected.
- 5 Search the **Look in** drop-down list for required target folder.
- 6 Select the required file, and then click **Open**.
The selected file path displays in the **File Path** entry box. You can also double-click the cell to enter the required path.
- 7 Click the **Scope** cell to specify a scope (either: **Global**, **Local**, or **Shared**) for the resource from the drop-down list.



For more information regarding scope of resources, refer to *Chapter 3: Understanding the Vitesse Directory Structure*.

- 8 Double-click the **Description** cell and specify brief information about the selected resource.

All of the resource files specified in this list will be exported with the current project; and will also be available when this exported project is imported back to the Vitesse environment.



Click  to delete a selected row from the list of files.

Exporting a Project

To export a Vitesse project/resource opened in the design area:

- 1 Select File>Export from the menu bar to open the *Export* dialog.



You can also open the *Export* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <E> on the keyboard.

- 2 The **File Name** entry box indicates the name of the project to be exported. You can also type a different name, if required. The exported project is packaged as a Vitesse Project Container file using the *.vpc extension.
- 3 In the **Export to** entry box, specify the required path for exporting the file. You can also click  to display the *Browse for Folder* dialog and select the required path.
- 4 Select the resources (used in the project) to be exported from the displayed list. You can also click the **Select All** button to select all of the resources.



Select All is a toggle button. It also enables you to deselect all of the resources by displaying the **Unselect All** toggle button.

- 5 When exporting a project, specify the required export output from the following:
 - **Export as VPC zip container** – exports the Vitesse project design (including the resources used) to the equivalent VIPP formats and packages them into a single VIPP Project Container file (*.vpc).
 - **Create “*client folder* \ *project folder*” with all resources** – exports the Vitesse project design (including the resources used) to the equivalent VIPP formats and saves them in the *client folder* \ *project folder* specified in **Step 3**. These options are not available when exporting stand-alone resources.



In the **Create “*client folder* \ *project folder*” with all resources** option:

- ***client folder*** – specifies the name of the target client folder where the project is being exported.
- ***project folder*** – specifies the name of the target project folder where the project is being exported

- 6 From the **Overwrite** area, select any one of the following:
 - **Prompt** – prompts you to confirm overwriting any existing files (resource and/or project) with the same name.
 - **Overwrite** – overwrites the existing files with the same name without prompting.
 - **Skip** – skips the export of files already existing with the same name.



See the “Managing Unmapped Fonts” section below for information on the **Export Fonts** functionality.

- 7 Click **Export**.

Managing Unmapped Fonts during Export

We recommend mapping the TrueType fonts used in a design to PostScript fonts, before exporting the design. However, if fonts are not mapped, Vitesse also enables you to convert these fonts to the PostScript format at the time of exporting a project. These PostScript fonts and the ones applied to objects in a design are then packaged as separate resource files with the exported project.

Converting True Type Fonts to the PostScript Format

To convert the True Type fonts to the PostScript format and then package the PostScript fonts:

- 1 Click the **Export Font** button in the *Export* dialog.

The *Export Fonts* dialog displays with a list of the unmapped True Type fonts used in the design. The list will be empty if unmapped True Type fonts are not used in the design.
- 2 Select the required font from the list.
- 3 Click **Next** to convert the selected fonts.

The converted True Type fonts are displayed as a list of PostScript fonts and ready for packaging.



The PostScript font files associated with an exported project are placed in a **fontlib** directory. **fontlib** is created whenever required, at the user-specified location for export.

Packaging the PostScript Fonts into the VPC

- 1 Select the required PostScript fonts from the list.
- 2 Click **OK** to package the selected PostScript font files into the VPC.

Chapter 36: Vitesse Administrative Options

This chapter describes activities and tasks for opening Vitesse in administration and user mode. It also provides information about operating Vitesse in administration mode. Included are instructions and information for the following:

- Vitesse User Types
- Opening Vitesse in Administration Mode
- Setting Administrator Preferences
- Setting up the Working Folder
- Limiting Designer Usage
- Setting Default Vitesse Mode
- Setting up Predefined Media
- Limiting Resource Insertion
- Enabling Field Insertion in Forms and Segments
- Setting VIPP Project Import Options
- Opening Vitesse in User Mode

Vitesse User Types

Vitesse enables you to control the behavior of the application for different users through the following two user modes.

- Administration
- User

Each user has a different set of permissions and supported functionality to operate Vitesse.

Administrator

The Vitesse administrator is the person who possesses the rights to use the complete Vitesse functionality without any restrictions. The administrator can create, edit, and modify projects and resources in Vitesse. Permissions to set or change preferences to provide limited functionality to the Vitesse User are also allowed to the administrator.

All the workspace options are enabled in the administration mode.

User

The Vitesse user is the person who possesses limited rights for using Vitesse. The user rights are set by the Vitesse administrator in the administration mode using the application preferences.

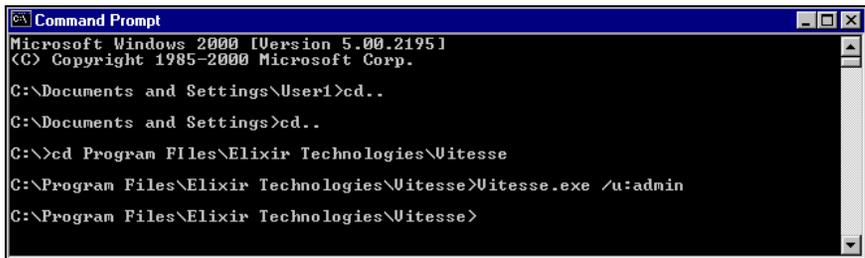
The Vitesse user is provided with a customized Vitesse workspace, depending on the preferences set by the administrator.

Opening Vitesse in Administration Mode

To open Vitesse in the administration mode:

- 1 Open the Windows *Command Prompt*.
- 2 In the *Command Prompt* window, use the DOS commands to set the Vitesse installation directory.
- 3 Specify the following syntax at the command prompt: *Vitesse.exe /u:admin*.
- 4 Press <Enter> on the keyboard.

Vitesse opens in the administration mode.



```
Command Prompt
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\User1>cd..
C:\Documents and Settings>cd..
C:\>cd Program Files\Elixir Technologies\Vitesse
C:\Program Files\Elixir Technologies\Vitesse>Vitesse.exe /u:admin
C:\Program Files\Elixir Technologies\Vitesse>
```

The image shows the process of opening Vitesse in administration mode using the Command Prompt.

You can also specify an existing project, form, segment, or Vitesse zip file that you want to open in Vitesse at the command prompt.

Opening a File in Vitesse Administration Mode

To open Vitesse in administration mode with the specified file:

- 1 Open the Windows *Command Prompt*.
- 2 In the *Command Prompt* window, use the DOS commands to set the Vitesse installation directory.
- 3 Specify the following syntax at the command prompt: *Vitesse.exe /u:admin "resource path and extension"*.

- 4 Press <Enter> on the keyboard.

Vitesse opens the selected file in the administration mode.



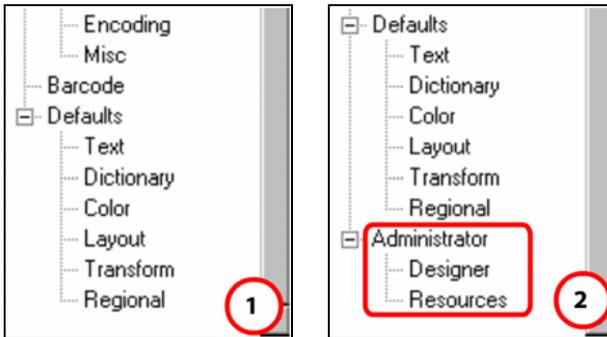
```
Microsoft Windows [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\User1>cd..
C:\Documents and Settings>cd..
C:\>cd Program Files\Elixir Technologies\Vitesse
C:\Program Files\Elixir Technologies\Vitesse>vitesse.exe /u:admin "FormLib\TrnAu
roraLM.evf"
C:\Program Files\Elixir Technologies\Vitesse>_
```

The image shows the process opening Vitesse in administration mode, with the specified file.

Setting Administrator Preferences

The Vitesse administrator can customize the features available for Vitesse user mode. These options are set using the **Administrator** category in the *Preferences* dialog; this category is not available in the user mode.



(1) User mode Preferences dialog (2) Administration mode Preferences dialog.

Setting up the Working Folder

A working folder is the directory where all Vitesse client folders are created. **Clients** folder is the default working directory for Vitesse. As an administrator, you can change the default working directory to create projects at a different location.



For more information on working directory, see *Chapter 3: Understanding the Vitesse Directory Structure*.

To set up the working folder:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **General** category to view the associated options.
- 3 Specify the required directory path in the **Working Folder** entry box. You can also click  adjacent to the **Working Folder** entry box to open the *Browse for Folder* dialog to specify the location.
- 4 In the *Browse for Folder* dialog, browse and select the required directory.
- 5 Click **OK** to return to the *Preferences* dialog.
The full path of the specified directory displays in the **Working Folder** entry box.
- 6 Click **OK**.
- 7 Projects will now be created at the specified working directory.



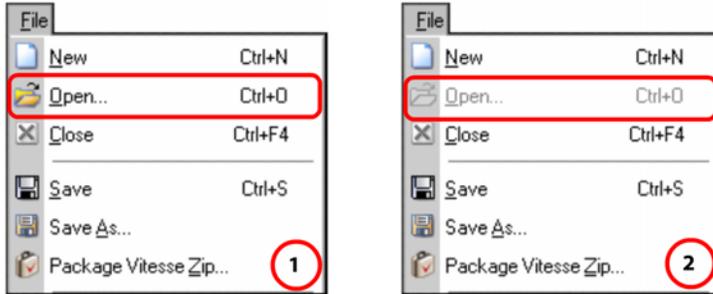
You can also open an existing project from any other location on the system besides the working folder. Selecting File>Save from the menu bar saves the existing project in the specified working folder.

Limiting Vitesse Functions

The Vitesse administrator can restrict functions such as creating, opening, saving, importing, or exporting projects for the Vitesse user.

To set restrictions on using the application functions for the user mode:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Designer** category under the **Administrator** node.
- 3 From the **Disable** area, select the required options from the following:
 - **Creating New Project** – restricts the Vitesse user from creating a new project, form, or segment while working in the user mode. Selecting this option disables the **New** option in the user mode workspace.
 - **Opening existing project** – restricts the Vitesse user from opening an existing project, form, or segment while working in the Vitesse user mode. Selecting this option disables the **Open** option in the user mode workspace.



The user mode workspace with (1) Open option enabled (2) Open option disabled.

- **Saving current project with new name** – restricts the Vitesse user from saving the current project, form, or segment with a new name or at a location other than the current location while working in the Vitesse user mode. Selecting this option disables the **Save As** option in the user mode workspace.
- **Packaging vitesse zip file** – restricts the Vitesse user from saving the current project, form, or segment in Vitesse zip format while working in the Vitesse user mode. Selecting this option disables the **Package Vitesse Zip** option in the user mode workspace.
- **Import** – restricts the Vitesse user from importing project and resources while working in the Vitesse user mode. Selecting this option disables the **Import** option in the user mode workspace.
- **Viewing recent files** – restricts the Vitesse user from viewing the recently used files in the File menu. The *New Project* dialog **Recent** tab is also removed from the user mode workspace.

4 Click OK.

The settings take effect the next time Vitesse is opened in user mode. These options have no effect on the administration mode workspace.

Setting Default Vitesse Mode

The Vitesse administrator can specify the default mode in which to open Vitesse. You can open Vitesse in the default mode, without specifying the operational mode (user or administration) at the time of opening either by using:

- The Windows Start menu.
- The Command Prompt without specifying the parameters “/u:admin” or “/u:user”.

To specify the Vitesse default operational mode:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Designer** category under the **Administrator** node.
- 3 From the **Default Vitesse Mode** area, select one of the following options:
 - **Administrator** – sets administration mode as the default Vitesse operational mode.
 - **User** – sets user mode as the default Vitesse operational mode.
- 4 Click **OK**.

The settings take effect the next time Vitesse is opened.



If the command line parameters “/u:admin” or “/u:user” are specified when opening Vitesse, then Vitesse opens in the corresponding mode irrespective of the default mode set using the *Preferences* dialog in the administration mode.



The **Administrator** category options in the *Preferences* dialog used for setting up the administration options are only available if Vitesse is opened using the “/u:admin” command line parameter. If the administration mode is set as the default operational mode and Vitesse is opened without specifying “/u:admin”, the application opens with the complete administration mode functionality other than the administrator *Preferences* dialog options.

Setting Permissions for the Export Folder

By default, a Vitesse user can save exported files at any location. However, the Vitesse administrator can set the target folder for the exported files. The user can only save the exported files in the specified target folder if the option has been set in the administration mode *Preferences* dialog.

To set the target export folder:

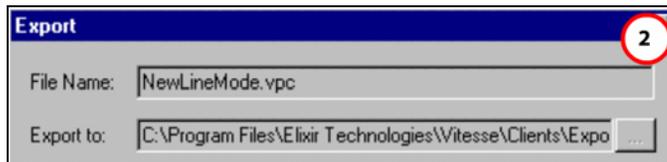
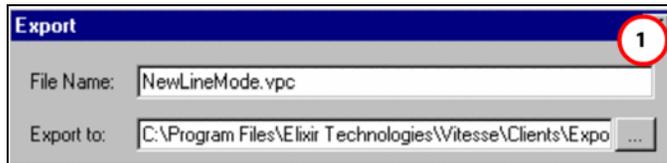
- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Designer** category under the **Administrator** node.
- 3 Select the **Disable editing of export folder** option; this enables the **Export folder** entry box and the associated browse button.

- 4 Click  to open the *Browse for Folder* dialog.
- 5 From the *Browse for Folder* dialog, select the required folder from the **Select Export Folder** list.
- 6 Click **OK** to return to the *Preferences* dialog.

The selected folder path is displayed in the **Export folder** entry box. You can also type the required export folder path in the **Export folder** entry box.

- 7 Click **OK**.

The **File Name** and **Export to** entry boxes in the *Export* dialog are disabled in the Vitesse user mode. The settings take effect the next time Vitesse is opened in user mode. These options have no effect on the administration mode workspace.



The Export dialog in the user mode workspace with (1) Disable editing of export folder option off (2) Disable editing of export folder option on.

Setting up Predefined Media

Vitesse provides the feature for using pre-defined media (paper) settings. These options are set in the administration mode. Using predefined settings reduces the time that is required for setting paper properties every time you create a project. As an administrator, you can create an XML file, in which you define the number of paper objects to be used in a design, and list the required media settings, pertaining to each paper object. You can also restrict the type of paper to be used in projects. Following image depicts the structure of an xml file containing the paper (media) settings:

```

<?xml version="1.0" ?>
- <VitesseMediaList Version="1.0">
  - <Paper>
    <Name>Tray5</Name>
    <Category>Standard</Category>
    <Standard>Letter</Standard>
    <Units>Centimeters</Units>
    <Height>0.0</Height>
    <Width>0.0</Width>
    <Orientation>Landscape</Orientation>
    <Type>Drilled</Type>
    <Weight>3.45</Weight>
    <Color>BLACK</Color>
    <IncludePaperDimensions>No</IncludePaperDimensions>
  </Paper>
</VitesseMediaList>

```

The `<Paper></Paper>` tag pair defines a paper object; all the sub tag nodes define the properties of the paper.

The XML file can contain any number of paper definitions, as required in the project design. This file can also be updated anytime; this action does not affect the paper objects already inserted in a project.

To setup the predefined media:

- 1 Using any text editor, create an xml file specifying all the required media options such as paper name, category, standard, dimensions, orientation, type, weight, and color as identified by their respective tags in the XML file. You can also choose to generate Document Structuring Conventions (DSC) %%DocumentMedia comments using the Include Paper Dimensions tag.



DSC comments are generated in a VIPP submission file (*.sub), which is created only on exporting a project. Printers process the DSC comments in this file to determine the document media requirements.

- 2 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 3 From the *Preferences* dialog, select the **Designer** category under the **Administrator** node.
- 4 Select the **Allow only predefined media** option; this enables the **Media file path** entry box and the associated browse button.
- 5 Click  to open the *Open* dialog.

