

4th Dimension

Design Reference



4th Dimension®
© 1985 - 1999 ACI SA. All Rights Reserved

4th Dimension Design Reference ***Version 6.5 for Mac[™] OS and Windows[®]***

*Copyright© 1985–1999 ACI SA/ACI US, Inc.
All rights reserved*

The software described in this manual is governed by the grant of license provided in this package. The software and the manual are copyrighted and may not be reproduced in whole or in part except for the personal licensee's use and solely in accordance with the contractual terms. This includes copying the electronic media, archiving, or using the software in any manner other than that provided for in the Software license Agreement.

4th Dimension, 4D, the 4D logo, 4D Server, 4D Runtime, 4D Compiler, 4D Write, 4D Draw, ACI, and the ACI logo are registered trademarks of ACI SA.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Apple, Macintosh, Power Macintosh, Laser Writer, Image Writer, and QuickTime are trademarks or registered trademarks of Apple Computer Inc.

Oracle is a registered Trademark of Oracle Corporation, Redwood Shores, California. Oracle Corporation and ACI are independent companies. Use of the Trademark "Oracle" authorized by Oracle Corporation authorized to highlight interoperability only.

All other referenced trade names are trademarks, registered trademarks, or copyrights of their respective holders.

IMPORTANT LICENSE INFORMATION

Use of this software is subject to its license agreement, which can be found on a separate card. Please read the License Agreement carefully before using the software.

Contents

Preface	XV
About the Manuals	xvi
About This Manual	xvii
Cross-Platform	xvii
Chapter Descriptions	xvii
Conventions	xviii
Chapter 1 4 th Dimension Basics	21
Starting 4 th Dimension	22
Creating a New Database	22
Opening an Existing Database	24
4 th Dimension Desktop Files	26
Selecting a different data file	28
Linking a Data File to a Structure File	29
Making Backups	29
The Environments	31
The Design Environment	31
The User Environment	32
The Custom Menus Environment	34
Changing Environments	36
Design Environment Editors	36
The Structure Editor	37
The Form Editor	37
The Method Editors	38
The Menu Bar Editor	40
The Password Access Editor	41
The List Editor	42
Picture Library	43
The Design Environment Interface	45
The Design Environment Menus	45
Contextual menus	49
4 th Dimension Editor Windows	50
The Toolbar	53
The Explorer	54
Working with the Explorer	54

Displaying Comments	56
Tables Page	58
Forms Page	60
Methods Page	62
Constants Page	65
Commands Page	66
Lists Page	68
Using Comments.	68
Modifying a Comment's Font Attributes	71
Inserting the Date, Time, or User Name.	71
Inserting Automatic Comments	72
Runtime Explorer.	75
Displaying the Window	75
Watch Page.	76
Process Page	77
Break and Catch Pages	78
Find Dialog Box	79
Searching in the Database	79
String Types and Scope of the Search	80
Searching Options	81
Setting Database Properties	82
General	83
Data Control	85
User Interface.	89
Design Environment	90
System Settings.	92
Display Formats and Entry Filters	95
Comments	95
Connections	96
Web Server I	97
Web Server II	101
Platform Interface	105
Platform Interface Settings.	105
4D Server Considerations	111

Chapter 2 Designing a Database Structure 113

Database Basics	114
Tables.	115
Creating a Database Structure	119
Using the Structure Editor	120
Selecting a Table Image	120
Scrolling the Field List	121
Resizing a Table Image.	122
Moving a Table Image	123

Creating a New Table	124
Setting Table Properties	125
Creating Fields and Setting Field Properties.	131
Creating New Fields	131
Field Types.	135
Field Attributes	140
Choices and Help	147
Setting the Color of the Field	148
Modifying Fields and Field Properties.	149
Renaming a Field	149
Changing a Field's Type.	149
Changing a Field's Attributes	150
Indexing or Reindexing a Field	150
Relating Tables.	152
Related Fields	154
The One Table and the Many Table.	156
Setting Relation Properties	158
Related Fields	159
Many to One Properties.	160
One to Many Properties.	161
Wildcard Choice.	162
Deletion Control	163
Color.	164
Creating a Relation Between Tables.	165
Setting Relation Properties	167
Removing Relations	169
Reestablishing Relations.	169
Modifying Relation Properties	169
Automatic and Manual Relations	170
Entering Data in Related Tables.	170
Using Wildcard Choice Lists	171
Relation Types	174
One to One Relations	174
Many to Many Relations	174
Analyzing Database Relations	181
Circular Relations	181
Multiple Links to the Same Table	182
Relations from Multiple Records	184

Chapter 3

Creating Forms	187
About Forms	188
The Form Wizard	188
The Form Editor	188
Forms, Tables, and Fields	189

Active Objects and Graphic Objects	192
Graphic Objects	192
Object Properties	193
The Form Wizard	194
The Form Editor	196
Creating a New Form.	197
Selecting Fields for the Form.	199
Reordering Fields	202
Grouping Fields	203
Removing Fields	205
Using the Form Wizard's Advanced Options	205
Adding Fields.	209
Customizing the Appearance of Form Objects	209
Customizing Buttons on the Form	212
Setting the Form Size.	215
Field Label Placement	216
Form Display Options	216
Adding a Subform to the Form	217
Creating the New Form	220
Setting the Appearance of Form Objects.	221
Fields and Field Labels	221
Using the Style Sheet Editor	222
Creating a Style Sheet	222
Applying a Style Sheet	225
Setting the Current Input and Output Forms	225
Deleting a Form	226
Renaming a Form	227

Chapter 4 **Form Editor Basics 229**

Using The Form Editor	230
Form Editor Window.	230
The Tools Palette	235
Form Editor Menus.	238
Displaying and Setting Form and Object Properties.	244
Opening a Form in the Form Editor	250
Setting Form Properties	251
Naming the Form	251
Setting Form Access	253
Setting the Platform Interface	255
Setting the Default Window Title	257
Assigning a Menu Bar to a Form.	258
Setting Sizing and Resizing Options	260
Form Events	262
Contextual On-line Help.	264

Managing Form Objects	265
Selecting Objects	265
Moving Objects	268
Resizing Objects	270
Using the Rulers	275
Creating Objects	276
Grouping Objects	278
Aligning Objects	280
Distributing Objects	286
Duplicating Objects	290
Copying Objects on a Form	292
Layering Objects	293
Deleting Objects	295
Optimizing the Appearance of Text and Picture Objects	295
Scaling a Form	296
Changing the Appearance of Objects	297
Platform Interface and Border Line Style	298
Working with Text Areas	299
Line Widths	304
Fill Patterns	305
Border Patterns	306
Foreground and Background Colors	307
Placing a Picture from the Picture Library	309
Modifying the Background of the Picture	310
Setting the Display Mode for a Static Picture	311
Dissociating a form Picture from its Library Source	314
Creating a Multi-page Form	314
Adding a Display Page to a Form	315
Moving from Page to Page	316
Deleting a Page	318
Adding Fields to a Blank Page	319
Adding Page Navigation Controls	319
Data Entry Order	320
Viewing and Changing the Data Entry Order	321
Setting the First Object in the Data Entry Order	321
Using a Data Entry Group	322
Restoring the Standard Data Entry Order	322
Viewing and Printing Forms	323
Saving Forms	323

Chapter 5 Working with Fields and Active Objects . . . 325

Active Objects Defined	326
Adding Fields to a Form	326
Modifying a Field in a Form	328

Changing a Field into a Variable and Vice-Versa	329
Inserting Dynamic Table and Field Names	329
Data Entry Controls	331
Setting the Enterable and Mandatory Attributes	332
Using Choice Lists	333
Using Entry Filters	335
Creating Custom Display Formats and Entry Filters	344
Setting Maximum and Minimum Values	348
Setting Default Values	349
Adding a Scroll Bar to a Text Object	352
Adding Help to a Field or Object	353
Display Formats	357
Date Field Formats	358
Time Field Formats	358
Number Field Formats	359
Alpha Field Formats	364
Boolean Field Formats	366
Picture Field Formats	369
Active Objects on a Form	372
Setting Object Properties	372
Creating an Active Object	374
Display Formats for Objects	376
Data Entry Controls for Enterable Objects	378
Setting the Tabable and Show Focus Properties	379
Assigning a Keyboard Equivalent	379
Enabling Drag and Drop	380
Types of Active Objects	381
Enterable and Non-enterable Variables	382
Buttons	382
Check Boxes and 3D Check Boxes	387
Radio Buttons, 3D Radio Buttons, and Radio Pictures	388
Picture Buttons	390
Menu/Drop-down Lists, Pop-up Menus, Drop-down Lists, and Scrollable Areas	397
Combo boxes	399
Hierarchical Pop-up Menus and Hierarchical Lists	400
Button Grids	401
Picture Pop-up Menus	403
Tab Controls	407
Graph Areas	410
Plug-in Objects	411
Indicators	413
Splitters	416
Duplicating on a Matrix	419
Using Object Methods with Fields and Objects	422

Object Events	423
Deleting an Object Method	428
Adding a Subform to the Form	428
Entering Data in a Subform	430
Data Entry Options for Subforms	431
Displaying Data in a Subform	432
Creating a Subform	432
Modifying a Subform	438

Chapter 6 Output Displays and Reports 439

Output Forms for Listing Records	439
Output Control Lines	442
Moving Output Control Lines.	445
Creating Output Forms	446
Using the Form Wizard's Advanced Options	448
Modifying an Output Form in the Form Editor.	452
Displaying Several Lines Per Record.	453
Modifying Output Forms for Reports	453
Printed Columnar Reports	453
One Record Per Page Reports	454
Using Subforms	454
Report with a Text Field.	454
Custom Mailing Labels	455
Creating Mail-Merge Documents	456
Basic Steps for Creating a Printed Report	458
Reports with Breaks	460
Creating Additional Control Lines	462
Initiating Break Processing	464
Reports with Subtotals.	465
Summary Reports	467
An Example Report	468
Printing Subforms, Pictures, and Text Fields	471
Printing Labels.	475

Chapter 7 Creating Methods. 479

4 th Dimension Methods.	480
Object Methods	480
Form Methods.	481
Triggers	481
Database Methods.	483
Project methods	483
Events	484
Database Methods.	486

Triggers	486
Form and Object Events	487
Introduction to Methods.	489
Examples	491
Where to Put an Object Method.	495
Creating Methods	496
The Method Editors	496
Creating an Object Method	497
Creating a Project Method	498
Creating a Trigger	499
Creating a Form Method.	501
Renaming a Project Method	502
Setting Access Privileges	502
Making a Method Invisible	503
Opening an Existing Method	504
Opening a Database Method, a Project Method or a Trigger	504
Opening an Object Method	505
Deleting a Project Method, a Form Method or a Trigger .	506
Clearing Unwanted Object Methods	507
Using the Listing Editor	507
Syntax Display Area	508
Writing a Method	511
Opening Methods and Forms	519
Managing the Listing Editor Window	521
Navigational Keyboard Shortcuts	522
Using The Flowchart Editor	523
The Flowchart Menu	524
Creating a Flowchart Method	524
Editing a Flowchart Method	528

Chapter 8 **Creating Custom Menus 531**

Designing Menus.	531
Creating Menus	533
Basic Steps for Creating Menus	533
Creating a Menu Bar	534
Adding Menus	536
Adding Menu Commands	538
Rearranging Menus and Menu Commands	540
Assigning Methods to Menu Commands	541
Working With Instances of a Menu	543
Creating Connected Menus	543
Modifying Connected Menus	544
Deleting Connected Menus	545
Enhancing Menus	546

Adding an Icon to a Menu Command 546
 Changing Font Styles 547
 Enabling and Disabling Menu Commands 548
 Adding Separator Lines 548
 Assigning Keyboard Equivalents 550
 Deleting Menus and Menu Commands. 552
 Previewing Menus and Adding a Splash Screen. 553
 Menus and Custom Applications 554

Chapter 9 Managing Password Access. 555

Access System Overview 556
 An Access Hierarchy Scheme 560
 The Designer and the Administrator 561
 Group Owners 563
 Giving Users Design Environment Access. 563
 Giving Access to the User Environment 564
 Defining a Default User 565
 Initiating the System 566
 Assigning Users and Groups. 566
 Adding Users. 567
 Creating Access Groups 569
 Assigning Users to Groups 570
 Removing Users from Groups 571
 Nesting Groups 572
 Removing Nested Groups 573
 Saving and Loading Groups. 574
 Assigning a Group To Database Objects 575
 Assigning Access To Record Operations. 575
 Assigning a Group to a Form 577
 Assigning a Group to a Project Method. 578
 Assigning a Group to a Menu Command. 579
 Assigning a Group to a Plug-in 580
 System Maintenance. 581
 Administrator and Group Owner Access 581
 Viewing Usage. 582

Chapter 10 Creating Lists 583

Designing Lists for Data Entry. 584
 Hierarchical Lists 586
 Required and Excluded Values 587
 Non-Sequential Ranges of Values 587
 Creating Lists 588
 Adding Items to Lists 589

Deleting Items and Lists	592
Adding a Small Icon to an Item	592
Adding a Reference ID to an Item	593
Specifying Ranges in a List.	594
Sorting a List	594
Specifying Font Attributes	595
Making a Choice List Modifiable	595
Making a Hierarchical List Modifiable or Enabling Tab Control Items	596
Setting the Minimum Height of a List.	597
Dragging a List into a Form	599

Chapter 11 Using the Picture Library 601

Managing and Viewing Pictures	601
Adding Pictures.	603
Importing a Picture File	603
Pasting a Picture from the Clipboard	603
Creating a New Picture.	604
Setting Picture Properties	604
Creating and Modifying Pictures	606
Saving and Cancelling Modifications	607
Creating Thumbnails.	607
Creating a Sequence of Frames	608
Size of the Frames	608
Viewing the Frames	608
Inserting and Deleting Frames.	609
Shortcuts for Inserting Picture Buttons and Pop-up Menus	611

Chapter 12 Managing Processes. 613

Processes	614
Processes Created and Managed by 4 th Dimension	615
Time-Sliced Execution	616
Starting a New Process	616
Starting a New Process Using New Process	617
Starting a New Process Using the Menu Editor	618
Starting a New Process Using Execute Method	620
Using The Process List	621
Process Number	623
Process Name.	623
Process Status.	624
Process Time	626
Controlling Process Execution.	626
Pausing and Resuming a Process.	627

	Aborting a Process 627
	Tracing a Process. 628
	Hiding a Process 628
	Bringing a Process to the Front 629
Appendix A	Segmenting Data Files 631
	Segmenting Data Files 631
	Segmenting a New Data File. 632
	Segmenting an Existing Data File 634
	Limiting Data Segment Size 636
	Adding a Data Segment While Indexing 637
	Deleting Data Segments 637
	Deleting a New Data Segment. 638
	Deleting an Existing Data Segment 638
	Missing Data Segments 638
	Reconfiguring Data Segments 639
Appendix B	Assigning a Help File to a 4D Database 641
	Setting a Database On-line Help 641
	File Formats 641
	Assigning the Help File to the Database 642
	Creating Contextual On-line Help 642
	Calling the On-line Help from a Database 643
Index. 645

Preface

4th Dimension is a powerful relational database application and development tool.

You can use 4th Dimension to manage your own data or develop custom applications for different kinds of database management tasks.

For example, you can:

- Create a database structure of tables and fields,
- Design forms for entering, modifying, and displaying records,
- Search and sort records,
- Create reports and labels from information in the database,
- Import and export data between 4th Dimension databases and other applications,
- Publish your database on the World Wide Web.

With 4th Dimension, you can enhance conventional data management tasks with the following features:

- The powerful Form Wizard that lets you create sophisticated forms and reports with only point-and-click operations,
- A password access system to protect sensitive data,
- The capability to create custom applications with your own menus, dialog boxes, toolbars, and buttons,
- A full-featured programming language that lets you incorporate routines written in other languages.

Novice users can quickly create databases and begin managing their data. Experienced users can customize their databases with

4th Dimension's development tools. More experienced developers can use 4th Dimension's powerful programming language to add sophisticated features and capabilities to their databases, including file transfer, communications, and World Wide Web capabilities.

When you create a custom database, you can use custom menus and dialog boxes, button palettes, toolbars, and multiple windows to enhance your databases and make users more productive.

About the Manuals

The manuals described below provide a guide to the features of both 4th Dimension and 4D Server. The only exception is the *4D Server Reference* which describes features exclusive to 4D Server and is included only in the 4D Server documentation package.

Discovering 4th Dimension leads you through the process of creating 4th Dimension databases. These examples provide hands-on experience to help you become familiar with the concepts and features of 4th Dimension and 4D Server.

The *User Reference* provides a description of the User environment — the built-in environment in which you can use databases to enter and manage data.

The *Design Reference* is a reference guide to the Design environment. It provides detailed descriptions of the operations that you use to build a database. You should use it in conjunction with the other volumes in your documentation package.

The *Language Reference* is the guide to 4th Dimension language. Use this manual to learn how to customize a database by incorporating 4th Dimension commands and functions.

The *4D Server Reference* is a guide to installing 4D Server and managing multi-user databases with 4D Server. This manual is included only in the 4D Server documentation package.

About This Manual

This manual describes the 4th Dimension Design environment. The Design environment is where you create a database before you enter data into records. This manual assumes that you are familiar with basic operations such as clicking, double-clicking, and choosing a menu command. To use the *Design Reference* effectively, you should do the following:

- Use *Discovering 4th Dimension* to work through the database examples as needed.
- Begin creating a database, referring to the *Design Reference* and *Language Reference* when you need to review a topic.
- Refer to the other manuals as needed.

Cross-Platform

This manual explains the use of 4th Dimension on both the Macintosh and Windows platforms. Although the concepts and functionality of both versions of 4th Dimension are nearly identical, the manual addresses any differences where necessary. Such differences include the graphical user interface and keyboard commands.

This manual contains graphics illustrating the Windows environment. If there is a substantial difference in the appearance of the Macintosh version of a window or dialog box, it is also presented.

Chapter Descriptions

This manual is divided into the following eleven chapters:

- Chapter 1, “[4th Dimension Basics](#),” introduces basic 4th Dimension operations such as starting 4th Dimension, setting database properties, using the Design environment menus, and using the 4th Dimension Explorer. It also describes the 4th Dimension environments and provides an overview of the Design environment editors.
- Chapter 2, “[Designing a Database Structure](#),” introduces the Structure editor and explains how to create tables, fields, and related tables.
- Chapter 3, “[Creating Forms](#),” introduces the Form Wizard and explains how to create forms.
- Chapter 4, “[Form Editor Basics](#),” explains how to use the Form editor to set form properties, create and modify form objects, and set object properties.

- Chapter 5, “[Working with Fields and Active Objects](#),” explains how to set and enforce business rules for fields and enterable objects and how to add custom interface elements such as drop-down lists, hierarchical lists, tab controls, and picture buttons.
- Chapter 6, “[Output Displays and Reports](#),” explains how to create a form for printing a report. It includes an explanation of how to create subtotals and other summary calculations using methods.
- Chapter 7, “[Creating Methods](#),” introduces the two 4th Dimension Method editors, the Listing editor and the Flowchart editor, and explains how to use them to create methods.
- Chapter 8, “[Creating Custom Menus](#),” explains how to use the Menu Bar editor to create custom menus. It also explains how to use connected menus to simplify menu management.
- Chapter 9, “[Managing Password Access](#),” explains how to use the Password Access editor to create a system that controls access to tables, table operations, forms, methods, menu commands, and plug-ins.
- Chapter 10, “[Creating Lists](#),” explains how to use the List editor to create lists.
- Chapter 11, “[Using the Picture Library](#),” explains how to create and manage pictures for use as buttons, icons, and background images.
- Chapter 12, “[Working With Processes](#),” introduces the concept of multi-tasking in 4th Dimension using processes. It explains how to start a process and how to use the Process List editor to view process information and control process execution.
- Appendix A, “[Segmenting Data Files](#),” explains how structure a data file so that you can store more than 2GB of data or use more than one volume to store the data file.
- Appendix B, “[Assigning a Help File to a Database](#),” explains how to install a custom help file in a 4th Dimension database.

Conventions

All the manuals in your documentation package, including this one, use certain conventions to help you understand the material.

The following explanatory notes are used:

Note Text emphasized like this provides annotations and shortcuts that will help you use 4th Dimension more productively.

4D Server Throughout the manual, 4th Dimension and 4D Server/4D Client are referred to simply as 4th Dimension. Differences between the operation of the two products are explained in 4D Server notes which provide information about using 4D Server/4D Client. This information is provided only when the operation of 4D Server/4D Client differs from that of 4th Dimension.

Warnings like this alert you to situations where data might be lost.

In addition, names of tables in a database are shown in brackets in the text to help distinguish them from the names of fields, forms, and other items. For instance, the Companies table is written as the [Companies] table.

Using Hypertext

If you are reading this manual on-line, you will find that the entries in the table of contents and the index are hypertext buttons. Hypertext links are printed in [blue](#). Click a table of contents entry or a page number in the index to jump to that page. Throughout the body of the manual, cross-references to other sections are also hypertext buttons. Click a cross-reference to jump to that section and use Adobe Acrobat's Back button to retrace your steps.

1

4th Dimension Basics

This chapter provides basic information about 4th Dimension and the Design environment. The chapter includes the following:

- Instructions for starting 4th Dimension,
- Instructions for managing 4th Dimension desktop files,
- Instructions for backing up 4th Dimension databases,
- A description of the three 4th Dimension environments,
- An overview of the Design environment editors you use to create a database,
- Instructions for performing standard 4th Dimension operations with menus, windows, and lists,
- Instructions for using the 4th Dimension Explorer,
- Instructions for using the 4th Dimension Runtime Explorer,
- Instructions for using the Search editor of the Design environment,
- Instructions for setting database properties.

Unless otherwise noted, all instructions and explanations apply to both 4th Dimension and 4D Server.

Starting 4th Dimension

When you start 4th Dimension, you can either create a new database or open an existing one. A 4th Dimension database consists of several desktop files. On Macintosh, the database is contained in two desktop files, the *Structure file* and the *Data file*. The Structure file contains all the specifications for the database and the Data file contains your data and any indexes you create. On Windows, resource files for both the Structure and Data files are also created, making a total of four required files.

When you create a new database, you have an option to create a new folder that contains these files. When opening an existing database, you have an option to change the Data file that will be opened with the structure or create a new (blank) Data file that will be opened with the existing structure.

Note You cannot open 4th Dimension without opening a database and you cannot open more than one Data file or database at the same time.

Creating a New Database



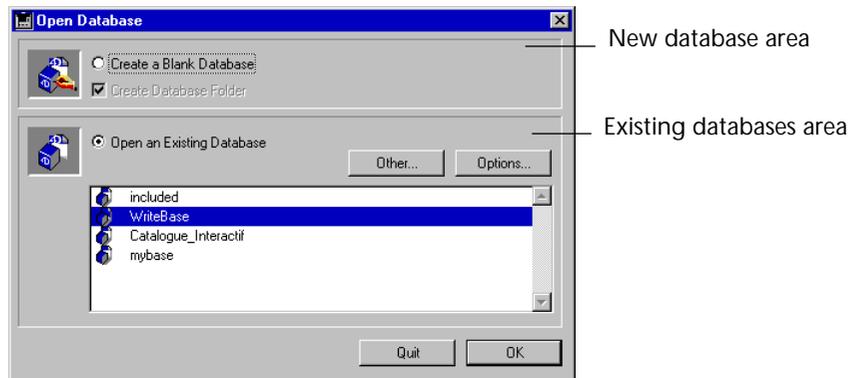
If you want to start 4th Dimension and create a new database, follow these steps:

- 1 Double-click the 4th Dimension application icon.

OR

Select the 4th Dimension application icon and choose Open from the File menu.

The Open database dialog is then displayed:



This dialog box lets you

- create a new database (i.e., a “blank” database), or

- open an existing database. Additional options allow you to check the integrity of the structure and change the data file.
- 2 Click the Create a Blank Database radio button.



If you check the Create Database Folder option, a folder will be created that has the same name as the database and the database files will be created in that folder.

- 3 Click the OK button.

4th Dimension displays a standard Save-file dialog box which allows you to enter the name and location of the database.

- 4 Enter the name of your database and select its location; then click the Save button.

You can use any name that is valid for your operating system.

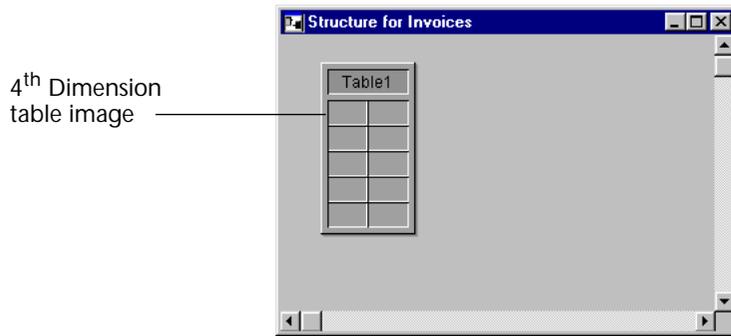
4th Dimension creates a new database with the name you specified.

4D Server For more information on creating a new database using 4D Server, refer to the *4D Server Reference manual*.

- If you selected the Create Database Folder option, the database files are saved in the new folder. The folder is located in the directory you specified in the Save-file dialog box.
- If you did not select this option, the database files will be created at the location you selected.

4D Server The files are stored on the server machine when you first create the database using 4D Server. You then modify the database design by accessing the database from any client machine using 4D Client. When you open the database from the client machine, the Structure window shown in the figure below is displayed.

4th Dimension displays the Structure window for the new database.

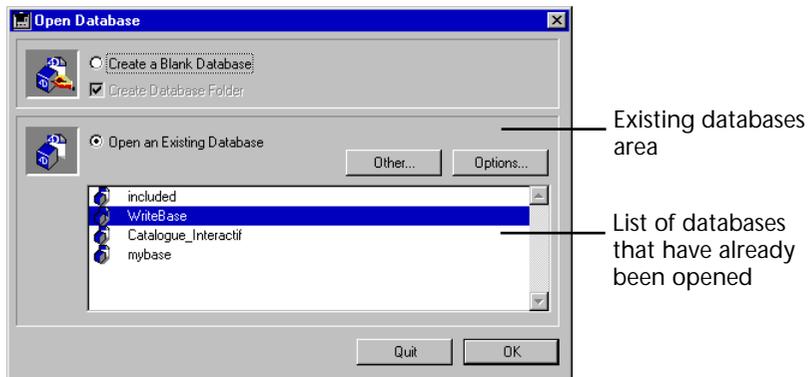


The Structure window displays the image of the first table. You can now begin adding fields to this table and creating additional tables. For detailed information about creating a database structure, see the section [“Creating a Database Structure” on page 119](#).

Opening an Existing Database

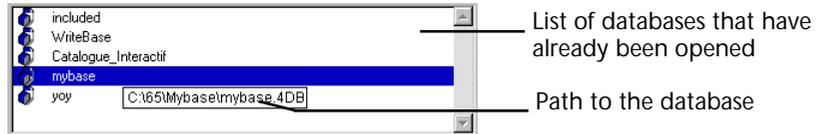
If you are starting 4th Dimension and want to open an existing database, follow these steps:

- 1 Double-click the desired structure file (On Windows, the .4DB file).
OR
Drag the structure file icon on the 4th Dimension application icon.
The database opens.
OR
- 1 Double-click the 4th Dimension application icon.
4th Dimension displays the Open Database dialog box:



4th Dimension stores the names and paths of all the databases it opens. The names of those databases are listed in the existing databases area of the Open Database dialog box. The databases are listed in chronological order, with the most recently opened database at the top of the list.

The path is displayed in a tip when you place the mouse cursor over the database currently selected in the list:



- 2 To open a database that is displayed in the list, double-click its name
OR
Select its name and click the OK button.
The database opens.

Notes To delete the reference to a database from the list, select it and press the Delete or Backspace key.
If you delete, move, or rename the database files using the file manager of your operating system, the reference to the database will be removed from the list.

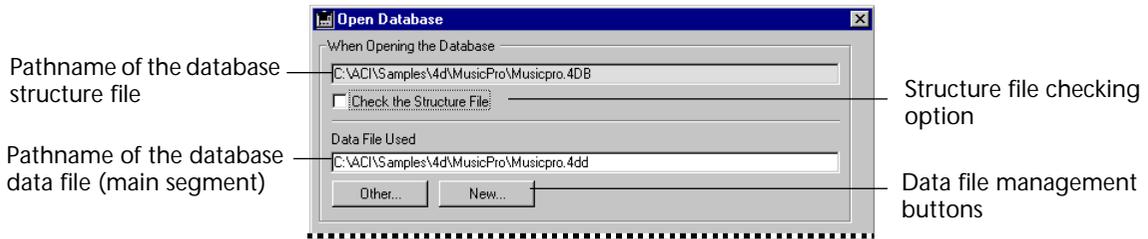
- 3 To open a database that is not available in the list, click Other.
A standard open file dialog appears, allowing you to select the structure file you want to open.

If a password is required, you are prompted to enter your password.
The database opens in the environment specified in the Database Properties dialog box. By default, the database opens in the Design environment.

4D Server For information about opening a database using 4D Client, refer to the *4D Server Reference* manual.

Options

The Options... button allows you to set additional options while opening a database. When you click this button, the following dialog box appears:



This dialog box displays the current name and access path of both the structure and data files of the database. If the data file is segmented, the information applies to the main segment.

- **Check the Structure File:** when this option is checked, the logical integrity of the structure file (tables, forms, menu bars, and so on) is checked when the database is launched. Once the check is performed, a “Check and Recover” window is displayed. That window displays the results of the tests performed.

For more information about this option, please refer to the *4D Tools Reference Guide*.

- **Other...:** clicking on this button displays a standard Open-file dialog box. You can then select a different data file that will be opened with the structure file.
- **New...:** when you click on this button, a standard Save-file dialog box is displayed. You can then create a new data file that will be opened with the structure file.

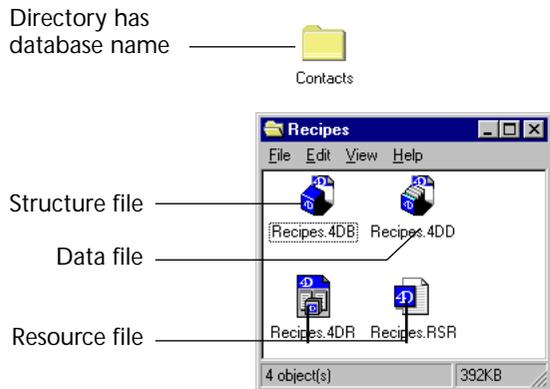
Note For more information about desktop files, refer to the next section.

4th Dimension Desktop Files

On Windows, 4th Dimension creates four files for each database: a structure file, a resource file for the structure, a resource file for the data, and a data file. They are initially placed in the same directory when the database is created. You may place these files in different locations.

The structure file contains all the specifications for the database structure (tables, fields, field properties), forms, methods, menu bars, password access groups, pictures, and lists. The structure resource file contains the Macintosh resources associated with the structure file. The data file contains the data that has been entered into records and any record-dependent information such as indexes. The data resource file contains the Macintosh resource files associated with the data file.

On Windows, the DOS name of each file has an extension that identifies whether it is a structure file, resource file, or a data file. The structure file has the same DOS name as the database, followed by “.4DB”. The structure resource file has the same DOS name as the database, followed by “.RSR”. The data resource file has the same DOS name as the database, followed by “.RSR”. The data file’s DOS name is the database name followed by “.4DD”.

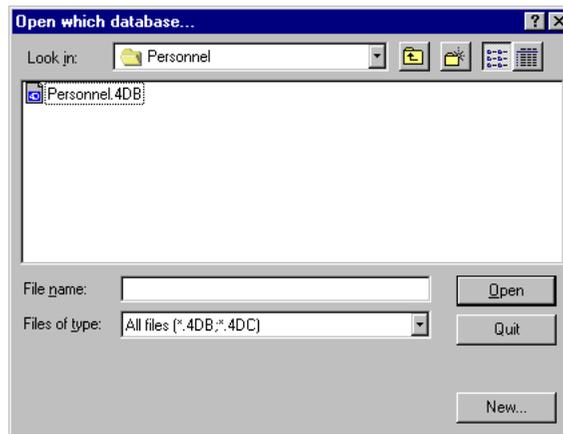


Note On Macintosh, the structure file does not have a suffix and contains both the design objects and the Macintosh resources. There is no .RSR or .4DR file on the Macintosh. The data file has the suffix “.data”. The database folder has the same name as the structure file.

In theory, you can use any data file with any structure file. The data file does not have to have the same name as the structure file, but the data contained in it must be compatible with the structure you want to use. That is, the data must fit into the fields; the number of fields in the structure must be at least as large as the number in the data file, and the data must agree with the field types.

On Windows, the .4DB and .RSR files must have the same names and be in the same directory. Otherwise, you will not be able to open the database.

When you open a 4th Dimension structure file, the application opens the data file in the directory that has the same name as the structure file followed by “.4DD”. If you change the name or location of the data file, the application will not be able to find the data file. An open-file dialog box will appear to allow you to select the data file you want to use or to create a new data file.



Selecting a different data file

When opening a database, you can choose to select a different data file since it is possible to use different data files with the same structure.

To select a different or a new data file, you can either:

- press the Alt key (Option key on Macintosh) when opening a database,
- or click the Other or New button in the Open Database dialog (for more information, refer to section [“Opening an Existing Database” on page 24](#)).

In both cases you can either create a new data file or open an existing one.

You can force 4th Dimension to allow you to specify a new or different data file by holding down the Alt key (on Windows) or Option key (on Macintosh) while opening any database to display the Open Data File dialog box. If you choose to create a new data file, 4th Dimension opens the database using the original structure but with no records.

When you use a different data file or create a new data file, 4th Dimension records the absolute path to that data file. The path specifies the location of a file. For example, if the data file “CONTACTS.4DD” is located in the CONTACTS directory in the WORK directory on the C volume, the path to the data file would be:

```
C:\WORK\CONTACTS\CONTACTS.4DD
```

Once you have used the Open Data File dialog box to locate a data file for a particular database, 4th Dimension subsequently opens that same data file in the same path, unless it finds a data file with the same name as the structure file followed by .4DD located in the same directory as the structure file. If it does not find *StructureName.4DD* in the same directory as the structure file, it then attempts to locate the data file based on the path you gave it.

If you move or rename the data file again, you will need to locate it again.

Note On Macintosh, the same data file located on the hard disk named “Hard Disk” would be accessed by the path:
Hard Disk:Work:Contacts:Contacts.data.

Linking a Data File to a Structure File

In most cases you would not want a user to be able to open a different data file. One important reason for this is that if the structure file is incompatible with the data file, the data file will be reconfigured to match the structure file. To avoid this, use the WEDD resource to lock a data file to the appropriate structure file. Locking a data file with a structure file prevents the structure file from opening a data file with a different WEDD resource. It does not, however, prevent the structure file from opening a data file without a WEDD resource. For more information about the WEDD resource, refer to “Customizer Plus” in the *4th Dimension Utilities Guide*.

Making Backups

As you work on databases, it is essential to develop a consistent method of backing up your work. In rare cases, unexpected interruptions such as a power failure or computer failure can damage a database. Although 4th Dimension can often recover your database after such damage, having a backup is necessary insurance.

After working in the Design environment, you should make a copy of the structure file (the file with the suffix “.4DB”) and the structure resource file (the file with the suffix “.RSR”). After entering or modifying data, you should make a copy of the data file (the file with the suffix “.4DD”) as well as the data resource file (the file with the suffix “.4DR”).

Note On Macintosh, the data file has the suffix “.data”.

The data file changes as new records are added and old records are modified or deleted. If a database is used infrequently, with perhaps only a few changes each day, backing up the data file once a week or even less often is probably sufficient. If a database is used often, a more structured backup system is needed.

For example, you may want to use the following system:

- ▶ To make daily backups:
 - 1 Make a backup at the end of each day. Use five separate disks, tapes, or other media — one for each day.
 - 2 At the end of the week, store the last backup permanently in a safe place.
 - 3 Reuse the first four backups from the previous week for the next week’s backups.

A backup system like the one above ensures that you always have at least five backup copies available.

You can also use the 4th Dimension plug-in, 4D Backup, to protect your database against damage. With 4D Backup, you create a backup of your database and a special log file that keeps track of changes to your database since the backup was made. If necessary, 4D Backup can restore the database to its state at a point in time before the damage occurred.

For information about creating a log file, refer to the *4th Dimension User Reference*. For information about using 4D Backup, refer to the documentation that comes with the plug-in.

The Environments

You work with 4th Dimension in three environments. Each environment is a system of editors, toolbars, windows, and pull-down menus that allow you to perform database operations. The three environments are:

- Design,
- User,
- Custom Menus.

The Design Environment

You use the Design environment to write databases. You implement all aspects of your database design in the Design environment. You use the Design environment to create tables and fields, define relations among tables, create forms for data entry, display, and printing, implement a password access system, create custom menus, attach methods to database objects, or set the database properties.

For example, you might want to keep track of information about each of the employees in a company. In the Design environment, you create an [Employees] table and add fields to that table to store employee data, such as the employee's name, job title, start date, and salary¹.

You might also add a [Departments] table that contains information about each department in the company. You could then create a *relation* between these tables that allows users to view or modify department information from an [Employees] record and view or modify the list of [Employees] in a department from a [Departments] record.

You can use the Design environment to do the following:

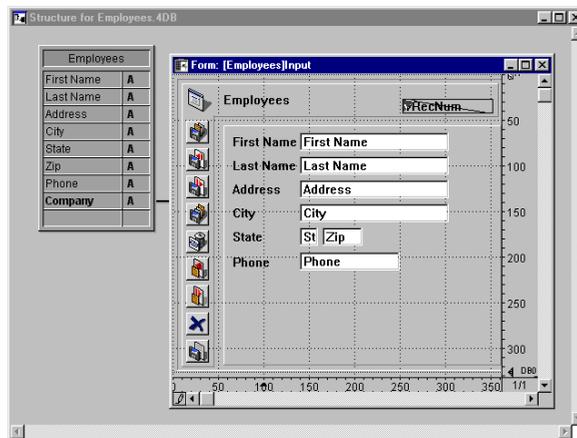
- Create tables and fields in which to store data,
- Establish relations between tables,
- Create forms for entering, displaying, and printing data,
- Create lists of choices that simplify and control data entry,
- Write methods to attach actions to database objects,

1. In the 4th Dimension documentation set, the names of tables appear in brackets. This is how table names appear in the Method editor.

- Create custom menus that use methods to automate database operations,
- Create and manage multiple processes, allowing users to perform multiple database operations at the same time,
- Specify database properties such as the default startup environment and the number of minutes between each automatic save of data,
- Set up a system of passwords to control access to information.

Each of these features is described in this manual.

The following illustration shows the Design environment being used to create a database. Complete information about each of the Design environment features is found in this manual.



From the Design environment, you can switch to the User environment to try out your database structure. To do this, choose User from the Use menu.