6 In the *Open* dialog, browse and select the required XML file.

7 Click **OK** to return to the *Preferences* dialog.

The selected file path is displayed in the **Media file path** entry box. You can also type the required XML file path in the **Media file path** entry box.

8 Click **OK**.

The behavior of Vitesse for inserting papers is modified for both the user and the administration mode.



For more information on inserting predefined papers, see “Adding a Predefined Paper” in *Chapter 17: Defining Page and Print Setup*.

Limiting Resource Insertion

The Vitesse administrator can restrict the user from inserting form, segment, and image resources to a project in user mode.

To limit options for inserting resources in the user mode:

1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.

2 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.

3 From the **Disable Inserting** area, select the required options from the following:

- **Image** – restricts the Vitesse user from inserting a new image resource into a project or resource while working in Vitesse user mode; selecting this option enables the **Data driven resources** option.
- **Form** – restricts the Vitesse user from inserting a linked or existing form resource into a project; and a new or existing form to the project resources, while working in Vitesse user mode; selecting this option enables the **Data driven resources** option.
- **Segment** – restricts the Vitesse user from inserting a linked or existing segment resource into a project; and a new or existing segment to the project segment resources, while working in Vitesse user mode; selecting this option enables the **Data driven resources** option.
- **Data driven resources** – restricts the Vitesse user from inserting data driven forms, segments, and images (using variables and fields) into a project while working in Vitesse user mode.
- **Data file** – restricts the Vitesse user from inserting a new data file into a project while working in Vitesse user mode.



The Insert toolbar in the user mode workspace with (1) Disable Inserting Data file option off (2) Disable Inserting Data file option on.

4 Click **OK**.

The settings take effect the next time Vitesse is opened in user mode. These options have no effect on the administration mode workspace.

Setting Resource Loading Options

By default, the form, segment, and image resources (with local, shared, or global scope) used in a project are available under the project resources (**Resources>Resource name** tree node) when a project is opened. However, the Vitesse administrator can set options to make all the resources (in the local, shared, or global resource folders) available under the project resources, even if they are not being used in a project.

To set resource loading options:

1 Select **Tools>Preferences** from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

2 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.

3 From the **Load Resources** area, select the required options from the following:

- **Local** – loads all the local resources for a project; selecting this option enables the **Don't load PS/EPS segments** option.
- **Shared** – loads all the resources from the project shared directory to the project resources even if the shared resources are not being used for the current project; selecting this option enables the **Don't load PS/EPS segments** option.

- **Global** – loads all the resources from the global form, image, and segment directories to the project resources even if the global resources are not being used for the current project; selecting this option enables the **Don't load PS/EPS segments** option.
 - **Don't load PS/EPS segments** – does not load PS\EPS segments from the local, shared, or global resource folders.
- 4 Click **OK**.

The behavior of Vitesse for loading resources is modified for both the user and the administration mode.



For more information on local, shared, and global resource directories, see *Chapter 3: Understanding the Vitesse Directory Structure*.

Limiting Resource Deletion

The Vitesse administrator can restrict the user from deleting form, segment, and image resources from a project in user mode.

To limit options for deleting resources in the user mode:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.



You can also open the *Preferences* dialog either by:

- Clicking  on the *Standard* toolbar.
- Or
- Pressing <Ctrl> + <K> on the keyboard.

- 2 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.
- 3 From the **Disable Removing** area, select the required options from the following:
 - **Image** – restricts the Vitesse user from removing image resources from a project while working in Vitesse user mode.
 - **Form** – restricts the Vitesse user from removing form resources from a project while working in Vitesse user mode.
 - **Segment** – restricts the Vitesse user from removing segment resources from a project while working in Vitesse user mode.
- 4 Click **OK**.

The settings take effect the next time Vitesse is opened in user mode. These options have no effect on the administration mode workspace.

Enabling Field Insertion in Forms and Segments

Vitesse enables you to add variables and fields in forms and segments using a predefined list (in Database and XML modes). As an administrator, you can create an XML file, in which you define the variables and fields to be used in a form or segment. Following image depicts the structure of an xml file containing the variables list for a database mode project:

```
<?xml version="1.0" ?>
- <VitesseVariableList Version="1.0">
  - <CustomerInfo>
    - <Variable>
      <Name>NAME</Name>
      <Type>Field</Type>
      <Mode>Database</Mode>
    </Variable>
    - <Variable>
      <Name>FNAME</Name>
      <Type>Field</Type>
      <Mode>Database</Mode>
    </Variable>
    - <Variable>
      <Name>VARnumber</Name>
      <Type>Variable</Type>
      <Mode>Database</Mode>
    </Variable>
```

Native and Line mode projects cannot use forms and segments that contain predefined fields. The predefined XML file which is to be used in native or line mode projects can only contain variables for inserting in forms and segments. Also, a single XML file cannot contain both database and xml mode fields. Following image depicts the structure of an xml file containing the variables list for an XML mode project:

```
<?xml version="1.0" ?>
- <VitesseVariableList Version="1.0">
  - <CustomerInfo>
    - <Variable>
      <Name>^Customer^Id</Name>
      <Type>Element</Type>
      <Mode>XML</Mode>
    </Variable>
    - <Variable>
      <Name>^Customer^@Name</Name>
      <Type>Attribute</Type>
      <Mode>XML</Mode>
    </Variable>
    - <Variable>
      <Name>VARname</Name>
      <Type>Variable</Type>
      <Mode>XML</Mode>
    </Variable>
```

The following table shows the valid values for each tag in the XML file:

Tag Name	Description
Name	Specifies the name of the Field , Tag , Attribute or Variable . No spaces or special characters are allowed. Variable Names must begin with VAR . Each name must be unique.
Type	Specifies the type of the variable, either: Element : a complete path for the XML tags must be specified. Field : name of the field is specified. Attribute : name of the attribute must contain @, as per VIPP rules.
Mode	Specifies the project mode, either Database or XML mode.

The XML file can contain any number of variables and fields, as required in the form or segment design. Whenever a form or segment is opened in Vitesse, loading the XML file with a list of variables and fields helps to merge data with text. These variables and fields become available in the **Merge data** drop-down in the *Merge Data* toolbar, thus making it possible to merge these variables with the text in a form or segment.

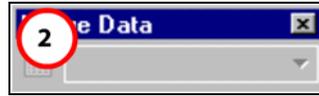
To enable field and variable insertion in forms and segments:

- 1 Using any text editor, create an xml file specifying all the required variable and field properties such as name, type, and mode as identified by their respective tags in the XML file.
- 2 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 3 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.
- 4 Select the **Allow field insertion in Form and Segment** option; this enables the **Fields file path** entry box and the associated browse button.
- 5 Click  to open the *Open* dialog.
- 6 In the *Open* dialog, browse and select the required XML file.
- 7 Click **OK** to return to the *Preferences* dialog.

The selected file path is displayed in the **Fields file path** entry box. You can also type the required XML file path in the **Fields file path** entry box.

- 8 Click **OK**.

The fields and variables defined in the XML file are displayed in the *Merge Data* toolbar drop-down list when designing forms and segments. This behavior of Vitesse is modified for both the user and the administration mode.



The Vitesse workspace with (1) *Allow field insertion in Form and Segment* option on (2) *Allow field insertion in Form and Segment* option off.



For more information on inserting predefined fields and variables, see *Chapter 9: Using the Text Tools*.

Setting VIPP Projects Import Options

When you import a VIPP project in the Vitesse environment, the resources referenced within the VIPP project are also imported. However, the VIPP source code includes only the resource name(s) of the referenced resource(s) and does not provide the full path where the resource is located. Therefore, prior to importing a VIPP project in Vitesse, you need to define a list of directories/folders for the required resources so that Vitesse locates the resources referenced in the imported project at the specified paths instead.

The **Folder** category in *Preferences* dialog provides the options for specifying the directories. This category consists of the different resource types that may have been used in an imported VIPP project. The resource types are identified by their respective sub categories under the **Folder** node in the *Preferences* dialog.

The Vitesse user can add new folders, delete existing folders, and reorder the folders for each of the resource type, by default. However, the Vitesse administrator can restrict the user from editing the resource look up folders.

To restrict the user from modifying the look up folders:

- 1 Select Tools>Preferences from the menu bar to open the *Preferences* dialog.
- 2 From the *Preferences* dialog, select the **Resources** category under the **Administrator** node.
- 3 Select the **Disable editing of lookup folders for resources in preferences** option.
- 4 Click **OK**.

All the sub categories under the **Folder** node in the *Preferences* dialog are disabled for the user mode workspace. The settings take effect the next time Vitesse is opened in user mode.

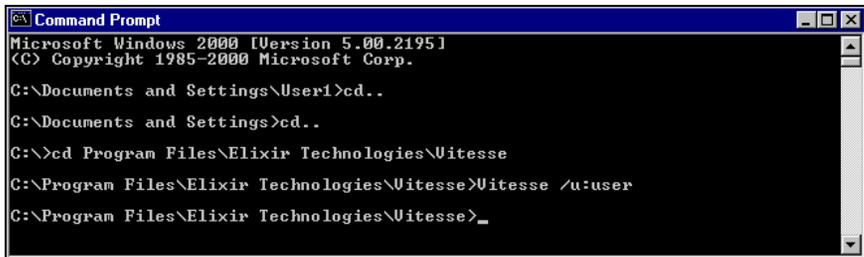
Opening Vitesse in User Mode

User mode provides a customized Vitesse workspace, depending on the preferences set by the administrator. The administrator can restrict options such as creating, opening, saving, or exporting projects in Vitesse user mode. The administrator can also setup rights for each project resource by setting preferences in the administrator mode. For example, restrictions can be set for loading form resources in a project.

To open Vitesse in the user mode:

- 1 Open the Windows *Command Prompt*.
- 2 In the *Command Prompt* window, use the DOS commands to set the Vitesse installation directory.
- 3 Specify the following syntax at the command prompt: *Vitesse.exe /u:user*.
- 4 Press <Enter> on the keyboard.

Vitesse opens in the user mode.



```
Command Prompt
Microsoft Windows [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\User1>cd..
C:\Documents and Settings>cd..
C:\>cd Program Files\Elixir Technologies\Vitesse
C:\Program Files\Elixir Technologies\Vitesse>Vitesse /u:user
C:\Program Files\Elixir Technologies\Vitesse>_
```

The image shows the process of opening Vitesse in user mode using the Command Prompt.

You can also specify an existing project, form, segment, or Vitesse zip file that you want to open in Vitesse, at the command prompt.

Opening a File in Vitesse Administration Mode

To open Vitesse in user mode with the specified file:

- 1 Open the Windows *Command Prompt*.
- 2 In the *Command Prompt* window, use the DOS commands to set the Vitesse installation directory.
- 3 Specify the following syntax at the command prompt: *Vitesse.exe /u:user "resource path and extension"*.
- 4 Press <Enter> on the keyboard.

Vitesse opens the selected file in the user mode.

```
Command Prompt
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\User1>cd..
C:\Documents and Settings>cd..
C:\>cd Program Files\Elixir Technologies\Vitesse
C:\Program Files\Elixir Technologies\Vitesse>Vitesse.exe /u:user "Clients\NewLineMode\NewLineMode.evd"
C:\Program Files\Elixir Technologies\Vitesse>_
```

The image shows the process opening Vitesse in user mode, with the specified file.



Vitesse user cannot open an existing project or resource from the command prompt if the **Opening existing project** option has been selected in the Vitesse administration mode.

Appendix A: Error and Warning Message Handling

This appendix lists and explains error and warning messages that can be encountered while performing specified Vitesse operations, along with explanations for each occurrence. Also included are messages and handling routines for VIPP source code errors. Information includes:

- Source Code Error Handling
- Error Messages
- Warning Messages

Source Code Error Handling

While compiling the application source code, error, warning, and/or informational messages may display in the *Message View* window:

Error Messages

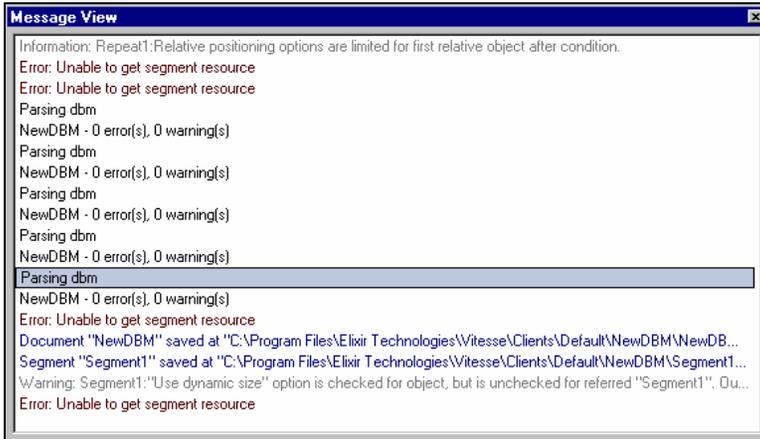
Error messages typically result from incorrect VIPP language implementation or missing source elements. Error messages are displayed in brown. You should not switch from **Source** view to **Design** view until all errors are corrected, otherwise, you may lose your changes. To resolve a compilation error, double-click an error message displayed in the *Message View* window. Source code in error will be automatically highlighted in the **Source** view; make the necessary changes to the source code, and then recompile by clicking  in the **Source** view.

Warning Messages

Warning messages typically result from illogical commands (for example, invalid variable names). Warning messages are displayed in gray and indicate non-critical source code irregularities. Ignoring warning messages will not result in loss of source code changes.

Informational Messages

Informational messages are displayed in gray. Other messages indicating the results after compilation are generated in blue.



Message View window displaying error, warning, and informational messages.

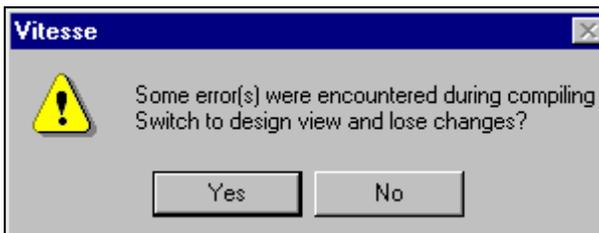
Viewing Source Code for Entire Application or External Resources

To view the source code for an entire application or for external resources used in an application:

- 1 Switch from **Design** view to **Source** view in the design area.
- 2 Click in the upper left corner of the **Source** view to view the drop-down list options.
- 3 Select either the *project name* or required external *resource node name* used in the project for viewing the application or resource code.

Handling Compilation Errors

When compilation errors occur, the *Message View* window displays the error. If you switch to the **Design View** at this stage, the following message displays:



Clicking **Yes** causes the source code to revert to the last valid compiled state and returns you to the design area; clicking **No** enables you to continue editing the source code.

Reverting to the Last Valid Compiled State

If errors cause the edited source code compilation to fail, you can restore the source code to the last valid compilation state as follows:

- Select View>Revert to Last State from the menu bar.
Or
- Click  in the upper left corner of the **Source** view.
Or
- Right-click in the **Source** view, then select **Revert to Last State** from the pop-up menu.

Error Messages

Import Error Messages

Error Message	Explanation
Parameter <Parameter_Name> is missing in command <Command_Name>.	This error implies that the expected parameter <Parameter_Name> is missing in the command <Command_Name>.
Parameter <Parameter_Name> must be <Parameter_Type>.	This error implies that the parameter <Parameter_Name> is invalid. Rectification: The parameter must be <Parameter_Type>.
Value of Parameter <Parameter_Name> is out of range <Range_Value>.	This error implies that the parameter <Parameter_Name> value is out of range. Rectification: The parameter value must be within <Range_Value>.
File <File_Name> not found.	This error implies that Vitesse cannot find the specified file <File_Name>.
Path <File_Path> does not exist.	This error implies that the specified file path <File_Path> is invalid.
Invalid file format.	This error implies that either the file format is not valid or the file does not contain a valid VIPP code.