When you switch from the Design environment to the User environment, the Design environment is still running and the Structure window goes to the background. If you do not want to view the Structure window, choose Exit Design Environment from the File menu in the Design environment. Otherwise, the Design environment will run concurrently with the User or Custom Menus environments.

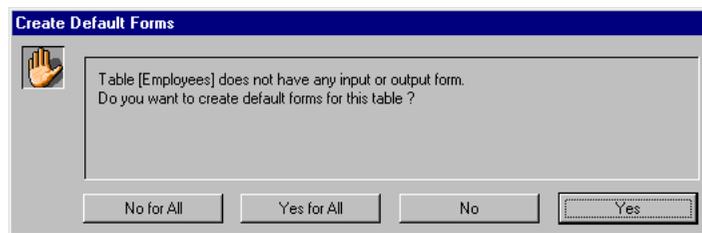
The User Environment

The User environment is a generic end-user database application that you can use to enter and manage data. You can use this environment for tasks such as entering data, searching and sorting records, importing

and exporting data, and printing reports and mailing labels. If you are in the process of developing a custom database application, you can use the User environment to enter or import a few records or test your methods before the custom database is finished.

The User environment includes generic editors that you need to import, enter, and export records, search and sort, and create reports, labels, and graphs. Also, the 4th Dimension programming language includes commands that let you incorporate the User environment's editors into your custom applications.

You can take advantage of the User environment to get a simple database up and running very quickly. To use the User environment, you only need to create your required tables and fields in the Design environment. If you switch to the User environment before you create forms for all tables, 4th Dimension will ask you whether you would like it to create them automatically.



Since the User environment has its own editors for all standard database functions, you do not have to build any of these functions yourself.

The User environment allows you to do the following:

- Enter and modify data,
- View and print data,
- Search and sort records,
- Create reports and labels,
- Create graphs,
- Import and export data,
- Execute methods,
- Start processes,
- Publish a database on the World Wide Web,

- Work with any 4th Dimension plug-ins installed in the database.

The following illustration shows the User environment being used to enter a new record.



From the User environment, you can switch to the Custom Menus environment or return to the Design environment by choosing either Custom Menus or Design from the Use menu.

Note You must have created at least one custom menu in the Design environment to be able to enter the Custom Menus environment. If you have not created a custom menu, the Custom Menus command is disabled or missing.

The Custom Menus Environment

The Custom Menus environment is the environment that you use to run a custom application — an application that uses 4th Dimension but has its own menu system and screen design. The application behaves exactly as if it were deployed and running on 4D Server and 4D Client. Use the Custom Menus environment to preview the operation of a custom application that will eventually be deployed.

Note The custom menu environment can be used only if at least one menu was created using the Menu editor.

In a custom application you control everything in the application, from the menus and forms it uses to the methods used to accept, process, and display data. You are responsible for providing menu

items and methods that manage basic tasks such as data entry and modification, searching and sorting, and reporting. You can utilize any or all of the generic tools from the User environment or create your own screens and editors¹.

The Custom Menu environment can be completely different for each application you create. From the user's standpoint, the Custom Menu environment is a complete application for a specific kind of information management. The following illustration shows an invoicing application running in the Custom Menu environment. It uses a floating palette that gives the user access to each of the application's modules. Each module runs in a separate process with its own menu system. The user can open as many modules as needed and move among them simply by clicking on their windows.



From the Custom Menu environment, you can return to the User environment by choosing Quit from the File menu.

After entering the User environment, you can return to the Design environment by choosing Design from the Use menu.

1. You cannot use the User Environment's Import and Export data dialog boxes in custom applications.

Changing Environments

As you have seen, you can change from one environment to another using the Use menu. A check mark in the menu indicates which environment you are in.

With the Use menu, you can move between the Design and User environments at any time. Once you have created a custom menu, you can enter the Custom Menus environment from the User environment. If no custom menu exists, Custom Menus in the Use menu is disabled (or does not appear at all).

If you have windows from more than one environment open at the same time, you can switch between environments by clicking their windows. When you click a window, 4th Dimension brings the appropriate window to the front of the screen and makes it the active environment. In this manner, you can switch directly from the Custom Menus to the Design environment without having to enter the User environment first.

Design Environment Editors

You use 4th Dimension editors to create and modify the various components of your database design. Each editor is dedicated to one aspect of a design.

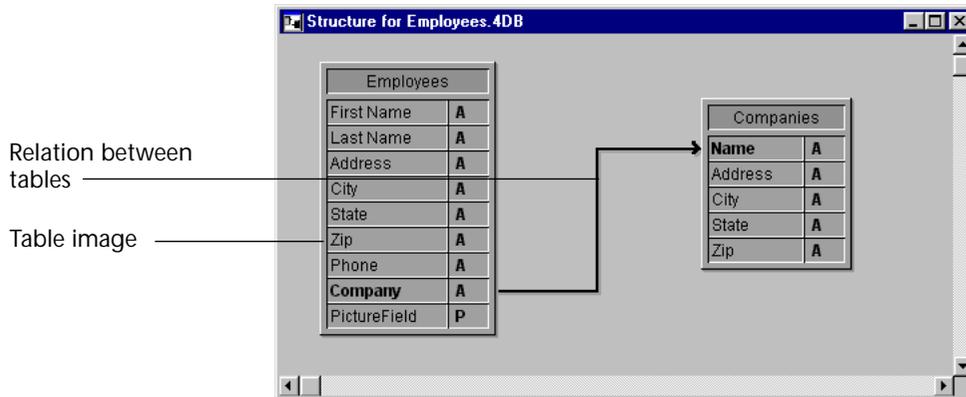
The Design environment contains the following editors:

- Structure editor,
- Form editor,
- Method editors,
- Menu Bar editor,
- Password Access editor,
- List editor,
- Process list,
- Break list,
- Picture library.

Each editor has its own window with appropriate tools, menu commands, and toolbars.

The Structure Editor The Structure editor is your starting point for all design operations. Use the Structure editor to create tables, fields, and relations between tables.

The Structure editor displays the images of the tables in the database and graphically shows the relations (if any) among the tables. The following illustration shows the Structure editor window.



Use the Structure editor to:

- Create tables and subtables in a database,
- Specify table properties,
- Create fields and subfields,
- Specify field types and properties,
- Relate tables and set relation properties,
- Establish access privileges to tables and fields,
- View the structure of a database.

Refer to [Chapter 2](#) for more information about using the Structure editor.

The Form Editor

You use the Form Wizard to create forms and the Form editor to modify forms — both on-screen forms and printed reports. The Form editor lets you create sophisticated forms for managing data.

Use the Form editor to:

- Add fields and variables to a form,

- Add buttons, combo boxes, drop-down lists, tab controls, and other interface objects to forms,
- Specify display formats and entry filters for data displayed and entered on the form,
- Add methods that enforce business rules during data entry or manage interface elements,
- Add graphic objects to the form — including text, lines, rectangles, and ovals — or paste in pictures or graphics that were digitized or created in another application,
- Specify fonts and font styles for objects containing text,
- Specify drag and drop actions,
- Specify automatic resizing and repositioning actions,
- Establish access privileges for forms.

Refer to [Chapter 3](#) and [Chapter 4](#) for a detailed discussion of how to create a form with the Form Wizard and how to modify forms with the Form editor. Refer to [Chapter 5](#) for information about using the Form editor to control data entry and add interface elements to data entry forms. Refer to [Chapter 6](#) for information about using the Form editor to design output forms, reports, mail-merge documents, and forms for mailing labels.

The Method Editors

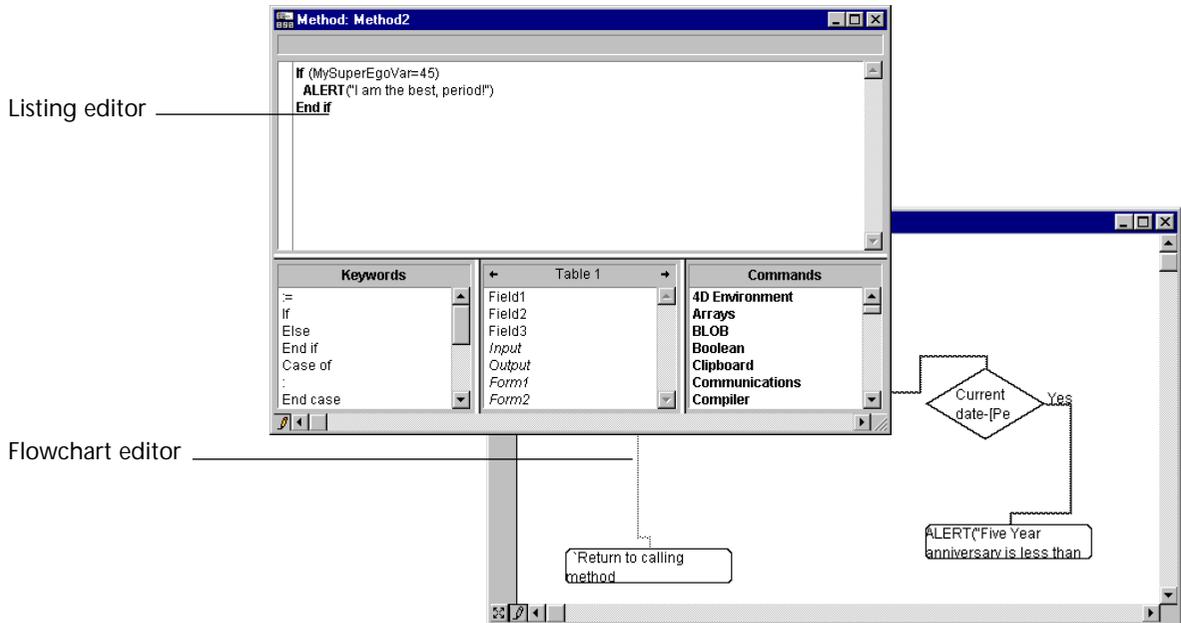
Methods are instructions that process data or perform actions. Methods can perform tasks such as calculating the values of fields or validating data as it is entered. Methods also perform actions such as displaying custom dialog boxes, starting processes, searching, sorting, displaying, or printing records.

4th Dimension provides a Pascal-like language for writing methods, two different Method editors, and a library of commands and functions.

The two Method editors are the Listing editor and the Flowchart editor. Either editor can be used to create methods. However, it is recommended that you use the Listing editor since only methods created with the Listing editor can be compiled. [Chapter 7](#) describes the use of each editor.

Note Except in sections that compare the Listing and Flowchart editors, this documentation set uses the term “Method editor” to refer to the Listing editor.

The figure below shows both editors being used to create methods.

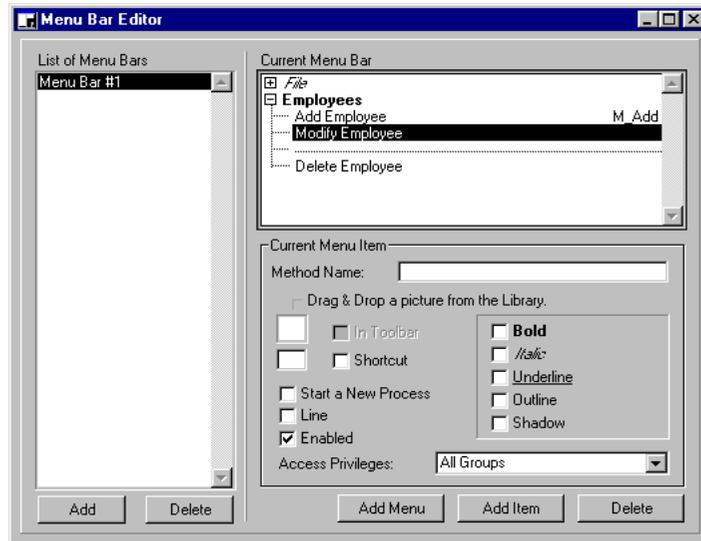


You can use either Method editor to:

- Write database methods that run automatically when particular worksession-related events occur,
- Write triggers that run automatically when particular database engine events occur,
- Create form methods that run automatically when a form is used,
- Create project methods that can be attached to custom menus, called by other methods, or executed by users in the User environment,
- Create object methods that are associated with fields or other form objects.

Refer to [Chapter 7](#) for overviews of each type of method and the Method editors. See the *4th Dimension Language Reference* for detailed information about writing methods.

The Menu Bar Editor When you create custom applications with 4th Dimension, you use the Menu Bar editor to create menu bars, menus, and menu commands. You can also attach menus to any form that you use for data entry. The figure below shows the Menu Bar editor being used to create a menu bar.



Use the Menu Bar editor to:

- Create menu bars,
- Create and modify custom menus and menu commands,
- Attach icons to menu commands to create a custom toolbar,
- Assign project methods to menu commands,
- Create connected menus that allow you to use the same menu definition in several menu bars,
- Preview the menus and menu bars as they will appear in the custom application,
- Include graphics for splash screens that display with menu bars,
- Establish access rights to menu commands,
- Set keyboard equivalents for menu commands,
- Enable or disable menu commands,
- Start a new process from a menu command.

Refer to [Chapter 8](#) for a detailed discussion of adding custom menus and menu bars to your applications.

The Password Access Editor

4th Dimension lets you add passwords so that you can control access to aspects of your databases. The figure below shows the Password Access editor.



Use the Password Access editor to:

- Create users and give them passwords,
- Allow designated people to add users and change user passwords,
- Place users in groups,
- Give groups access to parts of the database, such as the Design environment and particular forms, menu commands, methods, and plug-ins,
- Monitor database use by individual users.

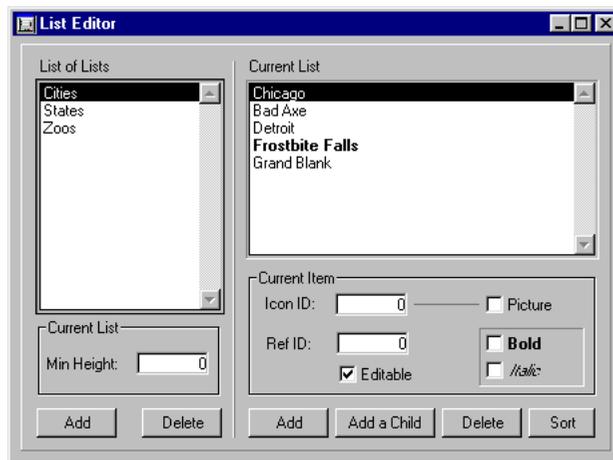
Refer to [Chapter 9](#) for a detailed discussion of the Password Access editor.

The List Editor

You use the List editor to create lists. Lists can be used for several purposes in a database. Here are the most common uses:

- You can attach a list to a field. A user can select an entry from a choice list instead of typing it. With a choice list, you prevent entry of misspelled words and incorrect data.
- You can use a list to specify the items in pop-up menus, drop-down list boxes, combo boxes, scrollable areas, or tab controls.
- You can create hierarchical lists to populate hierarchical lists, hierarchical pop-up menus, or tab controls. These objects are then automatically defined using the list values.
- You can access the elements of a list in your methods or transfer the elements of an array to a list (and vice versa). By doing that, you can use a list to populate pop-up menus, drop-down list boxes, combo boxes, scrollable areas, or tab controls.

The figure below shows the List editor being used to create a list.



Use the List editor to do the following:

- Create choice lists,
- Add items to a choice list,
- Attach small icons to list items,
- Delete choice lists or individual items in a list,
- Sort choices in a list,

- Activate values for use in tab controls,
- Make values editable for use in a hierarchical list,
- Make a choice list modifiable by the user.

Refer to [Chapter 10](#) for a detailed discussion of lists.

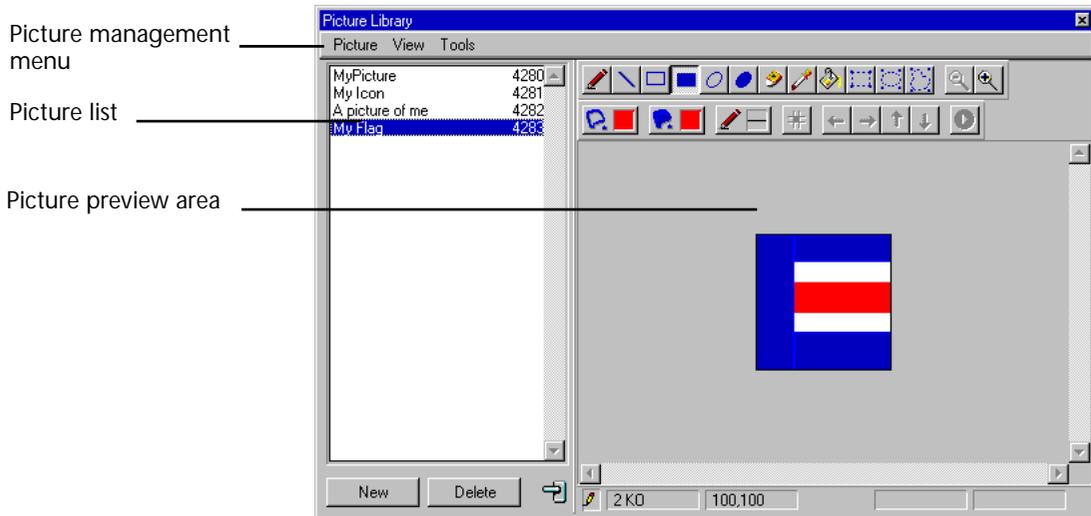
Picture Library

Use the Picture library to store graphics that you can use as design elements in forms, as toolbar or list icons, picture menu items, or picture buttons. With the Picture library, you can use a graphic in several places in your database but you need to store it in only one place. When you update an image in the Picture library, all references to the image are updated automatically. This feature can reduce the size of your Structure files and make changes to the database easier to manage.

The Picture library also includes a 2D Paint editor that allows you to create or retouch pictures. It is an ideal environment to design buttons and icons.

In addition to the features mentioned above, the Picture library also includes integrated functions that can create or edit row by column arrays of pictures that are used to create picture buttons or picture menus.

The Picture library is shown in the following illustration.



The Picture library is used to:

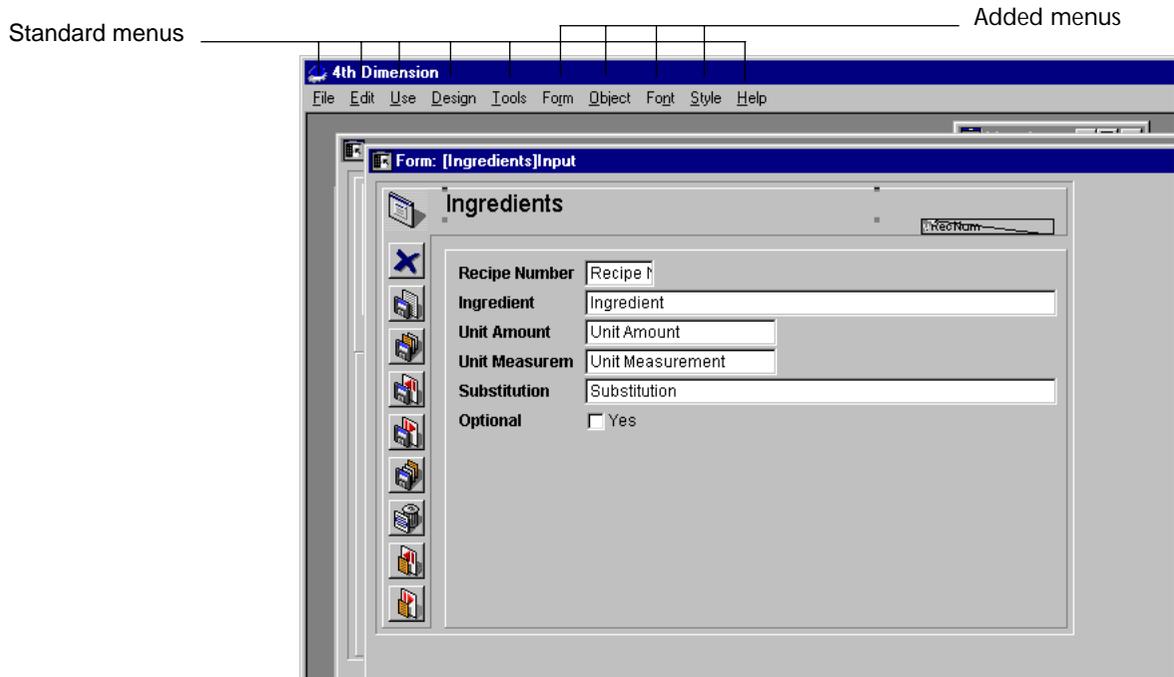
- Store and preview the pictures of the database structure,
- Create or import new pictures,
- Retouch pictures,
- Define and preview tables of thumbnails,
- Drag and drop pictures to the Form editor, the List editor, or the Menu bar editor,
- Delete pictures.

For more information about the Picture library, refer to [Chapter 11](#).

For more information on inserting a picture in the Form editor, refer to [“Placing a Picture from the Picture Library” on page 309](#), [“Picture Pop-up Menus” on page 403](#), [“Picture Buttons” on page 390](#), [“Adding an Icon to a Menu Command” on page 546](#), and [“Adding a Small Icon to an Item” on page 592](#).

The Design Environment Interface

In the Design environment, you interact with 4th Dimension using its menus, context-sensitive menus, toolbars, and windows. This section describes how to use these interface elements.



When several editor windows are open, the menus belonging to the frontmost editor window appear. You choose menu commands from these menus as you would in any application. For complete instructions for choosing from a menu, see the documentation that came with your computer.

The Design Environment Menus

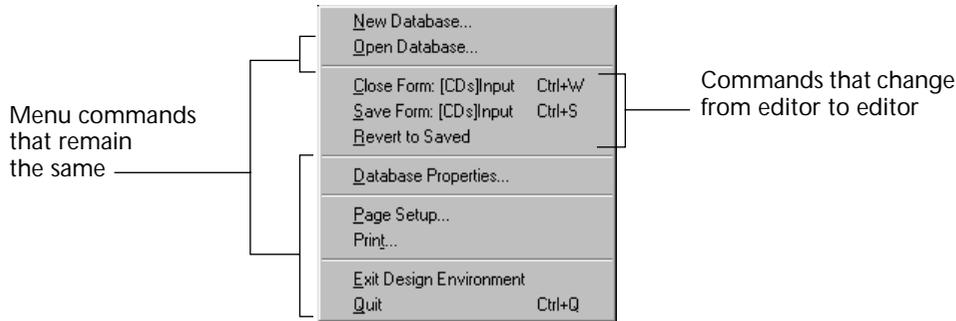
The Design environment has five permanent menus:

- File
- Edit
- Use
- Design
- Tools

Additional menus are active for different editors.

The File Menu

In the design environment, the File menu provides standard file operations.



These File menu commands remain the same for every editor:

- **Open Database** You can open an existing database at any time. 4th Dimension automatically saves changes to the current database before opening the next database.
- **Database Properties** You can set database properties at any time. For complete information, see the section [“Setting Database Properties” on page 82](#).
- **Page Setup** You can set the specifications for printing at any time. Page setup information is stored with each form.
- **Print** You can print the contents of any editor window at any time.
- **Exit Design Environment** You can put away the Design environment at any time. Exit Design Environment puts away all the Design environment windows and switches to the User environment. From the User environment, you can return to the Design environment by choosing Design from the User menu.
- **Quit** You can quit 4th Dimension at any time. 4th Dimension automatically saves your work before quitting.

These File menu commands change depending on the active editor:

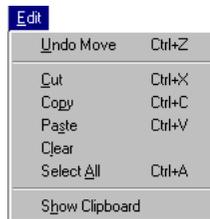
- **Close *EditorName*** You can close editor windows at any time. 4th Dimension saves the contents of each window before closing it. If the Structure window is frontmost, choosing Close Structure closes the Structure window. If there are no other Design environment windows open, it also switches to the User environment. If there are other

Design environment windows open, it only closes the Structure window. To close all Design environment windows, use Exit Design Environment.

- **Save *EditorName*** You can save the contents of an editor's window at any time without closing the window. 4th Dimension automatically saves the contents of an editor when you close the window, change to a new environment, or exit the application.
- **Revert to Saved** You can revert to the last saved version of a form or method. This menu command replaces the contents of the Forms or Method editor with the last version saved.

The Edit Menu

The Edit menu provides standard editing operations.



These are the menu commands provided by the Edit menu:

- **Undo** Use Undo to go back one step while working with an editor. This menu command is useful when you make a mistake and want to redo something.
- **Cut, Copy, Paste** You can select something on the screen and either cut or copy it. In either case, a copy of the selected object is placed on the Clipboard. You can then paste the object into a new location in the same window or in another window.
- **Clear** Use Clear to erase a selected object. No copy is placed on the Clipboard.
- **Select All** Use Select All to select every object in the current editor. For example, use Select All before adjusting all the elements of a form.
- **Show Clipboard** You can view the current contents of the Clipboard at any time. Sometimes you will want to view the Clipboard prior to pasting its contents.

The Use Menu

The Use menu lets you switch environments.

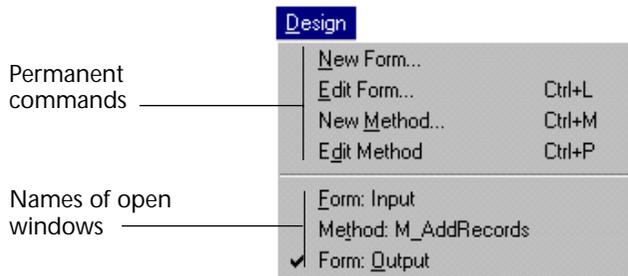


A check mark appears to the left of the current environment. To switch to another environment, choose the environment from the Use menu.

The Custom Menus environment is not available until you create at least one custom menu.

The Design Menu

The Design menu has a divider that separates the permanent menu commands from the list of open windows:



The four permanent commands let you create or edit forms and methods:

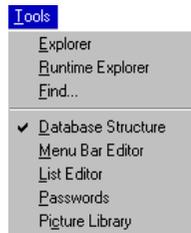
- **New Form** Opens the Form Wizard, ready for you to create a new form. For information about creating new forms with the Form Wizard, see [“Creating a New Form” on page 197](#).
- **Edit Form** Opens the Forms page of the Explorer, ready for you select a form to open. For information about opening forms, see the section [“Forms Page” on page 60](#).
- **New Method** Opens a blank Method editor window, ready for you to create a new project method¹. For information on creating a new method, see the section [“Using the Listing Editor” on page 507](#).
- **Edit Method** Opens the Methods page of the Explorer, ready for you to open an existing method. For information on opening methods, see the section [“Methods Page” on page 62](#).

1. If you have not chosen a default Method editor, the New Method command displays a dialog box in which you can choose a Method editor and name the method. For information on choosing a default Method editor, see the section [“Design Environment” on page 90](#).

The menu commands below the divider bring a Design environment window to the front. This list depends on the number and names of the open Design environment windows. A check mark indicates the active window.

The Tools menu

The Tools menu provides access to the Explorer and the various Design environment editors:



The Explorer menu command displays the Explorer. For information about the Explorer, see the section [“The Explorer” on page 54](#).

The Runtime Explorer command displays the Runtime Explorer window. For more information about the Runtime Explorer window refer to [“Runtime Explorer” on page 75](#).

The Find command displays the Find window which allows you to search for objects in the structure. For more information about structure searches, refer to [“Find Dialog Box” on page 79](#).

The middle set of menu commands opens the corresponding editor windows. The bottom set of menu commands opens the Process list and the Break list. For information on each editor, see the following chapters or sections:

Editor	Reference
Structure editor	Chapter 2
Menu Bar editor	Chapter 8
List editor	Chapter 10
Password editor	Chapter 9
Picture Library	Chapter 11

Contextual menus

In most Design environment editors, you can use contextual menus to execute specific actions on objects or to open dialog boxes.

► To use a contextual menu:

- 1 On Windows, click on an object or area with the right mouse button. On MacOS, Control-click on an object or area.

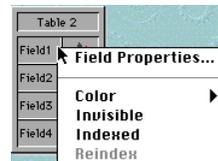
The contextual menu associated to the object or area appears.

Contextual menu associated with the fields in the Structure window

Windows



MacOS



Note On MacOS, when you press the Control key, the pointer changes to a contextual menu pointer , indicating that a contextual menu is available.

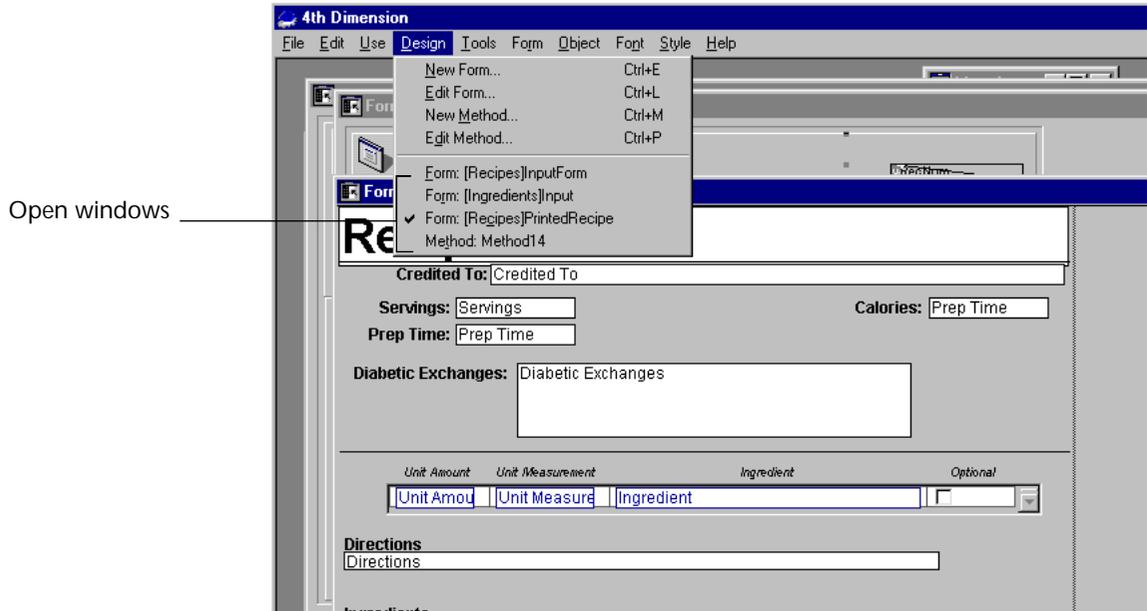
- 2 Select a command in the contextual menu, as you would for any other menu.

Commands in contextual menus vary according to the current editor as well as the object on which you clicked. As its name indicates, a contextual menu only includes menu items appropriate for a specific context. The contextual menu items perform functions that are identical to those in the standard menus, buttons, and keyboard shortcuts.

4th Dimension Editor Windows

Each 4th Dimension editor is displayed in a separate window. There may be several editor windows open at once. You can have more than one editor open at one time and some of the editors can display several

windows at the same time. A list of open windows is displayed at the bottom of the **Design** menu.



You can move between the open windows as you work. Only one editor window is active at any time. The active editor determines which menus are added to the right of the menu bar.

To make a window active, you can click anywhere in the window or you can choose the window's name from the **Design** menu. You can move a window anywhere on the screen by dragging its title bar.

You can expand the window to full-screen size by clicking the maximize box in the upper-right corner of the window. You can make the window any size you want by dragging the Size box in the lower-right corner of the window. On Windows, you can drag the edge of a window to resize it.

You close a window by clicking the Close box (on Macintosh) or double-clicking the Control-menu box in the upper left corner of the window (on Windows), or by choosing **Close *EditorName*** from the File menu.

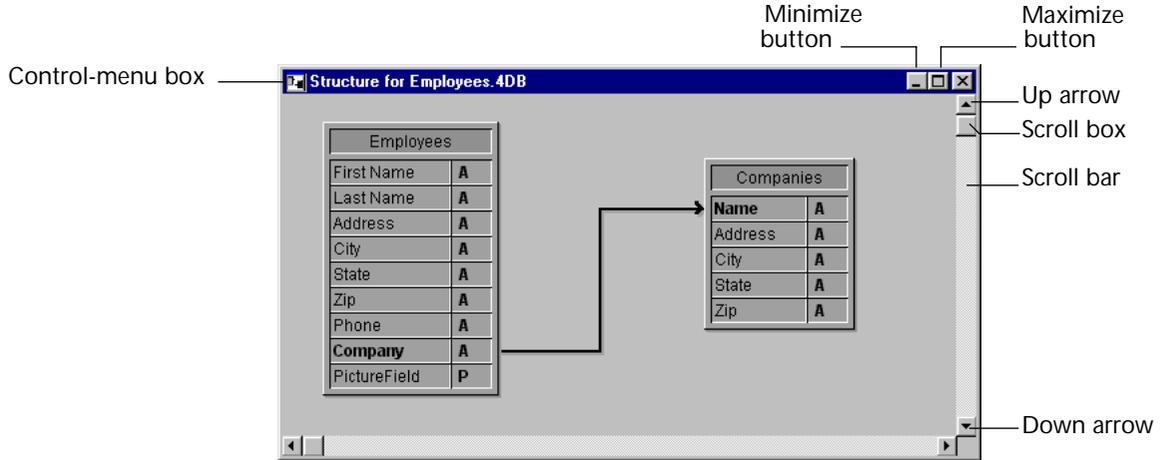
To close all the open windows of the Design environment, choose **Exit Design Environment** from the File menu.

To close all the Design environment windows except the structure window, press the **Alt** key while double-clicking the Control-menu box

(on Windows) or clicking the Close box (on Mac OS) of the frontmost editor window.

Scrolling

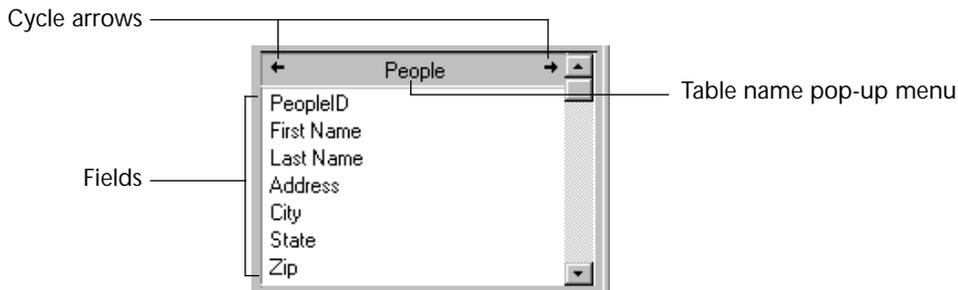
As you build a database, the contents of some of the editors can become so large that it is difficult to view the entire structure or form.



You scroll the window with the scroll bars as you would in any application.

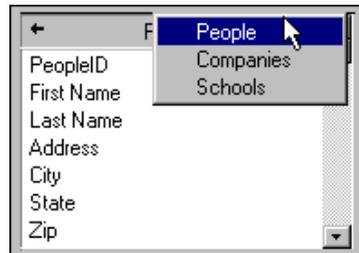
Displaying a Different Table

In some editors, such as the Method editor, 4th Dimension allows you to change which table is displayed in two ways: by clicking either of the cycle arrows on the table name bar or by choosing a table from a pop-up menu. The following figure shows the list of fields in the Method editor. You can use either the cycle arrows or the table pop-up menu to display the fields from a different table.



Click either arrow to cycle through the tables in the database. 4th Dimension displays the tables in the order in which they were created. When you cycle through all the tables, the first table is displayed again.

You can also change tables using the pop-up menu. If you click on the table name between the cycle arrows and hold the mouse button down for a moment, a pop-up menu of all the tables appears.



You can then choose one of the table names to list the fields in that table.

The Toolbar

The Design environment displays a toolbar just below the standard menubar. The toolbar provides convenient access to all the Design environment menu commands.

The Toolbar contains buttons for all the standard menu items. The buttons are arranged in groups, with each group corresponding to a menu. Each Design environment editor has its own toolbar; the toolbar buttons correspond to the editor's menu commands. Toolbar buttons are grouped by menu and appear in the same left-to-right order as the menus. When you are working with an editor window that adds commands to the standard Design environment menu bar, the corresponding buttons appear in the toolbar to the right of the standard toolbar buttons. The following illustration shows a Design environment toolbar:



To obtain help for a button, position the pointer over a button. The following illustration shows help for a Structure menu button.



If you don't want to use the toolbar, you can hide it by deselecting the Show Toolbar database property in the General Page of Database Properties. For more information, see [“General” on page 83](#).

When you create custom menus, you can create custom toolbars and add buttons corresponding to your custom menu items. For information on creating custom toolbars, see the section “[Adding an Icon to a Menu Command](#)” on page 546.

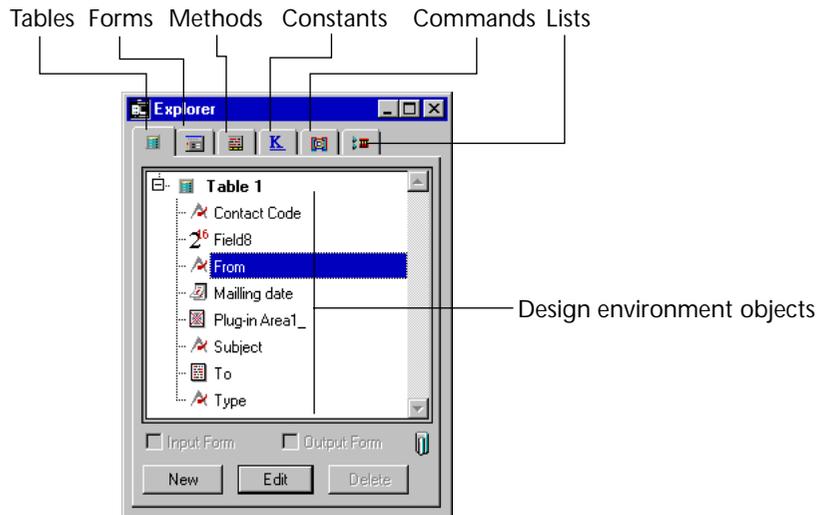
The Explorer

The Explorer is a window that gives you convenient access to tables and fields, forms, methods, constants, built-in 4th Dimension commands, and lists. You can display the Explorer at any time by:

- Choosing Explorer from the Tools menu,
- Pressing **Ctrl+Space** bar (on Windows) or **Command-Space** bar (on Macintosh).

Working with the Explorer

The tabs at the top of the Explorer let you access different groups of Design environment objects. The Explorer has separate pages for tables, forms, methods, constants, commands, and lists.



Displaying Different Explorer Pages

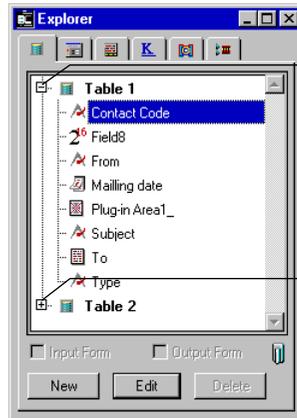
Click a tab to display either the Tables, Forms, Methods, Constants, Commands, or Lists page. When you display a particular page, the appropriate Design environment objects are listed in the Explorer window. On most pages, the objects are displayed as a hierarchical list.

Expanding and Collapsing Hierarchical Lists

You can expand an object in the list by clicking the plus sign (on Windows) or the triangle (on Macintosh) to its left or by highlighting the object and pressing the right arrow key. You collapse an expanded item clicking its downward-pointing triangle (on Macintosh) or minus sign (on Windows) or by highlighting it and pressing the left arrow key. You can also expand or collapse an object by double-clicking the object.

The following illustration shows both expanded and collapsed topics.

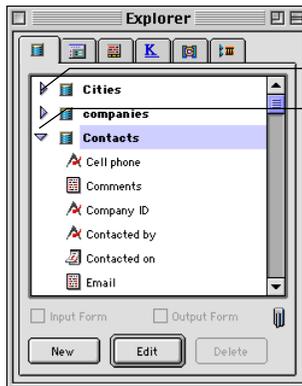
Windows



Click a minus sign to collapse a topic

Click a plus sign to expand a topic

Macintosh



Click an arrow to expand a topic

Click a downward pointing arrow to collapse a topic

Renaming a Form or Method

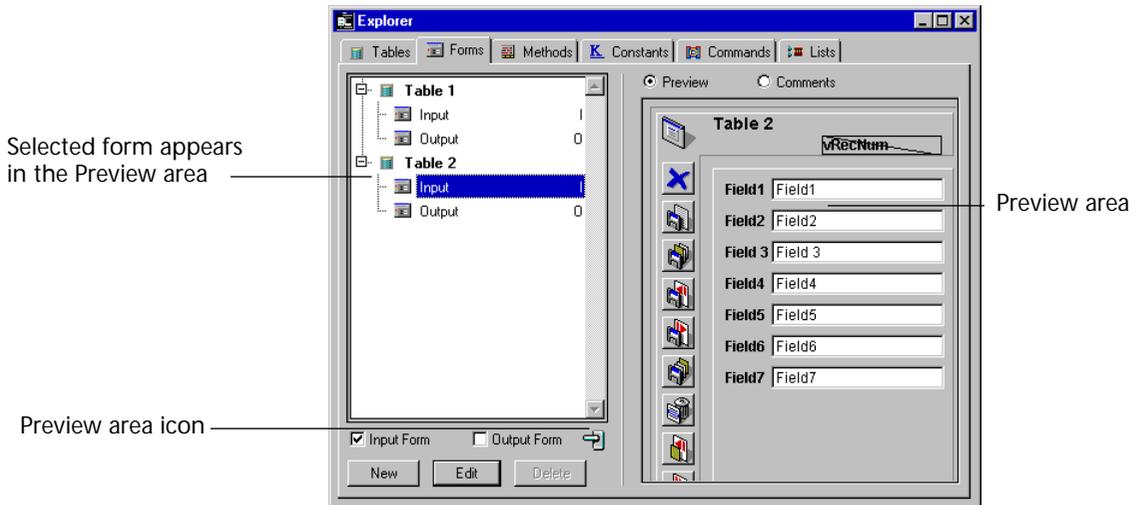
If you need to rename a form or method, hold down the Command key (on Macintosh) or the Ctrl key (on Windows) and click the name of the form or method. The text becomes editable. Make your changes and then click anywhere outside the text area to save your changes.

The Explorer always lists forms and methods alphabetically. If the new name changes the sort order, 4th Dimension will resort the list when you click outside the entry area.

Displaying and Hiding the Preview area

You can display or hide the Explorer preview area by clicking the Preview area icon . The Preview window lets you preview table images, forms, and methods and provides additional information about commands and constants. You also can use the Preview area to enter and read comments about certain database objects. See the section, “Using Comments” on page 68 for more information.

The following illustration shows a form being previewed.



To hide the Preview area, click the Preview area icon again.

Note When the Preview area is shown, labels are added to the tabs. The labels automatically disappear when you hide the Preview area.

Resizing the Explorer Window

You can resize the Explorer by dragging the lower-right corner of the window.

Note If the Preview area is not displayed, you cannot display it by resizing the window. You must use the Preview area button.

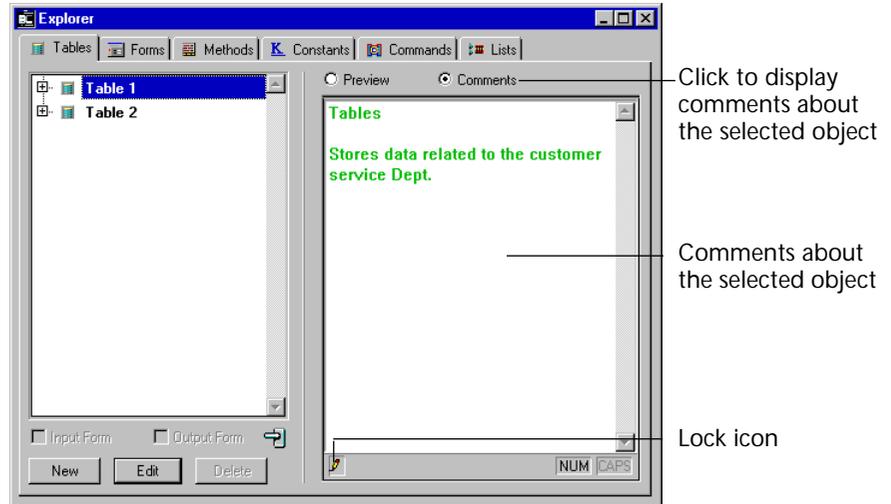
If the Preview area is not displayed, you can only expand the Explorer window vertically. If the Preview area is displayed, you can expand the Explorer window both horizontally and vertically.

Displaying Comments

4th Dimension allows you to assign comments to the following database objects: methods (object methods, database methods, project

methods, and triggers), forms, tables, and fields. You enter comments into the Preview area of the Explorer.

To display an object's comments, select the object in the object list and click the Comments button located above the Preview area.



The use of comments is described in detail in the section [“Using Comments” on page 68](#).

Using the Delete, New, and Edit buttons

The Delete, New, and Edit buttons below the list can be used to create, modify, or remove Design environment objects. These buttons are disabled automatically whenever an action is not possible. For more information on using these buttons, refer to the section on the appropriate Explorer page.

Using Drag and Drop

In many instances, you can use drag and drop to add a database object to an editor window. For example, you can add a field to a form by dragging a field name from the Tables page of the Explorer to an open form in the Form editor. When you are working with the Method editor, you can add the names of tables, forms, fields, project methods, constants, and commands as well as their syntax to a method using drag and drop.

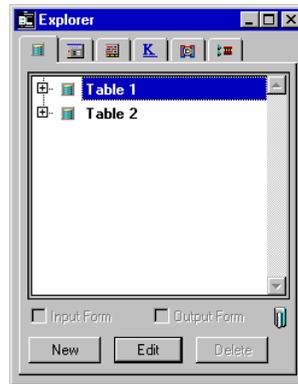
The sections that describe each Explorer page give specific information on the drag and drop options for that page.

Each Explorer page is described briefly in the following sections. More detailed information is presented in the chapter that deals with the appropriate topic.

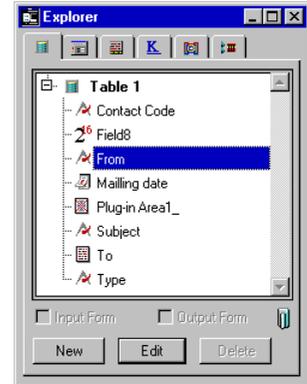
Tables Page

The Tables page lists all the tables and fields in the database. It can be used as an alternative to the Structure window to access table and field properties. When a table is expanded, the fields in the table are shown.

Collapsed View



Expanded View



The field type is indicated by an icon or a letter. Double-click a field name to open the Field Properties window. For more information on field properties, see [“Creating Fields and Setting Field Properties” on page 131](#).

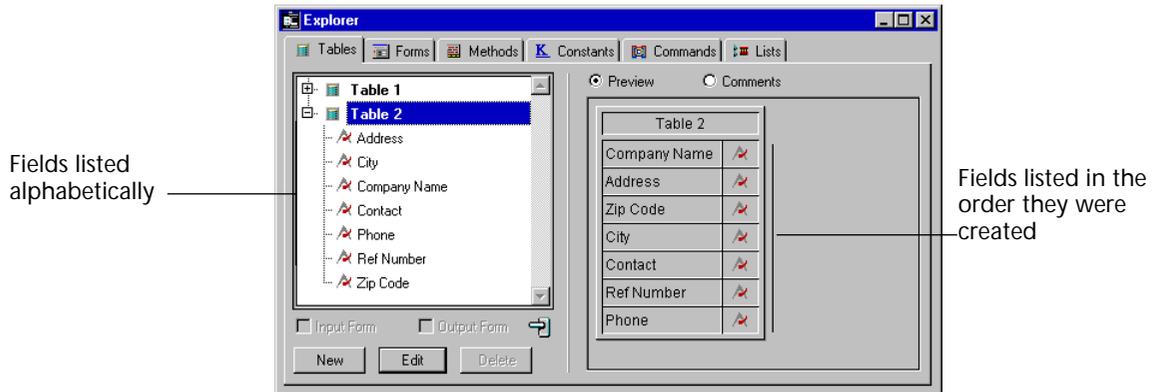
Using Drag and Drop

You can add a field to a form by dragging the field name from the Tables page onto the form. You can add either a table name or a field name to a method by dragging the name to the method. When you do so, the name appears using the correct syntax. For example, if you drag the field “First Name” in a [Customers] table, it appears in the Method editor as “[Customers]First Name”.

You can create a relation between two tables by dragging a field from the Tables page of the Explorer to the field that uniquely identifies the other table in the relation. For more information, see the section [“Creating a Relation Between Tables” on page 165](#).

Viewing a Table Image You can bring a a table image in the Structure editor window into view by double-clicking the table name in the Explorer. When you double-click a table name, 4th Dimension centers the table image you clicked in the Structure editor window. This feature is useful if you have a large structure and would otherwise need to scroll the Structure editor window to view a particular table image.

Previewing a Table Image You can also preview the table image for a selected table. To preview a table image, highlight a table and click the Preview area icon  (if the Preview area is not already displayed). The following illustration shows a table image being previewed.

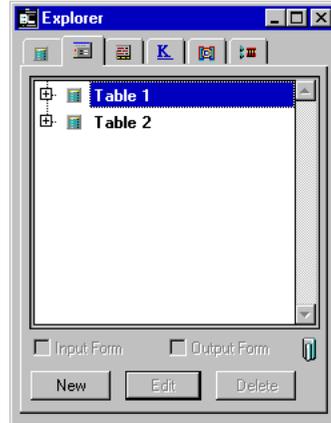


Setting Table Properties You can view or modify a table's properties from the Tables page. Double-click a table or highlight a table in the hierarchical list and click Edit. The Table Properties window for the selected table appears. For information on setting table properties, see the section [“Setting Table Properties”](#) on page 125.

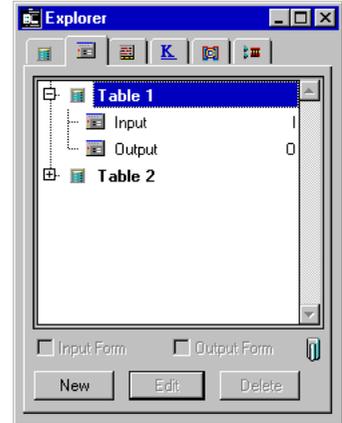
Forms Page

The Forms page lists all the tables and forms in the database. When a table is expanded, the forms for that table are shown:

Collapsed View



Expanded View



The Forms page is displayed automatically when you choose **Edit Form** from the **Design** menu.

Creating a New Form

To create a new form, highlight the desired table and click **New**. The Form Wizard appears, ready for you to create the new form. For more information about using the Form Wizard, see [“Creating a New Form” on page 197](#).

Note You can also create a new form by choosing **New Form** from the **Design** menu.

Editing a Form

To edit an existing form, expand the table containing the form, highlight the desired form, and double-click the form or highlight it and click **Edit**. You edit forms using the Form editor. For information about editing forms, see [Chapter 4, “Form Editor Basics” on page 229](#) and [Chapter 5, “Working with Fields and Active Objects” on page 325](#).

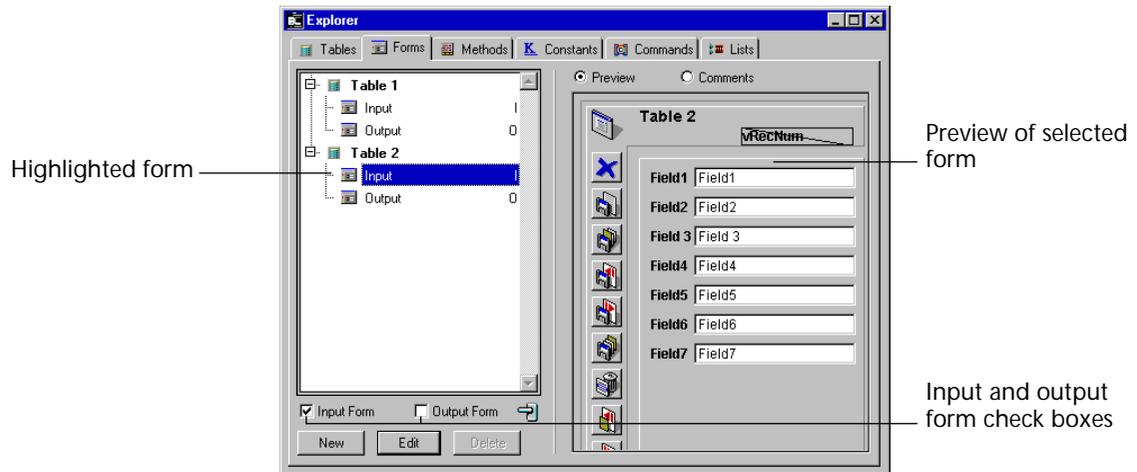
Deleting an Existing Form

To delete an existing form, highlight the undesired form and click **Delete**.

Note You cannot delete a form if it is either the current default input or output form for the table.

Previewing a Form

To display the Preview area, click the Preview area icon . Highlight a Form to preview it:



If you highlight a Table name in the Forms page, a preview of the table image from the Structure editor will appear in the Preview area.

Designating a Form as the Current Input or Output Form

When the Preview area is shown, you can select the default input and output forms. The default forms are used when you switch to the User environment. Click the desired form name in the hierarchical list and then click either the Input Form or Output Form check box under the form list. For information on default input and output forms, see [“Setting the Current Input and Output Forms” on page 225](#).

Using Drag and Drop

You can add a form name to a method by dragging. When you do so, the form name appears using the correct syntax. For example, if you drag the form “Input” in the [Companies] table, it will appear in a method as “[Companies]Input”.

You can add a subform to another form by dragging the name of the List form from the Forms page of the Explorer to the open form in the Form editor. You can add a Detail subform by holding down the Shift key and dragging the name of an input form from the Forms page of the Explorer to the subform area on the form. For more information, see the section [“Adding a Subform to the Form” on page 428](#).

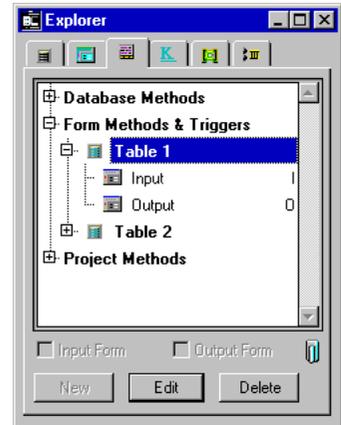
Methods Page

The Methods page lists the database, project, table (trigger), and form methods for the database. The three types of methods are grouped into the categories, Database, Project, and Trigger/Form.

Collapsed View



Expanded View



The Method page is automatically selected when you select **Edit Method** from the **Design** menu.

Creating a New Method Here is a description of how to create each type of method.

- **Project method** To create a new Project method, highlight the Project Methods item in the hierarchical list and click the New button.

Note You can also create a new project method by choosing **New Method** from the **Design** menu.

- **Trigger** To create a Trigger, expand the Form Methods & Triggers element in the hierarchical list and then highlight the desired table. Click the New button.
- **Form method** To create a Form method, expand the Form Methods & Triggers element in the hierarchical list, expand the desired table, and then highlight the desired form. Click the New button. You can also create a form method by double-clicking the form in the Method page of the explorer window.

Note You can also create a new form method by choosing **Form Method** from the **Form** menu that appears when the Form editor is active.

- **Database method** You cannot create new Database methods. Instead, you add code to an existing blank Database methods. To do so, expand

the Database method item and double-click the desired Database method or highlight it and click the Edit button.

If you have selected a default Method editor in the Database Properties¹ dialog box, a new untitled Method opens.

Note Even if you have selected a default method editor in the databases properties, 4th Dimension will still first gives you a choice of editors when creating a project method.

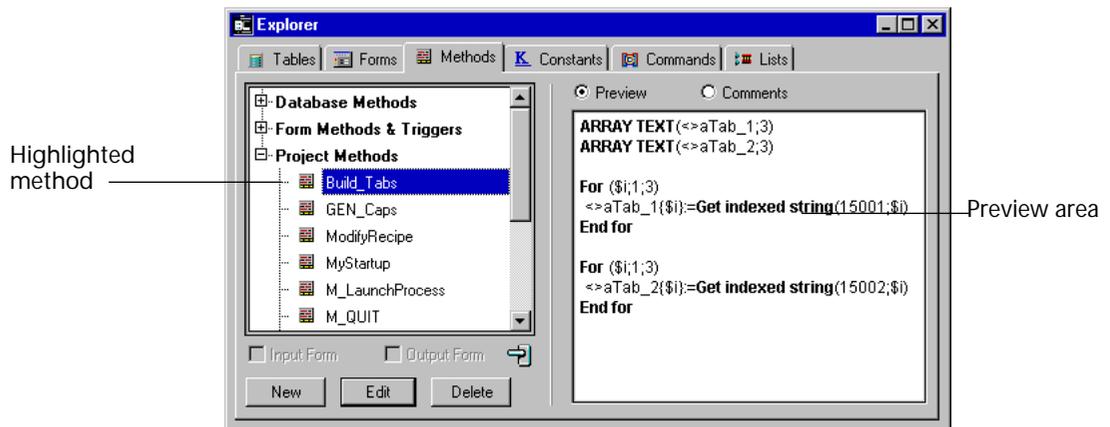
If you have not selected a default Method editor, 4th Dimension first gives you a choice of editors. You can enter the method name in the choice dialog.

Modifying a Method

To modify an existing method, double-click its name or highlight it and click Edit. The method opens in the Method editor.

Previewing a Method

Click the Preview area icon  to display the Preview area and highlight the method you want to preview.

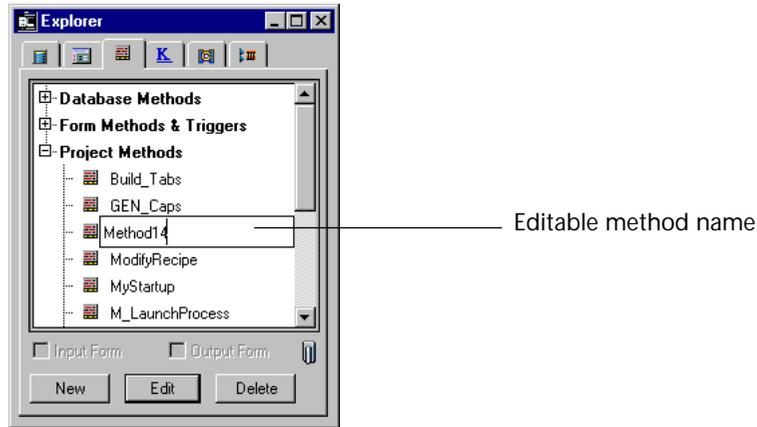


Once the Preview area is open, you can preview other methods by highlighting them in the hierarchical list.

1. For more information on selecting a default Method editor, see the section [“General”](#) on page 83.

Renaming a Method

Hold down the Command key (on Macintosh) or the Ctrl key (on Windows) and click a method name. The text becomes editable.



Enter the new name and click anywhere outside the entry area to save the new name. 4th Dimension then resorts the list of methods alphabetically.

Using Drag and Drop

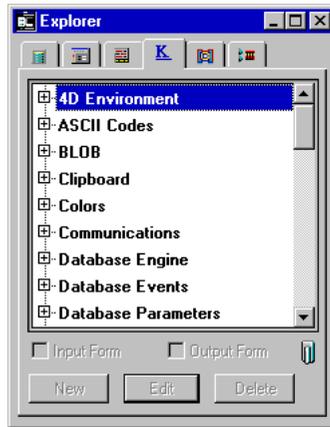
When you are writing a method, you can add the name of another project method using drag and drop. Highlight the desired method name in the Explorer and drag it to the Method editor window.

When you are creating custom menus, you need to assign a project method to each menu command. You can do so by dragging a project method name from the Explorer to the Menu Bar editor. For more information, see the section [“Assigning Methods to Menu Commands” on page 541.](#)

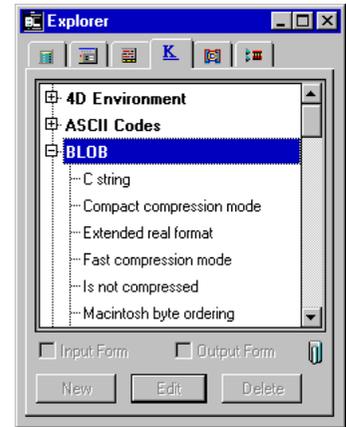
Constants Page

The Constants page contains a hierarchical list of all the constants that can be used in methods.

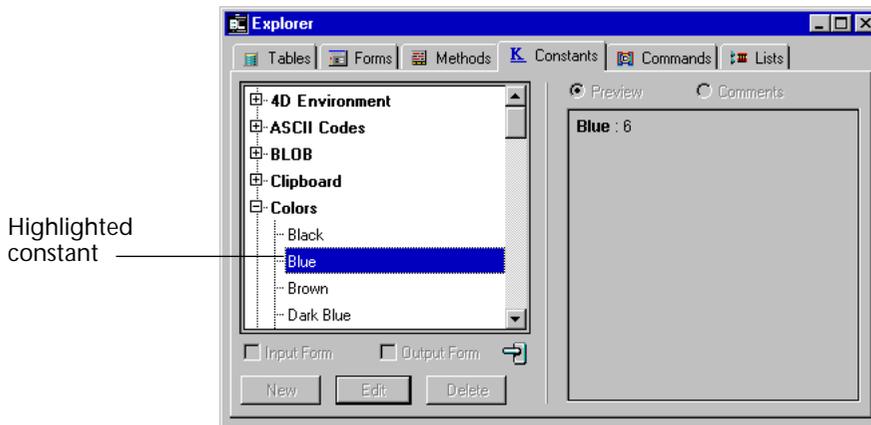
Collapsed View



Expanded View



If the Preview area is displayed, it shows the value of the highlighted constant.



For information on using constants in methods, see the section “Constants” in the *4th Dimension Language Reference*.

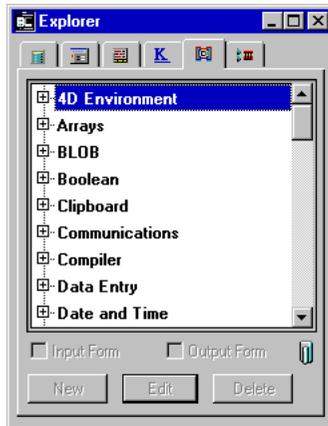
Using Drag and Drop

You frequently use constants in your methods. Instead of typing the constant you can add a constant to a method from the Explorer. Highlight the desired constant and drag it to the Method editor window. When the Method editor parses the line of code, the constant will be underlined.

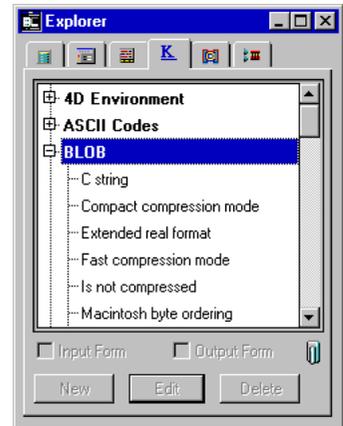
Commands Page

The Commands page displays all built-in 4th Dimension commands, grouped by category. It is equivalent to the list of commands shown in the bottom-right scrollable area of the Method editor.

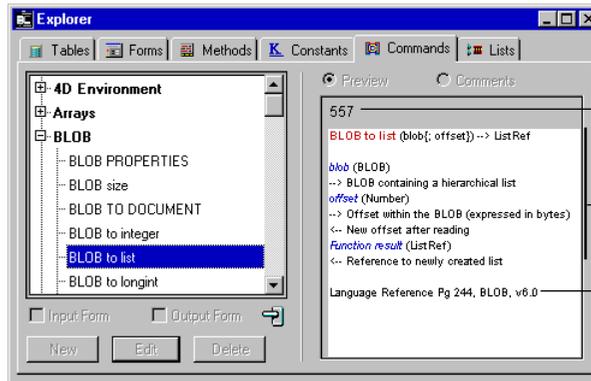
Collapsed View



Expanded View



When the preview area is selected, it shows the token of the selected command, a description of its syntax as well as the page number you should refer to in the *Language Reference* manual.



Token

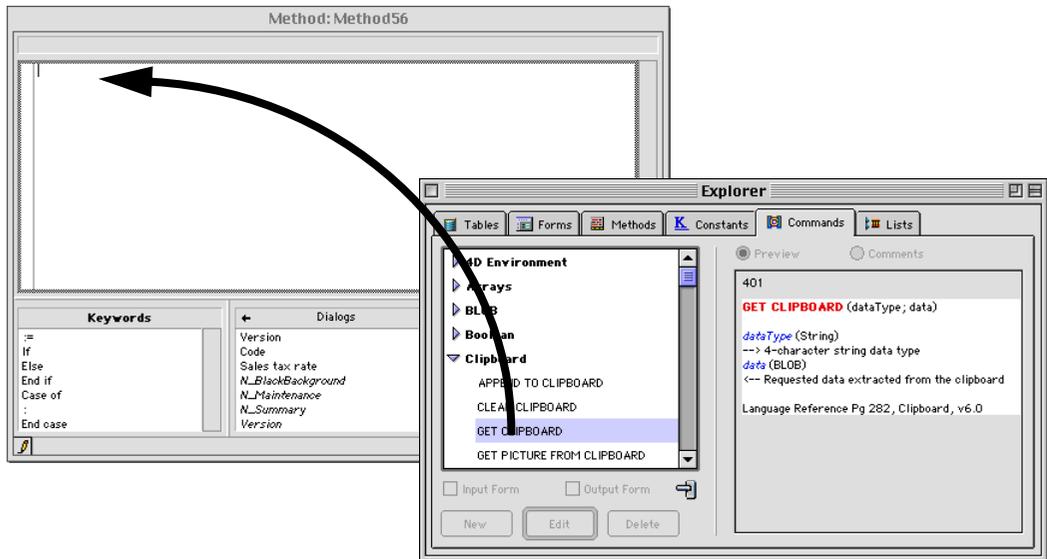
Syntax and parameter description

Location of the command description in the Language Reference manual

The token is used as a parameter to the Command name function. Command name returns the text of the command whose token you pass to it. Command name is useful in databases that require localization. For more information on the role of Command name, see the description of this command in the *4th Dimension Language Reference*.

Using Drag and Drop

Using drag and drop, you can add a command to a method. Highlight the desired command and drag it to the Method editor window. By default, the command and its syntax are dragged at the same time:



If you do not want to drag the syntax at the same time, press the Alt key when dragging the command.

Note The system that allows you to preview and drag the command syntax uses the “4D Help” file. To take advantage of this feature you should make sure the 4D Help file is neither renamed nor moved.

- On Mac OS, the 4D Help file should be located in the ACI folder of the Preferences folder of your system folder or in the 4D application folder.
- On Windows, the 4D Help.rsr file should be located in the “Windows/ACI”, where the Windows folder is the system files folder, or in the 4D application folder.

Lists Page

The Lists page lists all the lists that you have created using the Lists editor. There is no expanded view of lists.



You can modify an existing list or create a new list. To modify a list, click the name of the list or double-click it; to create a new list, click New. For more information on how to create a list, refer to [Chapter 10, “Creating Lists” on page 583](#).

Using Drag and Drop

You can create a hierarchical list or a drop-down menu in the Form editor by dragging a list from the Explorer window into the Form editor. The list is then automatically assigned to the object created in the Form editor.

- To create a hierarchical list, drag a list from the Explorer window into the Form editor window.
- To create a drop-down menu, drag a list from the Explorer window into the Form editor window while holding down the Shift key.

Using Comments

The Explorer allows you to write comments about objects in your database. Using comments is particularly appropriate for databases being developed by multiple programmers and is good programming practice generally. Your comments are displayed in the Preview area in the Explorer. The following objects accept comments:

- methods (database methods, project methods, form methods and triggers),
- forms,
- tables and subtables,

- fields.

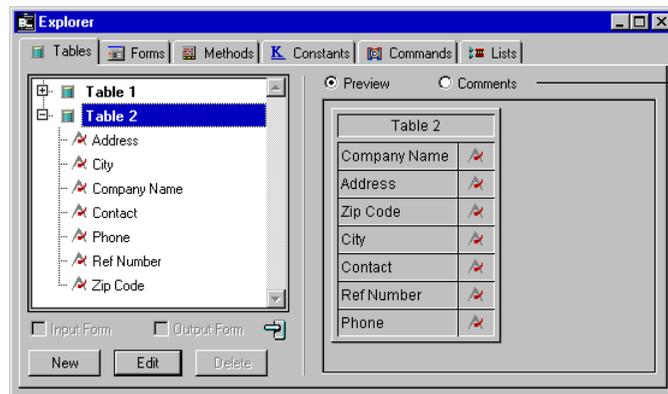
A comment can be entered as styled text (i.e., the characters can have different font styles or colors, etc.) that can be modified and viewed at any time in Design environment. It can contain a description of an object to which it is associated as well as any information necessary to understand how the object functions in the database. The comments you create are stored in the database's structure.

Moreover, 4th Dimension allows you to generate automatic comments, which means that 4th Dimension automatically enters comments when an object is created or modified.

Note Comments generated by 4th Dimension 6.5 are compatible with 4D Insider's comments.

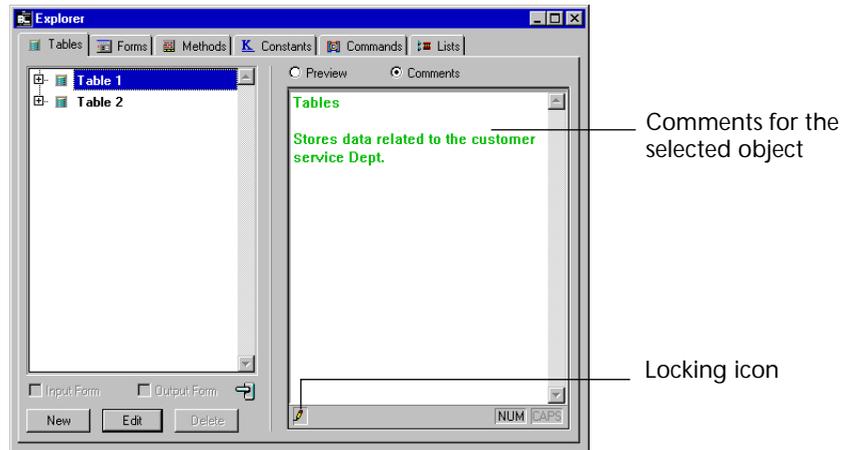
Associating a Comment to an Object

You create, view, and modify comments from the Explorer. To access the Comments area for an object, select the object and then click the Comments button located above the Preview area.



Option to view comments

When the Comments option is selected, the Preview area is replaced by the Comments area.



You can enter up to 32 KB (32,000 characters) of text for each object. The Comments area has a vertical scrollbar that allows you to scroll through the text.

4D Server The Locking icon, located in the bottom left corner of the area, indicates if the comment is already being edited by another user. If this is the case, the pencil has a slash through it and the comment can only be viewed.

Note The NUM and CAPS buttons to the right of the Locking icon indicate whether the NUM or CAPS Lock functions on your keyboard are engaged.

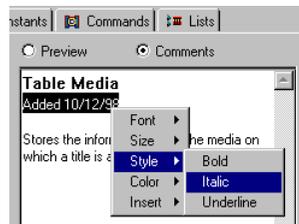
- ▶ To view, create, or modify a comment:
 - 1 Select the object (method, table, or field) to document in the left portion of the Explorer.
 - 2 Click the Comments radio button, if it has not already been selected. The Preview area changes to an editable text entry area.
 - 3 Enter or modify the text in the comments area. The text is saved as soon as you click outside of the entry area. You can use the standard text editor commands (Copy, Paste, Select All, etc.) available in the Edit menu or by using keyboard shortcuts in the Com-

ments area. You can also navigate the text in the comments area by using keyboard shortcuts as you would for any other text area.

Modifying a Comment's Font Attributes

You can change the font attributes (font, font style, font size, or color) of selected text using a contextual menu.

- ▶ To modify the font attributes of selected text:
 - 1 In the Comments area, select the text that you want to modify.
 - 2 On Windows: Click in the area with the right mouse button.
or
On MacOS: Control+click in the area.
 A hierarchical pop-up menu appears:



- 3 Select the font attributes that you want to apply to the text.

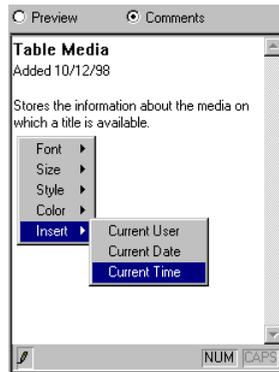
Inserting the Date, Time, or User Name

You can insert the date, time and user name (as defined in the 4th Dimension Password editor) into your comments.

Note These shortcuts are particularly appropriate for the automatic generation of comments (see below), but can also be used for comments about an individual object.

- ▶ To insert the date, time, or user name in a comment:
 - 1 In the Comments area, place the insertion point where you want to insert the information.
 - 2 On Windows: click in the area with the right mouse button.
Or
On MacOS: Control+click in the area.

A hierarchical pop-up menu appears:



- 3 Select the value you want to insert from the Insert menu.

The selected information is inserted immediately into the Comments area with its current value.

If the database doesn't have a password system, the user name will not appear.

Note This shortcut helps you enter data, but does not allow you to insert variables. Information inserted in this way can only be updated manually. To insert automatic comments, please refer to the following section.

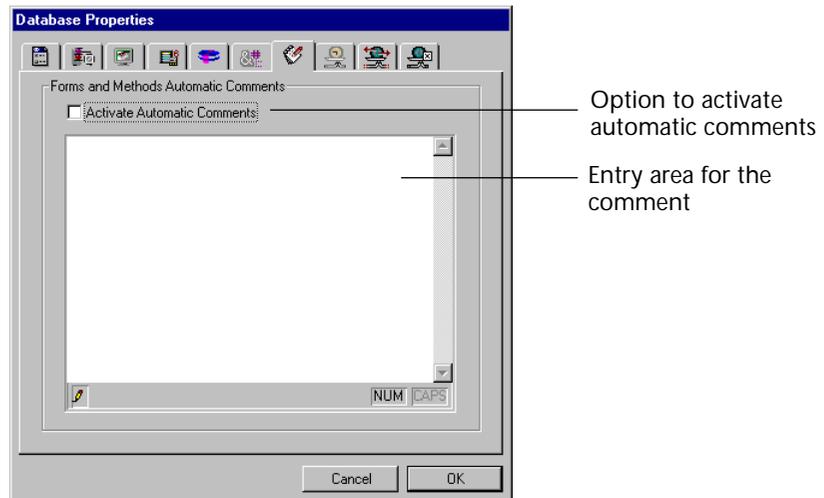
Inserting Automatic Comments

You can activate an automatic commenting system that can only be used for methods and forms in the database.

When this system is activated, a comment is automatically associated with every method or form created or modified in the database. An automatic comment can consist of both static text (such as "Modified by") and variables (such as the current date, current time, and user name).

- To activate the automatic comments system:
 - 1 In the Database Properties dialog box, click the Comments tab.

The following page appears:



Option to activate automatic comments

Entry area for the comment

- 2 Select the “Activate Automatic Comments” option.
- 3 Enter the information that you want to be inserted automatically in the entry area below.

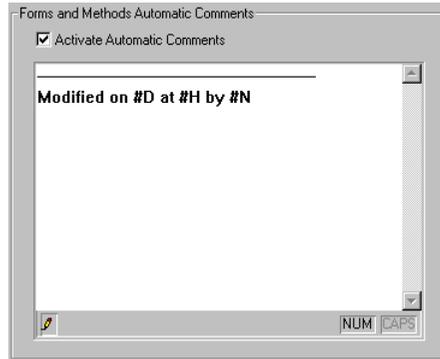
You can use variables that will be updated before being inserted in the comments:

- #D for the date
- #H for the time
- #N for the current user

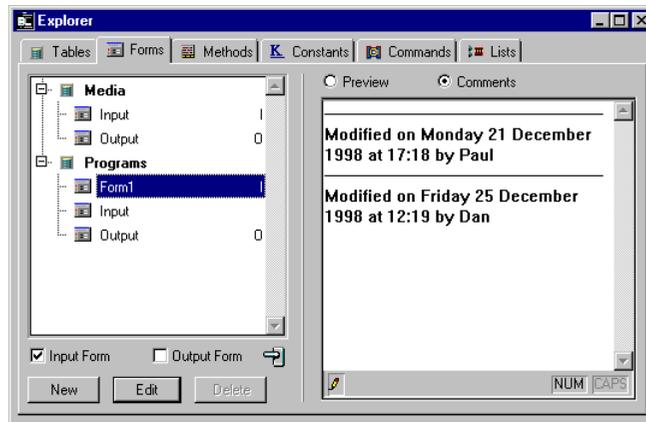
Note If your database doesn’t have a password system, #N returns an empty string.

You can also insert variables using the contextual menu (refer to step 2 in the [paragraph “Inserting the Date, Time, or User Name”, page 71](#)).

For example, if you enter the following automatic comment:



The automatic comment will be added to the comments for all new methods and forms as well as all existing methods and forms that are modified after automatic commenting is turned on:



4D Server Automatic comments can be modified by any client workstation that has access to Database Properties. You can also modify these parameters from the server station. Every modification made to the Automatic Comments page is immediately taken into consideration by each client workstation as soon as an object is modified and its comments are accepted.

Runtime Explorer

The Runtime Explorer window allows you to view the behavior of the different structural elements in your database and to verify that the available resources are operating as expected. The Runtime Explorer is particularly useful in your database's development and analysis phase. For that reason, you will want to familiarize yourself with the 4th Dimension language and the process of developing a customized application before learning about the Runtime Explorer.