Error Message	Explanation
Variable <Variable_Name> is undefined.	This error implies that the variable <Variable_Name> is not defined for the current application.
Font not found.	This error implies that the specified font is not found in the Vitesse font map.
Expression <Boolean_Expression> is not defined.	This error implies that the conditional expression <Boolean_Expression> is not defined for the current application.
Unexpected end of file.	This error implies that an unexpected end of file is encountered due to the missing symbols <},),>,]>.
Block syntax error.	This error implies that the expected block command <SETFORM or SCALL, or BCALL or FCALL or FSHOW or XGFRESDEF> is missing from the code.
Unable to set BACK form.	This error implies that the duplex printing command is not allowed with simplex paper.
<BAT KEY/ GEP KEY> missing in command.	This error implies that the expected text background attribute (BATKEY) or graphic attribute (GEPKEY) is missing from the command.
<Command_Name> is not allowed in <Vitesse Document Mode>.	This error implies that the command <Command_Name> is not supported by the current Vitesse project mode <Document Mode_Name>.
Source Error.	This error implies that project source code irregularities are encountered.
Unable to get segment resource.	This error implies that Vitesse cannot find the specified segment resource.

Warning Messages

Import Warning Messages

Warning Message	Explanation
Begin page should come before any marking command on page.	This warning implies that the Begin page command should occur before any marking command on the current page.
<Command Name> not supported in <Vitesse>.	This warning implies that the command <Command Name> is not supported by Vitesse in a Vitesse project mode. For example, CASE is not supported in FORM mode (forms).
<Command_Name> not supported.	This warning implies that the command <Command_Name> is not supported by Vitesse.
<Variable_Name> replaced in <Command_Name>.	This warning implies that the variable <Variable_Name> is replaced in the command <Command_Name>.
Missing parameter <Parameter_Name> in special comments.	This warning implies that a parameter <Parameter_Name> is missing from the Elixir comments.
Unexpected symbol <Symbol>.	This warning implies that an unexpected symbol <> is encountered in the code.
Unable to get NAME of the resource node.	This warning implies that Vitesse cannot locate a resource in the project tree resource while converting the current application.
Unsupported parameters were used with graphic text.	This warning implies that unsupported options are used with graphic text objects.

Export Warning Messages

Warning Message	Explanation
<Node> “Auto Size on, Conversion may not match.”	This warning implies that the Auto size feature may not be correctly translated when exporting projects. Rectification: Switch off the Auto size option before exporting projects.
<Node> “Graphics text is converted with multiple commands, with rotation applied output may not match.”	This warning implies that the output rotation of the graphic text object with multiple commands may not be translated correctly. Rectification: Use the same settings for all elements in the graphic text object.
<Node> “Invalid Field name, Missing '^.’”	This warning implies that a missing character ‘^’ is found in the XML field data settings.
<Node> “Text is converted with multiple commands, with Data merging output may not match.”	This warning implies that text object with multiple formatting may not be translated correctly when used with the Merge data command. Rectification: Use the same settings for all elements in text object.
“Unable to open file, <File_Name>.”	This warning implies that Vitesse cannot find the specified file. Rectification: Close and restart the application.
<Node> Chart's Height Negative.	This warning implies that the chart height cannot be a negative value. Rectification: Specify a positive value.
<Node> Condition Test missing.	This warning implies that the condition does not contain a test statement. Rectification: Specify a test statement.

Warning Message	Explanation
Constant field name missing in <Node> field node	This warning implies that data is missing for the constant field. Rectification: Add field data.
Data field name missing in <Node> field node.	This warning implies that data is missing for the data field. Rectification: Add field data.
<Node> Data Missing In chart.	This warning implies that data is missing in charts. Rectification: Specify data for the chart object.
<Node> DDG Chart should have variable mode of fields positioning.	This warning implies that records in which charts are used should have variable spacing setting set to Variable .
<Node> Duplicate record. Converted Code may not be correct.	This warning implies that RPE Key is mapped to multiple records for which the converted code may not be correct.
Field horizontal displacement is being ignored-Output will not match.	This warning implies that field horizontal displacement is not allowed in Line mode projects.
Field type missing in <Node> field node.	This warning implies that the field type value is not found. Rectification: Reinsert the field type value.
<Node> Font does not Exist Using Default Font.	This warning implies that the specified font is not found in the Vitesse font map. Rectification: Specified font is replaced with the default font.
<Node> Graphic text object in linked frame. Conversion may not match.	This warning implies that graphic text object is not allowed in linked frame.
Invalid data string encountered in	This warning implies that an invalid data string is encountered while

Warning Message	Explanation
<Command> conversion.	converting the current application.
Invalid data switch. Using // as default.	This warning implies that data switch is not specified in preferences.
<Node> Invalid-Hold Operator In Numeric Comparison.	This warning implies that an invalid operator is used for numeric comparison. Rectification: Specify the correct operator.
JDT missing in the project.	This warning implies that Vitesse cannot find the logical page for export operation.
Multiple pages are in DBM mode; only paper <NODE> will be used for conversion.	This warning implies that you cannot use more than one paper with NUP settings.
<Node> No Marking Command On Page.	This warning implies that Vitesse cannot locate a printable object on a page while converting the current application.
<Node> No Vertical Alignment Option Selected.	This warning implies that field vertical alignment is not set.
NOT Operator not allowed in PCD Test.	This warning implies that you cannot use the NOT operator for page level conditions inserted at the record level.
<Node> Pages can only be switched conditionally.	This warning implies that the Set Page action must be used in a condition.
<Node> Pages can only be switched conditionally on first Page.	This warning implies that the Set Page action can only be used on the first page.
<Node> Record tag name is missing.	This warning implies that the record tag is not specified.
<Node> Test missing in condition.	This warning implies that condition test statement is not specified for IF/ELSE IF nodes in the project tree.

Warning Message	Explanation
<Node> Text justification may not match with the Text justification of VIPP.	This warning implies that the applied text justification may not match the VIPP text justification.
<Node> Text object width is 0.	This warning implies that the text object width is set to 0. Rectification: Recommended text width must be greater than 0.
<Node> The type of transform applied on Rich string may not produce correct result.	This warning implies that the specified transform type is not compatible with rich string.
<Node> Transform(s) applied on DDG Field.	This warning implies that you cannot use transform(s) on field that is used in DDG index in Line mode projects.
<Node> Transforms Applied on Image or Segment Type Field.	This warning implies that the specified transform is not allowed for Image or Segment type fields.
<Node> Unable to find <Command> drawing object.	This warning implies that Vitesse cannot find a drawing object while converting the current application.
<Node> Unable to find condition node.	This warning implies that Vitesse cannot find the condition node while converting the current application.
<Node> Unable to find condition object.	This warning implies that Vitesse cannot find the condition object while converting the current application.
<Node> Unable to find text node.	This warning implies that Vitesse cannot find the text node while converting the current application.
<Node> Unable to find text object.	This warning implies that Vitesse cannot find the text object while converting the current application.
Unable to get <Command> resource.	This warning implies that Vitesse cannot find the required resource while converting the current application.

Warning Message	Explanation
<Node> Unable to get Action Object.	This warning implies that Vitesse cannot find the drawing Action object while converting the current application.
<Node> Unsupported parameters were used with graphic text.	This warning implies that specified graphic text parameters are not allowed.
<Node> Number of lines of text is different after applying data merging.	This warning implies text formatting may change as a result of merging variable data with text.
<Node> Enable frame overflow option is checked for object, but is unchecked for referred segment. Output may not match.	This warning implies that if you select the Enable frame overflow option for a segment resource, refer the segment on a page, and then insert a frame in the referenced segment, you need to set the Enable frame overflow option in the referenced segment as well; otherwise, the output may not match while exporting projects.
<Node> Use dynamic size option is checked for object, but is unchecked for referred segment. Output may not match.	This warning implies that if the Use dynamic size option is checked for a segment resource and the segment is referred on a page, you need to set the Use dynamic size option in the referenced segment as well; otherwise, the output may not match while exporting projects.
<Node> Image cannot be the last child of dynamically sizeable segment.	This warning implies that an image used in a dynamically sizeable referenced segment should not be the last relatively positioned object in the segment, as Vitesse cannot determine the image size at run time.
<Node> Overlapping records used; output may not match:	This warning implies that conversion may not match if overlapping records are used in Line mode projects.
<Node> Only forms, conditions, actions, records,	This warning implies that in XML mode, only forms, conditions,

Warning Message	Explanation
and end-tag segment objects are allowed on page; otherwise, conversion may not match.	actions, records, and end-tag segments are allowed on a page; insertion of all other objects on page is allowed under the records.

Text Warning Messages

Warning Message	Explanation
In the Paragraph Text combination of Batkey and TAB is not allowed.	This warning implies that you cannot use a combination of BAT key and Tab option in Paragraph text.
In the Paragraph Text combination of Transform and TAB is not allowed.	This warning implies that you cannot use a combination of Transform and Tab option in Paragraph text.
In the Paragraph Text combination of Batkey and Word justification is not allowed.	This warning implies that you cannot use a combination of BAT key and Word Justification option in Paragraph text.
In the Paragraph Text combination of Transform and Word justification is not allowed.	This warning implies that you cannot use a combination of Transform and Word Justification option in Paragraph text.
In the Paragraph Text combination of Batkey and Character justification is not allowed.	This warning implies that you cannot use a combination of BAT key and Character Justification option in Paragraph text.
In the Paragraph Text combination of Transform and Character justification is not allowed.	This warning implies that you cannot use a combination of Transform and Character Justification option in Paragraph text.
In the Paragraph Text combination of Batkey, Right Alignment and First line indentation is not allowed.	This warning implies that you cannot use combinations of BAT key, Right Alignment , and First Line Indentation options in Paragraph text.
In the Paragraph Text combination of Transform, Right Alignment and First line indentation is not	This warning implies that you cannot use combinations of Transform , Right Alignment , and First Line Indentation options in Paragraph

Warning Message	Explanation
allowed.	text.
In the Paragraph Text combination of Batkey, Center Alignment and First line indentation is not allowed.	This warning implies that you cannot use combinations of BAT key, Center Alignment , and First Line Indentation options in Paragraph text.
In the Paragraph Text combination of Transform, Center Alignment and First line indentation is not allowed.	This warning implies that you cannot use combinations of Transform , Center Alignment , and First Line Indentation options in Paragraph text.
In the Paragraph Text word justification and TAB combination is not allowed.	This warning implies that you cannot use a combination of Word Justification and Tab option in Paragraph text.
In the Paragraph Text character justification and TAB combination is not allowed.	This warning implies that you cannot use a combination of Character Justification and Tab option in Paragraph text.
In the Paragraph Text Paragraph justification and TAB combination is not allowed.	This warning implies that you cannot use a combination of Paragraph and Tab options in Paragraph text.
In the Paragraph Text Batkey and different font size combination is not allowed.	This warning implies that you cannot use a combination of BAT key and different font Size options in Paragraph text.
In the Paragraph Text Transform and different font size combination is not allowed.	This warning implies that you cannot use a combination of Transform and different font Size options in Paragraph text.
In the Paragraph Text TAB and Data Merging combination is not allowed.	This warning implies that you cannot use a combination of Tab and Merge Data options in Paragraph text.
In the Paragraph Text different font size and Data Merging combination might have slight wrapping differences.	This warning implies that a combination of Merge Data and different font Size options in Paragraph text may cause wrapping differences.

Warning Message	Explanation
In the Normal Text combination of TAB and Word justification is not allowed.	This warning implies that you cannot use a combination of Tab and Word Justification options in Normal text.
In the Normal Text combination of TAB and Character justification is not allowed.	This warning implies that you cannot use a combination of Tab and Character Justification options in Normal text.
In the Normal Text combination of TAB and Paragraph justification is not allowed.	This warning implies that you cannot use a combination of Tab and Paragraph Justification options in Normal text.
In the Normal Text combination of BatKey and TAB is not allowed.	This warning implies that you cannot use a combination of BAT key and Tab options in Normal text.
In the Normal Text combination of Transform and TAB is not allowed.	This warning implies that you cannot use a combination of Transform and Tab options in Normal text.
In the Fancy Text TAB is not allowed.	This warning implies that you cannot use the Tab option in Fancy text.
COLW Batkeys should be applied to the whole paragraph with the same color and either the paragraph is left aligned or each line's width equal to the object width.	<p>This warning implies that:</p> <ul style="list-style-type: none"> ▪ Paragraph is either left aligned or each line width is equal to the text object width. <p>And</p> <ul style="list-style-type: none"> ▪ Current BAT key uses COLW (Column Width) system variable and should be applied to the entire paragraph with the same color.
Using more indentation than the width of the object may not produce the same results as expected.	This warning implies that indenting the text beyond the text object width may not translate the text correctly.
Using Superscript or SubScript with a Batkey may not produce the	This warning implies that text objects containing combinations of Superscript and/or Subscript , and

Warning Message	Explanation
same results as expected.	BAT key options may not translate correctly.
Width of the object should not be less than or equal to zero.	This warning implies that the text object width should be set to greater than zero.

Glossary

The following is a list of the basic Vitesse terms used in this guide.

action – An object that prompts printer for execution of specified job requirements and/or sets or changes application variables.

alignment – The horizontal and/or vertical justification of objects placed in the design area.

AND – A logic operator equivalent to the sentential connective “and;” used when all the conditional logic statements return a true value.

arrangement – The placement of objects (i.e. Bring to Front, Send to Back, Bring Forward and Send Backwards) in the design area.

barcode – A special identification code representing binary information read by an optical scanner; code may include vertical bars of differing widths, numbers and letters, or a combination of both.

chart – A graph representing data.

condition – Standard logical tests (i.e. IF, ELSE, ELSEIF) used to make the required processing decisions.

crosshair – Intersecting lines used in lieu of the cursor for accurate object placement in the design area.

Database mode – A VIPP project mode used for processing files containing field delimited records; allows for creation of database-enabled documents such as mailers, promotional flyers, and insurance contracts.

data file – A flat file comprising data in the form of text and/or numbers.

data group – A grouping of prefixed records and fields having the same prefix family, and stored and processed as a unit.

data window – A Vitesse window that displays the active application data file.

DDC – Dynamic Document Composition based on real time composition of the document at the printer or imaging device.

design area – A Vitesse window that displays the active application.

delimiter – A character such as ‘.’ or ‘;’ used as a separator between records and fields, and between fields.

directory – A folder in which saved files reside.

Else – Evaluates a condition if the preceding **If** (and **ElseIf**) statement is false.

ElseIf – Evaluates multiple **If** conditions; executes only if the preceding **If** or **ElseIf** statements are false.

exporting – The process of converting applications and associated resources from a Vitesse or Elixir environment to VIPP.

fast keys - See *keyboard shortcut*.

field – An element of a record in which one piece of information is stored.

form – A single-page document used for background reproduction on each page of a document or job.

font – A complete set of print characters of a particular family, size, typeface, and orientation.

frame – A virtual container that controls the page layout by defining areas where text, data, or objects are to be placed before actually placing them on the page.

graphic ruler – A tool used to check object dimensions and alignment, and help ensure precise placement of the object in the design area.

grid – An overlay pattern of evenly spaced horizontal and vertical lines used for precise object positioning.

guidelines – A predefined set of horizontal and/or vertical lines used to align objects.

icon - A small, rectangular-shaped symbol used to represent an application, shortcut, file, folder, or peripheral device.

If – Represents the first conditional logic test and evaluates a condition as true or false.

importing files – The process of converting VIPP application and/or associated VIPP resources, or Xerox native resource formats, from VIPP to Vitesse or Elixir environment.

JDT – A Job Descriptor Ticket file that contains the VIPP commands used to define the required presentation of the line printer data.

keyboard shortcut - *Also called fast keys*; a special one- or two-key combination that allows execution of an operation without using the mouse.

landscape – A horizontal print orientation, narrower on the sides than on the top.

Line mode – A VIPP project mode used to create and manage legacy type data streams (line data or print-ready data) for transactional type applications such as telephone bills, credit card statements, etc.

mask – A predefined syntax used to format data. For example, @@@, @@@.

Master Doc – A project tree node containing logical page definitions used for an application, including all objects mapped to the logical page.

media – A project tree node that allows for insertion of paper definitions used for an application.

merge data – The merging of variable data picked from a data file with static text.

Messages window – A Vitesse window that displays informational, warning, and error messages in brown, black, and blue text respectively.

Native mode – A VIPP project mode used for creating, positioning, and printing static data such as text, images, lines, boxes, forms, and segments on a physical page.