Displaying the Window

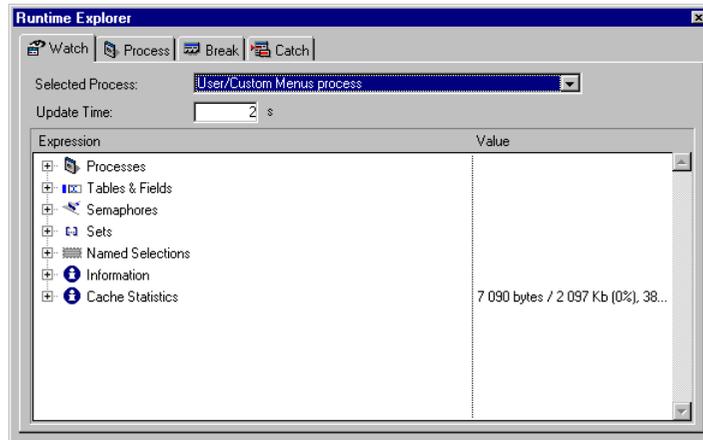
The Runtime Explorer window is accessible in all of 4th Dimension's environments: Design, User, and Custom Menus (compiled or interpreted).

Note In the Custom Menus environment, only the Designer and Administrator have access to the Runtime Explorer window.

The Runtime Explorer can be displayed in two types of windows: in a standard window (in the Design environment only) or in a floating palette (in all environments). The floating palette always remains in front of other open windows. The standard window behaves as any other Design environment editor window.

- ▶ To display the Runtime Explorer in a standard window (in the Design environment):
 - Choose Runtime Explorer from the Tools menu.
- ▶ To display the Runtime Explorer as a floating palette (from the Design, User, or Custom Menus environments):
 - 1 On Windows, press Ctrl+Shift+F9.
On MacOS, press Command+Shift+F9.
or
Hold down the Shift key and choose Runtime Explorer from the Tools menu (Design environment only).

The Runtime Explorer window has four pages, that you can access by clicking on the following tabs: Watch, Process, Break, and Catch.



Watch Page

The Watch page is a debugger and displays information about code execution.

Note The information displayed in this page is identical to the information in the debugger's Watch pane. For more information about the Watch pane, refer to the "Debugger" section in the *Language Reference* manual.

- **Selected Process:** this drop-down list contains all the processes that are being executed in the database. It allows you to select the process(es) that you want to observe.
- **Update Time:** in this area, you can define a value (in seconds) that indicates how often the information in the page will be updated.

The Expression column displays the names of the objects and expressions. The Value column displays the current value of the objects and expressions. These columns can be resized, one in relation to another. To do so, click on the separation line and drag it to the left or right.

By clicking on a value in the right column, you can modify the object's value, if the object allows you to.

The Expression column is a multi-level hierarchical list that is organized by theme. The themes are as follows:

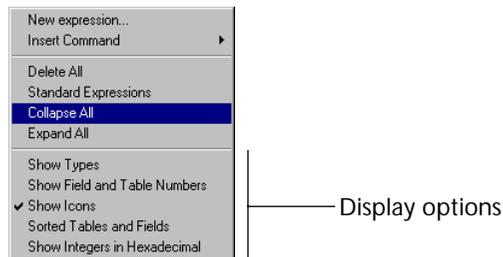
- **Processes:** allows you to view the list and the current status of the database's running processes.

- **Variables:** allows you to view the list of the database's interprocess variables as well as the list of the selected process's process variables.
- **Tables and fields, Semaphores, Sets, Named Selections, Information:** the information provided in these themes is identical to the information provided by 4th Dimension's debugger. For more information, refer to the "Debugger" chapter in the *Language Reference* manual.
- **Cache statistics:** allows you to obtain information about 4th Dimension's cache. This list is also available in 4th Dimension's debugger. For more information about 4th Dimension's cache, refer to "[System Settings](#)" on page 92.

To delete an expression or a theme, select the corresponding line and press the Delete key.

You can also add a New Expression or a 4th Dimension Command, or perform global actions: Delete All, display all the Standard Expressions, Collapse All or Expand All.

To do this, select the corresponding command in the contextual menu which appears when you click the Right mouse button (under Windows) or Control+click (under MacOS) in the window :



In addition, several display options are available in the lower part of the contextual menu. For more information, please refer to chapter "Debugger" in the *4D Language Reference* manual.

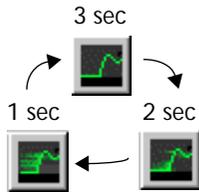
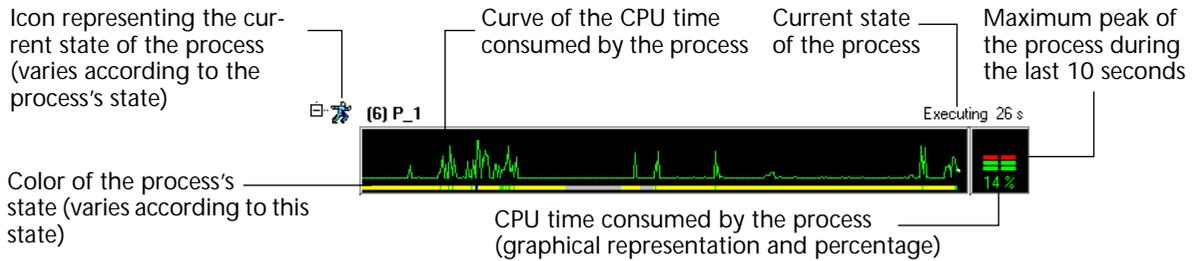
Process Page

The Process page allows you to graphically view the CPU time consumed by each process as well as the state of each process.

You can show or hide the graphical display of each process by clicking on the expansion icon to the left of the process's name. You can also show or hide all the icons by clicking the Show and Hide buttons in the window's tool bar.

The use of the process page is described in [Chapter 12, “Managing Processes”](#) on page 613.

Here is a description of the information displayed:



You can modify the frequency at which data should be updated, which can be every one, two, or three seconds. To do so, click successively on the icon in the top right portion of the window. The faster the update time, the more CPU time is consumed by the Runtime Explorer process. The number of processes that are represented graphically also influences the CPU time consumed by the process.

Note No CPU time is consumed for a process when its graphical representation is closed.

When you click in the graphical area, a vertical line appears where you clicked and a tip indicates the state of the process at that instant. By holding down the mouse button and moving it from side to side, you can view changes of the process's state.

The process management commands are now accessible by using the tool bar's buttons in the window: Resume , Pause , Abort , Trace , Hide , Show , Bring to Front .

Break and Catch Pages

The Break page allows you to view and manage break points that you have placed in your code.

The Catch page displays the break points defined in the database in relation to commands (or expressions).

The general functioning of these pages is identical to the debugger's Break List. For more information, refer to the “Debugger” chapter in *4D Language Reference* manual. The commands in the Break List menu are replaced by the buttons in the window's tool bar.

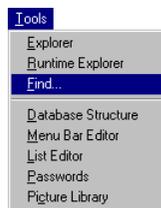
Note You can also define break points directly in the Method editor (refer to [“Using the Listing Editor”](#) on page 507).

Find Dialog Box

The Find dialog box allows you to search for a string in part of or in the entire database structure. You can, for example, search for the string “MyVar” only in methods. The search can go as far as searching objects in all the database’s forms.

Searching in the Database

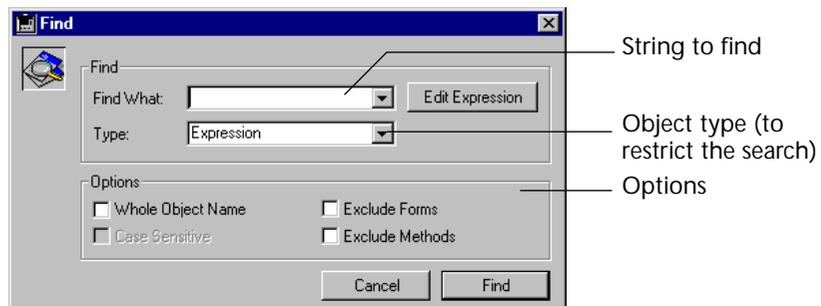
You access the Find dialog box in the Design environment.



► To find a string:

- 1 In Design environment, choose Find... from the Tools menu.

The Find dialog box appears:



- 2 In the “Find What” area, enter a string to find.

Note The at-sign (@) is considered a character. It is not possible to use it as the wildcard character in the Design environment Find dialog.

- 3 If desired, specify the object type to which you want to restrict the search.
 - Restricting the search to a certain object type makes the search faster. If you don’t want the search to be limited to an object type, choose All in the Type menu.
 - If you select Expression in the Type menu, the Edit Expression button becomes enabled. It allows you to define the expression to search for directly in the Formula Editor.

The object types are described in the section “String Types and Scope of the Search” on page 80.

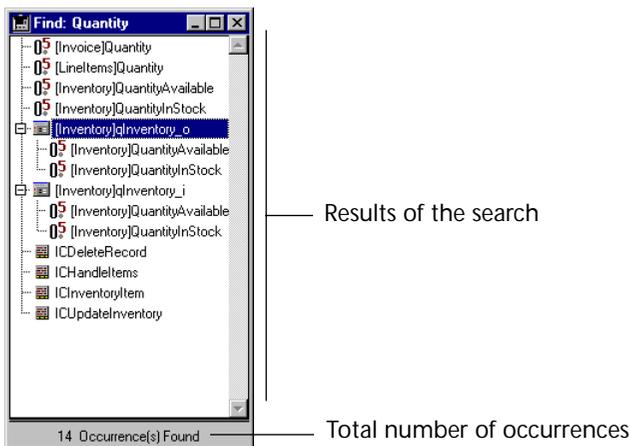
4 Define (if necessary) the search options.

These options are described in the section “Searching Options” on page 81.

5 Click the Find button or press the Enter key.

The search begins.

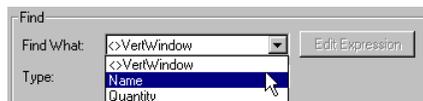
Once the search has finished, the occurrences found appear in a new window and are presented in a resizable hierarchical list.



You can double-click on a line in this window to view the object in its editor.

If you do several searches, each search opens its own result window, leaving previous result windows open.

Once you have executed a search, the value entered in the Find What area is saved in memory. This value, as well as all the other values entered during the same session, can be selected from the combo box:



Using the combo box, you can quickly do the same search many times.

String Types and Scope of the Search

The Find dialog box allows you to limit the search to a particular object type or search across all object types:

- an expression, for example “total:=Sum([Accounts]Total)”
- a variable, for example “\$vpPicture1”
- a table or field name, for example “[Clients]Name”
- a form object name, for example “Background”
- a comment, for example “Modified on”

You can also search among all object types.

By default, the search will be done throughout the entire structure of the database. You can, however, exclude methods and/or forms from the search.

Depending on the designated object type, the search will be done among the following types of objects:

- forms (which can also be excluded),
- methods (which can also be excluded),
- menus and menu commands in custom menus,
- lists,
- tables and fields (as well as subtables and subfields),
- comments.

The following table presents the structure elements in which you can search, depending on the different object types:

		Places in which to search					
		Forms and form names	Methods and method names	Menus/menu commands	Lists	Tables and fields (Structure window)	Comments
Object Types	Expression		X				
	Variable	X	X				
	Table or field name	X	X			X	
	Form object name	X	X				
	Comment						X
	All	X	X	X	X	X	X

Searching Options

You can select different options presented to you as check boxes. Depending on the type of search you specify, some options may be disabled:

■ **Whole Object Name**

When this option is selected, the search is limited to the exact occurrences of the searched object name or expression. In this case, for example, if you are searching for “client”, 4th Dimension will not find “clients” or “myclient”.

By default, the option is not selected, which means that searching for “var” will find “Myvar”, “variation”, etc.

■ **Case Sensitive**

This option is selectable only if the option **Whole Object Name** is selected. When **Case Sensitive** is selected, the search takes into account the case of the characters as they have been entered in the Find dialog box. Therefore, if you search for “MyVar”, 4th Dimension won’t find “myVar”.

■ **Exclude Forms**

When this option is selected, the search is done throughout the database, except in forms and form names.

■ **Exclude Methods**

When this option is selected, the search is done throughout the database, except in methods and method names.

Setting Database Properties

You can set database properties in the Design environment. The settings apply only to the current database — not all databases opened by that copy of 4th Dimension. The only exception to that rule is the **Display the Toolbar** option which will apply to the copy of 4th Dimension you set it for.

The specifications that you select in the Database Properties dialog box take effect immediately except for those that cannot take effect until the database is opened again (such as Startup environment).

4D Server Object locking occurs when two or more users try to modify the settings in the Database Properties dialog box at the same time. Only one user can use the Database Properties dialog box at a time. For more information about object locking, refer to the section [“4D Server Considerations” on page 111](#).

- To display the Database Properties dialog box:
 - 1 Choose Database Properties from the File menu.

OR

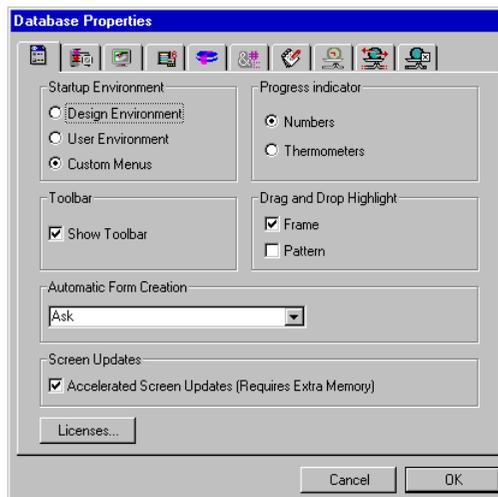
Click the background of the Structure editor (on Windows) or press the Control key while clicking the background of the editor window (on Mac OS) then select Database Properties from the contextual menu.

The Database Properties dialog box has ten pages: General, Data Control and Access, User Interface, Design Environment, System settings, Formats and filters, Comments, Connections, Web Server I, and Web Server II. Use the tabs to move from page to page.

General

General properties allow you to specify the environment that is displayed when the database is opened, choose the type of Progress indicator, the default Method editor, the presence of the toolbar, the graphical appearance of Drag and Drop operations, and the screen redraw mode. The General page also allows you to access the license management dialog.

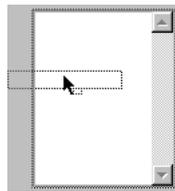
The General page is shown in the following illustration:



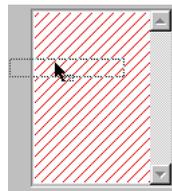
- **Startup Environment** Select the environment in which you want the database to open. By default, 4th Dimension opens the database in the Design environment, unless the database has been compiled or your user password prevents you from accessing the database in the Design

environment. For information about the password access system, see [Chapter 9](#).

- **Progress Indicator** Select a progress indicator. You can tell 4th Dimension to display the progress of an operation (such as sorting or indexing) through a numeric display or a graphic thermometer. The thermometer is slower, but easier to read. The numbers are faster, but they do not always refer to the number of records processed. For example, when 4th Dimension performs a sort, the numbers actually show the number of comparisons made.
- **Show Toolbar** This property controls whether the toolbar is displayed in the Design and User environments. This setting applies to each database opened with 4th Dimension.
- **Drag and Drop Highlight** These options let you set the appearance of the drop area (the area that is receiving a dragged object). The Drop area takes on the appearance you specify when a dragged object is on top of it and it is “eligible” to receive the dragged object. These options apply to all the 4th Dimension’s environments. You can choose to highlight the area with a frame or a pattern (or both). A frame is a dotted-line marquee; a pattern is a diagonal pattern of red lines. The following illustration compares the effects of these two options.

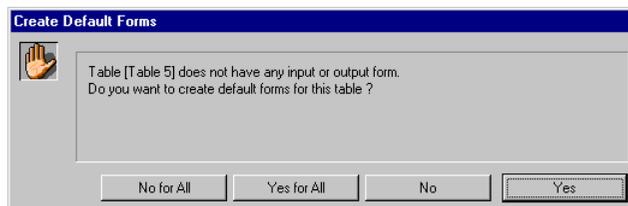


Frame



Pattern

- **Automatic Form Creation** When you create a table in Design environment and go to the User environment, 4th Dimension tells you that no form has been created for the new table gives you the option of creating default input and output forms automatically:



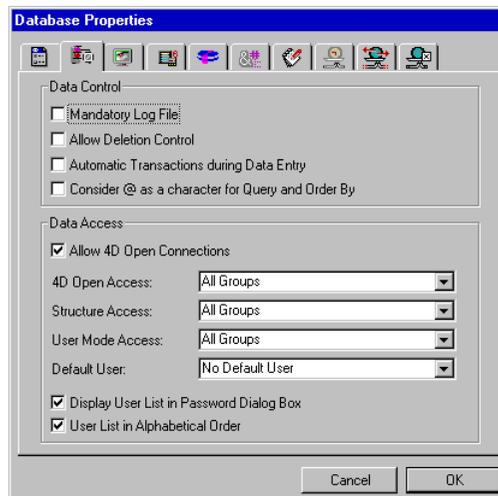
You can modify this feature so that 4th Dimension doesn't display this alert dialog box.

The Automatic Form Creation drop-down list offers three options:

- **Never:** the alert dialog box doesn't appear and no default form is created.
 - **Ask:** the alert dialog box appears when no form for the table has been created.
 - **Always Yes for All:** the alert dialog box doesn't appear, but default forms are created for all the tables automatically.
- **Screen Updates (Requires Extra memory)** This option creates an offscreen copy of the screen in order to enable faster screen redraw. The amount of memory used by that bitmap will depend on your screen settings. The formula used to assess the amount of memory required is as follows:
Memory amount (in KB) = (Screen width x Screen height x Pixel depth)
 - **Licenses** This button allows you to access the license management dialog. For more information on how to use this dialog, refer to the 4D Product Line Installation guide that came with your ACI product.

Data Control

The Data Control page lets you specify several options regarding data integrity and access management. The Data Control page is shown below:



- **Mandatory Log File** Requires that the database be opened with a log file. The log file is designed to keep track of changes to a database since the last backup. The 4D Backup plug-in is required to use this feature.
- **Allow Deletion Control** Allows the you to use the Deletion control options in the Relation Properties window for each relation you define. If the Allow Deletion Control check box is not selected, the Deletion control options are disabled. For more information, see the section [“Relation Types” on page 174](#).
- **Automatic Transactions During Data Entry** Automatically start a multi-transaction when an input form is first opened if the form has a subform. *This option is intended only for 4D First users who upgrade to 4th Dimension. It should not be used for 4th Dimension custom applications (i.e., databases that work in the Custom Menus environment).*
- **Consider @ As a Character for Queries and Order By** This option allows you to set how the at sign “@” will be interpreted when used in a query or a sort. This setting applies only when the at sign is located in a word — *the first and last characters are excluded*; it is designed to allow searching of email addresses without interfering with the normal use of the wildcard character, @, in ‘begins with’ and ‘ends with’ searches.

When this option is not checked, the at sign is used as the wildcard character can be placed internally. For example, ‘F@w’ finds words that begin with ‘F’, contain any number of letters, and end with a ‘w’. For more information, about the use of the at sign, refer to the *4th Dimension User Reference Manual*.

When the option is checked, the at sign is regarded as a single character—the at sign “@”. This setting is especially useful searching email addresses, where the @ sign is used internally.

This option has an influence on searches, sorts, string comparisons as well as data stored in tables and data found in memory, like arrays. Fields and variables of type alpha (indexed or not) and text are concerned with how the @ character is interpreted in searches and sorts.

For searches, it is important to note that if the search criteria begins or ends with @, the “@” character will be treated as a wildcard. Only if the “@” character is placed in the middle of a word (for example: bill@cgi.com) will the Query editor treat it differently.

This option can also have an influence on the behavior of the commands in the “Object Properties” theme that accept the wildcard char-

acter (“@”) in the object parameter. Please refer to 4th Dimension’s *Language Reference* manual.

If you modify this setting, you have to quit and reopen the database to make the change effective. Once the database is reopened, all of the database’s indexes are automatically re-indexed.

- **Allow 4D Open Connections** Gives the specified group the ability to log on to 4D Server from a 4D Open application. 4D Open is the API (Application Programming Interface) that allows non-4D Client applications to connect to 4D Server.
- **Structure access** Gives the specified group the ability to enter the Design environment. For information about users and groups, see [Chapter 9, “Managing Password Access” on page 555](#).
- **User Environment Access** Gives the specified group the ability to access the User environment. Any user who does not belong to the specified group will not be able to access the User environment. Any attempt from an unauthorized user to switch to the User environment will result in the 4th Dimension application quitting.
 - The Designer and Administrator always have access to the User environment even if they are not explicitly part of the group that has access to the User environment.
 - A user that has access to the Design environment always has access to the User environment, even if the user is not explicitly part of the group that has access to the User environment.

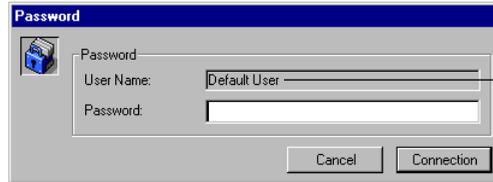
For more information about passwords, refer to [Chapter 9, “Managing Password Access” on page 555](#).

- **Default user** When a user is defined as the default user, each user that opens the database or logs onto it has the access privileges and restrictions defined for the Default User. The default user is not required to enter a user name. Moreover, if you have not associated a password with the Default User, the Password dialog box no longer appears and the database opens directly.

This option simplifies access to the database while maintaining a complete data control system.

- If you have associated a password with the Default User, a dialog box appears when the database is opened. The default user’s name

appears in the User Name area and the Default User's password must be entered:



Name defined for the Default User

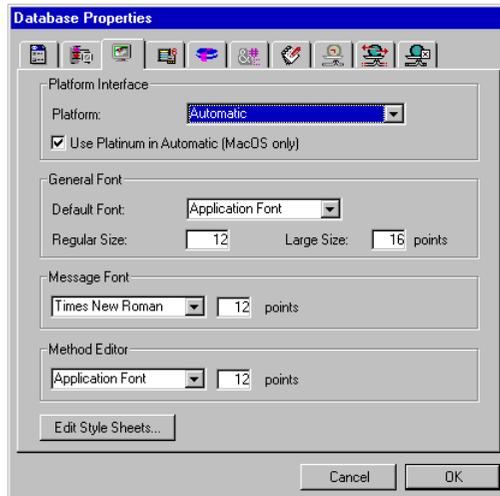
- If you haven't associated a password with the Default User, the above dialog box doesn't appear.

Note When connecting to a database for which a default user was defined, you can still connect as administrator or Designer. To do so, press the Shift key while opening the database or connecting to it.

- **Display list of users in Password dialog box** Gives users the opportunity to choose their name from a list when logging onto a Password-protected database. If this check box is selected, a user can select his or her name from a list of user names in the Enter Password dialog box. If you deselect this option, each user must enter both his or her name and password in the Enter Password dialog box. For more information about the two versions of the Enter Password dialog box, see the section [“Access System Overview” on page 556](#).
- **User List in Alphabetical Order** Sorts the list of users in the Password dialog box.

User Interface

User Interface properties allow you to customize the appearance of the user interface. The User Interface page is shown below:



- **Platform Interface** The Platform Interface property provides a way to control the appearance of forms and form objects in the database.

Five platform interface options are available:

- Automatic (default option)
- MacOS
- Windows NT 3.5.1
- Windows 95
- Platinum

If you choose Automatic, you can then choose between the Mac OS (System 7 or above) interface and the “Copland” interface. If you check **Use Platinum in Automatic (Mac OS only)**, 4th Dimension will use the Platinum interface rather than the System 7 Macintosh interface whenever the database runs on a Macintosh or Macintosh clone computer.

For complete information on Platform Interface options, see the section [“Platform Interface” on page 105](#).

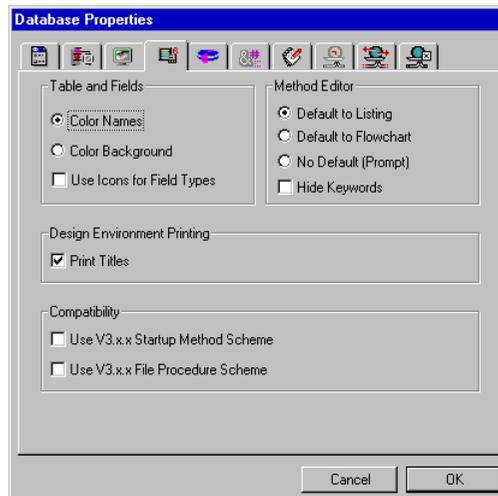
- **General Font** This area allows you to set the default font that will be used for the platform selected in the Platform drop-down list. You can set the small and large sizes of the default font. The Default font and

default font size are used in the Structure editor as well as the Method editor.

- **Message Font and Font size** This allows you to specify the font and font size that will be used for messages.
- **Method editor** This area allows you to set the font for the Method editor as well as its character size.
- **Edit Style Sheets** A style is a font, font size, and style specification. You can use style sheets to specify font attributes when creating or editing forms. When you click the Edit Style Sheets button, 4th Dimension displays the Styles dialog box. For more information about creating and using styles, see the section [“Creating a Style Sheet” on page 222](#).

Design Environment

The Design Environment page allows you to set the appearance of the tables in the Structure editor, the default Method editor, compatibility options, and to control printing from the Design environment. The Design Environment page is shown below:



- **Color Names or Background** These options let you choose between coloring either the table and field names or the table image in the Structure editor. You use the Color page of the Table and Field Properties windows to assign a color to either the names or the table image background. For information on setting the color, see the sections [“Setting the Color of the Table Image” on page 130](#) and [“Setting the Color of the Field” on page 148](#).

- **Use Icons for Field Types** Select this check box to display icons rather than letters to designate the field type in the second column of the table image. The following illustration shows the effects of this property.

Letters

Field Types	
Alpha Field	A
Text Field	T
Real Field	R
Integer Field	I
Long Int Field	L
Date Field	D
Time Field	H
Boolean Field	B
Picture Field	P
Subtable Field	*
BLOB field	X

Icons

Field Types	
Alpha Field	
Text Field	
Real Field	
Integer Field	
Long Int Field	
Date Field	
Time Field	
Boolean Field	
Picture Field	
Subtable Field	
BLOB field	

For more information on field types, see the section [“Field Types” on page 135](#).

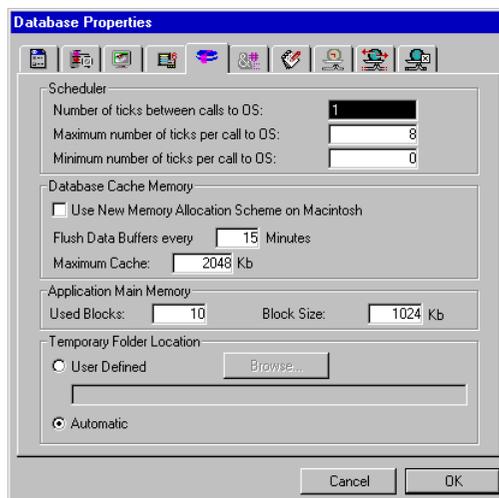
- **Method Default** Select a default Method editor for creating methods. 4th Dimension provides two editors for creating and modifying methods: the Listing editor and the Flowchart editor. Unless you specify a default editor in the Database Properties dialog box, 4th Dimension asks you to select an editor when you create a new method. 4th Dimension always gives you the option of selecting an editor for a new project method. When displaying the editor selection dialog box, the radio button of the default editor is selected. See [Chapter 7](#) for more information about creating and editing methods.
- **Hide Keywords** Select whether to hide keywords in the Method editor. By default, 4th Dimension displays scrollable lists of keywords, fields and forms from each table, and 4th Dimension commands and user-written methods. If you select the Hide Keywords check box, 4th Dimension displays the Method editor with the lists hidden. You can bring them into view by dragging the window divider. For more information on working with the scrollable lists, see the section [“Managing the Listing Editor Window” on page 521](#).
- **Print Titles** Select whether or not to print window titles when you print from the Structure, Form, Method, and Password Access editors. Unless you specify otherwise, 4th Dimension automatically prints window titles, the date on which the window was printed, and the page number. If you deselect the Print Titles check box, 4th Dimension omits the title when printing from these editors.

- **Use V3.x.x Startup Method Scheme** The Use V3.x.x Startup Method Scheme property allows you to let the *STARTUP* method be called automatically on startup, as in earlier releases of 4th Dimension. The Database methods (new in version 6) are active only if this property is *not* selected. If you are converting an old database and want to use the new Database methods architecture, you can copy your *STARTUP* procedure into the On Startup database method and deselect this property. For more information about database methods, see [“Database Methods” on page 483](#).
- **Use V3.x.x File Procedure Scheme** If this property is selected, Table methods (triggers) run according to the rules established for File procedures in earlier releases of 4th Dimension. File procedures were executed for input layouts only. They executed before the Layout procedure was executed for every input layout. File procedures were also executed whenever anything in the input layout was used (e.g., a button was pressed or data was entered in a field). This property can be used for either converted databases or databases created with the current release of 4th Dimension.

For information on the new trigger architecture, see the section [“Triggers” on page 481](#).

System Settings

The System Settings page lets you optimize the performance of 4th Dimension:



- **Scheduler** This area allows you to modify the number of ticks between calls from 4th Dimension to the operating system while running your database in interpreted mode. It also allows you to define the maximum and minimum number of ticks per call to the operating system.
- **Database Cache Memory** This area allows you to set the database's cache memory.
 - The **Use New Memory Allocation Scheme on Macintosh** check box allows you to take the database's cache memory from the Multi-Finder of the Macintosh system instead of the amount of memory allocated to 4th Dimension in the Get Info dialog box. The memory allocated to 4th Dimension will be used if not enough memory is available at the Finder level.

You must restart your Macintosh for this memory allocation scheme to take effect.

- **Flush Data Buffers Every ... Minutes** Specify a time period to save data automatically. 4th Dimension saves your data at regular intervals. You can specify any time interval between 1 and 120 minutes. As a default, 4th Dimension saves your data every 15 minutes. The application also saves your data each time you change to another environment or exit the application.

When you anticipate entering a lot of data, consider setting a short time period between saves. In case of a power failure, you will lose only the data entered since the previous save.

If each save involves a pause for disk activity, it's a sign that you should adjust the setting. The long pause means that 4th Dimension is saving many records to disk. A shorter period between saves is thus more efficient.

- **Maximum cache** Maximum size of cache memory (expressed in Kb)
- **4D Main Memory** This area allows you to define how much memory to allocate to 4th Dimension for your database while running under Windows.
- **Temporary Folder Location** This area allows you to select the location on your disk where you want 4th Dimension to store the temporary

files that are created while the database is executing. 4th Dimension mainly uses temporary files for transactions and named selections.

By default, temporary files are managed automatically, as in previous versions of 4th Dimension:

- on Windows, 4th Dimension places temporary files on volume C,
- on MacOS, 4th Dimension places temporary files on the local volume that has the most available free space.

► To modify the location of the temporary folder:

1 Click the “User Defined” radio button.
The Browse... button becomes enabled.

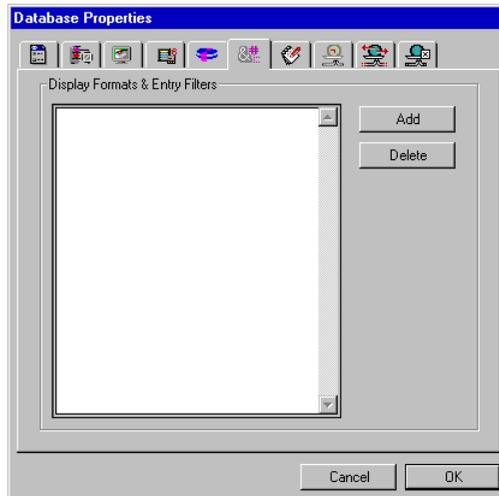
2 Click the Browse... button.
A standard open file dialog box appears.

3 Search for and select the location where you want to place the temporary folder and accept the dialog box.

Temporary files for named selections and transactions will then be written to the location indicated. This access path is stored in 4th Dimension’s preferences file.

Note If the access path is incorrect, the Automatic mode is reactivated (without the option being modified in the Database Properties dialog box). It is the database developer’s responsibility to make sure that the access path (names of the volumes or folders) is not modified.

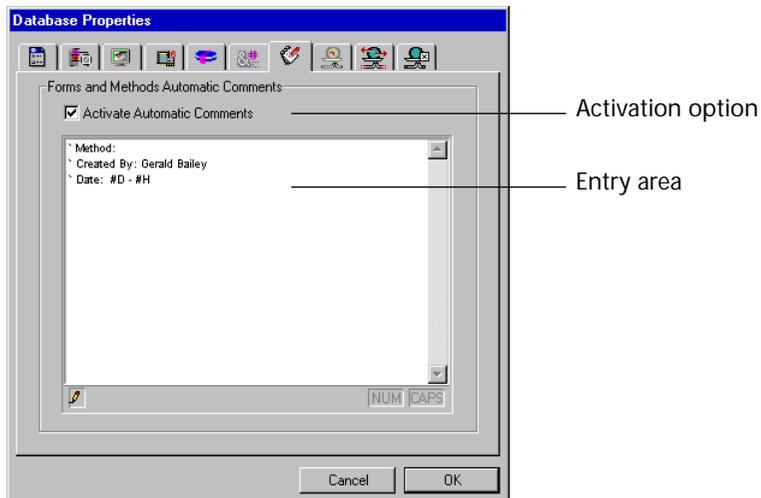
Display Formats and Entry Filters The Display Formats and Entry Filters page lets you create custom display formats and entry filters.



After you create a custom format or filter, you can refer to it by name instead of having to recreate the format or filter every time. For information on custom formats and filters, see the section [“Creating Custom Display Formats and Entry Filters”](#) on page 344.

Comments

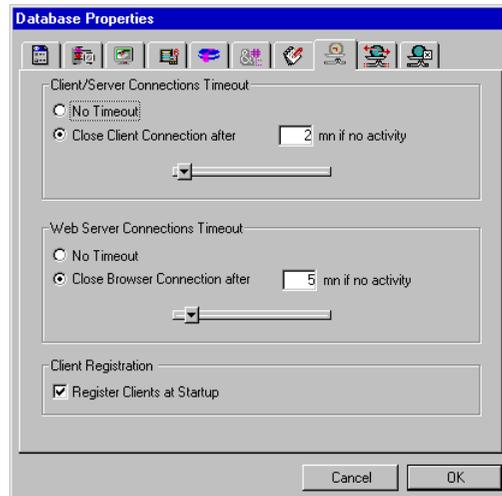
The Comments page allows you to activate and define an automatic comment system for your database.



When the **Activate Automatic Comments** option is selected, comments that are entered in the entry area will appear for each method or form that is either created or modified in the database. For more information, refer to [“Inserting Automatic Comments” on page 72](#).

Connections

The **Connections** page contains options for managing databases running on 4D Server/4D Client installations and databases used as Web servers.



With the **Connections** page you can set different timeout values for each type of client.

- **Client/Server Connections Timeout**
 - **No timeout** This property eliminates activity control. An inactive client will remain connected indefinitely.
 - **Close Client Connection after XX minutes** The server will close the connection to a client that does not send any requests to the server within the specified amount of time.

If you choose the latter property, you can set the timeout by entering a value into the entry area or dragging the slider control.

Note These two properties do not apply to network protocols other than TCP and IPX.

- **Web Server Connections Timeout**
 - **No timeout** This property eliminates the activity control of browsers connected to the database.
 - **Close Browser Connection after XX minutes** The server will close the connection to a browser that does not send any requests to the server within the specified amount of time.

If you choose the latter property, you can set the timeout by entering a value into the entry area or dragging the slider control.

- **Register Clients at Startup** This options lets 4D Server register Clients directly when they connect to the 4D Server database. Once a client is registered, it can perform any processing that was requested by the server or another client.

This option is mainly designed for the use of the **Execute on Client** function when run from the **Execute Method** dialog box in the **User** environment (for more information about the **Execute Method** dialog box, refer to the *User Reference Manual*).

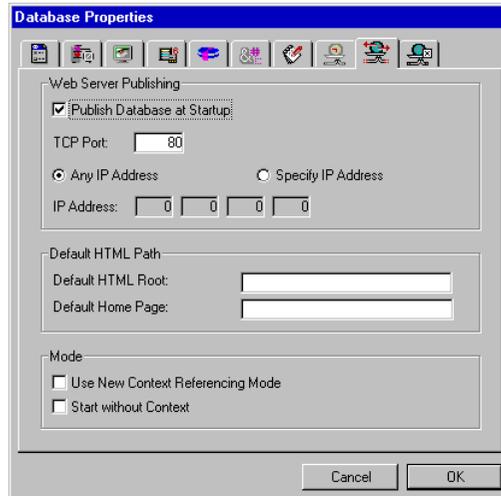
If you want to implement a sophisticated task allocation system, you should consider using the language commands that were designed for that purpose.

You can set the **Register at Startup** option on both the server or client machines. In both those cases it will apply to each client that connects to the database since it is stored in the structure file of the database. If this option is modified, you have to quit and reconnect all the clients that were already connected.

Web Server I

The operation of the 4D Web server can be set in the database properties. The two pages dedicated to those settings are the pages **Web Server I** and **Web Server II**.

The Web Server I page allows you to set the publishing options, the default HTML path, and the context referencing mode.



- **Publish Database at Startup** Choose whether to publish the current database on the Web. If this check box is selected, 4th Dimension translates forms into HTML and acts as a Web server. Using a Web browser, users can perform most database functions as 4th Dimension clients. For complete information on publishing a 4th Dimension database on the web, see the section “Web Server” in the *4th Dimension Language Reference*.
- **TCP Port** Set the TCP port to be used if the database is to be published on the Web. If you specify 0, the default value is used. The default value is 80. Setting the TCP port allows you to run several Web servers on the same computer. To do so, select different TCP ports for each Web server. This option also allows you to let the OS (i.e., Windows NT) provide Web services on port 80, while you are using 4th Dimension as a Web server on another port.
- **Specify IP address** This option allows you to specify the IP address on which the Web server will receive the HTTP requests. By default, the server answer on all the IP addresses (Any IP Address option).

When you select the Specify IP Address option, the “IP Address” area is enabled so that you can enter a specific address, like “194.166.100.101”. In this case, the server only responds to queries sent to this address.

his feature is for 4D Web Servers located on machines with multiple TCP/IP addresses. It is, for example, frequently the case of most Internet host providers.

- **Default HTML Root** This option allows you to define the folder in which 4th Dimension will search for the static HTML pages and the pictures to send to the browsers.

Moreover, the HTML root folder defines the hierarchical level above which the files will not be accessible. This access restriction applies to URLs sent to Web browsers as well as to 4th Dimension's Web server commands, such as SEND HTML FILE. If a URL is sent to the database by a browser or if a 4th Dimension command tries to access a file located above the HTML root folder, an error is returned indicating that the file has not been found.

By default, the HTML root folder is the one that contains the structure file of the database. Be careful because in this case there are no access restrictions (users can access all the volumes).

To set a folder as the HTML root folder, enter its path in the Default HTML Root entry area. The access path entered in this dialog box is relative because it is established from the folder containing the structure of the database.

For multi-platform compatibility, the 4D Web server uses the following syntax in pathnames:

- folders are separated by a slash ("/")
- the access path must not end with a slash ("/")
- to "go up" one level in the folder hierarchy, enter ".." (two periods) before the folder name
- the access path must not start with a slash ("/") (except if you want the HTML root folder to be the database's folder, see below).