NOT – Reverses the logical value of a condition evaluation.

OR – A logic operator equivalent to the sentential connective “or;” used when any conditional logic statement returns a true value.

page – An object representing a logical page.

paper – The physical attributes of a logical page, such as size, orientation, positioning, and partitioning.

portrait – A vertical print orientation, narrower on top than on the sides.

Properties View window – A Vitesse window used for setting design object attributes.

record – A discrete grouping of related data fields.

relative positioning – A technique that positions the current object relative to the previous object.

snap to grid – A feature used for fast, precise object placement by automatically aligning objects with the nearest preset grid.

snap to guidelines – A feature used for fast, precise object placement by automatically aligning objects with the nearest preset guideline.

Source view – A Vitesse window that displays VIPP source code generated for the active application.

Project Tree View window – A Vitesse window that displays the hierarchical structure of an active application.

variable – A container that stores information.

VIPP – an open language Xerox print stream, which is based on PostScript and works by extending PostScript programming language, enabling you to efficiently create Xerox printer applications.

VIPP form – A single page VIPP Native mode resource used as a background overlay in an application.

VIPP segment – A VIPP Native mode or PostScript resource that can be reused at specific application page locations.

VPC – VIPP Project Container that enables archiving of all project resources into one file.

XJT – An XML Job Ticket file that processes XML data.

XML mode – A document mode used for processing data files containing tags, attributes, and values.

zoom – Magnification settings of design area objects.

Index

A

- about
 - logical page, 289
- about this guide, I
- actions
 - deleting, 602
 - inserting, 601
 - modifying
 - variables, 613
 - properties,setting, 602
 - removing, 602
 - understanding, 600
 - using, 600
- adding
 - barcodes, 209, 637
 - charts, 561
 - colors, 215
 - using,model, 217
 - using,spectrum, 215
 - conditions
 - project tree, 537
 - data file, 377, *See* inserting data file,
 - dictionary files, 194
 - format masks, 202
 - frames, 499
 - images, 117
 - key-value pairs, 91, 101, 265
 - modifications, 266
 - objects
 - barcode, 641
 - box. *See* drawing objects box
 - chart. *See* inserting objects graphic text
 - line. *See* drawing objects line
 - page. *See* inserting objects page
 - path. *See* drawing objects path
 - polygon. *See* drawing objects
 - static table. *See* drawing objects
 - static table
 - text. *See* inserting objects text
- page borders, 294
- page numbers, 297
- pages, 290
- papers, 275
- path nodes, 138
- predefined papers, 276
- text
 - tab stops, 188
 - variables, 620
- adjusting
 - fields
 - size, 343, 430, 489
- administrator
 - preferences,setting, 723
- advanced vertical control
 - using, 364
- advanced vertical spacing
 - skip, 364
- aligning
 - graphic text, 185
 - image fields, 330, 418, 474
 - ImageRef, 686
 - bottom center, 687
 - bottom left, 686
 - bottom right, 687
 - center center, 687
 - center left, 687
 - center right, 687
 - top center, 686
 - top left, 686
 - top right, 686
 - normal fields
 - contents, 324, 405, 467
- objects. *See also* organizing objects
 - bottom, 167
 - center horizontally, 167
 - center vertically, 167

- evenly space horizontally, 168
- evenly space vertically, 168
- height, 169
- left, 167
- right, 167
- same width, 168
- top, 167
- SegRef
 - bottom center, 667
 - bottom left, 667
 - bottom right, 667
 - center center, 667
 - center left, 667
 - center right, 667
 - top center, 667
 - top left, 667
 - top right, 667
- SegRefs, 667
- text, 185
- alignment
 - image
 - bottom center, 331, 419, 475
 - bottom left, 331, 419, 474
 - bottom right, 331, 419, 475
 - center center, 331, 420, 475
 - center left, 331, 420, 475
 - center right, 331, 420, 475
 - top center, 330, 419, 474
 - top left, 330, 419, 474
 - top right, 330, 419, 474
 - normal
 - center, 324, 405, 467
 - justify, 324, 405, 467
 - left, 324, 405, 467
 - left trimmed, 406
 - right, 324, 405, 467
 - right trimmed, 406
 - string, 405
 - text
 - setting, 189
- Alignment toolbar, 166
- American National Standards Institute. *See* ANSI
- ANSI, 364
- application
 - viewing
 - source code, 738
- applying
 - 3D effect, 571
 - default regional preferences, 269
 - patterns
 - fill, 223
 - line, 222
 - regional preferences to numeric
 - formats, 266
- Arrangement toolbar, 169
- arranging
 - objects, 169
 - windows
 - design area, 55
- array values
 - deleting, 517
 - removing, 517
- array variables
 - using, 519
- ASCII, 40, 42, 239, 349
- assigning
 - objects
 - properties, 161
 - variables
 - field value, 623
- associating
 - barcode fonts
 - to barcode types, 640
- autosizing
 - objects
 - graphic text, 183
 - text, 183
- auto-sizing
 - data
 - columns, 306
- auto-sizing
 - data
 - columns, 383

B

- Back tool, 697
- background attributes
 - setting

- text, 183
- background color
 - setting
 - charts, 572
- barcodes
 - inserting, 641
 - properties,setting, 641
 - value
 - using, static text, 651
 - adding, 637
 - associating fonts
 - to types, 640
 - types, 209
 - Code 128, 638
 - Code 2 of 5, 638
 - Code 25, 638
 - Code 3 of 9, 638
 - Code 39, 638
 - data
 - specifying, 646
 - deleting, 641
 - EAN 128, 639
 - EAN13, 639
 - EAN8, 639
 - European Article Number, 639
 - formatting, 209
 - naming, 641
 - non-retail, 638
 - positioning, 641
 - Postal Numeric Encoding Technique, 639
 - PostNet, 639
 - properties,setting, 641
 - size, 645
 - types, 646
 - removing, 641
 - resizing, 645
 - setting,properties
 - absolute position, 642
 - position, 641
 - relative position, 643
 - types
 - selecting, 646
 - understanding, 637
 - Universal Product Code – A, 640

- UPCA, 640
 - using, 641
 - constants, 648
 - fields, 649
 - graphic text, 649
 - merge data, 649
 - text, 649
 - variables, 647
- begin page segments, 662
- BMP, 117, 679
- booklets
 - creating, 602
- bookmarks
 - properties,setting, 632
- border
 - setting
 - charts, 576
- box
 - drawing, 131
 - positioning, 131
 - properties,setting, 131
 - corner radius, 132
 - line style, 158
 - line width, 152
 - repeating, 134
 - size, 148
 - splits, 133
 - repeating, 134
 - resizing, 148
 - setting,properties
 - position, 131
 - splitting, 133
 - using, 131
- Box tool, 126, 131
- Browse for Folder dialog, 79, 82, 88, 96, 719, 724, 727

C

- caching
 - forms, 93
 - images, 121
 - resources, 93
 - segments, 99
- cell margins

- defining, 325
- cells
 - specifying
 - fill color, 413
 - fill pattern, 413
 - line color, 412
 - line pattern, 412
 - margins, 408
- change history
 - maintaining, 266
- Change RPE Prefix tool, 384
- changing
 - frames
 - size, 502
 - placement, 502
 - images, 118
 - paper orientation, 279
 - paper size, 278
 - prefix, 384
 - project resource scope, 80
 - referenced forms, 659
 - referenced images, 687
 - referenced segments, 670
 - rulers
 - unit of measurement, 58
- characters per line. *See* CPL
- chart data
 - manual mapping
 - using, field data, 588
 - using, fields
 - DDG Index, 584
 - using, manual mapping, 588
 - using, predefined variables, 587
- charts
 - defining
 - page level, 562
 - record level, 562
 - deleting, 563
 - inserting, 562
 - mapping
 - constants, 598
 - data, 584
 - database mode, 593
 - non-prefixed data, Line mode, 589, 591
 - variables, 597
 - XML mode, 595
 - naming, 563
 - positioning, 564
 - properties, setting, 563
 - 3D effect, 571
 - background color, 572
 - bar space setting, 578
 - borders, 576
 - color table, 573
 - data values, 578
 - fitsizing, 577
 - legends, 575
 - maintaining chart ratio, 580
 - merging values, 579
 - pie chart, 581
 - position, 564
 - scale color, 576
 - size, 568
 - spacing, 568
 - text color, 576
 - types, 563
 - value format, 579
 - Y-axis range, 580
 - removing, 563
 - resizing, 568
 - setting
 - absolute position, 564
 - relative position, 565
 - spacing, 568
 - types
 - selecting, 563
 - understanding, 561
- choosing
 - commands, 73
 - data fields
 - using delimited data, 628
 - using fixed field data, 627
 - data lines
 - using conditions, 624
 - using line number, 627
- clearing
 - messages, 54
- Close tool, 697
- CLX, 194

- Code 128 barcodes, 638
- Code 2 of 5 barcodes, 638
- Code 25 barcodes, 638
- Code 3 of 9 barcodes, 638
- Code 39 barcodes, 638
- code objects
 - setting
 - color, 704
 - font, 706
- Color Selector dialog, 215, 219, 295
 - categories
 - model, 217
 - spectrum, 215
- color table
 - setting
 - charts, 573
- colors
 - adding, 215
 - using.models, 217
 - using.spectrum, 215
 - deleting, 220
 - editing, 219
 - models,using, 217
 - selecting
 - background, 214
 - foreground, 213
 - setting
 - defaults, 224
 - spectrum,using, 215
 - using, 211
 - working, 211
 - palettes, 215
- column view
 - using, 383
- Command Prompt window, 722, 735
- commands
 - choosing, 73
- compilation errors
 - handling, 738
- Compile tool, 702
- compiling
 - source code, 702
- conditions
 - adding
 - project tree, 537
 - defining, 534
 - page level, 536
 - record level, 537
 - managing
 - forms, 557
 - segments, 557
 - mapping
 - data,Database mode, 548
 - data,Line mode, 542
 - data,page level, 542
 - data,record level, 544
 - data,XML mode, 550
 - non-prefixed data,Line mode, 542
 - prefixed data,Line mode, 546
 - operands, 534
 - operators, 534
 - comparison, 534
 - logical, 535
 - understanding, 533
 - using
 - constants, 556
 - data, 542
 - system variables, 554
 - variables, 552
- constant back, 282
- constant data
 - using, 324, 405, 467, 687
- constants
 - using
 - as barcode data, 648
- control codes. *See* PCC
- conventions, IV
 - display, IV
 - typographic, V
- converting
 - TIFF files, 715
- copying
 - objects, 164
- CPL, 240, 296
- creating
 - booklets, 602
 - data dictionary, 385
 - document set, 606
 - dynamic tables, 407, 468
 - font styles, 106

- format masks. *See also* adding format masks
- forms, 87
- frames, 498
 - using property sheet, 499
- objects
 - barcodes. *See* inserting objects barcodes
 - chart. *See* inserting objects chart
 - graphic text
 - line. *See* drawing objects line
 - page. *See* inserting objects page
 - static table. *See* drawing objects static table
 - text. *See* inserting objects text
- PCC table, 359
 - custom, 359
- preliminary project, 39
- projects
 - database mode, 233
 - line mode, 239
 - native mode, 229
 - XML mode, 259
- segments, 96
- text box, 174
- variables, 514
 - array, 517
 - bool, 520
 - number, 516
 - runtime, 616
 - string, 514
- XML file
 - with predefined variables, 178
- zebra effects, 295
- zsort
 - database mode, 308
- crosshair
 - displaying, 68
 - hiding, 68
 - using, 68
- Custom Zoom dialog, 71
- customer support, VII
- customizing
 - numeric formats, 266
- cutting

objects, 164

D

- data
 - background color
 - selecting, 380
 - columns
 - auto-sizing, 306, 383
 - data groups, 393
 - database, 317
 - delimited records, 233
 - database mode
 - delimited, 41, 629
 - database resident, 37
 - fields, 391, 450
 - font
 - selecting, 381
 - foreground color
 - selecting, 380
 - formatting, 336, 424, 481
 - legacy line, 37
 - line, 349
 - line mode
 - delimited, 40, 239, 349
 - line numbered, 40, 239, 349
 - non-delimited, 40, 239, 349
 - prefixed, 40, 239, 349
 - mapping, 317, 391, 449
 - mapping techniques
 - drag and drop, 317, 392, 450
 - Insert menu, 317, 392, 450
 - Insert toolbar, 317, 392, 450
 - non-prefixed, 396
 - processing order
 - understanding, 394
 - records, 391, 449
 - using
 - fixed field, 402
 - XML, 37, 42
 - XML mode
 - attributes, 42
 - tags, 42
 - values, 42
- data area

- defining, 386
- viewing, 387
- Data Base Master, 699, 709, 712
- data dictionary
 - creating, 385
- data file
 - adding, 304, 377, 441
 - editing, 379
 - inserting, 304, 377, 441
 - structure
 - database mode, 307
 - using, 321, 327, 332, 401, 416, 421, 466, 472, 476
- data formats
 - ASCII, 40, 42, 239, 349
- data grid
 - setting, 240, 296
- data groups
 - deleting, 437
 - inserting
 - line mode, 436
 - removing, 437
 - understanding, 434
- data switch
 - setting, 197
- Data View toolbar, 386
- Data View window, III, 52, 57, 307, 317, 349, 366, 377, 379, 387, 392, 393, 396, 402, 442, 450
 - tabs
 - Data, 52, 305, 307, 377, 378, 383, 386, 387, 443
 - Source, 52, 305, 377, 379, 443
 - using, 305, 377, 442
- database data
 - processing order
 - understanding, 318
- database mode, 37, 40
 - creating
 - zsort, 308
 - data
 - delimited, 41, 629
 - mapping, 317
 - processing order, 318
 - data file
 - structure, 307
- Data View window
 - viewing data, 305
- fields
 - image, 327
 - inserting, 320
 - normal, 320
 - static text, 336
- mapping
 - fields, 319
- projects
 - creating, 233
 - using, 233
- selecting
 - paper orientation, 235
 - paper size, 234
 - sample data file, 237
- specifying
 - data file delimiters, 238
 - delimiter, 305
- trimming
 - fields, 316
- understanding, 303
- variables
 - assigning field value, 629
- viewing
 - Data View window, 305
- working
 - with data, 317
- database mode data
 - understanding, 303
- Database Mode dialog, 233, 234, 235, 237, 238
 - tabs
 - Data, 237, 238
 - General, 234
 - Layout, 234, 235
- DBF, 41, 303
- DBM, 37, 699, 709, 712
- DDC, 38, 76, 717
- decrementing
 - variables, 619
- DEF files, 711
 - xgf.def, 711
- default

- tab value
 - setting, 189
- default regional preferences
 - applying, 269
- Define Area tool, 386
- defining
 - cell margins, 325
 - charts
 - page level, 562
 - record level, 562
 - conditions, 534
 - page level, 536
 - record level, 537
 - data, 246
 - data area, 386
 - fields
 - horizontally, 350
 - vertically, 350
 - media properties, 275
 - page setup, 275, 289
 - print setup, 275
 - RCD conditions
 - skip, 369
 - skip
 - RCD conditions, 369
 - test data, 39
- Delete dialog, 541
- Delete tool, 276, 290, 320, 395, 401, 437, 456, 462, 563, 602, 641
- deleting
 - actions, 602
 - array values, 517
 - barcodes, 641
 - charts, 563
 - colors, 220
 - conditions. *See* removing conditions
 - data groups, 437
 - dictionary files, 195
 - fields, 320, 401, 462
 - format masks, 203
 - frames, 499
 - key-value pairs, 91, 101
 - objects, 165
 - pages, 290
 - papers, 276
 - path nodes, 139
 - pcc definitions, 363
 - RCD conditions, 373
 - records, 395, 456
 - skip definitions, 369
 - test conditions, 373
 - text
 - tab stops, 188
 - variables, 520
 - VFU keys, 376
- delimited data, 248
- delimited files
 - column view, 383
- delimiters
 - fields, 317
- design area
 - full screen view, 72
 - pan view, 71
- tabs
 - Design View, 48, 702, 737, 738
 - Source View, 49, 350, 613, 699, 700, 701, 702, 703, 704, 706, 737, 738
- using, 48
- windows
 - arranging, 55
 - zooming, 70
- dialogs
 - Browse for Folder, 79, 82, 88, 96, 719, 724, 727
 - Color Selector, 215, 219, 295
 - Custom Zoom, 71
 - Database Mode, 233, 234, 235, 237, 238
 - Delete, 541
 - Export, 719, 727
 - Export Fonts, 720
 - Export Text, 192
 - Find, 198, 701
 - Form, 87
 - Import, 711
 - Import Text, 191
 - Insert Data File, 237, 245, 304, 377, 441
 - Insert Form, 526, 656

- Insert Image, 117, 529, 680
 - Insert Paper, 277
 - Insert Segment, 102, 528, 663, 675, 676
 - Line Mode, 239, 240, 244, 253, 257
 - Native Mode, 229, 230, 231
 - New Project, 87, 96, 229, 233, 239, 259, 691, 693
 - Open, 118, 180, 718, 728, 733
 - Preferences, 54, 60, 79, 83, 109, 121, 180, 194, 202, 224, 269, 277, 300, 381, 530, 699, 704, 715, 723, 734
 - Print, 697
 - Properties, 697
 - Replace, 199, 701
 - Save, 54
 - Save As, 707, 708
 - Save As Zip, 708
 - Segment, 96
 - Select dictionary file, 194
 - Source Export, 703
 - Spelling, 193
 - XML Mode, 259, 260, 261
 - dictionary files
 - adding, 194
 - deleting, 195
 - dictionary files list
 - rearranging, 195
 - directory structure
 - Vitesse, 77
 - display conventions, IV
 - displaying
 - crosshair, 68
 - grid, 60
 - DJDE, 522
 - document
 - setting
 - flow, 256, 357
 - document control printing options
 - setting, 604
 - document flow
 - cyclic, 256, 357
 - two page, 256, 358
 - Document Services Platform. *See* DocuSP
 - document set
 - creating, 606
 - Document Structuring Conventions. *See* DSC, *See* DSC, *See* DSC
 - documents
 - batch conversion, 44
 - properties, setting, 265
 - DocuSP, 606
 - print jobs, 606, 607
 - DocuSP printers, 606
 - drawing
 - objects, 125
 - box, 131
 - line, 127
 - path, 136
 - polygon, 141
 - static table, 144
 - Drawing toolbar, 71, 125, 127, 131, 136, 141, 144, 174, 498, 674
 - drawing tools
 - using, 125
 - DSC, 277, 282, 728
 - dynamic behavior of segments, 101
 - Dynamic Document Composition, 76, 717, *See* DDC
 - Dynamic Job Descriptor Entry. *See* DJDE
 - dynamic tables
 - creating, 407, 468
 - dynamically sizing
 - SegRefs, 672
- ## E
- EAN 128 barcodes, 639
 - EAN13 barcodes, 639
 - EAN8 barcodes, 639
 - editing
 - colors, 219
 - data file, 379
 - font styles, 109
 - source code, 702
 - text, 175
 - Elixir training, VIII
 - Elixir Vitesse Document, 691, 707, *See* EVD
 - Elixir Vitesse Form, 691, 708
 - Elixir Vitesse Segment, 691, 708

- Elixir Vitesse Zip, 691, 692, 708
- Elixir website support, VIII
- ELX, 709, 712
- embedded forms, 655
- embedded segments, 661
- enabling
 - field insertion
 - forms, 732
 - segments, 732
 - forms
 - field insertion, 732
 - frame overflow
 - segment fields, 510
 - segments, 509
 - SegRefs, 510
 - segment overflow, 508
 - segments
 - field insertion, 732
- end page segments, 662
- EPS, 95, 101, 557
- error handling
 - source code, 737
 - compilation, 738
- error messages, 739
 - import, 739
- errors
 - compilation
 - handling, 738
- European Article Number barcodes, 639
- EVD, 44, 691, 707
 - saving, 707
- EVF, 87, 691, 708
 - saving, 708
- EVS, 95, 691, 708
 - saving, 708
- EVZ, 691, 692, 708
 - saving, 708
- Export dialog, 719, 727
- export folder
 - setting
 - permissions, 726
- Export Fonts dialog, 720
- Export Text dialog, 192
- Export tool, 719
- exporting
 - images, 698
 - preferences, setting, 714
 - buffer size, 716
 - comment level, 715
 - converting TIFF files, 715
 - default measurement unit, 714
 - maximum form number, 716
 - using elixir comments, 715
 - projects, 709, 714, 717
 - managing un-mapped fonts, 720
 - packaging other files, 718
 - Rich Text Format, 192
 - source code, 703
- Extensible Markup Language, 42, 259, *See* XML
- extensions
 - .BMP, 117, 679
 - .CLX, 194
 - .DBM, 709
 - .DEF, 711
 - .ELX, 709, 712
 - .EPS, 95, 557
 - .EVD, 691, 707
 - .EVF, 87, 691, 708
 - .EVS, 691, 708
 - .EVZ, 691, 692, 708
 - .FNT, 109
 - .FRM, 81, 87, 109, 699, 703, 709, 711
 - .FSL, 709, 712
 - .FSL, 109
 - .JDT, 81, 699, 703, 709, 711, 712
 - .JPEG, 117, 679, 715
 - .JPG, 117, 679
 - .LM, 349
 - .NM, 699, 709, 712
 - .PCX, 117, 679
 - .PM, 95
 - .PNG, 117, 679
 - .PS, 95
 - .RTF, 191
 - .SEG, 81, 95, 699, 703, 709, 712
 - .SUB, 277, 728
 - .TIF, 117, 679
 - .TIFF, 715
 - .TLX, 194

- .VPC, 709, 712, 717
- .VPF, 81, 709, 712, 717
- .XJT, 699
- .XML, 441
- external projects
 - rich text
 - using, 191
- extracting
 - delimited sub-string, 204
 - fixed length sub-string, 203
 - variable length sub-string. *See also*
 - extracting delimited sub-string

F

- field, 588
- fields
 - adjusting
 - size, 343, 430, 489
 - defining
 - horizontally, 350
 - vertically, 350
 - deleting, 320, 401, 462
 - delimiters, 317
 - inserting
 - database mode, 320
 - line mode, 400
 - XML mode, 461
 - mapping
 - database mode, 319
 - line mode, 400
 - XML mode, 461, 462
 - number
 - using, 403
 - positioning, 337, 425, 483
 - removing, 320, 401, 462
 - scaling
 - image, 345, 433, 491
 - segment, 345, 433, 491
 - setting
 - absolute position, 338, 425, 484
 - justification, 325, 414, 469
 - relative position, 339, 426, 485
 - rotation, 344, 433, 490
 - size, 341, 429, 488

- spacing, 431, 489
- specifying
 - borders, 408
 - image, 327, 416, 471
 - segment, 332, 421, 476
- trimming, 316
- types
 - image, 327, 416, 471
 - normal, 320, 401, 465
 - segment, 332, 421, 476
 - static text, 336, 423, 481
- using
 - as barcode data, 649
 - borders, 407, 468
- Find dialog, 198, 701
- Find tool, 701
- finding
 - text, 198
- finishing operations
 - specifying, 607
- Fit space
 - fit RPE space, 577
 - fixed size, 577
- FNT, 37, 109
- folder preferences
 - directory
 - adding, 82
 - deleting, 83
 - directory list
 - rearranging, 83
 - setting, 81
- folders
 - Clients, 77
 - Config, 78
 - Data Files, 78
 - Default, 77
 - FormLib, 78
 - fShared, 77
 - ImgLib, 78
 - MisLib, 78
 - project, 77
 - SegLib, 78
 - Vitesse, 77
 - xgf, 78
- fonts

- mapping
 - legacy fonts, 110
 - VIPP fonts, 105, 112
- setting
 - default font, 114
 - viewing options, 115
- styles
 - creating, 106
 - editing, 109
 - managing, 106
 - name, 107
 - saving, 109
 - size, 107
 - subscript, 108
 - superscript, 108
 - type, 107
 - width, 107
- types
 - legacy, 109
 - PostScript, 105
 - supported in Vitesse, 105
 - True Type, 105
 - VIPP, 105
 - working, 105
- Form dialog, 87
- format masks
 - adding, 202
 - creating. *See also* format masks
 - adding
 - deleting, 203
 - editing, 203
 - moving down, 203
 - moving up, 203
- formatting
 - barcodes, 209
 - data, 336, 424, 481
 - number string, 201
 - objects
 - graphic text, 189
 - text, 183
 - page numbers, 297
 - color, 298
 - font, 298
 - position, 299
 - start value, 299
 - time string, 206
- FormRefs
 - changing, 659
 - referencing forms using variables, 659
- forms, 37
 - caching, 93
 - creating, 87
 - EVFs, 87
 - FRMs, 81, 87
 - inserting, 656
 - project design, 657
 - project resource, 656
 - resources, 37
 - saving, 708
 - selecting
 - height, 89
 - width, 89
 - setting
 - details, 91
 - properties, 89
 - specifying details, 91
 - types
 - embedded, 655
 - inline, 655
 - linked, 655
 - referenced, 656
 - using, 87, 655
 - variables, 530
- Frame tool, 126, 498, 502
- frames
 - adding, 499
 - changing
 - placement, 502
 - size, 502
 - creating, 498
 - using property sheet, 499
 - deleting, 499
 - inserting, 499
 - positioning
 - objects, 505
 - previewing
 - using borders, 500
 - properties, setting
 - style, 501
 - width, 500

- removing, 499
- understanding, 495
- using, 495
 - conditions, 508
 - field, 506
 - number, 511
 - SegRef, 510
 - text, 506
- FRM, 37, 81, 87, 109, 699, 703, 709, 711
- FSL, 37, 109, 709, 712
- Full Screen tool, 72, 697
- Full Screen toolbar, 72

G

- generating
 - PDF bookmarks, 632
- GEP Keys, 162
- getting the answers you need, VII
- global resources, 75
- graphic text
 - aligning, 185
 - autosizing, 183
 - formatting, 189
 - aligning, 185
 - line spacing, 184
 - paragraph spacing, 185
 - rotation, 190
 - inserting, 174
 - joining, 193
 - merging data, 178, 181
 - positioning, 175
 - properties,setting, 175
 - setting,properties
 - autosize, 183
 - formatting, 189
 - joining, 193
 - merge data, 178, 181
 - position, 175
 - rotation, 190
 - using
 - as barcode data, 649
- graphic text objects
 - formatting
 - setting text rotation, 190

- Graphic Text tool, 126, 174
- grid
 - displaying, 60
 - hiding, 60
 - setting
 - color, 62
 - spacing, 63
 - style, 63
 - snapping, 60
 - using, 59
- guides
 - moving to the design area, 65
 - related, IX
 - removing from design area, 66
 - setting
 - style, 67
 - color, 66
 - snapping, 66
 - turning off, 64
 - turning on, 64
 - using, 64

H

- help, VIII
 - help pop-ups and jumps, VIII
 - related guides, IX
 - What's this? help, VIII
- hiding
 - crosshair, 68
 - grid, 60

I

- image fields
 - aligning, 330, 418, 474
 - specifying
 - data
 - using the data file, 327, 472
 - using variables, 473
 - using
 - project resources, 329, 473
- ImageRefs
 - aligning, 686
 - changing, 687

- positioning, 681
- resizing, 683
- rotating, 684
- scaling, 685
- sizing, 683
- using variables, 687

images

- adding, 117
- caching, 121
- changing, 118
- exporting, 698
- inserting
 - project design, 679
- previewing
 - as faded template, 119
 - white as transparent, 120
- previewing, low resolution, 121
- referencing, 679
- replacing, 118
- setting
 - path, 118
 - properties, 118

IMG, 37

import

- preferences, setting
 - using preferences from DEF file, 711

Import dialog, 711

- tabs
 - Existing, 711

Import Text dialog, 191

Import tool, 711

importing

- preferences, setting, 709
 - comments, 710
 - text merge, 710
- projects, 709, 711
- Rich Text Format, 191
- VIPP projects, 81, 734

incrementing

- variables, 619

indenting

- text, 186

inline forms, 655

inline segments, 661

inline VIPP code

- inserting, 613

Insert Action tool, 601

Insert barcodes tool, 641

Insert Chart tool, 563

Insert Data File dialog, 237, 245, 304, 377, 441

- tabs
 - Existing, 304, 441

Insert Data File tool, 304, 377, 441

Insert Data Group tool, 437

Insert Field tool, 320, 401, 462

Insert Form dialog, 526, 656

- tabs
 - Existing, 527, 656
 - New, 527, 656

Insert Form tool, 526, 656

Insert Image dialog, 117, 529, 680

- tabs
 - Existing, 117, 529, 680

Insert Image tool, 529

Insert Page tool, 290

Insert Paper dialog, 277

Insert Paper tool, 275, 277

Insert Record tool, 395, 456

Insert Segment dialog, 102, 528, 663, 675, 676

- tabs
 - Existing, 102, 528, 663
 - New, 102, 528, 663

Insert Segment tool, 528, 663

Insert toolbar, 95, 102, 117, 126, 275, 277, 290, 304, 317, 320, 377, 392, 395, 401, 437, 441, 450, 456, 462, 526, 528, 529, 537, 563, 601, 641, 656, 663, 674, 680

inserting

- actions, 601
- conditions. *See* adding conditions
- data file, 377. *See* adding data file, *See* adding data file
- data groups
 - line mode, 436
- fields
 - database mode, 320
 - line mode, 400

- XML mode, 461
- forms, 656
 - project design, 657
 - project resource, 656
- frames, 499
- images
 - project design, 679
- inline VIPP code, 613
- key-value pairs, 91, 101, 265
- modifications, 266
- objects
 - barcode. *See* inserting objects
 - barcodes
 - chart, 562
 - graphic text, 174
 - text, 173
- papers, 275
- path nodes. *See* adding path nodes
- predefined papers. *See* inserting
- records
 - line mode, 395
 - XML mode, 455
- segments, 663
 - project design, 664
 - project resource, 663
- text
 - tab stops. *See* adding text tab stops

introduction, 37

J

- JDT, 37, 81, 349, 387, 699, 703, 709, 711
- Job Descriptor Ticket, 387, 699, 703, 709, 711, *See* JDT
- job descriptor tickets, 81
- JPEG, 117, 679, 715
- JPG, 117, 679
- justification
 - horizontal
 - center, 325, 414, 470
 - justify, 325, 414, 470
 - left, 325, 414, 470
 - right, 325, 414, 470
 - vertical
 - bottom, 326, 415, 470

- center, 326, 415, 470
- top, 326, 415, 470

K

- key-value pairs
 - adding, 91, 101, 265
 - deleting, 91, 101
 - inserting, 91, 101, 265
 - removing, 91, 101

L

- labels
 - specifying
 - fields, 584
 - storing
 - fields, 585
- LCDS, 709
- Legacy Elixir Format, 709
- Legacy Elixir Format, 712
- Legacy Font Mapper, 109
- legacy fonts
 - mapping, 110
 - using, 109
- legends
 - charts
 - setting, 575
- LGO, 37
- limiting
 - resource
 - deletion, 731
 - insertion, 729
 - Vitesse functions, 724
- line
 - properties, setting, 127
 - size, 129, 157
 - width, 151
 - setting
 - spacing, 184
 - setting, properties
 - position, 127
- Line Condition Data Stream, 709
- line mode, 37, 40
 - data