For example, if you want the HTML root folder to be the "Web" subfolder in the "4DDatabase" folder, enter "4DDatabase/Web". If you want the HTML root folder to be the database folder, but the access to the folders above to be prohibited, enter "/" in the area. For unrestricted access to the local volumes, leave the Default HTML Root area empty.

Note When the HTML root folder is modified in the Database Properties dialog box, the cache is cleared so as to not store files whose access is restricted.

- **Default Home Page** You can set a default home page for all the browsers that connect to the database. This page can be static or semi-dynamic¹.

If you specify a default home page, this page is sent to each browser that connects to the database, no matter which mode (contextual or non-contextual) has been defined for the Web sessions. Unlike previous versions of 4th Dimension, the current menu bar is not sent to the browser in contextual mode.

By default, no home page is defined. If you do not specify a custom home page, the behavior of the Web server will differ depending on the startup mode:

- If the Web Server starts up in contextual mode (by default), the current menu bar — by default, menu bar number 1 — is sent as the home page, as in previous versions of 4th Dimension.
- If the Web server starts up in non-contextual mode, only the On Web Connection database method is called. It's up to you to process the query programmatically.

To define a file as the default home page, enter its relative path in the Default Home Page entry area. For example, if you want the default home page to be “MyHome.htm”, and it is located in the “Web” folder, enter “Web/MyHome.htm”.

Note For more information on the syntax you should use, refer to the description of the SET HOME PAGE command in the *Language Reference* manual.

- **Use New Context Referencing Mode** When this option is selected, 4D Web server places the context number in the basic URL of the documents being sent, which speeds up the transmission of web pages. By default, 4D Web Server sends the number of the current context to the browser for each of the user's action. For instance, if a page contains two paragraphs and a picture, 4th Dimension will send the context number three times.
- **Start without context** This option is enabled only when the Use New Context Referencing Mode option is selected. When this option is selected, the Web server is placed in non-contextual mode. By default

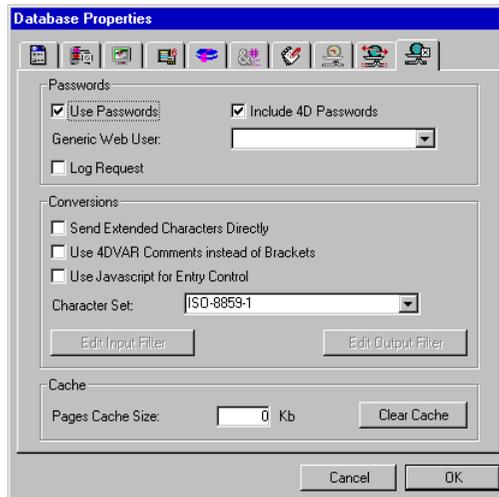
1. For more information, please refer to the *Language Reference* manual.

this option is deselected, which puts the Web server in contextual mode.

Note For more information about contextual and non-contextual modes, refer to the Web Server chapter of the *Language Reference* manual.

Web Server II

The Web Server II page lets you configure the security, the character conversion, and the web server cache:



- **Use Passwords:** activates the Web server's password system. For each connection, a dialog box appears on the browser so that the user can enter his name and password.

These two values, as well as the connection parameters (IP address and port, URL...) are sent to the On Web Authentication database method so that you can process them.

- **Include 4D Passwords:** This option is enabled only when the Use Passwords option is selected. It allows you to use, instead of or in addition to your own password system, 4th Dimension's password system (as defined in the Password editor).

Note The Web access control system consists of these options and the On Web Authentication database method. For more information about this database method, please refer to the *Language Reference* manual.

- **Generic Web User** Using this menu, you can designate a user, previously defined in the 4th Dimension Password editor, as a "Generic

Web User.” In this case, each browser that connects to the database can use the access authorizations and restrictions associated to this generic user. You can therefore simply control the browser’s access to the different parts of the database.

By default, the Generic Web user is the Designer and the browsers have full access to the entire database.

Note Do not confuse this option, which allows you to restrict the browser’s access to different parts of the database (tables, menus, etc.), with the Web server’s connection control system, managed by the password system and the On Web Authentication database method.

All the Web browsers that are authorized to connect to the database will benefit from the access authorizations and restrictions associated to the Generic Web user. The only exception to this operation is when the Include 4D Passwords option is selected and the user who connects is defined in 4th Dimension’s Password editor. In that case, the Generic Web User option is ignored and the users connect under their own access privileges.

- **Directly Sending ASCII Characters** By default, the 4D Web server converts the extended ASCII characters in the dynamic and static Web pages according to HTML standards before sending them. They are then interpreted by the browsers.

This option allows you to set the Web server so that the extended ASCII characters are sent “as is”, without converting them into HTML entities. This option increases speed on most foreign operating systems (especially the Japanese system).

- **Use 4D VAR Comments Instead of Brackets** This option allows you to define the notation to use when inserting 4th Dimension variables on static pages.
 - When the option is checked, the syntax you need to use is the standard HTML notation (`<!--4DVAR MAVAR-->`)¹.
 - When the option is not checked (default value), the syntax you need to use is the notation with square brackets (`[MAVAR]`) — which is a proprietary solution.

1. A space character must be inserted between 4DVAR and the variable name.

- **Using Javascript for Data Entry controls** When this option is checked, part of the browser's entry control is taken over by automatic Java scripts.

On the browser, the data entry controls and the data types (fields or variables) to which they can be applied are as follows:

- minimum value (for numeric values)
- maximum value (for numeric values)
- mandatory value (for numeric and alphanumeric values)

Generated Javascripts, which are small in size, display alert dialog boxes without preventing the user from accepting a data entry (it is still 4th Dimension's responsibility).

Actually, if a data entry area contains an incorrect value, an alert message is displayed on the browser when the user clicks a button (OK, Cancel, etc.):



Once the alert dialog box is validated, if the user clicks the button again, the button's action is then taken into account. The complete data entry control is done on the Web server (as in previous versions of 4th Dimension), in the User and Custom Menus environments.

- **Character Set** This option is designed for countries that use specific character sets such as Japan or Korea. It allows you to choose the character set 4th Dimension uses to communicate with browsers. The value selected will determine the conversion of ASCII characters that 4D Web server performs when sending or receiving an HTML document (dynamic or static pages).

The default character set for the US version of 4th Dimension is ISO-8859-1, which corresponds to the standard Occidental (Latin 1) encoding. That encoding is widely used in western Europe and the US.

The values available in 4th Dimension are the following:

- ISO-8859-1: Occidental (Latin 1) encoding
- Shift_JIS: Japanese encoding
- Big5: Chinese encoding

- **euc-cr**: Korean encoding
- **x-user-defined**: any user-defined encoding (for more information, refer to the paragraphs below).

4th Dimension lets the browser know which encoding is to be used. It is therefore not necessary to set that setting in each browser. The only setting that requires the browser to be preset to is **x-user-defined**.

The **x-user-defined** encoding is designed for countries that use a specific alphabet (Island, Greece, and so on). This setting permits the use of any specific encoding.

When this value is selected, the buttons **Edit Input Filter** and **Edit Output Filter** are enabled. Those buttons allow you to edit the input and output conversion tables for ASCII characters.

When this option is selected, 4th Dimension does not let the browser know about the encoding it is using and it therefore should be selected by the user when he connects to the database.

Note To ensure database compatibility, the default values of the filters are identical to those used in 4th Dimension version 6.0.6 (from the “MapC” — if defined with Customizer Plus).

- **Page Cache Size** This option activates the cache of 4D Web Server. This cache allows you to load static pages, GIF images, JPEG images (<100 kb) and style sheets (.css files) in memory, as they are requested. Using the cache allows you to significantly increase the Web server’s performance when sending static pages.

By default, the cache of the static pages is not enabled (its size is equal to 0).

The value you set depends on the number and size of your Web site’s static pages, as well as the resources available on the host machine.

Once the cache has been enabled, the 4D Web server looks for the page requested by the browser first in the cache. If it finds the page, it sends it immediately. If not, 4th Dimension loads the page from disk and places it in the cache.

When the cache is full and additional space is required, 4th Dimension “unloads” the oldest pages first, among the least demanded ones.

- **Clear Cache** Clicking this button clears the cache of the pages and images that it contains (if, for example, you have modified a static page and you want to reload it in the cache).

Platform Interface

The Platform Interface property lets you display any form using the conventions of the graphical user interface (GUI) of a selected platform. Specifying a Platform Interface for a form or object does not actually modify the form or object. The selected platform interface affects only the way the forms are displayed on the screen. A form can look like a Mac OS, a Platinum, a Windows NT 3.5.1, or a Windows 95 screen depending on the option you choose.

The Platform Interface option in Database Properties sets the platform interface for all forms in the database. You can also set the platform interface for individual forms and for individual objects on a form. At the form and form object level, you can choose to inherit the setting from the next highest level or override it with a custom setting.

The Platform Interface for individual forms is set in the Form Properties window. You can set the platform interface for the form when you are creating it using the Form Wizard or you can change it using the Form editor. For information on setting the platform interface when the form is being created, see the section [“Customizing the Appearance of Form Objects” on page 209](#). For information on modifying the platform interface settings of a form using the Form editor, see [“Setting the Platform Interface” on page 255](#). For information on changing the platform interface of individual objects, see [“Setting a Text Object’s Platform Interface and Appearance” on page 302](#).

Platform Interface Settings

The Automatic option is the default and displays forms as they appear on the host platform: If you use the database on Macintosh, the forms look like Macintosh screens; if you use the database under Windows 95, the forms will look like Windows 95 screens, and so on.

Note If you select the Use Platinum in Automatic (Mac OS only) option and run the database on Mac OS with the Automatic interface option selected, the Platinum interface will be used.

Choosing another option allows you to handle the following situations:

- You are designing forms on one platform and you would like to see how they will look on the other platforms or operating systems.

- No matter which platform you are using, you want your layouts to always look the same, overriding the default GUI of the platform.
- You would like to control the platform interface using the language. The command SET INTERFACE allows you to choose the platform interface. With SET INTERFACE, you can simply let the user(s) of the database choose the GUI they like.

The Platform Interface property affects only 4th Dimension forms. It does not change standard 4th Dimension dialog boxes such as the Query and Order By editors. 4th Dimension always displays standard dialog boxes using the platform's GUI regardless of the Platform Interface setting. On Macintosh, standard dialog boxes are displayed on a white background and the controls use the Macintosh system control definitions. On the Windows NT 3.5.1 platform, standard dialog boxes are displayed on a gray background and controls use the 3D effects Windows methods if the Windows 3D effects DLL (CTL3D32.DLL) is installed. Otherwise they are displayed on a white background with gray buttons and non-3D check boxes and radio buttons. Note that on Windows 95, 3D effects are always available. Buttons and graphical elements (such as surrounding rectangles) are displayed using the colors set in the Colors Windows Control panel.

4th Dimension displays your forms according to the current Platform Interface setting rather than the actual platform's GUI on which the database is run (except if the Automatic option is selected.)

This setting acts on the following objects and form properties:

- Fields
- Buttons (push buttons)
- Thermometers and dials
- Check boxes
- Radio buttons
- Hierarchical lists
- Pop-up menus and drop-down lists
- Objects whose foreground and/or background colors are automatic
- Form background color

Here are descriptions of each setting:

MacOS setting

- Buttons are displayed as MacOS round rectangle buttons.
- Check boxes and radio buttons are displayed as MacOS controls.
- Thermometers, rulers and dials are displayed as MacOS controls.
- Hierarchical lists are displayed according to Mac OS standards.
- Pop-up menus/drop-down lists are displayed according to Mac OS standards:



- The automatic foreground color of layout objects is set to black.
- The automatic background color of layout objects is set to white.
- Each form's background color is white.

Platinum setting

- Buttons are displayed as Platinum buttons.
- Check boxes and radio buttons are displayed as Platinum controls.
- Thermometers, rulers and dials are displayed as Platinum controls.
- Hierarchical lists are displayed according to the Platinum standard.
- Pop-up menus/drop-down lists are displayed according to the Platinum standard:



- The automatic foreground color of layout objects is set to black.
- The automatic background color of layout objects is set to gray.
- Each form's background color is gray.

Windows NT 3.5.1 setting

- Buttons are displayed as 3D effects push buttons (using the Windows NT 3.5.1 method).
- Checkboxes and radio buttons are displayed as regular Windows NT 3.5.1 controls.
- Thermometers, rulers and dials are displayed as Windows controls.
- Hierarchical lists are displayed according to Windows standards.
- Pop-up menus/drop-down lists are displayed according to Windows standards:



- The automatic foreground color of layout objects is set to black.
- The automatic background color of layout objects is set to white.
- Each form's background is white.

Windows 95 setting

- Buttons are displayed as 3D effects push buttons (using the Windows 95 method).
- Check boxes and radio buttons are displayed as 3D effects (using the Windows 95 method).
- The automatic foreground color of form objects is set to the Button Text color chosen by the user in the Windows Color Control panel.
- The automatic background color of form objects is set to the Button Face color chosen by the user in the Windows Color Control panel.
- Each form's background is set to the Button Face color chosen by the user in the Windows Color Control panel.

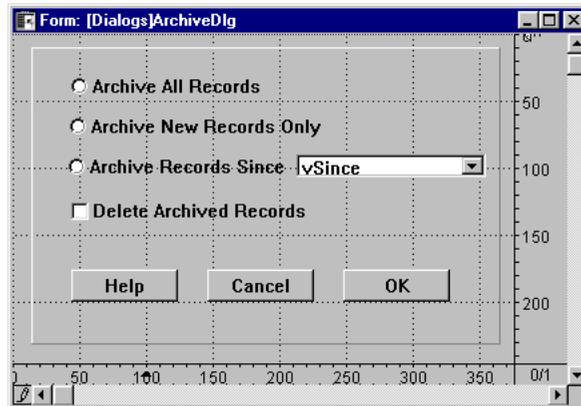
Note For more information about the automatic foreground and background colors of objects, see [“Foreground and Background Colors” on page 307.](#)

On Windows, if you change colors in the Windows Color control panel while a 4th Dimension session is running, the program immediately updates the forms displayed in all running processes according to the new color settings. If you choose Windows 95 as the Platform Interface setting on a Macintosh, the automatic foreground and background colors are dark gray and light gray, respectively.

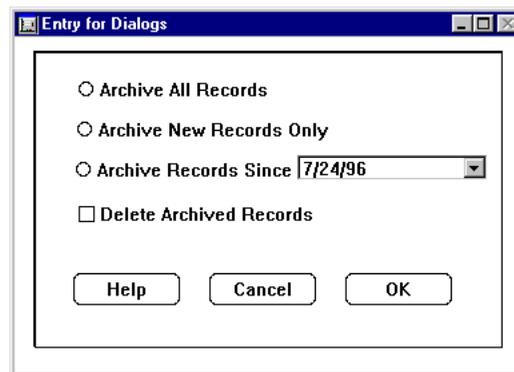
On any platform, changing the Platform Interface setting (using the Database Properties dialog box or the SET INTERFACE command) has an immediate effect: all the forms in any environment and in any process are redrawn to reflect the selected option.

4D Server Under 4D Server, several 4D Client stations can concurrently use different Platform Interface settings.

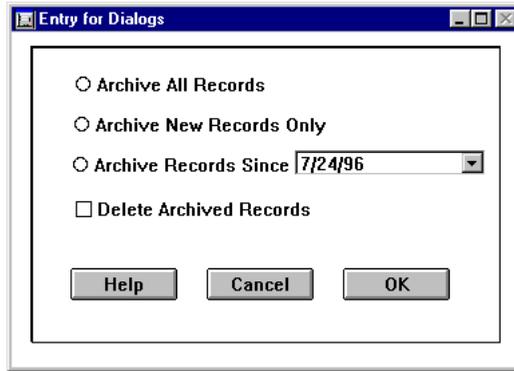
- ▼ This example illustrates how a form appears under the MacOS, Windows 3.1, and Windows 95 settings. The form contains buttons (push buttons), radio buttons, and check boxes:



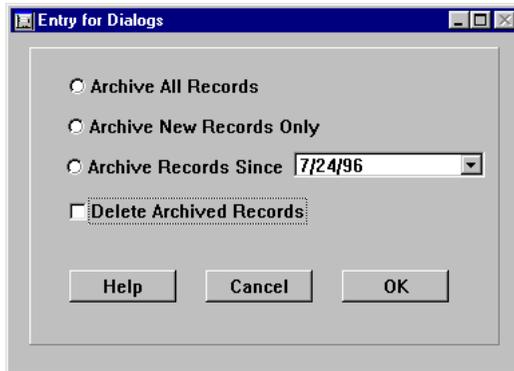
MacOS setting



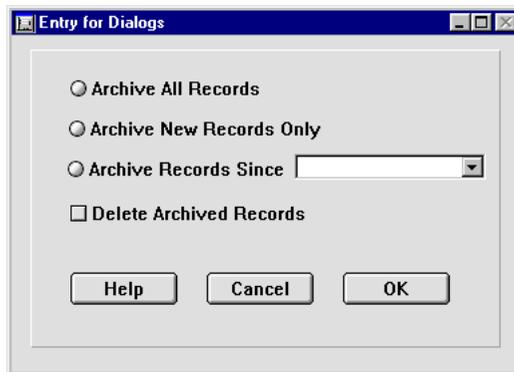
Windows NT 3.5.1
setting



Windows 95
setting



Platinum
setting



4D Server Considerations

4D Server supports the continuing development of a database that is currently in use by allowing you to modify the design of a database in the Design environment while users continue to work with the database in the User or Custom Menus environments. Changes to the design are visible to users as soon as they are saved (when you close the editor window or choose *Save EditorName* from the File menu).

In addition, 4D Server allows several users to work on the database design concurrently. 4D Server protects against users erasing each other's changes in the Design environment by providing an object-locking system in which structure objects are locked while they are being modified by a user. Structure objects include the following: field definitions, table definitions, forms, methods, menus, passwords, database properties, and lists. During the time that an object is locked, other users cannot modify it. An object is unlocked when it is closed.

For more information about how object locking works with a particular structure object, refer to the appropriate chapters in this manual.

2

Designing a Database Structure

This chapter tells you how to use 4th Dimension's Structure editor to create and modify database structures. The structure of a database consists of tables and fields. If a database has more than one table, the structure might include the relations between the tables.

The structure of a database is like the foundation of a house — it provides the basis for everything else. This chapter gives a general description of databases and of how to design database structures to meet different information management needs.

This chapter tells you how to:

- Manipulate table images in the Structure window,
- Create tables and set table properties,
- Create fields and set field types and properties,
- Relate tables.

Database Basics

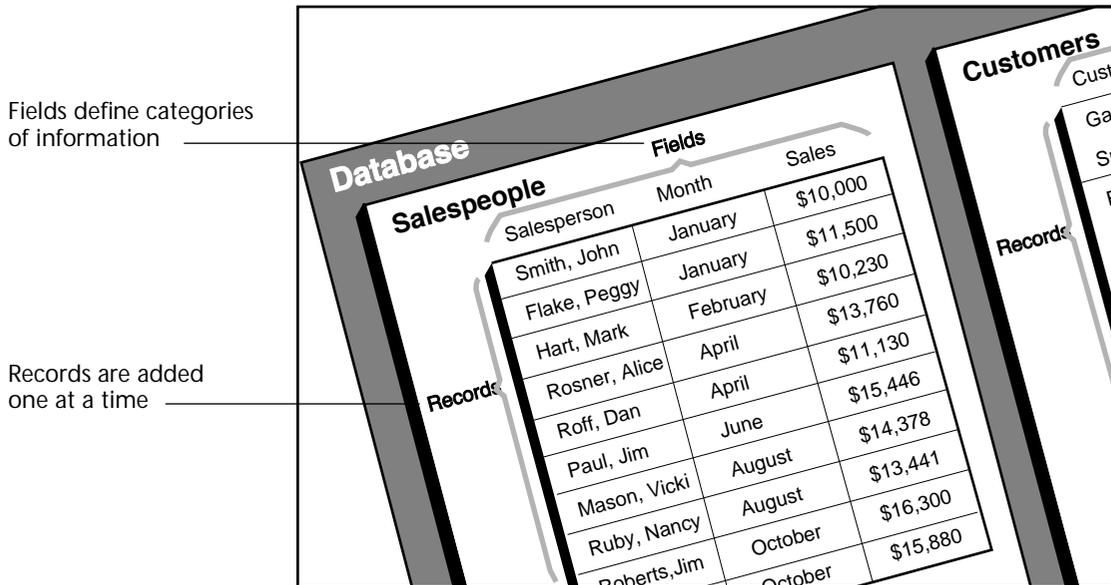
A database is any collection of information, organized so that it can be used efficiently. A telephone directory is a good example of a database. So is a dictionary, a calendar, or a cookbook.

The information in a database is organized in the form of records. Each record contains all of the information about one person or thing in the database. For example, each record in a telephone directory contains one person's name, address, and telephone number.

Each record contains fields. A field is used to store a particular piece of information. For example, in the telephone directory database, one field contains the person's name; a second field contains the person's address; and a third field contains the person's telephone number. Every record contains each of these fields and every record can have information in these fields.

A field name usually identifies the information that goes into the field. A field name is usually something like Name, Address, or Phone Number. Each field has a field type that identifies the kind of information that can be entered in a field: numbers, dates, alphanumeric characters, and others. Because each field contains a specific type of data, you can perform calculations and other operations on the information in the fields. For example, numbers from two fields can be added. A date in one field can be compared to a date in another field. A person's first name (stored in one field) can be displayed in front of the last name (stored in another field) to make the first line of an address label.

All the records together make up a *table*. Each database can contain many tables. The following figure shows how these concepts are related.



4th Dimension can reorganize records and perform calculations on the information so that the information is useful. For example, 4th Dimension can calculate the total values in a field and present the total in a report. It can calculate a total for each salesperson and display a graph that compares sales figures.

Tables

When you create a new database, 4th Dimension automatically creates one table and names it *Table1*. You can rename the table and add fields to it. To use the database, you must add at least one field.

4th Dimension allows you add additional tables to the structure and establish relations among tables. This gives you the ability to create a structure that meets your exact needs.

Single-Table Structures

Some databases use only one table. You use a single table for a single category of information such as people, companies, or inventory. You can have as many fields in a table as you need (up to 511). Non-

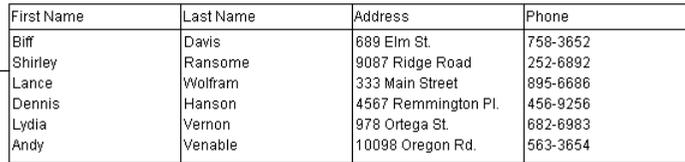
relational database applications sometimes refer to a single table database as a *flat-file database*.

Table image in the Design environment



People	
FirstName	A
LastName	A
Address	A
Phone	A
State	A

Records in the User environment



First Name	Last Name	Address	Phone
Biff	Davis	689 Elm St.	758-3652
Shirley	Ransome	9087 Ridge Road	252-6892
Lance	Wolfram	333 Main Street	895-6686
Dennis	Hanson	4567 Remmington Pl.	456-9256
Lydia	Vernon	978 Ortega St.	682-6983
Andy	Venable	10098 Oregon Rd.	563-3654

In the figure above, every person's record needs the same types of data. The database grows in accordance with the number of people stored.

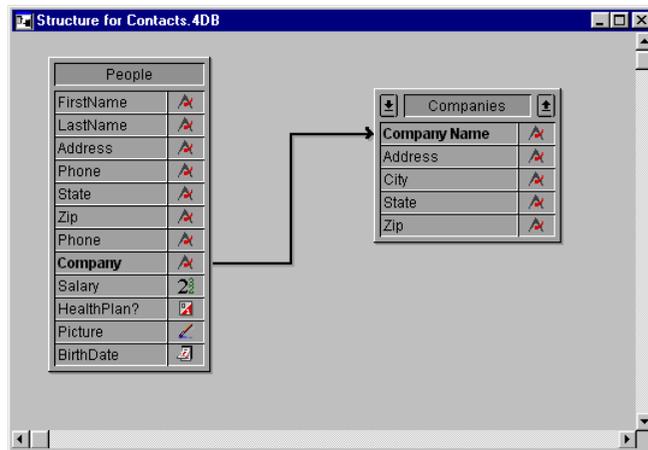
Multiple-Table Structures

A database can often store and access data more efficiently by using more than one table. A good rule to remember is that different types of information should be stored in different tables.

A database that keeps track of both people and companies is a good example. The records for the people and the companies are stored in different tables. If the address of a company changes, you need only change that company's record. You do not need to enter that new address for every person who works for the company.

With a single table, you would have needed to enter the address in each individual record; with two tables, you need to enter that information only once. When a company name is entered in a person's record, 4th Dimension can search for the company's record and automatically display the correct address.

The figure below shows the structure of a multiple-table database in which two tables are related. The arrow drawn between the Company field and the Company Name field shows that relationship.



The data for each person is stored in the [People] table. Data about each company is stored in the separate [Companies] table¹.

4th Dimension is called a *relational database* application because it can use multiple tables and relate them in various ways. For example, you can create a report for the [People] table that searches the [Companies] table and automatically displays and prints information about each person's company. The relationship between the tables allows information from each table to be available to the report.

You can also enter data directly into related tables. For example, an invoicing database can write information to a [Line Items] table from within an Invoicing screen. You can also write data to related tables using 4th Dimension's language.

For information on creating and using related tables, see the section [“Relating Tables” on page 152](#).

Sometimes you need a multiple-table structure in which tables are not directly related. It may be convenient to have one database store different kinds of information such as a contact list and an expense table.

1. In the 4th Dimension documentation, table names are shown in brackets. This is how they appear in the Method editor.

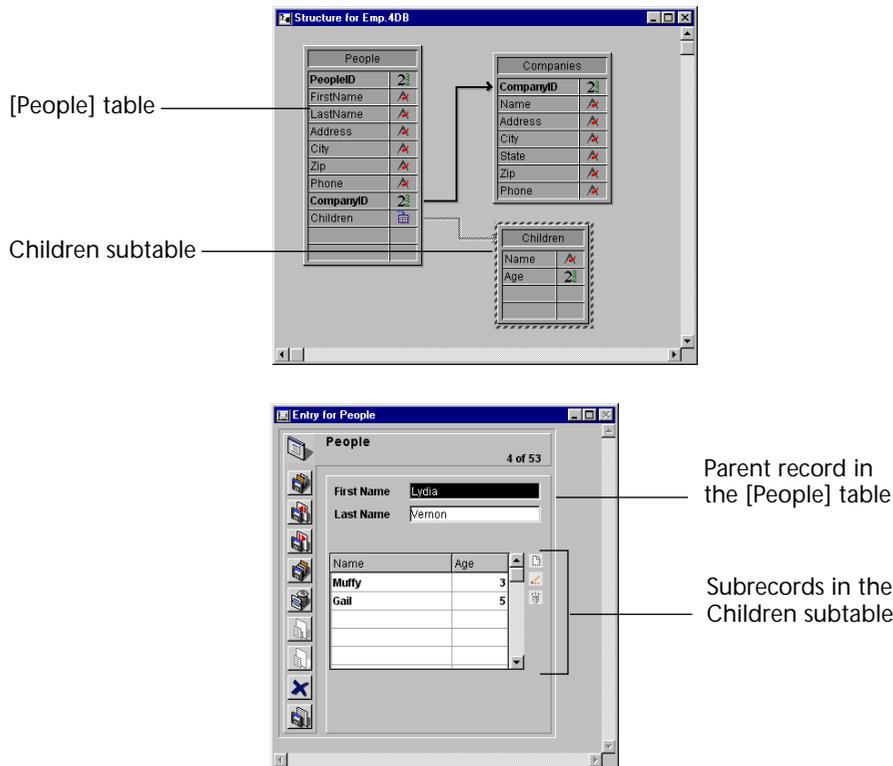
4th Dimension allows up to 255 tables in each database. A table can have up to 511 fields. Using multiple tables, virtually any kind of database structure is possible.

Subtables

Occasionally you will need to store a variable amount of information for each record in a table. 4th Dimension allows you to create a table structure that includes a subtable — a table subordinate to a parent table. Information in the subtable is available only when the parent record is in use.

For example, suppose you want to keep track of the children of the people in your table. Some people have no children and some people have many children. You can use a subtable to store this information.

The figure below shows a subtable and form that displays subrecords belonging to the parent record.



This subtable allows you to create as many subrecords as you need for each record in the [People] table. If a person has three children, you

create three subrecords for that record. If a person has no children, you create no subrecords.

A subtable structure is useful when you need to see and use the subrecords *only* when its parent record is being used. For example, using subtables you could easily find the average age of one person's children, but it would be difficult to find the average age of everyone's children. Similarly, it would be difficult to do a search across all children in the whole database (e.g., to find all five-year-old children). If performing these kinds of operations on the data is a requirement, it would be better to use a separate table for children, not a subtable.

Note Whenever a parent record is loaded, 4th Dimension loads all the subrecords belonging to that parent record. If there are many subrecords belonging to the parent record, this can be undesirable. In this case, it is recommended that you use a related table to store the information and that you control the loading of records using the language.

You cannot access information contained in a subtable from a 4D Open application. 4D Open is the API (Application Programming Interface) that allows non-4D Client applications to connect to 4D Server.

Creating a Database Structure

Every database has a structure that consists of at least one table and at least one field. These elements must exist before the database can store records.

► Typically, you create a structure with the following basic steps:

1 Create a new database.

For more information, see [“Creating a New Database” on page 22](#).

4th Dimension automatically creates the first table.

2 Rename the automatically created table (optional) and assign table properties.

For more information, see [“Renaming a Table” on page 127](#) and [“Setting Table Properties” on page 125](#).

3 Create fields for the table and assign field properties.

For more information, see [“Creating New Fields” on page 131](#).

- 4 Add additional tables and fields as necessary.
For more information, see [“Creating a New Table” on page 124](#).
- 5 Relate one table to another, if necessary, by drawing a line between fields in the two tables and setting relation properties.
For more information, see [“Relating Tables” on page 152](#).

The remainder of this chapter describes these steps in detail.

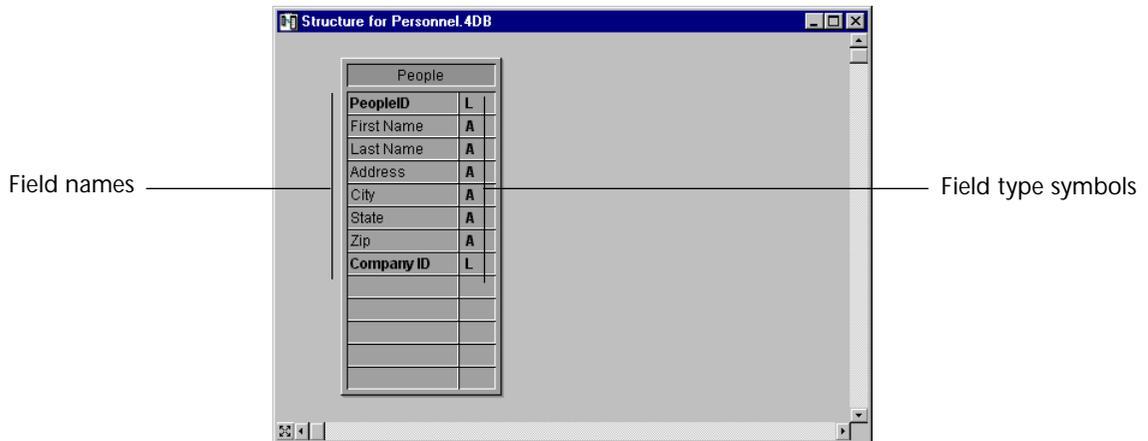
Using the Structure Editor

The Structure editor lets you manage the database structure — the tables and their relationships. It gives you control over such things as tables, table properties, fields, field properties, and table relations.

The Structure editor gives you a graphic view of a database’s structure and provides menus for performing design operations.

Each table is represented by a table image in the Structure editor. It shows the fields and their types. 4th Dimension automatically creates the first table. You can add tables as necessary.

The following illustration shows the Structure editor window with one table image.



Selecting a Table Image

To work with a table in the Structure editor window, you first need to select it¹. You can then move, or resize the table image.

- To select a table in the Structure editor.

1 Click the image of the table

OR

Double-click the table's name from the Tables page of the explorer window

OR

Right-click (Windows) an empty area of the Structure editor window or press the Ctrl key while clicking an empty area of the Structure editor window (Mac OS), then select the table from the Table List submenu of the contextual menu.

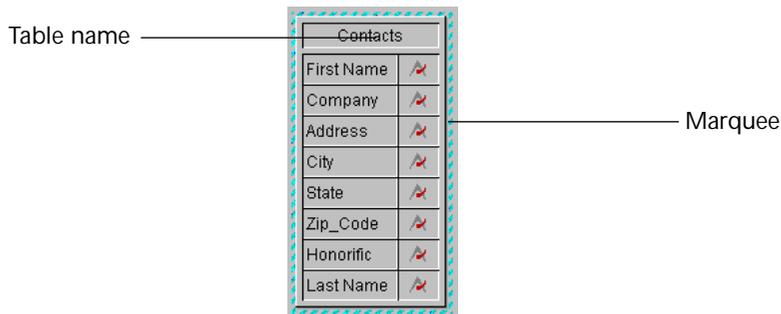
OR

Press the Tab key to select through the tables and subtables of the structures.

4th Dimension draws a flashing marquee around the table image and makes it the frontmost table¹. Subsequent actions affect the selected table.

Note Unless a table is selected, the Table Properties and New Field menu commands in the Structure menu are disabled.

The figure below shows a selected table.



Scrolling the Field List

As you add fields to a table, you may add more fields than are visible in the table image displayed in the Structure editor window. When this

1. If the desired table image is not in view, you can double-click its name in the Tables page of the Explorer to bring it into view.

1. When tables are superposed in the Structure editor window, their position determines the selection order when you navigate through the tables using the Tab key. 4D stores this position for each user between two sessions.

happens, 4th Dimension automatically adds scroll arrows to the title area of the table image.



Click a scroll arrow to scroll the field list upward or downward

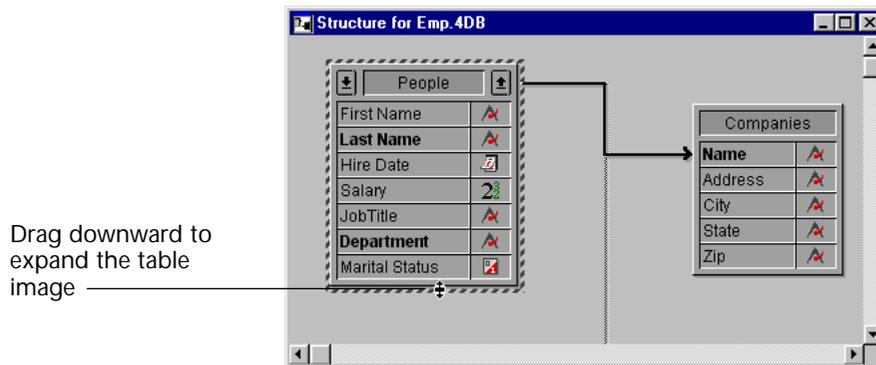
You can scroll the field list within the table image by clicking on a scroll arrow.

Resizing a Table Image

You can also expand a table image to show more fields in the table.

- ▶ To expand a table image:
 - 1 Position the pointer at the bottom edge of the table image until it changes to a table resizing pointer
 - 2 Drag the bottom edge down.

The following illustration shows a table image being resized.

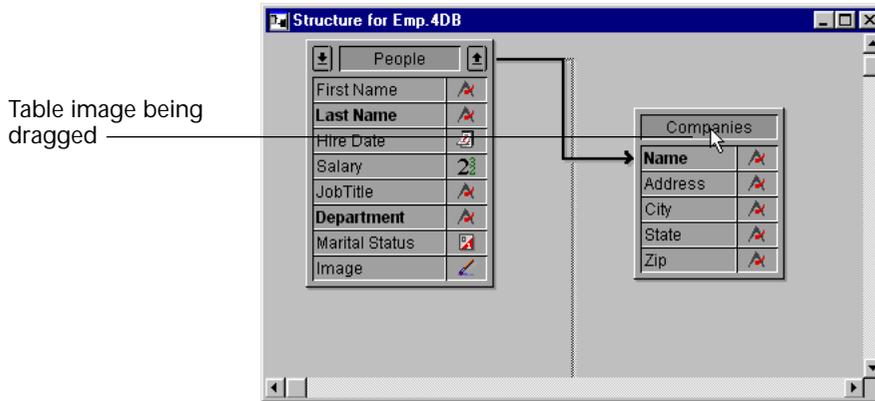


As you drag, the table image expands in increments of one field at a time. When all fields become visible, the scroll arrows in the title bar of the table image automatically disappear.

4D Server If you resize a table image when using 4D Server, the table appears resized for all users in the Design environment.

Moving a Table Image

As you add tables to your structure, you may need to move the table images around to make room or reorganize the table images. You can do so by dragging the table name area. The illustration below shows a table image being dragged.



- To move a table image, drag the table name bar.

Drag the table name bar only. Dragging other parts of the table image may produce different effects, such as creating a new table relation or changing the size of the table image.

If the table image that you move is related to another table or subtable, the connecting lines move correspondingly, maintaining their attachment to the other table or subtable.

4D Server If you move a table image when using 4D Server, the table appears in its new location for all users in the Design environment.

Creating a New Table

When you create a new database, 4th Dimension automatically creates the first table in the database. You can create additional tables at any time.

4th Dimension names the first table [Table1]. 4th Dimension names additional tables sequentially, up to [Table255]. You can rename the tables at any time. See “[Renaming a Table](#)” on page 127 for more information about naming tables.

You cannot delete tables. However, if you create an unwanted table, you can make it invisible to users (for more information, see the section “[Setting Table Properties](#)” on page 125). If you accidentally choose New Table, you can cancel the operation by immediately clicking in the menu bar.

- ▶ To create a new table:
 - 1 Choose New Table from the Structure menu.
 - OR
 - Press Ctrl+N (on Windows) or Command-N (on Macintosh).
 - OR
 - Click the New Table button in the toolbar .
 - OR
 - Right-click (Windows) an empty area of the Structure editor window or press the Ctrl key while clicking an empty area of the Structure editor window (Mac OS), then choose New table from the contextual menu.

The following dialog box appears.



If you want to cancel the operation, click the Cancel button.

- 2 Enter the name of the table you want to create and click the OK button.

You can enter up to 31 characters in the name entry area. The name can include any combination of characters, dashes, numbers and

underscores. 4th Dimension removes any character past the 31st character and also removes spaces that are located at the beginning or the end of the name.

4th Dimension creates a new table image. It becomes the selected table image in the Structure editor window.

3 Repeat steps 1 and 2 for each table you want to add to the database.

Setting Table Properties

You use the Table Properties window to set several properties of each table. With the Table Properties window, you can

- Rename a table,
- Set access privileges,
- Specify the events for which the table's trigger is active,
- Customize the appearance of the table image in the Structure editor window.

Displaying the Table Properties Window

The Table Properties window displays the properties of the selected table. If it is not displayed, you first need to open it.

► If the Table Properties window is not displayed:

1 Click on the table image whose properties you want to set.

A marquee surrounds the selected table.

2 Choose Table Properties from the Structure menu.

OR

Double-click the title bar of the table image.

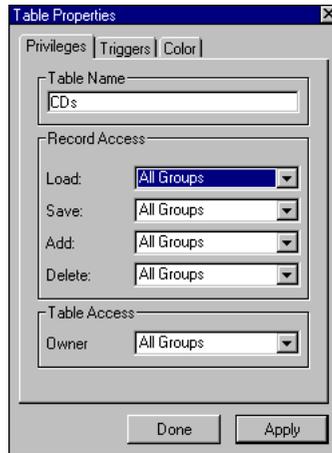
OR

Press Ctrl+R (on Windows) or Command-R (on Macintosh).

OR

Right-click (Windows) the title bar of the table or press the Ctrl key while clicking the title bar of the table (Mac OS), then select Table Properties from the contextual menu.

4th Dimension displays the Table Properties window, showing the properties for the selected table.



Note You can also display the Table Properties window by double-clicking the table name in the Tables page of the Explorer or highlighting the table name and clicking **Edit**.

- ▶ If the Table Properties window is displayed but shows the properties of another table:
 - Click on the table image whose properties you want to set. For more information about selecting a table image, refer to [“Selecting a Table Image” on page 120](#). Once you have selected the table image, the Table Properties window reflects the properties of the newly selected table.

Renaming a Table

You will usually want to rename a table so that the table name identifies the information it will contain. For example, if [Table1] will contain customer records, you might rename it [Customers].

You can rename tables at any time. If you have used the old table name in a method, 4th Dimension automatically changes it to the new name *provided the method is closed*. If the method is open, you must make the changes yourself.

Do not use the same name for two tables in the same database. If you inadvertently create a duplicate table name, 4th Dimension recognizes only the table you created first, ignoring any tables subsequently created with the same name. If you attempt to give the same name to two different tables, an alert is displayed that allows you to cancel the name entry.

- To rename a table, enter a name in the Table Name area of the Table Properties window.

You can enter up to 31 characters in the Name area. The table name must begin with a letter. The remainder of the table name can contain any combination of letters, numbers, spaces, and underscores.

4th Dimension truncates table names longer than 31 characters and removes spaces at the beginning or end of the name.

Setting Record and Table Access Privileges

The drop-down lists in the Record Access area allow you to assign groups access to various operations in the User and Custom Menus environments. Members of the group assigned Load privileges can view records in the table, members of the group assigned Add privileges can add records to the table, and so on.

The Owner drop-down list in the Table Access area allows you to give a group the ability to modify the table properties in the Design environment.

For complete information about creating a password access system and assigning access privileges, see [Chapter 9, “Managing Password Access” on page 555](#).

Setting Trigger Events A trigger is a method that runs automatically when certain events occur at the database engine level. The events are:

- Saving a new record,
- Saving a record,
- Deleting a record,
- Loading a record.

You create the trigger by creating a Table method. In the Triggers page of the Table Properties window, you specify when you want the trigger to run.

For more information about creating and editing triggers, see the section [“Triggers” on page 481](#).

► To set the events for which the trigger is active:

- Click one or more trigger check boxes.

The trigger you enter in the Method editor will run only when the events you check occur.

Note In the Design Environment page of the Database Properties dialog box, you can choose to execute triggers according to the rules for File procedures in previous releases of 4th Dimension. For more information, see the section [“Design Environment” on page 90](#).

Setting Table Attributes You can set the following two table attributes:

- **Invisible table** This check box allows you to make a table invisible in the User and Custom Menus environments. Use this option when a table is no longer being used in the database.

Note The Invisible property can also be set using the contextual menu that appears when you right-click the title bar of the table image (on Windows) or when you press the Ctrl key while clicking the title bar of the table image (on Mac OS).

Making a table invisible allows you to limit the operations that a user can perform on a table by making the table and its fields invisible in all editors and some dialog boxes that appear in the User and Custom Menus environments.

These editors and dialog boxes include the following:

- All query editors,
- Order By editor,
- Chart editor,
- Label editor,
- Quick Report editor,
- Import and Export dialog boxes,
- Apply Formula dialog box.

In each of these editors, the user is unable to see or choose the table or any of its fields. For instance, the user cannot include any fields from an invisible table in a report or label.

Note When using the editors, users have the option of saving their specifications (e.g., the query or sort they created) to disk files. In this case, any invisible tables or fields specified will be used in the operation. In addition, users can type the names of invisible tables and fields in the Apply Formula dialog box.

In the Design environment, invisible tables and fields are displayed in italics in the Structure editor window.

4D Server Object locking occurs when more than one person tries to modify the same table's properties at the same time.

If a user is modifying a table's properties, the properties are locked and cannot be modified by other users. The properties remain locked until the first user finishes modifying the properties (i.e., by clicking the Cancel or Apply buttons).

- **Completely Deleted** This option in allows you to optimize the deletion of a selection of records that are deleted with the DELETE SELECTION command.

When 4th Dimension deletes a selection, the tags of the deleted records are also cleared. A tag is a header attached to a record that carries information about that record.

When both records and tags are deleted, the process is significantly slower than only deleting the records. Therefore, you may not always wish to have the tags cleared automatically. You can choose whether or not this action is automatic.

To speed up the deletion of a large selection of records with the DELETE SELECTION command, uncheck the Completely Deleted check box. Deselecting this option tells 4th Dimension not to clear tags when it deletes.

This option cannot be set using the language.

By default, 4th Dimension deletes records and tags. If you deselect this option, 4th Dimension will not delete the record tags. Therefore, if you perform a recover by tags, you will recover the deleted records or whatever has been put in the holes corresponding to the deleted records. For this reason, it is recommended that you deselect this option only for tables that are not important, such as temporary tables. If you deselect this option and you must repair by tags, you can avoid recovering the deleted records by compacting the data table before performing the recover by tags (see the documentation for 4D Tools for information about compacting).

Setting the Color of the Table Image

The third page of the Table Properties window lets you set the color of either the table name or the background of the table image in the Structure editor. You use a Design Environment property in the Database Properties dialog box to determine whether the color you choose applies to the name or to the background. For more information, see the section [“Design Environment” on page 90](#).

The Color property lets you use color to help organize the structure of a large database. For example, you could use one color for all tables that relate to customers and customer records another color for tables that relate to inventory and inventory records.

► To set a color:

- In the Color page of the Table Properties dialog, deselect Default Color and click the desired color in the Color palette.

When you click **Apply**, either the table name or the table background changes to the color you selected (depending on the Database Property you selected).

OR

Right-click (Windows) the title bar of the table or press the Ctrl key while clicking the title bar of the table (Mac OS), then select a color from the Color palette.

The color is applied immediately.

- ▶ To set the color back to the default color.
- 1 In the Color page of the Table Properties dialog, select the default Color options.
The default color is applied when you either click the **Apply** or **Done** button.

Creating Fields and Setting Field Properties

For each table, you need to create the fields that hold the data you want to store and manage.

When you create a field, you assign it a field type that describes the kind of information that will be stored in the field. 4th Dimension uses the field type to perform different kinds of operations on the contents of the field. For example, if a field will contain a date, you will want to create it with a Date field type. Subsequently, 4th Dimension can compute date values, such as length of service or qualification for benefits. In addition, 4th Dimension can sort records in chronological order using the dates in this field. Field types are described in detail in the section [“Field Types” on page 135](#).

In addition to the field type, each field in a table can possess any of six attributes. Attributes determine conditions for entering, displaying, or modifying data in the fields. They are described in the section [“Field Attributes” on page 140](#).

With the exception of Subtable fields, after you create a field, you can return to it to change the field type or any of its attributes.

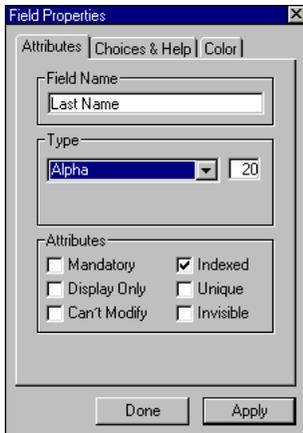
Creating New Fields Each time you add a new field to a table, you

- Name the field,
- Specify the field type,
- Set field properties (optional).

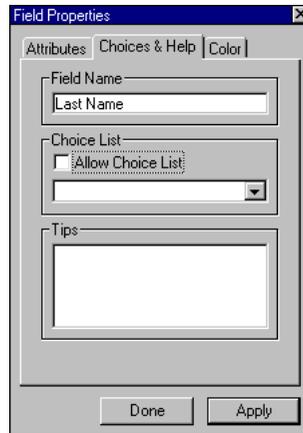
You can add up to 511 fields to a table or subtable. 4th Dimension adds fields to the table in the order that you create them. You cannot reorder or delete fields. You can, however, order the fields any way you want in the input and output forms you will use to enter and manage the data.

Refer to [Chapter 3](#) for information about creating forms.

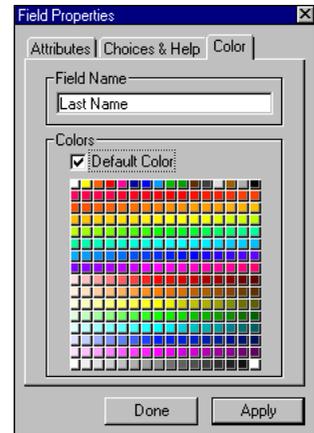
You create new fields and set field properties using the Field Properties window. The Field Properties window has three pages, Attributes, Choices & Help, and Color. The pages are shown in the following illustration.



Attributes
Set field name, type, and attributes.



Choices & Help
Assign a choice list and enter a help message.



Color
Set the field color in the table image.

You cannot delete fields. However, if you create an unwanted field, you can make it invisible to users by choosing the Invisible attribute. For more information, refer to “Field Attributes” on page 140.

- To create a field:
 - 1 Select the table image in which you want to create a field in the Structure editor and select New field from the Structure menu. Double-click an empty row in the table image below the existing field names.



Double-click on a blank area to create a field

OR

Select the table image in which you want to create a field and press the Ctrl+F (on Windows) or Command-F (on Macintosh). Choose New Field from the Structure menu.

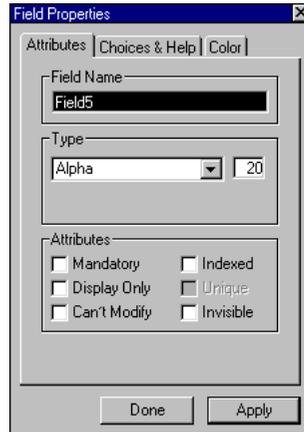
OR

Select the table image in which you want to create a field and click the New Field button in the Toolbar .

OR

Right-click (on Windows) the title bar of the table or press the Ctrl key while clicking the title bar of the table (Mac OS), then select New field from the contextual menu.

4th Dimension displays the Attributes page of the Field Properties window.



2 Type the field name in the Field Name area.

You can enter up to 31 characters in the Field Name area. The field name must begin with a letter. You can use any letters or numbers, the space, and the underscore. 4th Dimension truncates field names longer than 31 characters and removes spaces at the beginning or end of the name.

Do not use the same name for two fields in the same table. If you inadvertently create a duplicate field, 4th Dimension displays an alert that prevents you from reusing same field name in a table. Do not leave a field name blank or use any reserved words in the field name. Reserved words include command names (e.g. Date, Subtotal) and keywords (e.g. If, While, Repeat).

Tip Although you can include spaces in field names, entering a field name that does not include any spaces allows you to double-click to select the field name in the Method editor. Use the underscore character instead of a space.

3 Select a field type and, if necessary, modify the maximum field length.

For more information on field types and field lengths, see the section [“Field Types” on page 135](#).

- 4 Select any attributes (optional).

For more information, see the section [“Field Attributes” on page 140](#).

- 5 Click the Choices & Help tab and enter a help message for the field or assign a choice list (optional).

Balloon Help can be viewed only when the database is used on a Macintosh. A tip is displayed on all platforms. For more information, see the section [“Help” on page 147](#).

A choice list is displayed automatically when the field is selected for data entry or when the field is used in the Query editor. A choice list can be assigned in the Field Properties window or can be assigned on a form-by-form basis in the Form editor. For more information, see the section [“Using Choice Lists” on page 333](#).

- 6 Click the Color tab to assign a color to the field (optional).

The field color can be used in the Structure editor to distinguish fields. For example, you can use a color to highlight the field that uniquely identifies each record. The color is applied to either the text of the field name or the background rectangle containing the field name, depending on the Color Field Names or Background database property. For more information, see the section [“Setting the Color of the Field” on page 148](#).

- 7 When you are finished assigning properties, click the Apply button to save the field properties.

You do not have to click **Apply** separately on each page. Clicking **Apply** once after you have entered all your field properties saves the properties on all three pages.

When you click **Apply** after creating a new field, 4th Dimension automatically displays default properties for another new field. The default name is *FieldN*, where N is the sequence number of the field in the table. If you want to create the new field, modify the default properties.

When you are finished creating fields, click **Done** instead of **Apply**. When you click **Done**, 4th Dimension puts away the Field Properties window.

With the Field Properties window on-screen, you can modify the properties of any existing field simply by clicking the field in its table image or double-clicking its name in the Tables page of the Explorer. You can also cycle through tables by clicking one table image and

pressing the Tab key. You can navigate through the fields using the following keys.

- Up and Down arrow keys: to go up and down in the field list of the selected table.
- Home and End keys: to go at the top or bottom of the field list of the selected table.
- PgUP and PgDn: to display the next or previous sequence of fields in the selected table image.

4D Server Object locking occurs when more than one user tries to modify the same field definition at the same time. Field properties are modified in the Field Properties window, which appears when you choose either New Field or Field Properties from the Structure menu.

If a user is modifying a field's properties, the properties are locked and cannot be modified by other users until the first user unlocks the field properties by clicking the Done button).

In addition, while the field properties are locked, other users cannot modify the properties of other fields in that table. Other users can, however, modify the table properties and the properties of other fields and tables in the database.

Field Types

You must specify a field type for each field. Field types affect how 4th Dimension manipulates and stores data in a field and how you enter or display data in forms. 4th Dimension supports the following field types:

- Alpha Alphanumeric text between 2 and 80 characters,
- Text Text up to 32,000 characters,
- Real Floating point number in the range of 1.9E-4951 to 1.1E4932,
- Integer Number in the range of plus or minus 32,767,
- Long integer Number in the range of plus or minus 2,147,483,647,
- Date Date between the year 100 and the year 32,767,
- Time Time in hours:minutes:seconds format,
- Boolean A field that can only take the values TRUE or FALSE,
- Picture A PICT image,

- **BLOB** Any binary object such as a graphic, another application, or any document.

You set the field type in the Field Properties window. Here are complete descriptions of each field type.

Alpha

An Alpha field contains alphanumeric characters (letters and numbers), punctuation marks, and special characters such as the asterisk (*), percent sign (%), hyphen (-), and so on. Use an Alpha field to contain any information that must be treated as text and does not exceed 80 characters in length.

An Alpha field can be indexed (whereas a Text field cannot be indexed). For information about indexing, see the section [“Indexed” on page 142](#).

Alpha is the most common field type. Typically, you use this field type for names, addresses, telephone numbers, postal codes, and so forth. During data entry, an Alpha field accepts any character, number, punctuation mark, or special character.

Zip codes are best placed in an Alpha field for two reasons: Numeric fields do not display leading zeros and some zip codes contain a hyphen. The general rule for deciding between a numeric field type or an alphanumeric field type is make it an alphanumeric field unless it will be used in a numerical calculation or searched or sorted based on numeric values.

You can set the maximum length of an Alpha field to be between 2 and 80 characters long. For example, if you use a field for state abbreviations, you could limit it to 2 characters.

You can concatenate two or more Alpha fields. For instance, you might want to join a person’s first name and last name for the first line in a label form. You can do so using a one-line method, such as:

```
FullName:=First_Name+" "+Last_Name
```

The variable FullName can be displayed or printed. You can also extract part of the information for use in another place (extraction of a substring). The substring can be displayed or printed.

Text

A Text field can hold up to 32,000 alphanumeric characters. You use a Text field to hold blocks of text longer than 80 characters such as comments or descriptions.

In an input form, a Text field can be given a vertical scroll bar. In a printed report, the Text field area can expand as necessary to print all the information, even if it covers several pages.

During data entry, Text fields provide basic text editing features: scrolling, word wrapping within the area set for the field display, double-clicking to select a word, moving the insertion point with the arrow keys, and standard cut, copy, and paste operations. A Text field accepts a carriage return during data entry to create a new paragraph (an Alpha field does not).

You can paste text into Text fields, including text from word processors.

You cannot index a Text field, but you can perform a search based on characters in the field.

Another way to store text with a record is to use the 4D Write plug-in. With 4D Write, you can use different font attributes, paragraph alignments, and other word processing features that are not available in standard Text fields. For more information about using 4D Write, refer to the documentation that comes with 4D Write.

Real

A Real field stores real numbers, that is, decimal numbers (price, salary, expenses, and so on). Real number fields can hold any number in the range of 1.9E-4951 to 1.1E4932.

Note In the United States, the decimal separator in real numbers is the decimal point (.). In other countries, it is often a different character such as the comma (,). If you are using a non-U.S. version of 4th Dimension, real numbers are stored based on the decimal separator specified in the operating system of your machine.

Integer

Use an Integer field type for any field that stores whole numbers, that is, numbers without decimals (record number, invoice number, and so on). Integer fields can contain whole numbers between $\pm 32,767$.

Long Integer

Use a Long Integer field type for any field that stores whole numbers that are too large for an Integer field. They can contain whole numbers (no decimal) between $\pm 2,147,483,647$.

Date

Use a Date field to store date values such as Start Date, Date Purchased, Birthdate, and so on. A Date field can store any date value (month, day,

year) entered in a *MM/DD/YYYY* format between the year 100 and the year 32,767.

Note In the United States, dates are specified in the month/day/year (*MM/DD/YYYY*) format. Other countries use different formats such as *DD/MM/YYYY* for British systems and *YY/MM/DD* for Swedish systems. If you are using a non-U.S. version of 4th Dimension, 4th Dimension stores the date based on the date format specified by the operating system of your computer.

Time Use a Time field type to manage times such as Current Time, Meeting Time, Billed Time, and so on. A Time field can store any time value entered in *HH:MM:SS* format.

Boolean Boolean fields (sometimes called logical fields) contain TRUE or FALSE values.

You can format a Boolean field as either a check box or as a pair of radio buttons. A check box that contains a check is TRUE; empty, it is FALSE. Either the first radio button is selected (TRUE), or the second button is selected (FALSE).

You should name a Boolean field so that you can ask the question, “Is *field name* true?” This question is useful for searching because during a search, 4th Dimension looks for a TRUE and FALSE value in a Boolean field. For example, you might want to name a field “Male” instead of “Sex.” Your search condition can then be written “Male is equal to true,” instead of “Sex is equal to true.”

Picture Picture fields are used for several purposes in 4th Dimension. Creating a Picture field allows you to save the following types of data.

- **Pictures** You can store digitized photographs, diagrams, maps, and illustrations created using a graphics application. Some graphic applications store extra information with pictures that may provide special instructions for output devices such as a PostScript™ printer. This information “tags along” when the picture is copied or pasted into a Picture field and is used by 4th Dimension when printing the picture to an appropriate output device.
- **Data from 4th Dimension plug-ins** You can store data created with some of the 4th Dimension plug-ins — 4D Chart, 4D Write, 4D Draw, and 4D Calc. For more information about using a 4th Dimension plug-in a Picture field, refer to the documentation that comes with the plug-in.

- **Data from third-party plug-ins** You can also store data from third-party plug-ins (if the plug-in supports saving data). For more information about storing data from a plug-in in a Picture field, refer to the documentation included with the 4th Dimension Plug-ins Kit.

Note Data that comes from third-party plug-ins can also be stored in Blob fields.

For more information on this feature, refer to the documentation that comes with the plug-in.

Blob

Blob (Binary Large Object) fields store binary documents of any kind. For example, you can store documents created by other applications, scanned images, or other applications. A Blob can be as large as 2 gigabytes. When you are working with a record that contains a Blob field, the entire Blob is loaded into memory. You can use a Blob field to store entire desktop documents within your database. You can also write the contents of a Blob field to a desktop document. For example, you can use a Blob field in a document management system that stores documents in the database and delivers them to users upon request.

You use Blob commands in 4th Dimension's language to manage Blob fields. Use the DOCUMENT TO BLOB and BLOB TO DOCUMENT commands to read and write documents to and from Blob fields. The commands COMPRESS BLOB, EXPAND BLOB, and BLOB PROPERTIES let you work with compressed Blobs. For more information about working with Blobs, see the section on Blobs in the *Language Reference*.

The contents of a Blob field is not displayed on-screen since a Blob can represent any type of data.

Subtable

A Subtable field is a field that associates a subtable with each individual record in a table. For example, a [People] table could include Children as a Subtable field. The subtable associated with the field — also called Children — could contain fields that store data on each person's children (their names, ages, birth dates, and so on). A single subtable can contain up to 32,767 subrecords and each subrecord can have as many as 511 subfields.

The record to which a subtable is attached is called the parent record and the table that contains the parent record is called the parent table. Each subtable has its own set of fields, called subfields.

You can use subtables to manage a variable number of subrecords. For example, you may want to create a student table in which each record contains data about the student's honors and the dates of the awards. Instead of having to create a number of fields such as Honor 1, Honor 2, and so on, you can create a subtable to which you can add subrecords as the awards accumulate.

In many cases, however, it is best to use a related table instead of a subtable. Subtables have several limitations that related tables do not. First, a subtable cannot be viewed without opening the parent record. Second, it is difficult to use information across subrecords. For instance, you could easily produce a sorted list of each student's honors but it would be difficult to produce a sorted list of all honors awarded to all students. If you need to generate this type of sorted list, you should use a related table.

In general, you should not use a subtable to store information that you will need to search on, access directly, or use for calculations.

Because 4th Dimension loads subtables into memory when it loads their parent records, the number of subtables and subrecords is limited by the amount of available memory. A good rule of thumb is to allow no more than 100 subrecords per parent record. If processing speed is of concern, allow no more than 25.

You add subfields to a subtable the same way that you add fields to tables. You cannot create more than one level of subfields.

After you define a field as a Subtable field, you cannot assign it another field type. However, you can make the field invisible in the User or Custom Menus environment (for more information, see the section [“Invisible” on page 143](#)).

Field Attributes

Field attributes determine conditions for entering, displaying, or modifying data in the field. Each field can have several attributes.



You set the field attributes in the Attributes page of the Field Properties window. Attributes that cannot be selected for a particular field type are disabled. A description of each of the field attributes follows.

Note The Invisible and Indexed attributes can also be set for a field using the contextual menu that appears when either right-clicking a field in the structure window (on Windows) or pressing the Ctrl key when clicking the field in the Structure editor window (on Mac OS).

Mandatory

When the Mandatory attribute is set for a field, the user must enter a value in that field during data entry. 4th Dimension does not accept a record that contains an empty mandatory field. You would set the Mandatory attribute for a field that contains essential information for your database. The field that uniquely identifies each record is a good candidate for the Mandatory attribute. Social Security numbers, invoice numbers, certain dates, or employee numbers might need to have the Mandatory attribute set to protect the integrity of the records.

You can also set the Mandatory attribute for a field in a particular form. If you select the Mandatory attribute in the Structure editor, you cannot deselect it on a particular form. However, you can apply the Mandatory attribute on a form to a field that does *not* have this attribute in the Structure editor. For information about setting the Mandatory attribute for a field in a form, see the section [“Setting the Enterable and Mandatory Attributes” on page 332](#).

Display Only

The user cannot enter values from the keyboard into a field that has the Display Only attribute set. You must use a default value for such a field or write a method that inserts a value in the field. A field with the Display Only attribute is useful for displaying values that you do not want database users to modify, such as calculated totals or a sequence number assigned by a method.

You can also make any field non-enterable on a particular form. For information about making a field non-enterable, see the section [“Setting the Enterable and Mandatory Attributes” on page 332](#).

Can't Modify

If the Can't Modify attribute is set for a field, 4th Dimension accepts the value initially entered in the field, but does not allow the user to modify the value after the record has been saved. The user can edit an entry in such a field only during the initial creation of the record, before the record is accepted. Once the user saves the record, the value in the field

not editable. The value can be modified by a method or by returning to the Design environment and removing the attribute.

Use Can't Modify for fields that must provide an audit trail such as Date Received, Date Paid, and so on. The Can't Modify attribute is often used for the field that uniquely identifies each record in the table.

Indexed

You should use the Indexed attribute for fields that you frequently use for searching and sorting. For example, you might index Last Name, Company name, or Product name if you plan to search for specific records or sort the records by these fields. You also use this attribute for fields that establish relations between tables. For more information, see [“Setting Relation Properties” on page 158](#).

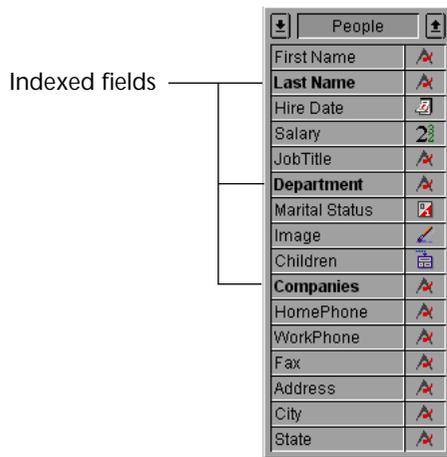
The Indexed attribute causes 4th Dimension to create an internal index table for the field. The table allows 4th Dimension to perform rapid searches and sorts on the field. When searching or sorting on an unindexed field, 4th Dimension moves through data sequentially, examining each record in order. An index allows 4th Dimension to search and sort without going through every record.

Note When selecting the Indexed attribute for a field in a table that has over 1,000 records, 4D allows you to choose between two indexing modes. For more information, refer to [“Indexing or Reindexing a Field” on page 150](#).

You can index Alpha, Real, Integer, Long Integer, Time, Boolean, and Date fields. As you add and delete records, 4th Dimension automatically updates its index table. If you assign the Indexed attribute to an existing field, 4th Dimension automatically indexes the existing data when you leave the Design environment. You can specify as many indexed fields as you want.

Do not index every field. An index increases the size of the database, using more space on disk. Using many indexes also increases the time needed to save a record since 4th Dimension updates the index table with each entry.

Indexed fields are displayed in bold type in the Structure window.



Unique

Use the Unique attribute when you want to be certain that each record has a different (unique) value in this field. The Unique attribute should be used for the field that uniquely identifies each record in the table. The Unique attribute is useful to validate fields that store Employee numbers, Social Security numbers, Purchase Order numbers, and so on.

If you want to set the Unique attribute for a field, you must first make it an indexed field. The Unique attribute is disabled unless the Indexed attribute is set.

The Unique attribute prevents duplication of empty values as well as actual entries. An empty field cannot be duplicated in another record.

Note If you apply the Unique attribute to a subfield (a field in a subtable), the attribute ensures that no subrecord contains a duplicate value. The Unique attribute applies to all subrecords, not just the subrecords for each parent record.

Invisible

You can make a field invisible in the User environment and custom applications by selecting the Invisible attribute for the field. Use the Invisible attribute if you inadvertently create an “extra” field or if a field is no longer needed in the current structure. The Invisible attribute hides the field from the user. A field with this attribute does not appear in all standard 4th Dimension editors and dialog boxes that appear in the User and Custom Menus environments.

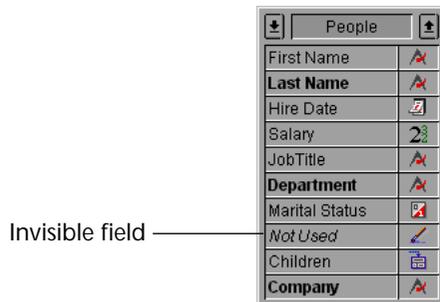
These editors and dialog boxes include the following:

- All query editors,
- Order By editor,
- Chart editor,
- Label editor,
- Quick Report editor,
- Import and Export dialog boxes,
- Apply Formula dialog box.

In each of these places, the user is unable to see or choose the field. For instance, the user cannot choose an invisible field for a report created with the Quick Report editor.

Note When using the editors, users have the option of saving their specifications (e.g., the query or sort they created) to disk files. In this case, any invisible fields specified will be used in the operation. In addition, users can type the names of invisible fields in the Apply Formula dialog box.

Invisible fields are displayed in italics in the Structure editor window.



Compression (Mac OS only)

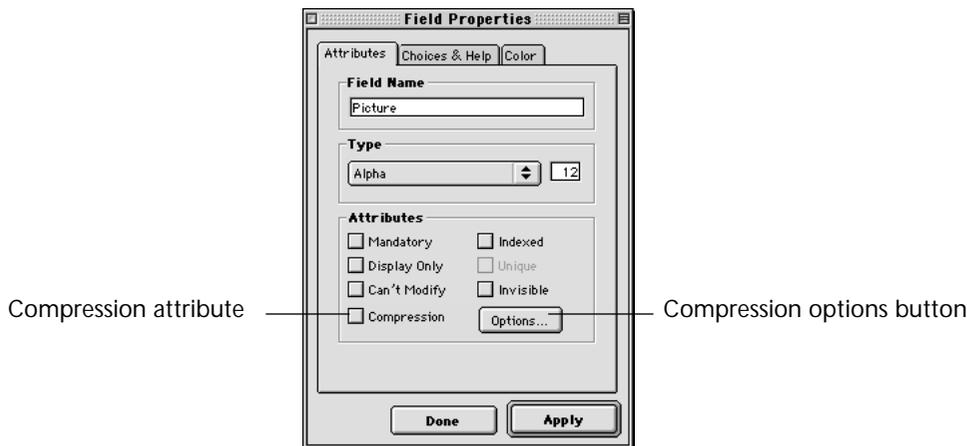
On Mac OS, you can apply a QuickTime™ compression to picture fields. Selecting that attribute lowers the disk space required for storing pictures.

On Windows, you cannot compress pictures using QuickTime compression routines. However it is still possible to unpack pictures that were compressed using QuickTime™ by using QuickTime for Windows. Refer to 4D's ReadMe files for the current compatibility information between QuickTime™ for Windows and 4th Dimension.

The Compression attribute can be selected under the following conditions:

- The operating system is Mac OS.
- The QuickTime™ extension (version 1.6 or later) is currently installed in the System folder.
- The field type is Picture.

The Compression attribute is accessible in the Attribute page of the Field Properties dialog.



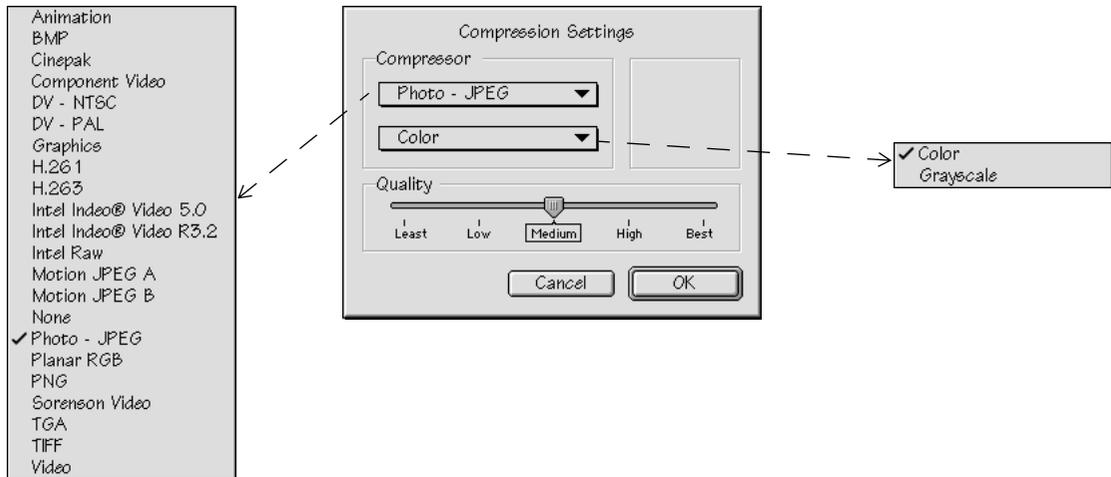
If you check the Compression Attribute without setting any compression options, the default compression is used.

To remove picture compression, uncheck the Compression box.

Note QuickTime™ compression cannot be used with picture fields used to store data from plug-ins.

The Options button allows you to set the QuickTime™ compression settings. The contents of the dialog that is displayed when you click

the Options button will depend on the version of QuickTime™ that is installed on your machine. The following dialog box is from QuickTime 4.



Note The settings you define in the Compression Settings dialog will apply to all the records of the table—for that field. However, the user is given the option to use different settings when pasting a picture into a field in the User environment. To do so, press the Option key while pasting the picture in the field to display the Compression Settings dialog.

The QuickTime extension manages several compression methods, shown in the above illustration. For more information about these methods, refer to the QuickTime documentation available online from Apple Computer.

Note For more information about storing pictures, refer to the 4th Dimension *Language Reference Manual*.

Choices and Help

The second page of the Field Properties window lets you assign a choice list to the field and enter a help message.

Choices

Use the Choices attribute if you want to display a choice list for entering information in the field. To use this attribute, you first need to create the choice list using the List editor.

Use the Choices attribute when you want to standardize entries in the field and avoid misspellings. Use a choice list for a field that has a limited number of valid entries or a limited number of usual entries. Using a choice list does not necessarily prevent the user from typing a different value (one that does not appear on the choice list). For more information about choice lists, see [Chapter 10](#).

You can also assign a choice list to a field on a particular form. However, when you assign a choice list only on a form, the list is not displayed in other editors and dialog boxes, such as the Query editor. For information about using a choice list in a form, see the section [“Using Choice Lists” on page 333](#).

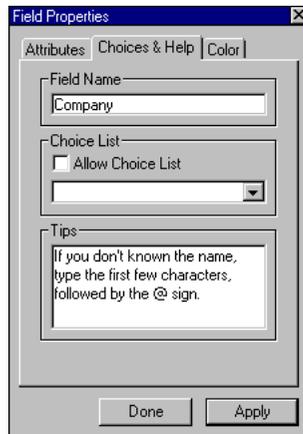
Help

You can provide users with additional information about a field by adding a Tip or Balloon Help to the field. When the database is used on a Macintosh with Balloon Help turned on, the Balloon Help appears next to the field whenever a user places the pointer over the field in any form in which the field is included. A Tip is displayed on all platforms.

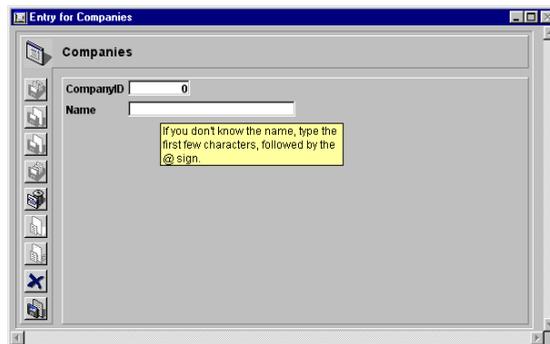
Note You can create Balloon Help when you are designing a database on Windows, but the Balloon Help can be displayed only on Macintosh running System 7.0 or greater, with Balloon Help turned on.

- ▶ To add a help message to a field:
 - 1 Click the Choices & Help tab in the Field Properties window. The Choices & Help page appears.
 - 2 Enter the text in the Help message area and click Apply.

The figure below shows text being entered into the message area. In this example, the help message is prompting the user about using the wildcard character during data entry.



When the user selects the field, the tip appears below the field, as shown below.



Setting the Color of the Field

The Color page of the Field Properties window lets you set the color of either the field name or the background of the row in the table image in which the field appears. You use a Design Environment property in the Database Properties dialog box to determine whether the color you choose applies to the name or to the background. For information on the Color Names or Background properties, see the section [“Design Environment” on page 90](#).

- ▶ To set a color:
 - In the Color page, deselect Default Color and click the desired color in the Color palette.

OR

Right-click (on Windows) the field name of the table or press the Ctrl key while clicking the field name of the table (Mac OS), then select the new color from the Color palette.

Modifying Fields and Field Properties

You can change a field's name or properties at any time, whether or not you have entered data into the field.

► To modify a field's properties:

1 Double-click the field name in the Structure editor.

OR

Select the field you want to modify and choose Field Properties from the Structure menu.

OR

Double-click the field name in the Tables page of the Explorer.

OR

Click the field name in the Tables page of the Explorer and click Edit. 4th Dimension displays the Field Properties window for that field. The window displays the field name and properties that you have previously assigned to the field.

2 Make any necessary changes.

3 Click Apply to save the changes to the field properties.

Renaming a Field

If you change the field name, 4th Dimension automatically updates the name anywhere the field is used (e.g, forms, methods, and disk files containing specifications from the 4th Dimension editors). Field labels on forms are not changed.

Field names are updated in methods as long as the methods are closed when the name is changed.

Changing a Field's Type

If you change the field type before entering any data into the field, 4th Dimension simply changes the field type. The only restriction is on the Subtable field type which cannot be changed to any other field type.

If you change the field type after entering data into the field, 4th Dimension converts the data to the new type if possible, when the data is loaded for the first time after the change. Data from a Picture field converted to any other type does not display. Data from a field converted to a Picture or Subtable field type does not display.

When you convert a field type, 4th Dimension retains the field's original value until you modify the record. For example, if a Text field contains text such as "over 10" and you change the field type to Integer, the modified field displays "10." If you change the field back to a Text field without editing the field value, 4th Dimension displays "over 10" again.

Changing a Field's Attributes

If you change field properties, the change does not usually affect data already entered into the field. For example, if you set the Unique attribute, only entries made after that are checked for uniqueness; field entries made up to that point may include duplicates. However, if you set the Indexed attribute, all field values (both old and new) are included in the index.

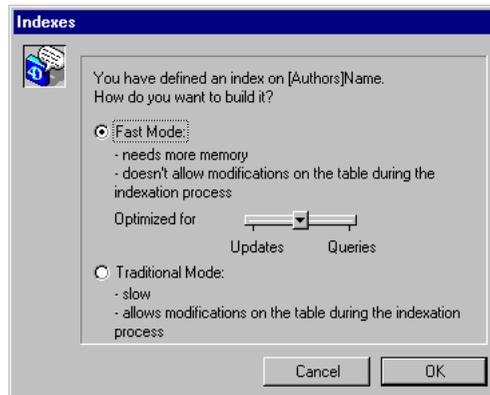
Indexing or Reindexing a Field

If you index a field (by selecting the Indexed attribute, by selecting the Reindex command of the Structure editor contextual menu or when defining a relation between two tables), both the new and old (created before the indexing) values will be indexed.