- delimited, 40, 239, 349
- line numbered, 40, 239, 349
- non-delimited, 40, 239, 349
- prefixed, 40, 239, 349
- understanding, 349
- data groups
 - inserting, 436
- fields
 - image, 416
 - inserting, 400
 - normal, 401
 - segment, 332, 421
 - static text, 423
- inserting
 - records, 395
- looping
 - page delimiter line, 254, 356
- mapping
 - fields, 400
 - records, 394
- matching
 - page delimiter, 252, 354
- presentation
 - listing, 350
 - Record Processing Entry. *See* RPE
 - RPE, 350
- printing
 - page delimiter line, 253, 355
- projects
 - creating, 239
 - using, 239
- selecting
 - paper orientation, 243
 - paper size, 242
 - sample data file, 245
- setting
 - data grid, 240, 296
 - document flow, 256, 357
 - margins, 244
- specifying
 - control codes, 257
 - delimiter, 249, 352
 - field type, 248
 - fixed field data, 248, 351
 - page delimiter, 251, 353
 - PCC, 257
 - prefixes, 247
 - Printer Carriage Controls, 257
 - Printer Control Characters, 257
 - variables
 - assigning field value, 623
 - working
 - with data, 391
 - with delimited data, 248
 - with page delimiters, 250, 353
- Line Mode dialog, 239, 244, 249, 254,
 - tabs
 - Data, 245, 247, 248, 249
 - General, 239
 - Layout, 240, 244
 - Page, 251, 257
- Line tool, 126, 127
- lines
 - drawing, 127
 - positioning, 127
 - resizing, 129
 - using, 127
- Lines Per Page. *See* LPP
- linked forms, 655
- linked segments, 661
- listing color
 - specifying, 389
- LM, 349
- loading
 - palettes, 215
 - XML file
 - predefined variables, 180
- local resources, 75
- locating
 - test data, 39
- logical operators
 - AND, 535
 - NOT, 535
 - OR, 535
- looping
 - line mode
 - page delimiter line, 254, 356
- LPP, 240, 296, 367

M

- maintaining
 - change history, 266
 - chart
 - ratio, 580
- managing
 - font styles, 106
 - forms
 - conditions, 557
 - segments
 - conditions, 557
 - unmapped fonts, 720
- manual mapping
 - field data, 588
- mapping
 - charts
 - constants, 598
 - non-prefixed data,Line mode, 589, 591
 - variables, 597
 - conditions
 - data, 584
 - data,Database mode, 548
 - data,Line mode, 542
 - data,XML mode, 550
 - non-prefixed data,Line mode, 542
 - prefixed data,Line mode, 546
 - conditions,page level
 - data, 542
 - conditions,record level
 - data, 544
 - constants
 - charts, 598
 - data, 317, 391, 449
 - conditions,page level, 542
 - conditions,record level, 544
 - data,Database mode
 - conditions, 548
 - data,Line mode
 - conditions, 542
 - data,XML mode
 - conditions, 550
 - database mode
 - charts, 593
 - fields
 - database mode, 319
 - line mode, 400
 - XML mode, 461, 462
 - legacy fonts to VIPP fonts, 110, 112
 - non-prefixed data,Line mode
 - charts, 589, 591
 - conditions, 542
 - prefixed data,Line mode
 - conditions, 546
 - records
 - line mode, 394
 - XML mode, 453, 456
 - variables
 - charts, 597
 - VIPP fonts to PC fonts, 105, 112
 - XML mode
 - charts, 595
- margins
 - bottom, 244, 293
 - left, 244, 293
 - right, 244, 293
 - setting, 244
 - specifying
 - line style, 410
 - line width, 409
 - top, 244, 293
- matching
 - line mode
 - page delimiter, 252, 354
- media, 275
 - defining
 - properties, 275
- menu, 72
- menu bar, 56
- merge data
 - using
 - as barcode data, 649
- Merge Data tool, 531
- Merge Data toolbar, 181, 530, 531, 733
- merging
 - data
 - graphic text, 181
 - text, 181
 - values, 579

merging data with static fields, 482
Message View window, 53, 702, 737

messages
error, 737
information, 737
warning, 737

messages
clearing, 54
error, 737, 739
import, 739
information, 737
saving, 54
warning, 737, 741
export, 742
import, 741
text, 747

metacode, 37

modes
database, 37, 40
line, 37, 40
native, 37, 40
XML, 37, 40

modifications
adding, 266
inserting, 266

modifying
text
tab stops, 188
variables, 520
using actions, 613

moving
guides to the design area, 65
objects, 69, 165

N

naming

barcodes, 641
charts, 563

native mode, 37, 40

projects
creating, 229
using, 229
selecting
paper orientation, 231

paper size, 230

Native Mode dialog, 229, 230, 231
tabs

General, 229
Layout, 230, 231

Native Mode Job, 699, 709, 712

navigating
workspace, 72

new features
batch converting
documents, 44
form behavior, 43
frame behavior, 43
predefined media and fields, 44
restricted mode operation, 43
segment behavior, 43

New Project dialog, 87, 96, 229, 233, 239,
259, 691, 693

tabs
Existing, 691
New, 87, 96, 229, 233, 239, 259
Recent, 693, 725

New tool, 229, 233, 239, 259

Next tool, 697

NM, 37, 699, 709, 712

nodes

sequencing, 166
non-prefixed data, 246, 350

non-retail barcodes, 638

normal fields

aligning
contents, 324, 405, 467
specifying
data, 321, 401, 466
using constant data, 324, 467
using the data file, 466
using variables, 322, 466

nudging

objects, 69

number up. *See* NUP

number variables

using, 516

NUP, 282

eight up, 283

four up, 283

- multi up, 283
- partitions, 282
- two up, 283

O

objects

- adding
 - barcode, 641
- aligning. *See also* objects organizing
 - bottom, 167
 - center horizontally, 167
 - center vertically, 167
 - evenly space horizontally, 168
 - evenly space vertically, 168
 - left, 167
 - right, 167
 - same height, 169
 - same width, 168
 - top, 167
- aligning to grid, 60
- aligning to guides, 66
- arranging, 169
- assigning
 - properties, 161
- autosizing
 - graphic text, 183
 - text, 183
- basic operations, 162
- copying, 164
- cutting, 164
- deleting, 165
- drawing, 125
 - Box, 126, 131
 - Frame, 126
 - Graphic Text, 126
 - Line, 126
 - Pan View, 126
 - Path, 126, 136
 - Polygon, 126, 141
 - Select, 126
 - static table, 144
 - Table, 126
 - Text, 126
 - Zoom, 126

- formatting
 - graphic text, 189
 - text, 183
- inserting
 - chart, 562
 - graphic text, 174
 - page, 290
 - text, 173
- joining
 - graphic text, 193
 - text, 193
- moving, 69, 165
- nudging, 69
- organizing. *See also* objects aligning
- pasting, 164
- placing
 - in design area, 58
- positioning. *See also* objects moving
 - barcodes, 641
 - box, 131
 - charts, 564
 - graphic text, 175
 - line, 127
 - path, 136
 - polygon, 141
 - static table, 144
 - text, 175
- relative, 496
- renaming, 166
- resizing, 165
 - barcodes, 645
 - box, 148
 - charts, 568
 - line, 129
 - path, 149
 - polygon, 149
 - static table, 150
 - text, 177
- selecting, 162
 - design area, 163
 - project tree, 163
- setting
 - line style, 156, 157, 158, 159, 160, 161

- line width, 150, 151, 152, 153, 154,
155
- size, 148
- setting attributes, 51
- spacing
 - charts, 568
- static, 496
- super nudging, 69
- using
 - barcode, 641
 - box, 131
 - line, 127
 - path, 136
 - polygon, 141
 - static table, 144
- Open dialog, 118, 180, 728, 733
 - tabs
 - Existing, 718
- Open tool, 691, 693
- opening
 - administration mode, 722
 - projects, in scope specific view, 81
 - resources
 - existing, 691
 - recent, 693
 - user mode, 735
- optimization
 - memory size
 - setting, 698
- options,setting
 - crosshair, 68
 - data grid, 300
 - default view, 54
 - grid, 60
 - guide, 65
 - margins, 300
 - printing, 272, 280
 - duplex, 280
 - tumble, 281
 - project information, 265
 - ruler, 58
 - simplex printing, 280
 - snap to grid, 61
 - snap to guide, 66
 - text

- justification, 195
- text attributes, 195
- organizing
 - objects. *See also* aligning objects
- orientation
 - inverse landscape, 90, 231, 236, 243,
262, 280, 287
 - inverse portrait, 90, 231, 236, 243,
261, 279, 287
 - landscape, 90, 231, 235, 243, 261,
279, 287
 - page
 - selecting, 292
 - portrait, 90, 231, 235, 243, 261, 279,
287
- overflow
 - segments, 43

P

- packaging
 - files with projects, 718
- page break, 250, 296, 353
- page control printing options
 - setting, 609
- Page Criteria Definition. *See* PCD
- page delimiters, 250, 353
 - line
 - looping, 254, 356
 - printing, 253, 355
 - matching, 252, 354
 - first line, 252, 354
 - last line, 252, 354
- page setup
 - defining, 275, 289
- pages
 - adding, 290
 - deleting, 290
 - properties,setting, 290
 - borders, 294
 - data grid, 296
 - formatting numbers, 297, 298, 299
 - margins, 293
 - name, 290
 - numbers, 297

- orientation, 292
- positioning numbers, 299
- selecting paper side, 291
- selecting papers, 291
- zebra effects, 295
- removing, 290
- palettes
 - loading, 215
 - saving, 221
- Pan View tool, 71, 126
- paper
 - selecting
 - height, 230, 234, 242, 260
 - orientation, 231, 235, 243, 261
 - size, 230, 234, 242, 260
 - width, 230, 234, 242, 260
- papers
 - adding, 275
 - back, 276
 - changing
 - orientation, 279
 - size, 278
 - deleting, 276
 - front, 276
 - inserting, 275
 - partitions, 282
 - printing, 280
 - constant back, 282
 - duplex, 280
 - NUP, 282
 - simplex, 280
 - tumble, 281
 - properties, setting, 278
 - removing, 276
 - selecting, 291
 - height, 278, 286
 - type, 281
 - width, 278, 286
 - setting
 - color, 282
 - weight, 281
 - working, 275
- paragraph
 - setting
 - spacing, 185
- pasting
 - objects, 164
- path
 - adding
 - nodes, 138
 - deleting
 - nodes, 139
 - drawing, 136
 - positioning, 136
 - nodes, 139
 - properties, setting, 136
 - line style, 159
 - line width, 153
 - nodes, 137
 - size, 149
 - resizing, 149
 - setting, properties
 - position, 136
 - using, 136
- Path tool, 126, 136
- patterns
 - fill
 - applying, 223
 - line
 - applying, 222
 - setting
 - defaults, 224
 - using, 221
- PCC, 250, 257, 358
 - setting
 - VFU table, 375
- table, 257, 358
 - ANSI, 257, 358
 - IBM 1403 EBCDIC, 257, 358
- tabs
 - SETPCC, 359, 363, 364
 - VFU, 375, 376
- definitions
 - deleting, 363
 - removing, 363
- table
 - selecting, predefined, 364
 - creating, 359
- PCD, 624
- PCX, 117, 679

- PDF, 38, 632
 - bookmarks, 38
- persistent variables
 - using, 525
- placing
 - objects
 - in design area, 58
- PM, 95
- PNG, 117, 679
- polygon
 - drawing, 141
 - positioning, 141
 - properties,setting, 141
 - line style, 160
 - line width, 154
 - shape, 142
 - size, 149
 - resizing, 149
 - setting,properties
 - position, 141
 - using, 141
- Polygon tool, 126, 141
- Portable Document Format. *See* PDF
- positioning
 - fields, 337, 425, 483
- ImageRefs, 681
- objects. *See* moving objects
 - barcodes, 641
 - box, 131
 - charts, 564
 - graphic text, 175
 - line, 127
 - path, 136
 - polygon, 141
 - static table, 144
 - text, 175
- page numbers, 299
- SegRefs, 666
- Postal Numeric Encoding Technique
 - barcodes, 639
- PostNet barcodes, 639
- postscript print controller, 38
- postscript segments, 101
- predefined
 - media, 44
 - setting, 727
 - variable list, 44
- predefined papers
 - adding, 276
 - inserting, 276
- Preferences dialog, 54, 58, 60, 61, 62, 63, 65, 66, 68, 69, 79, 83, 121, 180, 202, 269, 277, 286, 287, 300, 380, 381, 530, 698, 699, 704, 705, 706, 710, 715, 723, 724, 726, 728, 729, 730, 731, 733, 734
 - categories
 - Administrator, 180, 723, 724, 726, 728, 729, 730, 731, 733, 734
 - Barcode, 640
 - Color, 224, 225
 - Data, 380, 381
 - Default, 224, 225, 300
 - Defaults, 270, 286, 287
 - designer, 58, 60, 61, 62, 63, 65, 66, 68, 69
 - Designer, 724, 726, 728
 - Dictionary, 194
 - Encoding, 81
 - Export, 714, 715
 - Folders, 81, 734
 - FontMap, 113, 115
 - Form, 81
 - General, 79, 121, 198, 698, 724
 - Image, 81
 - Import, 710
 - JDT, 81
 - Layout, 286, 287, 300
 - Legacy Fonts, 109
 - Misc, 81
 - Project, 81
 - Regional, 270
 - Resources, 180, 729, 730, 731, 733, 734
 - Segment, 81
 - Source, 704, 706
 - Text, 196
 - Transform, 202
 - Tree View, 81
 - view, 54
 - Workspace, 81, 380, 381, 704, 706

- Preferences tool, 195, 269, 286, 287, 300, 380, 640, 698, 724, 726, 728, 730, 731
- preferences, setting
 - administrator, 723
 - export, 714
 - buffer size, 716
 - comment level, 715
 - converting TIFF files, 715
 - default measurement unit, 714
 - using elixir comments, 715
 - forms
 - maximum number, 716
 - import, 709
 - comments, 710
 - text merge, 710
 - using preferences from DEF file, 711
- prefix, 350, 398
 - changing, 384
 - length, 247
 - start byte, 247
- preliminary project
 - creating, 39
- previewing
 - frames
 - using borders, 500
 - images
 - as faded template, 119
 - white as transparent, 104, 120
 - images, low resolution, 121
 - postscript segments
 - as faded template, 102
 - projects, 691, 693
 - multi-up, 695
- Print dialog, 697
- Print Preview tool, 695
- print setup
 - defining, 275
- Print tool, 697
- Printer Carriage Controls. *See* PCC,
- Printer Control Characters. *See* PCC,
 - printing
 - collate, 272, 609
 - constant form on paper back, 282
 - cycle copy, 271
 - data values, 578
 - documents
 - using zsort, 309
 - line mode
 - page delimiter line, 253, 355
 - multiple copies of a document, 271
 - paper side, selecting, 291
 - papers, 280
 - duplex, 280
 - eight up, 283
 - four up, 283
 - multi up, 283
 - NUP, 282
 - simplex, 280
 - tumble, 281
 - two up, 283
 - several pages on one paper, 282
 - PRN, 101
 - project files
 - JDT, 81
 - VIPP, 81
 - project resource scope
 - changing, 80
 - project resources
 - using, 473
 - as field data, 329, 334, 418, 423, 478
- project tree
 - adding
 - conditions, 537
 - nodes
 - Back, 50
 - BeginPage, 662
 - Condition, 537
 - Else, 534
 - ElseIf, 533
 - End, 676
 - EndPage, 662
 - Field, 461
 - Font Styles, 51, 106
 - Fonts, 51
 - FormRef, 656, 658, 659
 - Forms, 656, 659
 - Front, 50
 - global, 80

- If, 533
- Images, 51, 117, 680
- Local, 80
- MasterDoc, 51
- Media, 50
- Orphan, 676
- Page, 51, 455, 536
- Paper, 50
- Record, 455, 461, 537
- Resources, 51, 106, 117, 656, 659, 663, 670, 680
- Segments, 51, 663, 670
- SegRef, 665, 667, 670
- shared, 80
- Start, 675
- SubDoc, 51
- Project Tree View window, 49, 72, 106, 117, 127, 131, 144, 174, 265, 290, 317, 327, 394, 450, 471, 563, 568, 663, 679, 718
- nodes
 - Chart, 563
 - Media, 275
 - Page, 562
 - Record, 562
 - Text, 175
- scope specific view, 79
- projects
 - creating
 - database mode, 233
 - line mode, 239
 - native mode, 229
 - XML mode, 259
 - exporting, 709, 714, 717
 - managing un-mapped fonts, 720
 - packaging other files, 718
 - importing, 709, 711
 - previewing, 691, 693
 - multi-up, 695
 - proof printing, 694
 - saving, 707
 - specifying
 - details, 265
 - testing, 691
 - using
 - database mode, 233
 - line mode, 239
 - native mode, 229
 - XML mode, 259
 - XML mode
 - using segments, 675
- proof printing
 - projects, 694
- properties
 - defining media, 275
 - setting
 - barcode, 641
- Properties dialog, 697
- properties,action
 - tabs
 - Transform, 602
- properties,field
 - tabs
 - Settings, 462, 473, 474
- properties,ImageRef
 - tabs
 - Settings, 684, 685, 686, 687
- properties,objects
 - FormRef, 659
 - changing, 659
 - using variables, 659
 - ImageRef, 680
 - alignment, 686
 - changing reference, 687
 - postion, 681
 - rotation, 684
 - scaling, 685
 - size, 683
 - using variables, 687
- records
 - line mode, 395
- SegRef, 666
 - alignment, 667
 - changing, 670
 - dynamic size, 672
 - postion, 666
 - rotation, 670
 - scaling, 669
 - using variables, 671
- properties,page
 - tabs