When indexing a field that has more than 1,000 records, 4th Dimension lets you choose between two indexing modes: the "traditional" mode, which is the mode used in previous versions of 4th Dimension, and the new "fast" mode, which in most cases is significantly faster.

The choice between the two modes is only given for tables that have over 1,000 records because below that value, the gain in speed is not

significant. The mode selection dialog is displayed each time you index or reindex a field.



To select a mode, click the corresponding button. If you select the Fast Mode, the optimization parameter must be set, depending on how the indexed field will be used. You use the slider to set the parameter to the desired value between two extremes:

- **Updates:** The index will be updated very frequently because data entry is being done.
- **Queries:** The index be updated rarely because the datafile is largely complete and the database is primarily used for searches, sorts, reports, etc.

The Optimization slider determines how the index is optimized, not its definition. Even if you place the slider to the extreme right for Queries, the index will still be modified when data is modified. However, its performance will not be optimal for data entry.

Which Mode to Choose? Each mode has its advantages and inconveniences:

Comparative table of the two index modes

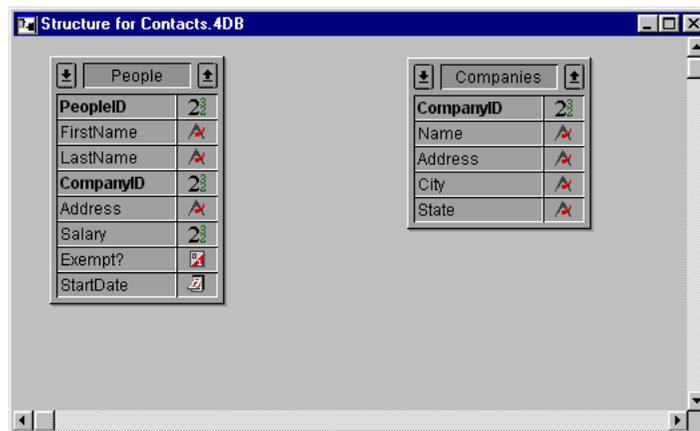
	<i>Advantages</i>	<i>Inconveniences</i>
<i>Fast Mode</i>	- Fast construction ¹ of the index - Optimization possibility according to usage	- Needs more memory - "Locks" any modifications to the table during the indexing process
<i>Traditional Mode</i>	- Needs less memory - Allows you to modify the table during the indexing process	- Slow construction of the index - No optimization possible

1. The increase in speed depends mainly on the number of records to index, the fragmentation of the data, and the available memory (RAM).

To summarize, the fast mode (which is selected by default) is more efficient and can be optimized. It should be used in most cases. To obtain the maximum increase in speed, the indexing must be done as often as possible in memory (you can verify, in the Evaluation page of the Runtime Explorer, if the memory cache is sufficient to contain the index). However, if you don't have enough memory or if you don't want a table in your database to be locked (in write mode) for a certain amount of time, you can choose the traditional mode.

Relating Tables

You will usually need to create structures in which several tables share information. For instance, suppose you create a database to keep track of people and their companies. The database structure, shown below, contains a [People] table for storing people information and a [Companies] table for storing company information.



Although useful, the information stored in each separate table does not fulfill your information tracking needs. When you are viewing a record from the [People] table, you need to be able to view information about the company for which that person works and when you are viewing a record from the [Company] table, you need to be able to view information about all the people who work for that company.

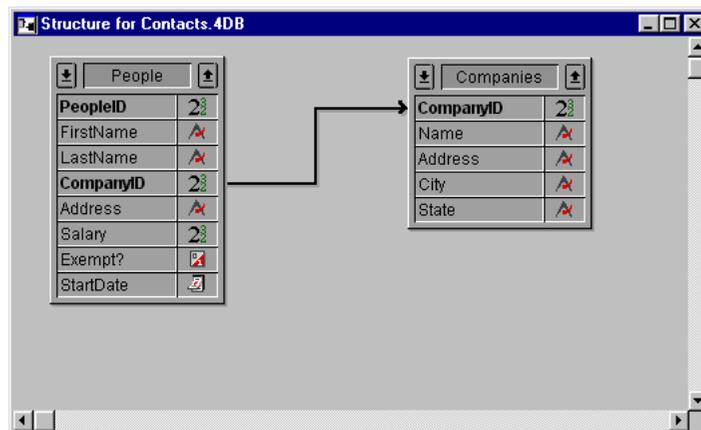
To allow two tables to share information in this way, the tables can be related to each other — in other words, a relation can be established between the data in each table.

In 4th Dimension, *table relations* allow data stored in one table to be accessed from another table. Tables that share information by means of a relation are called *related tables*.

Relating tables allows you to do the following:

- Store data efficiently,
- Update data in one place and have the change reflected everywhere the data is used,
- View related information,
- Perform queries and sorts in one table that are based on data in another table,
- Create, modify, or delete records in related tables.

The figure below shows a relation created between the [People] table and the [Companies] table in the Structure editor.



The [People] table contains one record per person. The [Companies] table contains one record per company. The relation between the two tables allows you to access, enter, modify, or delete information from both tables. For example,

- When a person's record is onscreen, you can view or modify the corresponding company information — the address, city, state, zip code, and company telephone number.

- When you add a new person, you can link the person's record to the appropriate company record (if the company is already entered), or, if the person's employer is not in the database, create the new company record while creating the person record. For more information, see [“Entering Data in Related Tables” on page 170](#).
- For each company, you can view or modify information for each person in the company — name, title, telephone number, and so forth. You can also add a person record from within the company record. For more information, see [“One to Many Properties” on page 161](#).

Related Fields

You are able to display information from related tables by means of the related fields — the fields that connect the two tables in a relation.

The basic purpose for relating tables is to instruct 4th Dimension which record or records to make current in one table based on which record is current in the other table. The related tables make use of data in two related fields to identify corresponding records. In the following example, the company name is stored in both the [People] table and the [Companies] table.

Related fields

First Name	Last Name	Title	Company
Biff	Davis	Salesperson	Howard Battery Co.
Andy	Venable	Engineer	Howard Battery Co.
Bryan	Pfaff	Secretary	Howard Battery Co.
Kathy	Forbes	Secretary	Howard Battery Co.

Company Name	City	St	Phone
Howard Battery Co.	Arcadia	CA	818-576-2534

The Company field in the [People] table and the Company Name field in the [Companies] table relate the two tables. The Company Name field in the [Companies] table is the *primary key field* for [Companies]. It uniquely identifies each company record. A primary key should have the Indexed and Unique attributes. The Company field in the People table is a *foreign key field*. Each value of the foreign key field matches exactly one value of the primary key field in the related table. A foreign key should have the Indexed attribute. If both the primary and foreign key fields do not have the Indexed attribute when you create the relation, 4th Dimension assigns this attribute automatically.

Each value in a foreign key field is equal to one value of the primary key field in another table. In this example, a value of the foreign key field in [People] matches exactly one value of the primary key field in [Companies]. The foreign key field is also indexed but its values are non-unique (e.g., several people may work for the same company).

In some database designs, the values of the primary key field are assigned by the database automatically — either by assigning a sequence number that 4th Dimension generates or by a user-written method. Such a procedure guarantees the uniqueness of the key field. For example, if the primary key field in the [Companies] table is a sequence number rather than the company name, it would be possible for users to enter several companies with the same name but different addresses. Also, if a company name changes, the user could make the change to the database without disturbing the relation between the two tables.

If the user is permitted to enter the value of the primary key field, you should use both the Unique and Can't Modify attributes to check for uniqueness of the initial entry and to prevent users from subsequently changing the entry to a non-unique value. If you elect not to use the Can't Modify attribute, you will need to take other measures to prevent users from creating “orphaned” records in any related tables by making changes to the values of the primary key field.

When relations are established, you can read and write values in one table while working in the related table. For example, when you enter a company name in a person's record, 4th Dimension searches for that company in the [Companies] table and displays the company address and phone number in that person's record. When you view a company's record, 4th Dimension searches in the [People] table for all the people who work at that company and displays their records in the company record.

These relations can be invoked automatically (i.e., with no programming on your part) or you can choose to use manual relations. In the latter case, you use methods to load and unload related records and control the creation, modification, or deletion of related records. Manual relations are sometimes preferable in complicated structures in which more than two tables are related to one another and you need to control the loading and unloading of related records.

You can choose to use automatic relations by selecting the appropriate properties at the time the relation between the tables is specified. For more information, see the section [“Automatic and Manual Relations” on page 170](#).

The One Table and the Many Table

When you create a relation between two tables, the table containing the primary key in the relation is called the *One table* and the table containing the foreign key in the relation is called the *Many table*. The tables are called the One table and the Many table because one record in the One table relates to many records in the Many table and many records in the Many table relate to one record in the One table. This type of table relation is called a *Many to One* relation.

In the relation between people and companies, the [Companies] table is the One table and the [People] table is the Many table. One company record relates to several people (i.e., all the people who work for that company) and several people relate to one company (i.e., the company for which they work). For instance, there may be one record for Acme in the [Companies] table but many records of people employed by Acme in the [People] table.

When any record in the [People] table is made current, 4th Dimension loads the corresponding single record from the [Companies] table. If any fields have been included from the [Companies] table, the values for these fields are automatically displayed. For information about including fields from other tables, see the section [“Selecting Fields from Related One Tables”](#) on page 200.

The figure below shows how the company name in a [People] table record specifies one record in the [Companies] table so that the [People] table record can display the company’s address and phone number.

The screenshot shows a window titled "Entry for People" with a "People" tab and "9 of 53" records. The form contains the following fields:

First Name	Kathy
Last Name	Forbes
Job Title	Secretary
Name	Howard Battery Co.
Address	245 Arcadia Ave.
City	Bad Axe
State	MI
Zip	48070

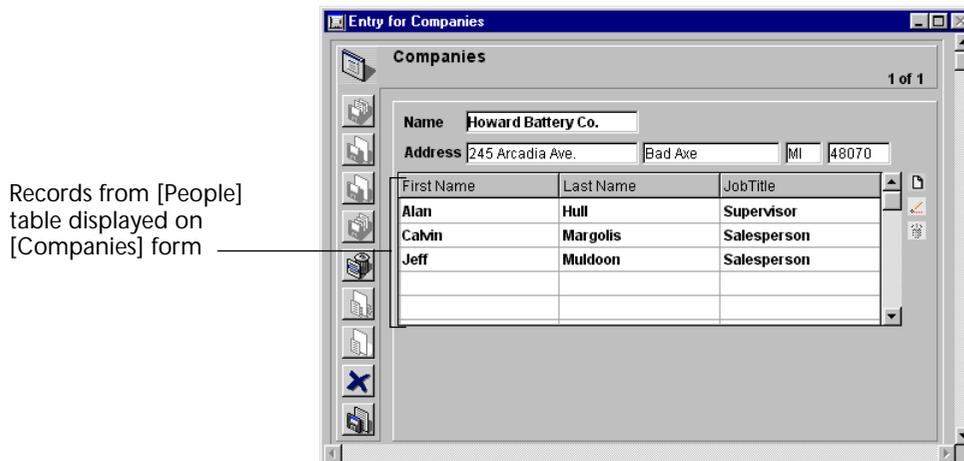
Company name specifies one company in [Companies] table

Data from [Companies] table displayed on [People] form

Conversely, when a record in the [Companies] table is made current, 4th Dimension creates a selection of records in the [People] table and displays them on the form. Because the relation specifies several records in the other table, the names and titles of many people can be displayed.

Note Only those records currently displayed on the form are loaded into memory.

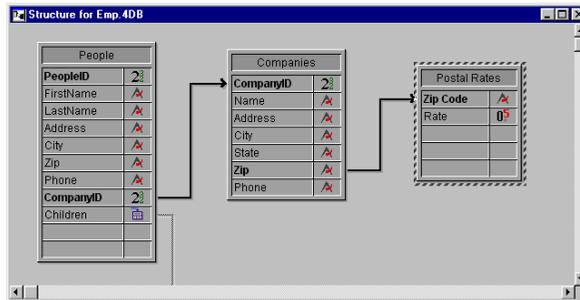
The figure below shows how a company name in a record in the [Companies] table specifies several records in the [People] table so that the [Companies] table record can display a list of people employed by that company.



The distinction between the One table and the Many table is specific to a particular relation. A table may be the One table in one relation and the Many table in another. A table in a relation need have only one primary key, but it can have several foreign keys.

For example, suppose you decide to send a package of sample merchandise to everyone in your [People] table. You add a [Postal Rates] table that contains zip codes and the postal rate for each zip code. Using this structure enables you to print an address label for each person that includes the amount of postage needed to mail the package.

The figure below shows the [Postal Rates] table added to the database structure.



The Zip Code field in the [Postal Rates] table is its primary key, so the [Postal Rates] table is the One table. The Zip field in the [Companies] table is the foreign key field for this relation. Since the Zip field is a foreign key, it can have non-unique values. The Zip field will contain duplicate Zip codes for companies that are near each other. The [Companies] table is therefore the Many table in relation to the [Postal Rates] table.

Whether a table is a One table or a Many table, therefore, depends on its relation to the other table. The [Companies] table is the Many table in relation to the [Postal Rates] table and it is the One table in relation to the [People] table.

Setting Relation Properties

You must have at least two tables in your database to create a relation. You create a relation by drawing a line between two fields.

The field where you start drawing must be a foreign key field in the Many table and the field where you end must be the primary key field in the One table. Using the Company database example in this chapter, you would start drawing in the [People] table and end in the [Companies] table. Remember,

You always draw a relation from the Many table to the One table.

The related fields must have the same field type. For example, the Company field in the [People] table and the Company Name field in the [Companies] table can be related because they are both Alpha fields.

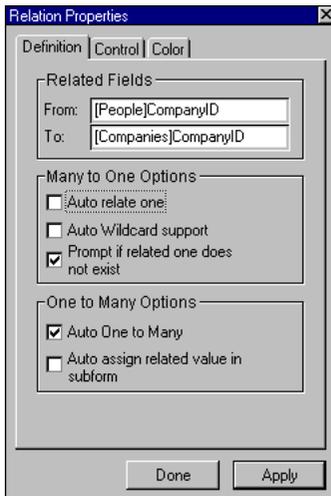
You can use these field types for the primary and foreign key fields:

- Alpha,
- Number (Real, Integer, or Long Integer),
- Time,
- Boolean,
- Date.

Primary and foreign key fields must be indexed. If the fields do not have the Indexed attribute set, 4th Dimension automatically indexes them when you switch to the User environment.

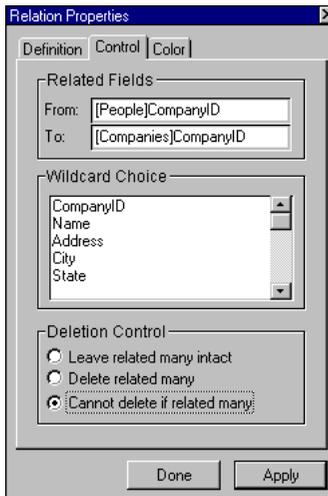
Note If you draw a relation between tables that have 1,000 records, you will be prompted to select an indexing mode. For more information, refer to [“Indexing or Reindexing a Field” on page 150](#).

You specify various properties of the relation with the Relation Properties window. It has three pages, Definition, Control, and Color.



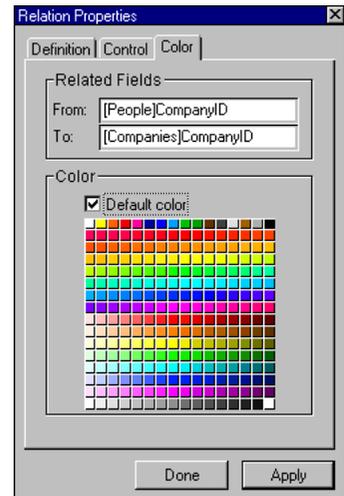
Definition Page

Specify the primary and foreign keys.
Set automatic relation options.



Control Page

Specify the wildcard choice field in the One table.
Set Deletion Control options.



Color Page

Specify the color of the relation line in the Structure editor.

Here are descriptions of the relation properties:

Related Fields

The Related Fields area identifies the foreign and primary key fields:

- The “From” field is the foreign key field in the Many table for this relation,
- The “To” field is the primary key field in the One table,
You draw the relation line *from* the foreign key field in the Many table *to* the primary key field in the One table.

Many to One Properties

The Many to One properties affect what happens when a record from the Many table is opened:

- **Auto Relate One** This check box establishes automatic relations from the Many table to the One table. For example, when a record from the [People] table is opened in the User environment, the related company in the [Company] table is selected. This allows 4th Dimension to display information about the company for which the employee works if you so desire.

If you deselect Auto Relate One, you can manage the loading and unloading of the related One record using commands in the language.

- **Auto Wildcard Support** This check box has the effect of invisibly appending the wildcard character (@) to any value entered in the foreign key field from the Many table when the user tabs or clicks out of the field. If the user enters a partial value, 4th Dimension looks for a matching value in the related One table. If 4th Dimension finds only one match, it automatically completes the entry. If 4th Dimension finds more than one possible match, the user is presented with a list of values from which to choose. For a complete description of this process, see “[Entering Data in Related Tables](#)” on page 170.
- **Prompt if related One does not exist** This check box forces 4th Dimension to display a dialog box that lets a user create the related One record if it does not exist. By default, when you enter a value in a related field from the Many table, 4th Dimension checks to see if a matching record already exists in the related One table. If 4th Dimension cannot find a match, the following dialog box is displayed:



This dialog box allows the user to create a corresponding record in the One table while you are entering a record in the Many table.

For instance, suppose that you have an Invoicing database that contains an [Invoices] table and a [Customers] table. If you enter an invoice in the [Invoices] table and the customer to whom the invoice belongs does not already have a record in the [Customers] table, 4th Dimension will ask you if you want to create the corresponding record in the [Customers] table when you validate the record in the [Invoices] table.

You can suppress this dialog box by unchecking the Prompt if related One does not exist check box. Suppressing this dialog box is useful when you need to manage the creation of the related One record using a method.

One to Many Properties

The One to Many properties control automatic relations in the other direction.

- The **Auto One to Many** check box establishes automatic relations from the One table to the Many table. For example, when a record from the [Company] table is opened in the User environment, the related records in the [People] table are loaded. This enables 4th Dimension to display the records of the people that work for the company in a subform.

When you create a relation, the automatic relation check boxes are already selected. If you want to turn off automatic relations from the Many table to the One table, deselect the **Auto Relate One** check box. If you want to turn off automatic relations from the One table to the Many table, deselect the **Auto One to Many** check box.

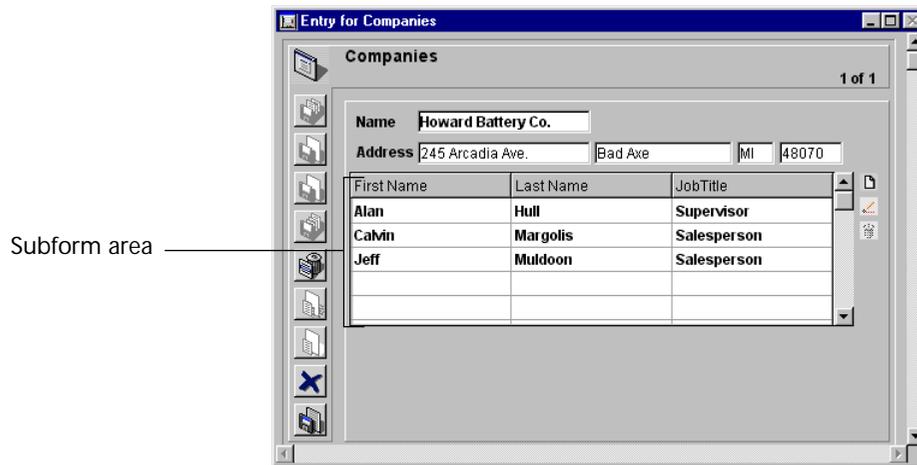
- **Auto Assign Related Value in Subform** is used to automatically assign the value of the primary key field in the One table to the foreign key field in the Many table during data entry. This option is available only if automatic One to Many relations are established.

This option affects data entry when an input form in a One table has a subform of a related Many table (for information on subforms, see [“Adding a Subform to the Form” on page 428](#)). If **Auto Assign Related Value in Subform** is selected, a user can add records to the subform (i.e., the related Many table) and have the relating value assigned automatically¹. This occurs as long as the record is added by typing

1. This works when the foreign key field is *not* displayed in the subform.

Ctrl+/ (Ctrl-Enter on Macintosh) or by pressing the Add to Subform button.

In the relation between the [Companies] table and the [People] table, the [Companies] table is the One table and the [People] table is a related Many table. Each company has one record in the [Companies] table and several records in the [People] table. The data entry screen for the [Companies] table is shown below.



If you want to add a person record from this screen, you need to assign the Company name to the foreign key field of the person's record. Otherwise, the new record would not be related to the correct record in the [Companies] table. **Auto Assign Related Value in Subform** does this automatically. If you deselect this option, you would need to make the assignment using a method.

Note If you double-click in a blank area of the subform and proceed directly to the input form to add a new record, or if you modify the relating field value in the One table after you have created records in the Many table, **Auto-assign Related Value in Subform** has no effect and you must either manually assign the relating field value or use the language.

Wildcard Choice

The Wildcard Choice list allows you to select an additional field to display in the Wildcard choice list (which appears when the user enters the 4th Dimension wildcard symbol (@) in the relating field during data entry). Usually, you will want to select the field that best identifies the

record. For more information about the wildcard choice field, see the section [“Using Wildcard Choice Lists” on page 171](#).

Deletion Control

The Deletion control options regulate record deletion in the Many table when a record is deleted in the One table. Normally, the user cannot delete records in a table unless it is the current table. This means, for example, that to delete records from the [People] table, it must first be made the current table. You make a table the current table by choosing it in the Choose Table/Form dialog box in the User environment.

When tables are related, 4th Dimension allows you to specify one of three special cases for records deleted from the One table.

The following deletion control options can be set only if the Allow Deletion Control check box in the Database Properties dialog box is checked. For more information, see [“Data Control” on page 85](#).

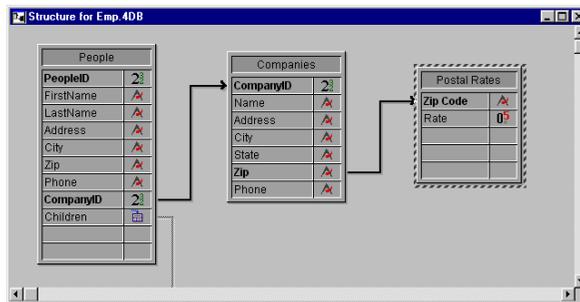
- **Leave Related Many Intact** Selecting this radio button allows the user to delete a record in the One table, leaving the corresponding records in the Many table intact. This leaves records in the Many table without any corresponding related record in the One table. The only effect is to render the information from the One table unavailable. No record from the One table is loaded when a record corresponding to the deleted record is loaded in the Many table.
- **Delete Related Many** Selecting this radio button instructs 4th Dimension to automatically delete all related records in the Many table when the user deletes a record in the One table. This property ensures that no related Many records become “orphaned” when the corresponding related One record is deleted.
- **Cannot Delete if Related Many** Selecting this radio button instructs 4th Dimension to prohibit the user from deleting a record in the One table if there are related records in the Many table. This property ensures that no records are mistakenly deleted.

Notice that you can freely delete records from the Many table, no matter which choice is made.

The Delete Related Many and Cannot Delete if Related Many radio buttons enforce what is called *referential integrity* in database theory. When referential integrity is in effect, 4th Dimension ensures that each record in a related Many table will always be associated with exactly one record in the related One table.

If you set the Deletion control option to either Delete Related Many or Cannot Delete if Related Many, 4th Dimension automatically adds the Indexed, Can't Modify, and Unique attributes to the primary key field in the One table. You cannot remove these attributes unless you first change the Deletion control setting to Leave Related Many Intact.

If you have several related tables, deletion control is activated for each relation as in a chain. For instance, suppose you have the structure shown below. If a Zip code is deleted from the [Postal Rates] table (a One table) and Delete Related Many has been selected for each relation, 4th Dimension first deletes the records for the corresponding companies in the [Companies] table and then deletes the records of all the people who work for those companies in the [People] table.



When confronted with contradictory Deletion control settings, 4th Dimension does not allow the deletion to occur. For instance, if Delete Related Many is selected for the relation between the [Companies] table and the [Postal Rates] table but Cannot Delete if Related Many is selected for the relation between the [People] table and the [Companies] table, no deletion will occur and the records in the [Companies] and [People] tables will remain intact.

The Deletion control choice is made when the relation is established. To change the choice, you can modify the relation's properties. For complete information about establishing and reestablishing relations, see the section [“Creating a Relation Between Tables” on page 165](#).

Color

The Color property controls the color of the relation line in the Structure editor. The color of the relation line does not signify any other property of the relation.

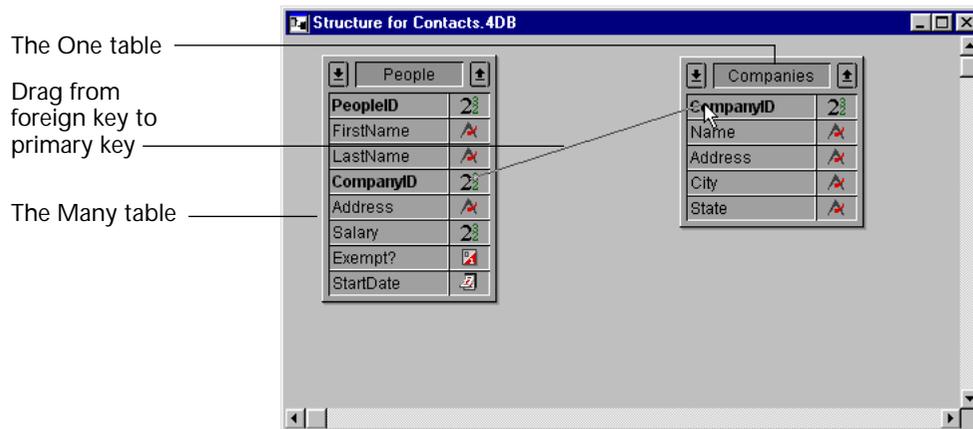
Creating a Relation Between Tables

You create a relation by dragging from the foreign key field to the primary key field. You can do this using either the table images in the Structure editor window or the Explorer.

It is convenient to use the Explorer to create the relation when your structure is large and the table images of the tables that you want to relate are not adjacent to each other. Using the Explorer method, only one of the two table images must be visible in the Structure editor window.

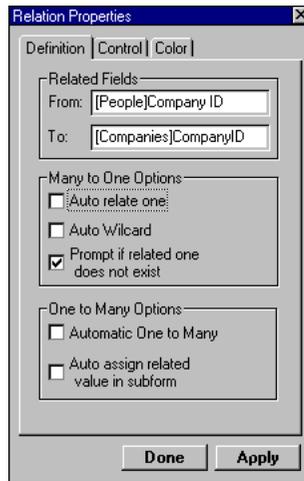
Tip To center a table image on-screen in the Structure editor window, double-click its name in the Tables page of the Explorer.

- ▶ To create a relation using the Structure editor window:
 - 1 In the Structure editor window, move the pointer over the foreign key field for this relation.
 - 2 Hold down the mouse button and drag toward the table to be related. As you move the pointer, 4th Dimension selects the field and draws a thin line, as shown below.



- 3 Drag to the primary key field in the One table and release the mouse button.

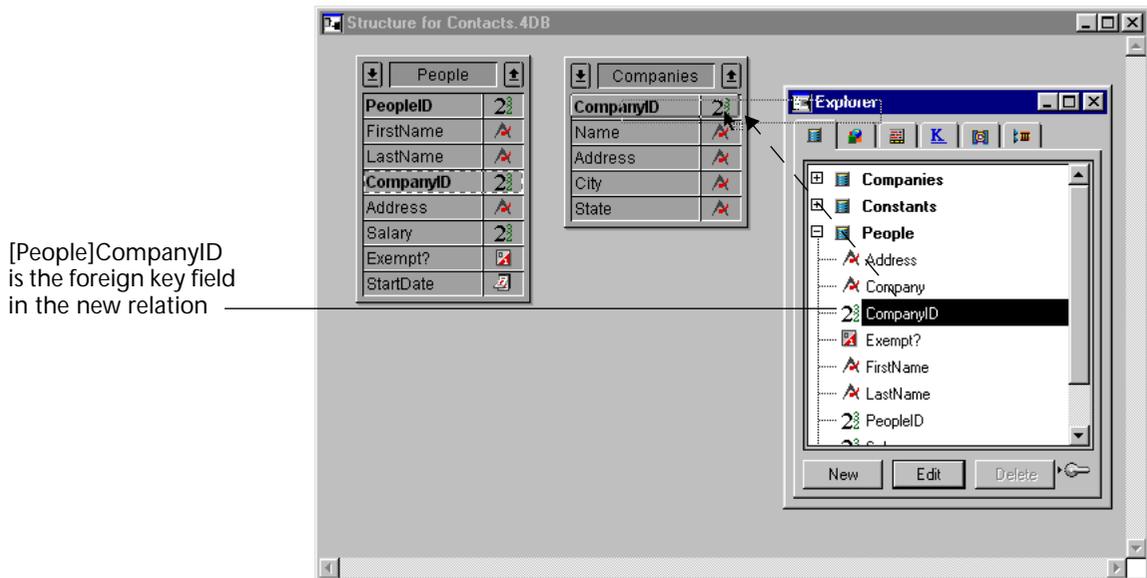
The Definition page of the Relation Properties window appears.



For information on setting properties for the relation, see the section [“Setting Relation Properties”](#) on page 167.

- ▶ To create a relation using the Explorer:
 - 1 Open the Explorer to the Tables page.
 - 2 Expand the table that contains the foreign key field.
This table will be the related Many table in the relation.
 - 3 Drag the foreign key field in the Explorer to the primary key field in the Structure editor window.

The following illustration illustrates the process.



The Definition page of the Relation Properties window appears.

Note If you hold down the **Shift** key, you can drag from the primary key field in the Explorer to the foreign key field in the Structure editor window.

For information on setting relation properties, see the following section.

Setting Relation Properties

This section describes the process of setting relation properties. The properties are the same regardless of the method you use to create the relation.

- ▶ To set relation properties:
 - 1 Check to make sure that the foreign and primary key fields are correct. Remember:
 - The “From” field is the foreign key in the Many table for this relation,
 - The “To” field is the primary key in the One table.
 - The primary and foreign key fields must be of the same data type.
 - 2 Select the desired automatic relation check boxes to establish automatic relations between the tables.

OR

Deselect the check boxes to establish manual relations.

In automatic relations, whenever a record from the One table is used, the related record or records in the other table is made the current selection for that table.

- 3 Click the Control tab and use the Wildcard Choice list to select the additional field to display in the Selection window.

For information on how the Wildcard choice field works during data entry, see the section [“Using Wildcard Choice Lists” on page 171](#).

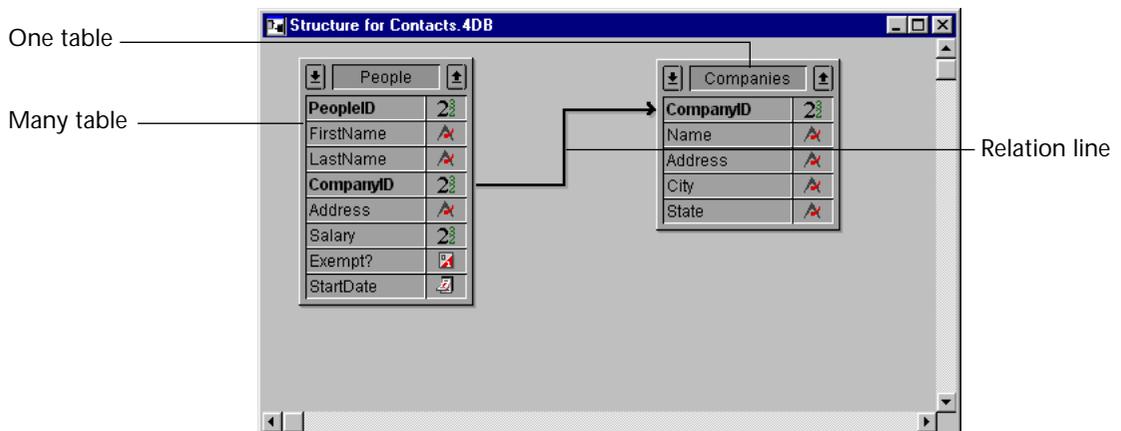
- 4 Select a Deletion control option by clicking the radio button for the choice you want.

The Deletion control radio buttons allow you to determine what happens when you delete a record in the One table. The Leave Related Many Intact radio button is the default. To be able to change this default option, you must have selected the Allow Deletion Control check box in the Database Properties dialog box. Otherwise, the deletion control options are dimmed. For more information about deletion control options, see the section [“Deletion Control” on page 163](#).

- 5 Click the Color tab and set the color of the relation line (optional).

- 6 Click the Apply button.

4th Dimension displays the Structure window with a line between the two fields, as shown in the following illustration.



The arrow on the relation line points to the One table.

- Removing Relations** 4th Dimension lets you remove a relation by deleting the line that relates the two fields.
- ▶ To remove a relation:
 - 1 In the Structure editor window, move the pointer over the name of the foreign key field in the Many table.
 - 2 Hold down the mouse button and drag over any empty area between table images.
 - 3 Release the mouse button.
 4th Dimension removes the arrow and the tables are no longer related.

Reestablishing Relations

You can reestablish any relation at any time. You would do so, for example, if you mistakenly draw the relation between the wrong fields or if you want to change a relation property. 4th Dimension lets you reestablish a relation simply by drawing the relation line again.

To reestablish a relation using the same two fields, draw the relation line again. 4th Dimension displays the Relation Properties window so that you can make any necessary changes.

To reestablish a relation using a different field in the Many table, first remove the faulty relation and then draw the correct relation line again.

Modifying Relation Properties

If you need to view or change the properties of a relation, you can do so without having to recreate the relation.

- ▶ To modify a relation's properties:
 - 1 Move the pointer to the relation line until it changes to a relation icon .
 - 2 Double-click the relation line.
 The Relation Properties window appears, showing the selected relation's properties.

Note The Relate One and Relate Many properties as well as the color property can be selected using the contextual menu that appears when you right-click the relation line (on Windows) or when you press the Ctrl key while right-clicking the relation (on Mac OS).

- 3 Modify the relation properties as necessary and click Apply.

Automatic and Manual Relations

Relations can be either automatic or manual.

In an automatic relation, whenever a record in a related table is made current, 4th Dimension selects the corresponding record or records. The record or records so specified can then be viewed, printed, modified, or used in searches and sorts. No programming is required.

In a manual relation, you control whether 4th Dimension loads the corresponding record or records into memory. To exercise this control, you use methods. For complete information about creating the methods that control related tables, see the *4th Dimension Language Reference*.

You would use a manual relation if you wanted to optimize the performance of specific applications that do not need all corresponding records loaded each time. For example, if your structure relates three or more tables together, you may want to control when related records are loaded into memory. You would also use a manual relation if you wanted to relate two tables with two separate relations. Only one automatic relation can exist between two tables. Any number of manual relations can exist between two tables.

Entering Data in Related Tables

You can display fields from the One table on a form for a related table. The user can use these fields to enter and edit the data directly while in the records of the related table.

If the relation is automatic, information entered into the related fields is automatically saved in the related field's table. For complete information, see the section [“Selecting Fields for the Form” on page 199](#).

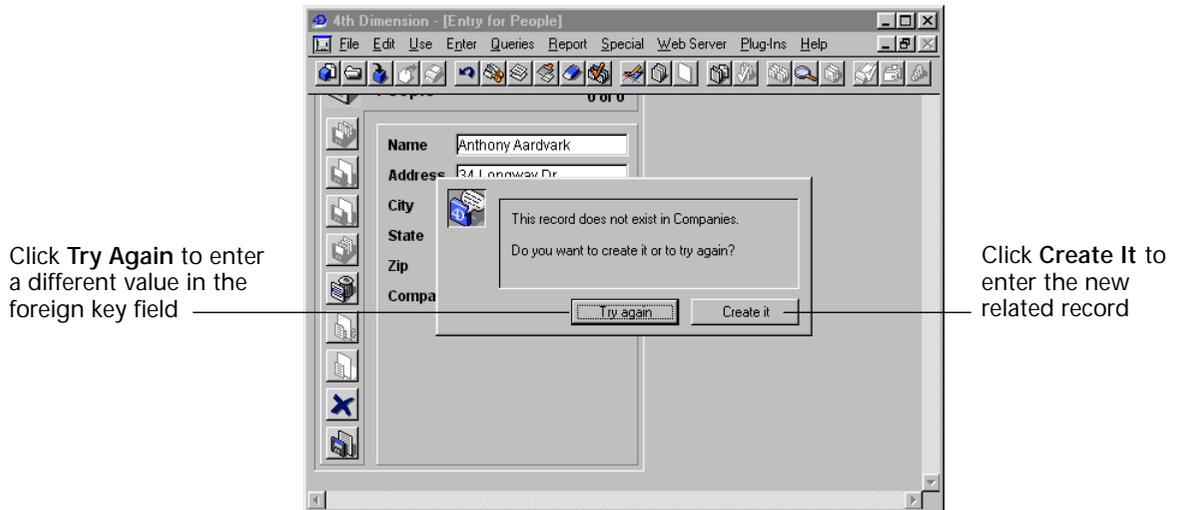
If the relation is manual, you use the language to display values in related fields and to save the information entered into related tables. You use methods to transfer and save any entered data.

The user enters records in the One table as in any other table — by typing information into an input form or by importing the data. For instance, you might have a subform in the [Companies] table that displays employee data from the [People] table. If automatic Relate Many relations have been established, any information you add or modify in the subform is automatically updated in the [People] table. For more information about subforms, see the section [“Adding a Subform to the Form” on page 428](#).

Occasionally, a user needs to create a new record for the One table while creating a record in the Many table. For example, suppose that,

while creating a record in the [People] table the user enters a company name that doesn't exist in the [Companies] table. If automatic Relate One relations have been established, 4th Dimension automatically gives the user the opportunity to create a new record in the [Companies] table.

If the Prompt if Related One does not Exist property is selected, 4th Dimension displays the following message when a user enters a company name that does not exist in the [Companies] table.



4th Dimension requires that the corresponding record exist in the related table. The chance to create a new record in the One table is automatically provided to the user.

For complete details on entering data into fields from related tables, see the *4th Dimension User Reference*.