- Frame, 499
- properties,paper
 - tabs
 - Settings, 282, 283
- properties,project
 - modifications, 266
- properties,SegRef
 - tabs
 - Settings, 670
- properties,setting
 - actions, 602
 - barcode, 641
 - barcodes, 641
 - absolute position, 642
 - relative position, 643
 - bookmarks, 632
 - box, 131
 - charts, 563
 - absolute position, 564
 - relative position, 565
 - data view, 380
 - default color
 - background, 224
 - foreground, 224
 - default colors, 224
 - default fill
 - pattern, 225
 - default line
 - pattern, 224
 - default patterns, 224
 - documents, 265
 - field
 - bar code types, 209
 - concatenate strings, 208
 - extract delimited sub string, 204
 - extract fixed sub string, 203
 - format number, 201
 - format time, 206
 - fields, 337, 425, 483
 - forms, 89
 - details, 91
 - orientation, 90
 - size, 89
 - graphic text, 175
 - bar code types, 209
 - concatenate strings, 208
 - extract delimited sub string, 204
 - extract fixed sub string, 203
 - format number, 201
 - format time, 206
- images, 118
 - path, 118
 - previewing as faded template, 119
 - previewing white as transparent,
 - 104, 120
- line, 127
- pages, 290
 - name, 290
- papers, 278, 286
 - media, 287
 - orientation, 286
 - size, 286
- path, 136
- polygon, 141
- records
 - line mode, 395
- segments, 97
 - caching, 99
 - clipping, 99
 - details, 100
 - dynamic behavior, 101
 - dynamic size, 101
 - enable frame overflow, 101
 - origin, 98
 - previewing as faded template, 102
 - size, 97
- source code, 704
- static table, 144
- text, 175
 - bar code types, 209
 - concatenate strings, 208
 - extract delimited sub string, 204
 - extract fixed sub string, 203
 - format number, 201
 - format time, 206
- properties,subdoc
 - tabs
 - PCC, 363, 369
- properties,SubDoc
 - Cycle Copy, 272

- regional, 266
- regional, 271
- properties, text
 - tabs
 - Settings, 531
- Property View window, 51, 72, 106, 132, 161, 221, 265, 282, 305, 321, 375, 389, 400, 465, 467, 474, 498, 517, 666, 718
 - tabs
 - Action, 603, 604, 607, 609, 613, 614, 616, 619, 622, 624, 627, 628, 630, 631, 632
 - Cell, 408, 410
 - Color, 130, 135, 140, 143, 147, 211, 213
 - Cycle Copy, 272
 - Font, 107, 109, 183
 - Frame, 498, 499, 500, 501, 503
 - General, 89, 131, 148, 165, 175, 280, 305, 351, 425, 490, 537, 563, 564, 569, 666, 677, 681, 683
 - General, 583
 - Misc, 578, 580, 581, 582, 583
 - Modifications, 266
 - Nodes, 138, 139
 - Other Files, 706, 718
 - Page Number, 297, 299
 - PCC, 359, 363, 364, 365, 369, 373, 375, 376
 - Properties, 91, 100, 265
 - Regional, 266, 271
 - Settings, 99, 102, 181, 282, 296, 320, 321, 344, 389, 396, 398, 456, 509, 527, 528, 529, 531, 571, 646, 670, 684,
 - Spacing, 184, 185, 325, 336, 414, 424, 469, 481
 - Transform, 201, 204, 209, 602, 616, 623
 - Variables, 514, 516, 517, 520, 525
- PS, 95, 101, 508

R

- RCD, 365, 369

- RCD conditions
 - deleting, 373
 - removing, 373
- rearranging
 - dictionary files list, 195
- Record Criteria Definition. *See* RCD,
- Record Processing Entry, 523, *See* RPE
- records
 - deleting, 395, 456
 - inserting
 - line mode, 395
 - XML mode, 455
 - mapping
 - line mode, 394
 - XML mode, 453, 456
 - properties, setting
 - line mode, 395
 - removing, 395, 456
 - specifying
 - prefix, 398
 - repeating lines, 396
 - RPE key, 398
 - starting line, 396
 - using
 - fixed spacing, 398
 - variable spacing, 398
- referenced forms, 656
- referenced images, 679
- referenced segments, 662
- referencing
 - forms, 526
 - images, 529, 679
 - segments, 528
- refreshing
 - projects, 72
- regional preferences
 - applying
 - to numeric formats, 266
 - setting, 266
- related guides, IX
- relative objects
 - understanding, 496
- removing
 - actions, 602
 - array values, 517

- barcodes, 641
- charts, 563
- conditions. *See* deleting conditions
- data groups, 437
- fields, 320, 401, 462
- frames, 499
- guides from design area, 66
- key-value pairs, 91, 101
- pages, 290
- papers, 276
- path nodes. *See also* deleting path
 - nodes
- pcc definitions, 363
- RCD conditions, 373
- records, 395, 456
- skip definitions, 369
- test conditions, 373
- text
 - tab stops. *See* deleting text tab stops
 - variables, 520
 - VFU keys, 376
- renaming
 - objects, 166
- repeat segments, 662
 - using, 674
- repeating
 - box, 134
- repeating actions, 172
- Replace dialog, 199, 701
- replacing
 - images, 118
 - text, 199
- resizing
 - ImageRefs, 683
 - objects, 165
 - barcodes, 645
 - box, 148
 - charts, 568
 - line, 129
 - path, 149
 - polygon, 149
 - static table, 150
 - text, 177
- resource
 - deletion
 - limiting, 731
 - insertion
 - limiting, 729
 - resource loading
 - options
 - setting, 730
 - resource scope
 - changing, 80
 - global, 80
 - Local, 80
 - precedence, 76
 - shared, 80
 - understanding, 75
 - resources
 - caching, 93
 - forms, 37
 - global, 75
 - local, 75
 - opening
 - existing, 691
 - recent, 693
 - saving, 706
 - segments, 37
 - shared, 75
 - viewing
 - source code, 738
 - Restore Area tool, 387
 - Return to last valid state tool, 703
 - reverting
 - to last valid compiled state, 739
 - Rich Text Format, 191, *See* RTF
 - exporting, 192
 - importing, 191
 - rotating
 - ImageRefs, 684
 - SegRefs, 670
 - rotation text
 - setting, 190
 - RPE, 350, 398, 523
 - rulers
 - changing
 - unit of measurement, 58
 - displaying, 58
 - hiding, 58
 - using, 58

S

- Save As dialog, 707, 708
- Save As Zip dialog, 708
- Save dialog, 54
- Save tool, 707, 708
- saving
 - EVD, 707
 - EVF, 708
 - EVS, 708
 - EVZ, 708
 - font styles, 109
 - forms, 708
 - messages, 54
 - projects, 707
 - resources, 706
 - segments, 708
 - Vitesse Zip, 708
- saving palettes, 221
- scaling
 - fields
 - image, 345, 433, 491
 - segment, 345, 433, 491
 - ImageRefs, 685
 - SegRefs, 669
- scope specific view
 - global, 80
 - local, 80
 - opening projects, 81
 - shared, 80
 - using, 79
- searching
 - source code, 701
- SEG, 37, 81, 95, 508, 699, 703, 709, 712
- Segment dialog, 96
 - tabs
 - General, 96
- segment fields
 - enabling
 - frame overflow, 510
 - specifying
 - data
 - using the data file, 332, 476
 - using the variables, 333, 478
 - dynamic size, 334, 479
- segment overflow
 - enabling, 508
- segments, 37
 - caching, 99
 - clipping, 99
 - creating, 96
 - dynamic behavior, 101
 - enabling
 - frame overflow, 509
 - EPSs, 101
 - EVSs, 95
 - inserting, 663
 - project design, 664
 - project resource, 663
 - overflow, 43
 - understanding, 498
 - postscript, 101
 - viewing as faded template, 102
 - viewing white as transparent, 104
 - PRNs, 101
 - PSs, 95, 508
 - PSs, 101
 - saving, 708
 - SEG, 81, 95, 508
 - selecting
 - height, 97
 - width, 97
 - setting
 - details, 100
 - properties, 97
 - vertical spacing, 677
 - specifying details, 100
 - types
 - begin page, 662
 - embedded, 661
 - end page, 662
 - inline, 661
 - linked, 661
 - referenced, 662
 - repeat, 662
 - tag end, 676
 - tag orphan, 676
 - tag start, 675
 - understanding, 661
 - using, 95, 661

- variables, 530
- SegRefs
 - aligning, 667
 - changing, 670
 - dynamically sizing, 672
 - enabling
 - frame overflow, 510
 - positioning, 666
 - referencing segments using variables, 671
 - rotating, 670
 - scaling, 669
- Select dictionary file dialog, 194
- Select tool, 126
- selecting
 - barcode type, 646
 - chart type, 563
 - colors
 - background, 214
 - foreground, 213
 - data
 - background color, 380
 - font, 381
 - foreground color, 380
 - data file
 - delimiters, 238
 - database mode
 - paper orientation, 235
 - paper size, 234
 - sample data file, 237
 - line mode
 - paper orientation, 243
 - paper size, 242
 - sample data file, 245
 - menus, 73
 - native mode
 - paper orientation, 231
 - paper size, 230
 - objects, 162
 - design area, 163
 - project tree, 163
 - page
 - orientation, 292
 - paper orientation, 231, 235, 243, 261
 - paper side, 291
 - paper size, 230, 234, 242, 260
 - paper type, 281
 - papers, 291
 - PCC table
 - predefined, 364
 - predefined
 - PCC table, 364
 - sample data file
 - database mode, 237
 - line mode, 245
 - text
 - tabs type, 188
 - XML mode
 - paper orientation, 261
 - paper size, 260
- sequencing
 - nodes, 166
- setting
 - barcode
 - properties, 641
 - absolute position, 642
 - relative position, 643
 - box
 - corner, 132
 - splits, 133
 - character set encoding, 271
 - charts
 - absolute position, 564
 - background color, 572
 - border, 576
 - fit sizing, 577
 - legends, 575
 - properties, 563
 - relative position, 565
 - code objects
 - color, 704
 - font, 706
 - color table
 - charts, 573
 - custom zoom value, 71
 - data grid, 240
 - data switch, 197
 - default
 - tab value, 189
 - user modes, 725

- default attributes
 - colors, 224
 - patterns, 224
- default color
 - background, 224
 - foreground, 224
- default fill
 - patterns, 225
- default fonts, 114
- default line
 - patterns, 224
 - spacing, 196
- default page properties, 300
- default paragraph
 - spacing, 196
- default tab alignment, 196
- default tab value, 196
- default text
 - indentation, 197
- document control printing options, 604
- document flow, 256, 357
- document set options, 606
- export folder
 - permissions, 726
- fields
 - absolute position, 338, 425, 484
 - justification, 325, 414, 469
 - relative position, 339, 426, 485
 - rotation, 344, 433, 490
 - size, 341, 429, 488
 - spacing, 431, 489
- folder preferences, 81
- font list viewing options, 115
- form orientation, 90
- form size, 89
- FormRef.properties, 659
 - changing, 659
 - using variables, 659
- grid color, 62
- grid spacing, 63
- grid style, 63
- guide color, 66
- guide style, 67
- ImageRef.properties, 680
 - alignment, 686
 - position, 681
 - reference using variables, 687
 - rotation, 684
 - scaling, 685
 - size, 683
- import options
 - VIPP projects, 734
- line
 - spacing, 184
 - style, 156, 157, 158, 159, 160, 161
 - width, 150, 151, 152, 153, 154, 155
- line mode
 - data grid, 240, 296
 - document flow, 256, 357
 - margins, 244
- margins, 244
- nudge, 69
- object
 - size, 148
- optimization
 - memory size, 698
- options
 - crosshair, 68
 - data grid, 300
 - default view, 54
 - duplex printing, 280
 - grid, 60
 - guide, 65
 - margins, 300
 - printing, 272
 - simplex, 280
 - ruler, 58
 - snap to grid, 61
 - snap to guide, 66
 - text
 - justification, 195
 - text attributes, 195
 - tumble duplex printing, 281
- page control printing options, 609
- page margins, 293
- page properties, 290
- paper color, 282
- paper properties
 - media, 287

- orientation, 286
- size, 286
- paper weight, 281
- paragraph
 - spacing, 185
- path nodes, 137
- PCC
 - VFU table, 375
- permissions
 - export folder, 726
- pie chart
 - display type, 581
 - labels, 582
 - properties, 581
 - radius, 583
 - slice burst value, 582
- predefined media, 727
- project information, 265
- properties
 - documents, 265
 - forms, 89
 - images, 118
 - papers, 286
 - segments, 97
 - source code, 704
- properties,box, 131
- properties,data view, 380
- properties,fields, 337, 425, 483
- properties,path, 136
- properties,polygon, 141
- properties,static table, 144
- regional preferences, 266
- resource loading
 - options, 730
- scale color, 576
- segment origin, 98
- segment size, 97
- SegRef properties, 666
 - alignment, 667
 - position, 666
- SegRef,properties
 - changing, 670
 - dynamic sizing, 672
 - rotation, 670
 - scaling, 669
 - using variables, 671
- skip
 - VFU table, 375
- source view
 - background color, 705
- spacing
 - bars, 578
- spell checker
 - depth, 197
- tabs, 186
- text
 - alignment, 189
 - background attributes, 183
 - rotation, 190
 - tab stops, 187
- text color, 576
- value format, 579
- variables
 - array, 517
 - bool, 520
 - number, 516
 - string, 514
 - value, 614
- VFU table
 - PCC, 375
 - skip, 375
- VIPP projects
 - import options, 734
 - working folder, 723
 - working folders, 79
- shared resources, 75
- shortcut keys
 - using, 73
- shortcut menus
 - using, 73
- Show/Hide Data Window tool, 306, 378, 442
- Show/Hide Properties tool, 697
- skip
 - setting
 - VFU table, 375
 - tabs
 - RCD, 369, 373
 - SETSKIP, 365, 368, 369
 - VFU, 375

- skip definitions
 - deleting, 369
 - removing, 369
- Snap to Grid tool, 60
- Snap to Guide tool, 60
- source code
 - compiling, 702
 - editing, 702
 - error handling, 737
 - compilation, 738
 - exporting, 703
 - reverting
 - last compiled state, 703
 - searching, 701
 - viewing, 700
 - application, 738
 - external resources, 738
 - VIPP, 37, 38
- Source Export dialog, 703
- source view
 - setting
 - background color, 705
- Source View toolbar, 700, 702, 703
- spacing
 - bars
 - setting, 578
 - fixed, 400
 - objects
 - charts, 568
 - variable, 400
- specifying
 - barcode data, 646
 - cells
 - fill color, 413
 - fill pattern, 413
 - line color, 412
 - line pattern, 412
 - margins, 408
 - condition tests
 - Else, 538
 - ElseIf, 540
 - If, 538
 - logical operators, 539
 - AND, 539
 - NOT, 540
 - OR, 539
 - control codes, 257
 - data file
 - delimiter, 249, 305, 352
 - field type, 248
 - fixed field data, 248, 351
 - page delimiter, 251, 353
 - prefixes, 247
 - database mode
 - data file
 - delimiters, 238
 - delimiter, 305
 - dictionary files
 - spell checking, 194
 - fields
 - borders, 408
 - image, 327, 416, 471
 - labels, 584
 - segment, 332, 421, 476
 - values, 584
 - finishing operations, 607
 - form details, 91
 - frame border
 - style, 501
 - width, 500
 - line mode
 - control codes, 257
 - delimiter, 249, 352
 - field type, 248
 - fixed field data, 248, 351
 - page delimiter, 251, 353
 - PCC, 257
 - prefixes, 247
 - Printer Carriage Controls, 257
 - Printer Control Characters, 257
 - listing color, 389
 - margins
 - line style, 410
 - line width, 409
 - normal fields
 - data, 321, 401, 466
 - page numbers
 - start value, 299
 - PCC, 257
 - Printer Carriage Controls, 257

- Printer Control Characters, 257
- project details, 265
- records
 - prefix, 398
 - repeating lines, 396
 - RPE key, 398
 - starting line, 396
- segment details, 100
- static fields
 - data, 337, 482
- spell checker
 - setting
 - depth, 197
 - using, 193
- spell checking
 - specifying
 - dictionary files, 194
- Spelling dialog, 193
- splitting
 - box, 133
- Standard toolbar, 51, 164, 229, 233, 269, 300, 320, 380, 395, 401, 462, 541, 563, 602, 698, 701, 731
- static fields
 - merging data, 482
 - specifying
 - data, 337, 482
- static objects
 - understanding, 496
- static table
 - drawing, 144
 - positioning, 144
 - properties, setting, 144
 - cell spacing, 146
 - columns, 145
 - line style, 161
 - line width, 155
 - rows, 145
 - size, 150
 - resizing, 150
 - setting, properties
 - position, 144
 - using, 144
- status bar
 - viewing, 57
- storing
 - fields
 - labels, 585
 - values, 585
- string variables
 - using, 515
- strings
 - changing case, 207
 - concatenating, 208
 - extracting sub-string
 - fixed length, 203
 - variable length, 204
 - formatting
 - time string, 206
 - stripping blanks, 205
 - stripping characters, 205
 - transforming, 201
 - types
 - numeric, formatting, 201
- SUB, 277, 728
- subtracting
 - variables, 622
- super nudging
 - objects, 69
- support
 - customer, VII
 - Elixir website, VIII
- symbols used in this guide, VI
- system variables
 - using, 524

T

- Table tool, 126, 144
- tabs
 - design area
 - Design View, 48, 702, 737, 738
 - Source View, 49, 613, 699, 700, 701, 702, 703, 704, 706, 737, 738
 - Insert Form dialog
 - Existing, 656
 - New, 656
 - Insert Segment dialog
 - Existing, 663
 - New, 663

- selecting
 - type, 188
- setting, 186
- tag end segments
 - using, 676
- tag orphan segments
 - using, 676
- tag start segments
 - using, 675
- Tagged-Image File Format, 715
- test conditions
 - deleting, 373
 - removing, 373
- test data
 - defining, 39
 - locating, 39
- testing
 - projects, 691
- text
 - adding
 - tab stops, 188
 - aligning, 185
 - autosizing, 183
 - box,creating, 174
 - deleting
 - tab stops, 188
 - editing, 175
 - finding, 198
 - formatting, 183
 - aligning, 185
 - background attributes, 183
 - indenting, 186
 - line spacing, 184
 - paragraph spacing, 185
 - tab stops, 187
 - tabs, 186
 - indenting, 186
 - inserting, 173
 - joining, 193
 - merging data, 178, 181
 - modifying
 - tab stops, 188
 - positioning, 175
 - properties,setting, 175
 - size, 177
 - replacing, 199
 - resizing, 177
 - selecting
 - tabs type, 188
 - setting default tab value, 189
 - setting.properties
 - autosize, 183
 - formatting, 183
 - joining, 193
 - merge data, 178, 181
 - position, 175
 - tabs
 - aligning text, 189
 - using
 - as barcode data, 649
- text tools
 - using, 126, 173, 531
- TIF, 117, 679
- TIFF, 715
- TLX, 194
- toolbars, 57, 72
 - Alignment, 166
 - Arrangement, 169
 - Data View, 386
 - Drawing, 71, 125, 127, 131, 136, 141, 144, 174, 498, 674
 - Full Screen, 72
 - Insert, 95, 126, 275, 290, 304, 317, 377, 392, 401, 437, 462, 526, 528, 529, 563, 641, 656, 663, 674, 680
 - Merge Data, 181, 530, 531, 733
 - Source View, 700, 702, 703
 - Standard, 51, 87, 164, 229, 276, 286, 287, 290, 300, 380, 401, 541, 563, 602, 698, 701, 731
 - View, 58, 60, 64
 - Zoom, 70, 72
- tools
 - Back, 697
 - Change RPE Prefix, 384
 - Close, 697
 - Compile, 702
 - Create new font style, 107
 - Crosshair, 68
 - Define Area, 386

- Delete, 276, 290, 320, 395, 401, 437, 456, 462, 563, 602, 641
 - drawing, 125
 - Box, 126, 131
 - Frame, 126
 - graphic text, 174
 - Graphic Text, 126
 - Line, 126, 127
 - Pan View, 126
 - Path, 126, 136
 - Polygon, 126, 141
 - Select, 126
 - Table, 126, 144
 - Text, 126, 174
 - Zoom, 126
 - edit font style, 109
 - Export, 719
 - Find, 701
 - Frame, 498, 502
 - Full Screen, 72, 697
 - Grid, 60
 - Guides, 64
 - Import, 711
 - Insert Action, 601
 - Insert barcodes, 641
 - insert chart, 563
 - Insert Data File, 304, 377, 441
 - Insert Data Group, 437
 - Insert Field, 320, 401, 462
 - Insert Form, 526, 656
 - Insert Image, 117, 529, 680
 - Insert Page, 290, 563
 - Insert Paper, 275, 277
 - Insert Record, 395, 456
 - Insert Segment, 102, 528, 663
 - Merge Data, 531
 - New, 87, 96, 229, 233, 259, 697
 - Open, 691, 693
 - Pan View, 71
 - Preferences, 195, 269, 286, 287, 300, 380, 640, 698, 724, 726, 728, 730, 731
 - Print, 697
 - Print Preview, 695
 - Restore Area, 387
 - Return to last valid state, 703
 - Ruler, 58
 - Save, 707, 708
 - save font style, 109
 - Show/Hide Data Window, 306, 378, 442
 - Show/Hide Properties, 697
 - Snap to grid, 60
 - Snap to guide, 66
 - text, 173
 - Text, 531
 - zoom in, 70
 - Zoom In, 697
 - Zoom Out, 70, 697
 - Zoom to full extent, 70
 - Zoom to Height, 70
 - Zoom to Object, 70
 - Zoom to Width, 70
 - transforming
 - options
 - barcode types, 209
 - changing case, 207
 - concatenating strings, 208
 - extracting, delimited, 204
 - extracting, fixed, 203
 - format number, 201
 - formatting time, 206
 - stripping. *See also* truncating
 - truncating, 205
 - strings, 201
 - trimming
 - fields, 316
 - database mode, 316
 - typographic conventions, V
- ## U
- undersanding
 - segment types, 661
 - actions, 600
 - barcodes, 637
 - charts, 561
 - conditions, 533
 - data groups, 434
 - data processing order, 394

- database data processing order, 318
- database mode, 303
 - data file structure, 307
- Database mode, 41
- database mode data, 303
- frames, 495
- Line mode, 40
- Line mode data, 349
- Native mode, 40
- relative objects, 496
- resource scope, 75
- sample data file, 236, 245
- segment overflow, 498
- static objects, 496
- variables, 513
 - types, 513
- Vitesse directory structure, 76
- workspace, 47
- XML data processing order, 451
- XML mode, 42, 441
- Understanding
 - VIPP modes, 40
- undoing actions, 172
- Universal Product Code – A barcodes, 640
- UPCA barcodes, 640
- user modes
 - administration, 43, 722, 726
 - setting
 - default, 725
 - user, 43, 724, 726, 735
- using
 - actions, 600
 - advanced vertical control, 364
 - array variables, 519
 - color models, 217
 - color spectrum, 215
 - colors, 211
 - column view, 383
 - conditions
 - constants, 556
 - data, 542
 - system variables, 554
 - variables, 552
 - constant data, 324, 405, 467, 687
 - constants
 - barcode data, 648
 - crosshair, 68
 - data
 - fixed field, 402
 - data file, 321, 327, 332, 401, 416, 421, 466, 472, 476
 - Data View window, 52, 305, 377, 442
 - design area, 48
 - Document Structuring Conventions.
 - See* DSC
 - drawing tools, 125
 - field number, 403
 - fields
 - barcode data, 649
 - borders, 407, 468
 - image, 327, 416, 471
 - normal, 320, 401, 465
 - segment, 332, 421, 476
 - static text, 336, 423, 481
 - forms, 87, 655
 - frame number, 511
 - frames, 495
 - full screen view, 72
 - graphic text
 - barcode data, 649
 - grid, 59
 - guides, 64
 - legacy fonts, 109
 - menu bar, 56
 - merge data
 - barcodes, 649
 - merge data feature, 178
 - Message View window, 53
 - number variables, 516
 - objects
 - barcode, 641
 - box, 131
 - line, 127
 - path, 136
 - polygon, 141
 - static table, 144
 - patterns, 221
 - persistent variables, 525
 - project resources, 473

- as field data, 329, 334, 418, 423, 478
- Project Tree View window, 49
- projects
 - database mode, 233
 - line mode, 239
 - native mode, 229
 - XML mode, 259
- Property View window, 51
- records
 - fixed spacing, 398
 - variable spacing, 398
- repeat segments, 674
- rich text
 - external projects, 191
- rulers, 58
- scope specific view, 79
- segments, 95, 661
- shortcut keys, 73
- shortcut menus, 73
- spell checker, 193
- string variables, 515
- system variables, 524
- tag end segments, 676
- tag orphan segments, 676
- tag start segments, 675
- text
 - barcode data, 649
- text tools, 173
- toolbar, 57
- variables, 322, 329, 333, 403, 417, 422, 466, 473, 478, 513
 - barcode data, 647
 - to reference forms, 659
 - to reference images, 687
 - to reference segments, 671
- workspace tools, 56
- using, field data
 - manual mapping
 - chart data, 588
- using, fields
 - chart data
 - DDG Index, 584
- using, manual mapping
 - chart data, 588

- using, predefined variables
 - chart data, 587
- using, static text
 - barcode value, 651

V

- value format
 - charts, setting, 579
- values
 - merging, 579
 - specifying
 - fields, 584
 - storing
 - fields, 585
- Variable Data Intelligent Postscript Printware. *See* VIPP
- Variable Data Intelligent Postscript Printware Interactive Development Environment. *See* VIPPIDE
- variables, 513
 - adding, 620
 - assigning field value, 623
 - database mode, 629
 - line mode, 623
 - XML mode, 631
 - creating, 514
 - array, 517
 - bool, 520
 - number, 516
 - runtime, 616
 - string, 514
 - database mode
 - assigning field value, 629
 - decrementing, 619
 - deleting, 520
 - incrementing, 619
 - line mode
 - assigning field value, 623
 - modifying, 520
 - using actions, 613
 - referencing
 - forms, 526
 - images, 529
 - segments, 528

- removing, 520
- setting
 - array, 517
 - bool, 520
 - number, 516
 - PDF bookmarks, 632
 - string, 514
- subtracting, 622
- types
 - array, 513
 - bool, 514
 - number, 513
 - string, 513
 - system, 520
- understanding, 513
 - types, 513
- using, 322, 329, 333, 403, 417, 422, 466, 473, 478, 513
 - as barcode data, 647
 - to reference an image, 687
- value
 - setting, 614
- XML mode
 - assigning field value, 631
- Vertical Format Unit, 363, 366, 375, *See* VFU
- VFU, 360, 363, 366, 375
- VFU keys
 - deleting, 376
 - removing, 376
- View toolbar, 58, 60, 64
- viewing
 - data area, 387
 - source code, 700
 - status bar, 57
- VIPP, 37
 - source code, 37, 38
- VIPP fonts
 - mapping, 105, 112
- VIPP Form, 709, 711
- VIPP Project Container, 698, 709, 712, 717, *See* VIPP
- VIPP Project File, 81, 709, 712, 717
- VIPP projects
 - setting
 - import options, 734
- VIPP Segment, 709, 712
- VIPPIDE, 514, 616
- Vitesse
 - administrative options, 721
 - directory structure
 - client folder, 77
 - Clients, 77
 - Config, 78
 - Data Files, 78
 - FormLib, 78
 - ImgLib, 78
 - MisLib, 78
 - project folder, 77
 - SegLib, 78
 - understanding, 76
 - xgf, 78
 - font types, 105
 - legacy, 109
 - PostScript, 105
 - True Type, 105
 - VIPP, 105
 - introduction, 37
 - limiting functions, 724
 - new features, 43
 - users, 721
 - administrator, 721, 724
 - user, 721
 - workspace, understanding, 47
 - zip
 - saving, 708
- VPC, 37, 236, 245, 632, 698, 709, 712, 717
- VPF, 81, 709, 712, 717

W

- warning messages, 741
 - export, 742
 - import, 741
 - text, 747
- welcome, I
- What's this? help, VIII
- who should use this guide?, I
- windows
 - Command Prompt, 722, 735

- Data View, III, 52, 307, 317, 349, 366, 385, 386, 387, 392, 393, 396, 402, 442, 450
- Message, 702
- Message View, 53, 737
- Project, 718
- Project Tree View, 49, 117, 127, 131, 136, 174, 265, 275, 290, 317, 327, 392, 450, 471, 563, 568, 663, 679
- Project TreeView, 106
- Property, 718
- Property View, 50, 72, 91, 140, 206, 265, 320, 351, 398, 400, 401, 498, 514, 537, 588, 603, 641, 659, 687
 - tabs
 - Color, 215
- Property View, 214
- working
 - colors, 211
 - palettes, 215
 - in listing mode, 387
 - with control codes, 358
 - with database mode data, 317
 - with delimited data, 248
 - with fonts, 105
 - with frames, 497
 - XML mode, 512
 - with line mode data, 391
 - with page delimiters, 250, 353
 - with paper, 275
 - with PCC, 358
 - with printer carriage controls, 358
 - with printer control characters, 358
 - with regional languages, 271
 - with segments, XML mode, 675
 - with source code, 699
 - with system variables, 520
 - with XML mode data, 449
- working folder
 - setting, 723
- working folders, 77
 - setting, 79
- workspace
 - choosing commands, 73
 - design area, 48

- navigating, 72
 - understanding, 47
- workspace tools
 - using, 56

X

- Xerox
 - native font, 37
 - native format resources, 37
 - native forms, 37, 699, 703
 - native image, 37
 - native segments, 699, 703
 - VIPP compiler, 38
 - VIPP Thin Printer, 38, 632
- Xerox Form, 709
- Xerox FSL Format, 709, 712
- XJT, 37, 42, 441, 699
- XML, 37, 42, 259, 441
- XML data
 - processing order
 - understanding, 451
- XML Job Tickets. *See* XJT
- XML mode, 37, 40
 - data
 - attributes, 42
 - fields, 450
 - mapping, 449
 - processing order, 451
 - records, 449
 - tags, 42
 - values, 42
 - data file
 - attributes, 447
 - structure, 446
 - tags, 446
 - values, 447
- Data View window
 - data tree, 444
 - viewing data, 443
 - viewing multiple tag occurrences, 445
- fields
 - image, 471
 - inserting, 461

- normal, 465
- processing, 464
- segment, 476
- static text, 481
- inserting
 - records, 455
- mapping
 - fields, 461, 462
 - records, 453, 456
- projects
 - creating, 259
 - using, 259
- records
 - printing on new page, 459
 - processing, 461
- selecting
 - paper orientation, 261
 - paper size, 260
- understanding, 441
- variables
 - assigning field value, 631
- viewing
 - Data View window, 443
- working
 - with data, 449

- XML Mode dialog, 259, 260, 261
 - tabs
 - General, 260
 - Layout, 260, 261
- XML Mode Project, 699
- XML Processing Definition, 42
- XPD, 42

Z

- zebra effects
 - creating, 295
- Zoom to Width tool, 70
- Zoom In tool, 70, 697
- Zoom Out tool, 70, 697
- Zoom to full extent, 70
- Zoom to Height tool, 70
- Zoom to Object tool, 70
- Zoom tool, 126
- Zoom toolbar, 70, 72
- zooming, 70
- zsort
 - creating, 308
 - printing documents, 309