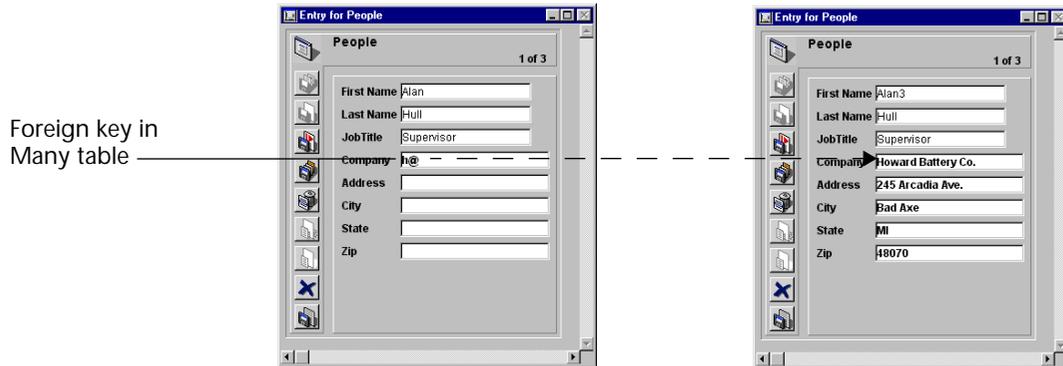
Using Wildcard Choice Lists

When tables are related, 4th Dimension allows the user to look up values in the One table when entering data into the foreign key field in the Many table. The user simply uses the standard wildcard character (@) in the related field. Doing so causes 4th Dimension to search for the corresponding entry in the related One table.

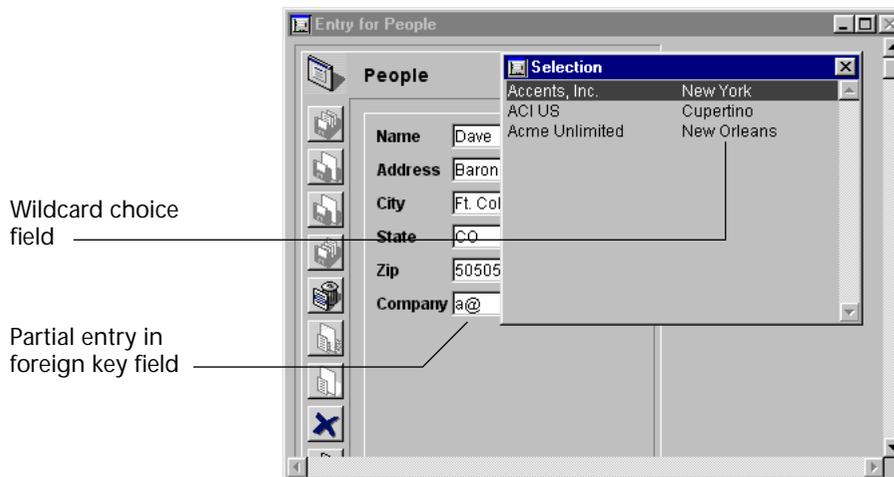
The wildcard character can be used in two ways: to complete a partial entry or to display a list of valid entries. When a list is displayed, the user can select the entry from the list. An additional field can be displayed with the related field.

For example, suppose the user is creating a record in the [People] table. Instead of typing ACI US, Inc. in the Company field, the user can type Ac@ and then press Tab to move to the next field. Because @ is the 4th Dimension wildcard character, this entry means “this value starts with “Ac” and is followed by anything else.” 4th Dimension looks in the related table for the record which matches this entry. If it finds one, it completes the entry and selects the next field in the data entry order.

The figure below shows how this use of the wildcard works.



If 4th Dimension finds more than one entry that meets the requirement, it displays a list of entries so that the user can select the correct one. The figure below shows such a list being displayed.

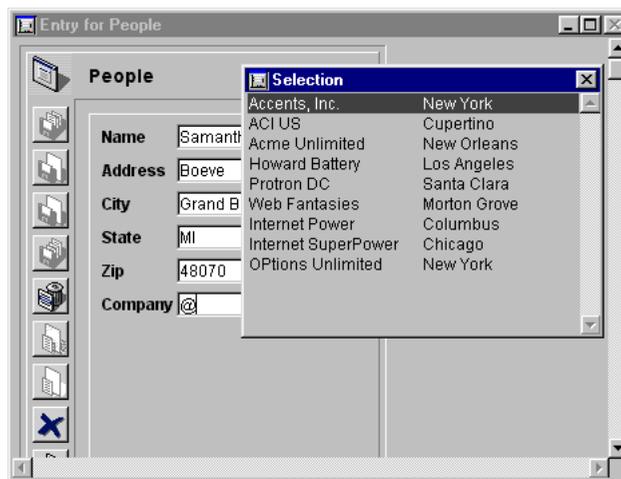


You can specify a second field for the list to help the user decide which company to select. The second field is the wildcard choice field you

selected in the Relation Properties window when you created the relation.

The figure above shows the list of companies displaying the city as well as the company name. This wildcard choice field assists the user who doesn't know whether the company is named ACI US or Acme Unlimited, but remembers that the company is located in Cupertino.

To see a list of all companies in the [Companies] table, the user enters @ only. 4th Dimension then displays a list of all the companies so that the user can select the correct one. The figure below shows a complete list of companies being displayed.



Relation Types

Until now, we have been discussing the process of creating the most common type of relation — a relation between a One table and a Many table — called a Many to One relation. However, you can also create One to One and Many to Many relations. These types of relations are described in this section.

One to One Relations

One to One relations are used only in special cases since tables that are related on a one-to-one basis could be combined into a single table. Here are some reasons to use a one-to-one relation:

- You have large BLOB fields in the database. BLOBs would slow down the database if they were loaded into memory when a record is made current. By placing the BLOBs in another table, you can load the BLOBs only when needed.
- You have a very large number of fields and need to divide them into logical groups. Separate tables can make the database faster and easier to use.
- You want to limit access to certain fields. If you use separate tables, you can assign different access privileges to each table.

Many to Many Relations

Sometimes you need to relate many records in one table to many records in another table. This is called a Many to Many relation.

An example of a Many to Many relation is a database that tracks class enrollment. Suppose that this database has two tables, [Students] and [Classes]. A student may enroll in many classes and a class may have many students. You want to see all the classes that a student has enrolled in and you want to see all the students enrolled in each class.

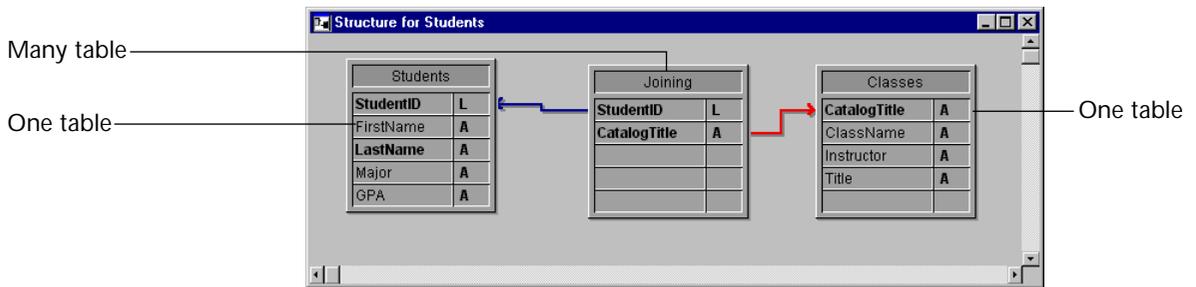
Other examples of Many to Many relations include the following.

- [Suppliers] and [Products] Each supplier provides many products and each product may be provided by several suppliers.
- [Employees] and [Account] Each employee works on many accounts and each account may be worked on by several employees.
- [Movies] and [Actors] Each movie involves several actors and each actor may appear in several movies.

You can use 4th Dimension to create automatic Many to Many relations. The key is to create an intermediate table which is related to the other tables using Many to One relations. You can then create input and output forms that handle all the necessary record tracking and data display.

This section describes how to use automatic relations to handle a Many to Many relation.

The figure below shows the enrollment database with three tables, [Students], [Classes], and [Joining]. This database structure is used throughout this section to explain how an automatic Many to Many relation works.



The [Students] table is a One table. It contains one record for each student, including the name, major, and GPA. The Student ID field identifies each student uniquely.

The [Classes] table is also a One table. It contains one record for each class, including the class name and the instructor. The CatalogTitle field identifies each class uniquely.

An intermediate table, [Joining], is the Many table for both of the other tables. It contains records for many students and many classes. Forms for this table are used for entering data into both of the other tables, and for displaying information in each of the other tables.

The use of three tables ensures that the data is stored efficiently. A student's complete record is stored only once. Each class has one record, stored only once. Records that relate students to classes are stored once for each enrollment. All of the information, however, is available in any combination.

Entering Data with Many to Many Relations

You use the intermediate table — in this example the [Joining] table — to enter and display information from both of the other tables. Each record that you enter in the [Joining] table is related to both of the other tables (a student and a class). The records from the [Joining] table contain only the two pieces of information that establish the relation: the student ID and the catalog title.

Here is an example of a new record being entered in the [Joining] table.

The screenshot shows a window titled "Entry for Joining" with a table view showing 0 of 0 records. The form contains the following fields:

Field	Value
StudentID	1
First Name	Jeffrey T.
Last Name	Spaulding
Major	Phys. Ed.
CatalogTitle	Journalism 354
Class Name	Distorting the News
Instructor	Fats Everready

Labels on the left side of the image point to specific fields:

- "Fields in [Students] table" points to the StudentID, First Name, Last Name, and Major fields.
- "Fields in [Classes] table" points to the Class Name and Instructor fields.
- "Fields in [Joining] table" points to the CatalogTitle field.

This record defines Jeffrey T. Spaulding as enrolled in a Journalism class. This record actually combines information from the other two tables.

A similar record exists for each class in which the student is enrolled. Only the Student ID and Catalog Title fields are actually stored in the [Joining] table. Each record catalogs a particular student taking a particular class.

Note When a record in the [Joining] table is loaded (as in creating such a record), it automatically creates a selection of records in the related tables. The selection consists of the corresponding student and class records. If you switch to either of the other tables, only a single record is displayed. To display all the records, choose **Show All** from the **Queries** menu.

The input form for this record is shown below. Notice that it contains fields from both the [Students] and [Classes] tables.

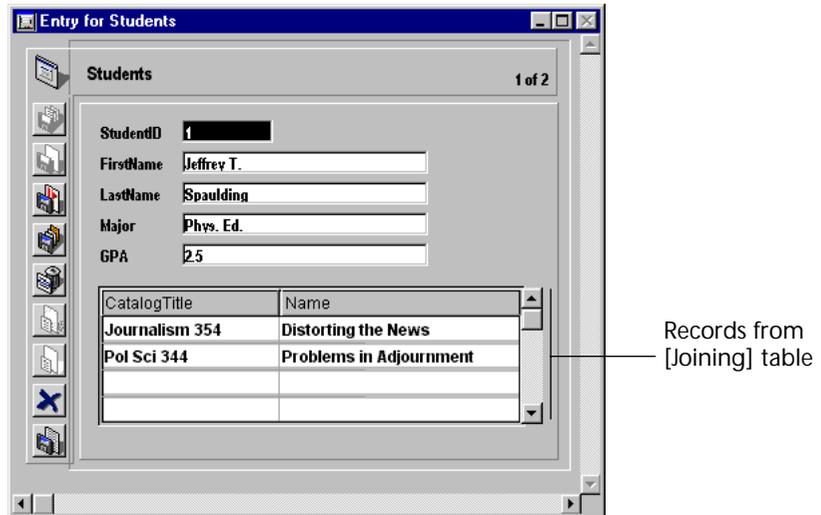
Data is entered only in the Student ID and Catalog Title fields. When a student ID is entered, 4th Dimension finds the student information in the related Students table and displays it in the Last Name, First Name, and Major fields. Likewise, when a Catalog Title is entered, 4th Dimension finds class information in the [Classes] table and displays it on the input form.

Displaying Information in a Subform

You can display information from these three tables using subforms. You can display all the classes a student is enrolled in on the student's record. You can display all the students enrolled in a particular class on the class record.

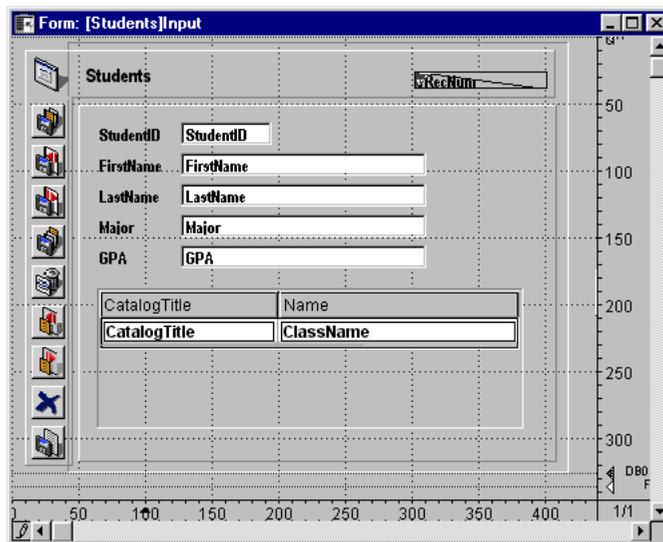
To display classes on a student's record, you use a subform. For information about creating subforms with the Form Wizard, see the section [“Adding a Subform to the Form” on page 217](#). For information on cre-

ating a subform using the Form editor, see [“Adding a Subform to the Form”](#) on page 428.



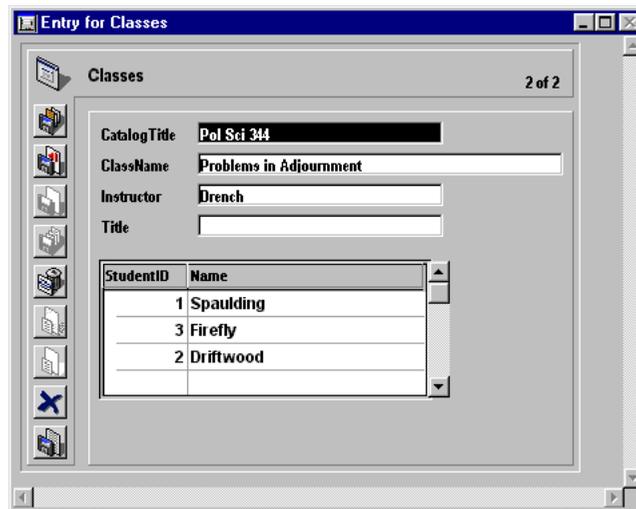
The record shown above is in the [Students] table. It shows information about the student at the top of the record. The information about the two classes that he is enrolled in is drawn from the [Joining] table where the enrollment information is kept.

Here is the form for this record:



Notice that the subform is for the [Joining] table, not the [Classes] table. The [Joining] table contains the records that relate the student's record to the class records. The subform contains the ClassName field from the [Classes] table. Because of the relation between the [Joining] and [Classes] tables, 4th Dimension can display the correct class name automatically.

Here is a record that shows the students who are enrolled in a class:



The screenshot shows a window titled "Entry for Classes" with a "Classes" subform. The subform displays the following information:

Field	Value
CatalogTitle	Pol Sci 344
ClassName	Problems in Adjournment
Instructor	Drench
Title	

StudentID	Name
1	Spaulding
3	Firefly
2	Driftwood

This is a record from the [Classes] table. It shows class information and lists the students enrolled in the class. The information about the students is also drawn from the [Joining] table since that table contains the records that relate the classes to the students enrolled in them.

Here is the form for the record shown above:

In the above examples of subforms, you can enter records in any of the fields shown. For example, to enter a new student into a class record, you simply tab to the last student record shown in the subform and press Ctrl+/ (on Windows) or Command+Tab (on Macintosh) to create a new record. When you enter the appropriate catalog title, the remainder of the information is entered in the record. For information about entering and deleting records in subforms, refer to the *4th Dimension User Reference*.

Creating Reports

Quick reports that include information from both the [Classes] and [Students] tables are typically generated from the intermediate table. If you create form reports for printing, you can use subforms to print information from either of the related One tables.

Here is a simple quick report that lists all the classes and the students that are enrolled in each class.

Class Name	First Name	Last Name
Auto Mech 101	Rufus T.	Firefly
Ceramics 211	Otis B.	Driftwood
	Jeffrey T.	Spaulding
Journalism 354	Jeffrey T.	Spaulding
Pol Sci 344	Jeffrey T.	Spaulding
	Otis B.	Driftwood
	Rufus T.	Firefly

For complete information about creating and printing quick reports, refer to the *4th Dimension User Reference*. For complete information about using forms for printing reports, refer to [Chapter 6](#) of this manual.

Analyzing Database Relations

The relations that you establish in a database play an important role in the operation of the database by controlling the flow of information between the tables.

If a record with an automatic relation is loaded from disk using an input form, the corresponding record or records from the related table are selected. If a relation selects only one record in a related table, that record is loaded from disk. If a relation selects more than one record in a related table, a new current selection of records is created for that table and the first record in the current selection is loaded from disk. The record that is loaded from disk is called the *current record* for the table.

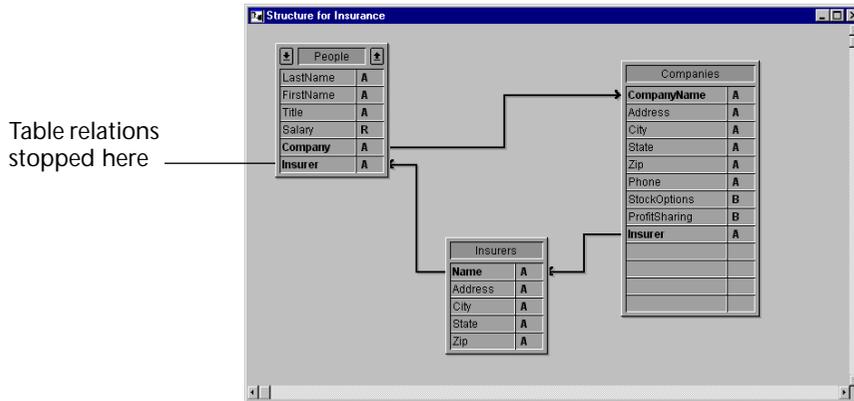
In the examples in this chapter, relations have been established between no more than three tables. In the real world, relations are often created between several tables and are activated one after the other, as in a chain. Each time a relation is activated, 4th Dimension creates a selection of records in the related table and loads a record from disk. The record that is loaded from disk becomes the current record for the table and — if the table has an automatic relation — 4th Dimension creates a selection and loads a current record in the next related table in the chain, and so on.

If the table relations have not been set up properly, the circulation of information between tables can become disorderly or corrupt. The following cases alert you to relational structures of which you should be aware.

Circular Relations

A circular relation is one in which table relations are set up so that the transfer of information will loop indefinitely. The figure below shows a circular relation in which the [People] table relates to the [Company]

table, which relates to the [Insurer] table, which relates back to the [People] table.



When a record in the [People] table is loaded from disk, 4th Dimension loads the related company record from the [Companies] table. This becomes the current record for the [Companies] table, which in turn loads the related insurer record from the [Insurer] table.

If the table relations were allowed to continue, the records related to this insurer (all the people insured by the company) would be selected in the [People] table and the first record in that selection would be the current record. Note that this current record may be different from the current record that started this progression. In this situation, 4th Dimension has no way of knowing which record is really the current record.

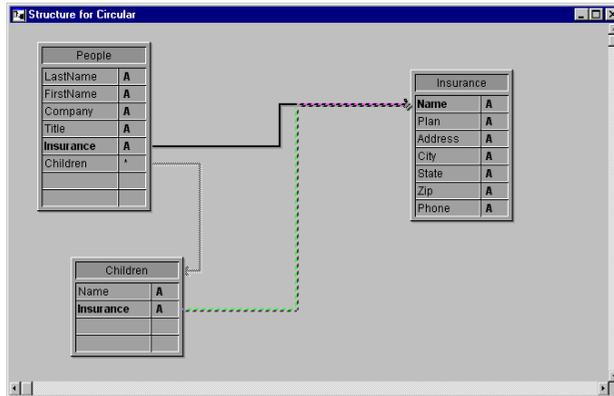
When 4th Dimension encounters this kind of circular relation, table relations are stopped at the last table in the chain. In this case, the relation between the [Insurer] table and the [People] table is not carried out.

Multiple Links to the Same Table

A similar conflict between current records occurs if you have more than one link to the same table.

Since you cannot have more than one current record at a time, you cannot manage an automatic table relation in which two or more tables are related to the same table.

The following illustration shows a database structure in which a table and its subtable both relate to the same table.



When a user is working with a record in the [People] table, the related record is loaded in the [Insurance] table and it is made the current record for that table.

However, there is also a relation between the [Children] subtable and the [Insurance] table. This means that another related record is loaded in the [Insurance] table based on the current record (the first record) in the [Children] subtable. If the child's insurance company is different from the parent's, this relational structure will cause problems.

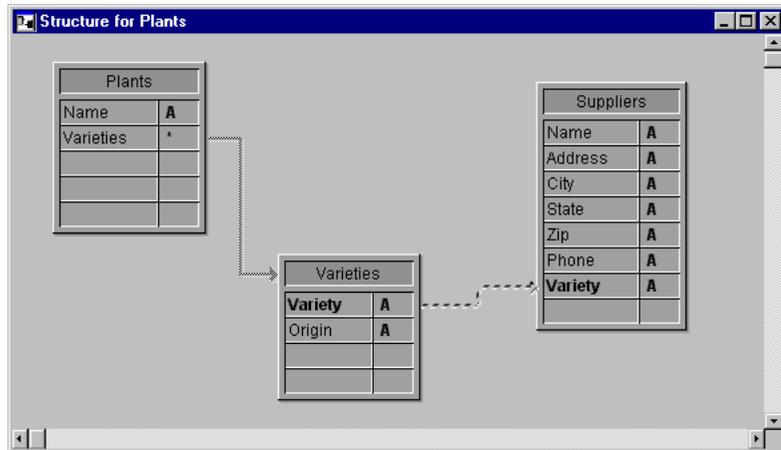
In this case, 4th Dimension does not stop the relations from proceeding. Both the relations are carried out, but not at the same time.

If you want to use this kind of structure, you must use manual table relations and control the relations using the commands described in the *4th Dimension Language Reference*.

Another example of a relational structure that cannot be managed by automatic relations is a structure in which one table has more than one relation to another table. Each time a user modifies either of the related fields in one table, the current record in the other table may change. In this situation, you cannot tell which relation is being activated.

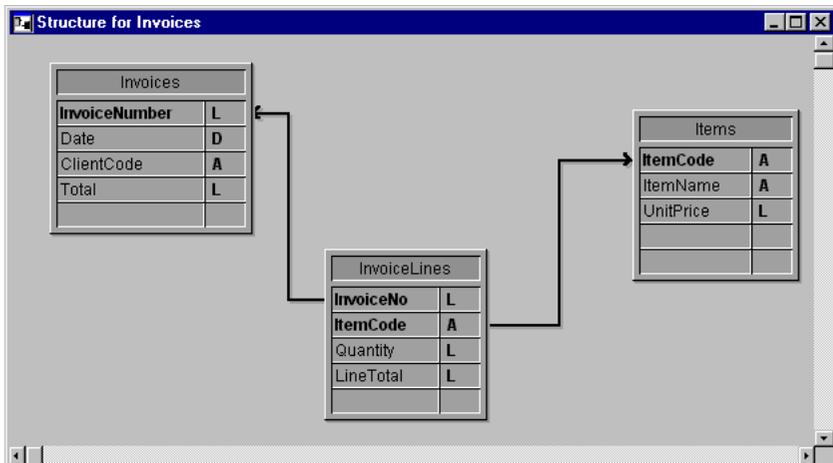
Relations from Multiple Records

Since there is only one current record in a table, relations are not established for all of the records in a selection. For instance, in the following figure, the records in the [Varieties] subtable are related to the records in the [Suppliers] table.



This structure implies that several varieties of plants are related to several suppliers and that the supplier records are loaded for all varieties of a plant species. However, the related supplier records are loaded for only the first record (i.e., the current record) in the subtable.

A similar case occurs in the Invoices database shown in the structure below.



When a record in the [Invoices] table is being used, a selection of records is created in the [InvoiceLines] table that contains all of the lines for that invoice. But the corresponding record in the [Items] table is selected only for the first item in the [InvoiceLines] table. The selection in the [Items] table does not include information about all the items in the invoice, only the first item.

However, if you place [InvoiceLines] in a subform in the [Invoices] table, *4th Dimension* calls each invoice line, one at a time, and activates the relationship for each one of them.

3

Creating Forms

Forms provide the interface through which information is entered, modified, and printed. A user interacts with the data in a database using forms and prints reports using forms.

Each table in your database generally has at least two forms. One form is for listing records on-screen and the other form displays one record at a time and is used for data entry and modification. The form that lists records is called the *output form* or *list form* and the form that displays one record at a time is called the *input form* or *detail form*. When you are viewing records using the list form, you can double-click a record to view the record using the current detail form.

This chapter covers the following topics:

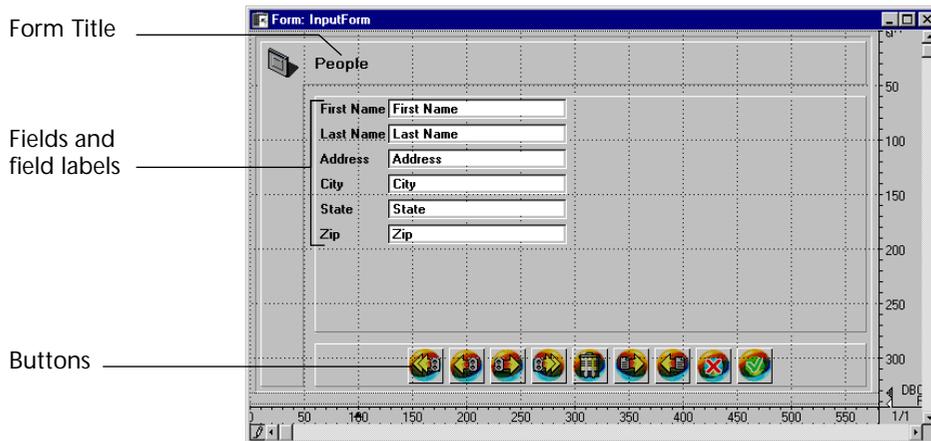
- Creating forms for data entry and display,
- Establishing default input and output forms,
- Deleting forms,
- Renaming forms.

For information about customizing forms, see [Chapter 4, “Form Editor Basics” on page 229](#) and [Chapter 5, “Working with Fields and Active Objects” on page 325](#). For information on creating and customizing list forms for listing records on-screen and for printing see [Chapter 6, “Output Displays and Reports” on page 439](#).

About Forms

The form is the interface object that you use for data entry, for listing records, for printing reports and mailing labels¹, and (in custom applications) for custom dialog boxes and palettes.

4th Dimension lets you create standard forms quickly. It also provides powerful tools that let you create forms that implement sophisticated interfaces. Your forms can provide exactly what your database needs. With only point and click operations, you can create a basic form as shown below.



4th Dimension has two tools for creating and modifying forms, the Form Wizard and the Form editor.

The Form Wizard

The Form Wizard is your starting place for creating any type of form. With the Form Wizard, you can create a new form by choosing the desired fields from a list and the desired form template from a drop-down list. Form templates control the appearance of forms. A template specifies such characteristics as form size, platform interface, font attributes, and buttons.

The Form Editor

The Form editor is an object-oriented drawing environment that lets you customize forms by manipulating objects on the form directly. For example, you can reposition objects, add objects not supported by the

1. You can also print reports and labels with the User environment's Quick Reports and Labels editors. These editors can also be added to a custom application.

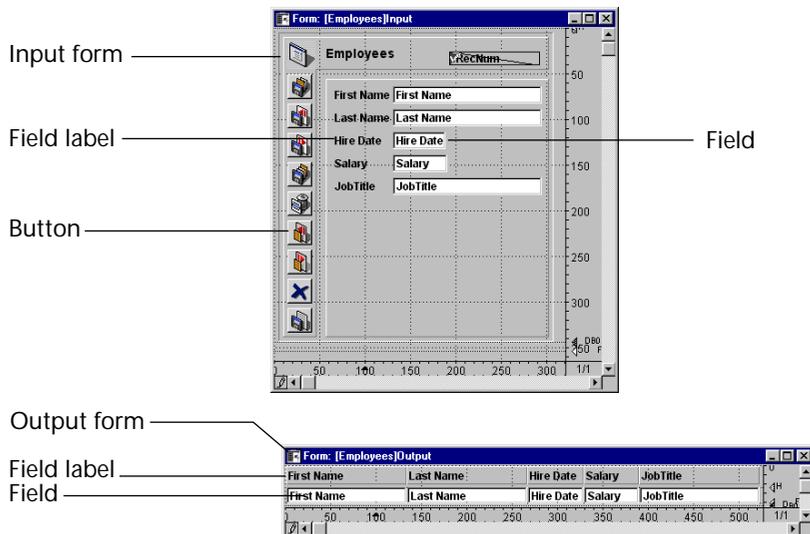
Form Wizard, create multi-page forms with tab controls, enforce business rules by specifying data entry constraints, specify form access privileges, associate a custom menu bar with a form, and write form and object methods that run automatically when the form is used.

This chapter discusses creating forms for data entry and display using the Form Wizard. [Chapter 4](#) and [Chapter 5](#) discuss the Form editor.

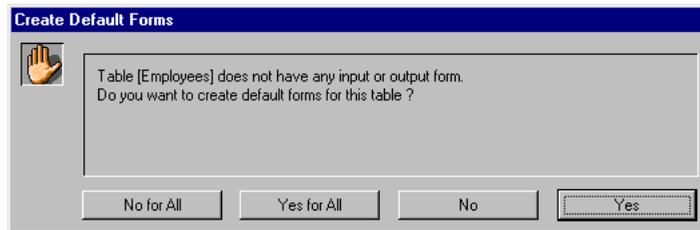
Forms, Tables, and Fields

Every form is attached to a table. The table to which a form is attached is called its *master table*. Each table must have at least one form so that information can be entered into fields and displayed on screen. Typically, a table has separate input and output forms. The input form is the one used for data entry. It displays one record per screen and typically has buttons for saving and canceling modifications to the record and for navigating from record to record (i.e., First Record, Last Record, Previous Record, Next Record). The output form displays a list of records, with one line per record. The results of queries are shown in the output form and the user can double-click a row in an output form to display the input form for that record.

The following illustration shows a typical input form and a typical output form.



If you switch to the User environment before creating a form for a table, 4th Dimension asks you if you want it to create default input and output forms for you.



Note With the Automatic Form Creation option in Database Properties, you can, for example, set 4th Dimension to automatically create default forms and therefore not display the Create Default Form dialog box. For more information, refer to “[General](#)” on page 83.

Click Yes to create default forms. You can always return to the Design environment and modify them or replace them with more sophisticated forms. Without making any modifications, you can start using these forms to enter and display data in your database.

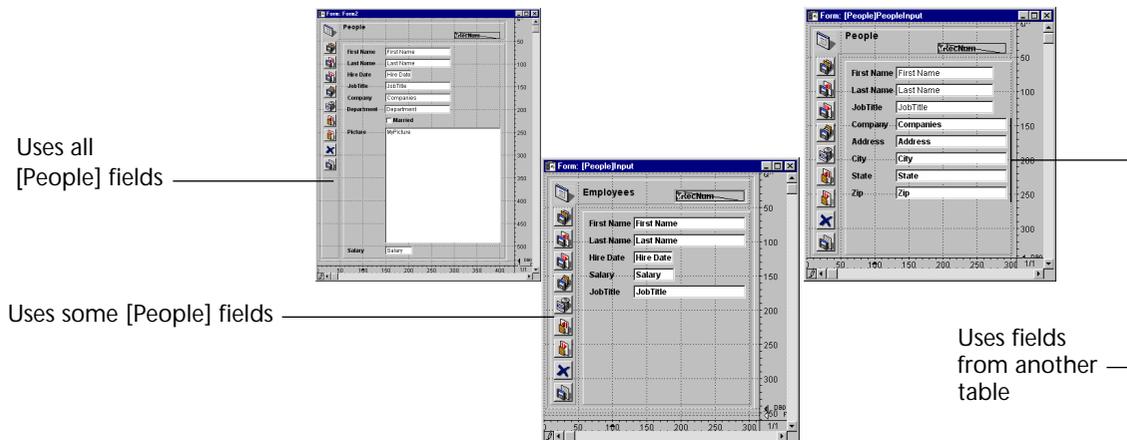
Your database can use a large number of forms that perform specific functions. In custom applications you can use the language to control which forms are the active input and output forms. For example, you may want to switch sets of forms depending on whether the user is using a monochrome or color monitor. You can also use the language to use different sets of forms for Web browsers and 4D Client users. When you write a custom application you can create forms for use as custom dialog boxes or floating palettes. In custom applications, you can also use multiple processes to allow users to work with several forms simultaneously.

A form can display fields from more than one table. You can place fields from a related One table on a form and allow users to enter values directly into the related One table. You can also include a *subform* that displays a list of records from a related Many table. A subform displays a list of records from another table or a subtable in the master table. With a subform, the user can view, enter, and modify records in another table. This is sometimes known as a *master-detail relationship*. For example, an invoicing application would use a subform on the invoicing input form that lets the user enter line items for the invoice.

Although the line items appear on the invoicing screen, the line item records are actually stored in a related Many table.

A form used for data entry can have more than one subform. For example, a contacts manager database can use a subform for telephone numbers, another subform for ToDo's, and another subform for prior contacts with the person. Each subform displays records from a different related Many table.

A particular form can use some of the fields in a table or all of the fields. You might have two input forms, for example — one for use by a clerk and one for use by supervisors — neither of which contains all the fields. You might use another group of fields for the screen display and yet a fourth group for a printed report.



Forms can be modified at any time, whether or not you have entered data into the database. Changes to a form do not affect the data stored on disk in any way.

Each form has one or more display pages in which fields and other enterable objects appear. If your fields don't fit on one page, you can create additional pages. When you create a multi-page form, you also add buttons or a tab control to allow users to move from one page to another.

Each form also has a background page (a page zero) on which you place objects that appear on all display pages. Use the background page to place background graphics, buttons, a tab control, and other graphic objects that define the “look” of the page, such as rectangles and labels.

Note When a multi-page form is used as an output form (e.g., when it is printed), only the first display page appears.

Active Objects and Graphic Objects

There are two kinds of objects in a form: active objects and graphic objects.

Active objects perform operations on data or provide a customized user interface. Active objects include the following:

- Fields, including fields from other tables,
- Enterable and non-enterable areas (variables) for entering or displaying data,
- Buttons (standard, 3D, Highlight, Invisible, or Picture) that perform actions,
- Tab controls,
- Combo boxes,
- Pop-up menus and drop-down lists,
- Hierarchical pop-up menus and hierarchical lists,
- Radio buttons and check boxes used for entering values into Boolean fields or variables,
- Thermometers, rulers, or dials that show relative values,
- Lists and hierarchical lists, that allow the user to select from the list or drag elements to or from the list,
- Graph and Plug-ins areas,
- Splitters for resizing form areas,
- Subforms that display data from other tables and subtables.

Graphic Objects

Graphic objects are geometric or textual elements that enhance the appearance of the form. Graphic objects include the following:

- Rectangles, ovals, and circles for enhancing the appearance of a form,
- Text, for labelling areas in the form,

- Grids, for aligning other objects,
- Graphics from applications other than 4th Dimension, including pictures stored in the Picture library.

All objects, whether active or graphic, are handled in the same way in the Form editor — they are created by being drawn or dragged and dropped; they can be selected and moved or resized; they can be duplicated, cut, copied, and pasted; they can be aligned to each other or to an invisible grid; and their appearance can be changed.

Object Properties

Each object has a set of properties. For graphic objects, properties include foreground and background color, line width and fill pattern, resizing and repositioning options, and font size and attributes. For active objects, properties may also include the object's relationship to data, the object's "action" when it is used, a help message, and the object's method. When the Form Wizard generates a form, it assigns appropriate default properties to both graphic and active objects¹; you can modify these properties in the Form editor.

Graphic objects have no impact on the data. You can create a graphic object on a form simply by drawing it and making any necessary modifications to its appearance. For example, when you create a text area, you draw the area, then you type the text you want to display. You can change the text at any time without affecting the data.

Active objects require instructions about their relation to the data or instructions on the actions that they are to perform. In simple cases, The Form Wizard does everything for you. Entry areas for fields are automatically associated with the appropriate field in the database structure and buttons automatically perform the appropriate actions. In other cases, you can specify additional instructions in the Form editor using the Object Properties window. For example, you can assign special instructions to a button by writing a method. The method remains attached to the object as one of its properties. If the object is copied and pasted, it retains all its properties, including its method.

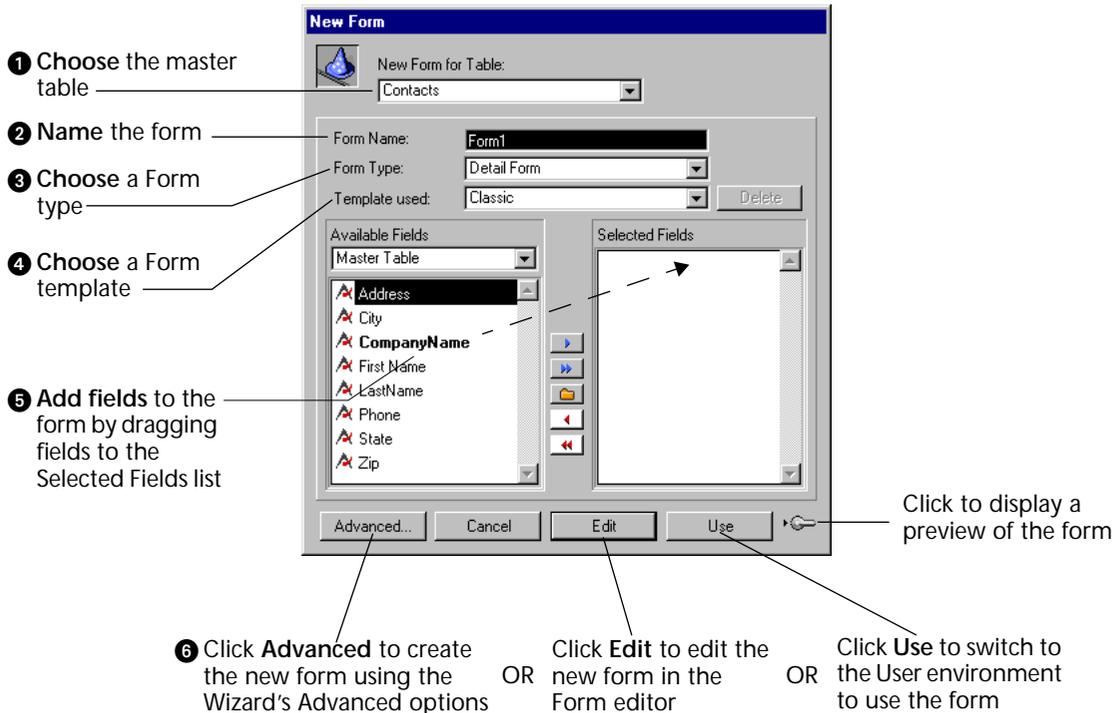
[Chapter 4](#) provides detailed information about working with the Form editor. [Chapter 5](#) provides complete information about active objects.

1. For example, the Form Wizard generates buttons that perform automatic actions and assigns appropriate resizing and repositioning options to decorative rectangles.

The Form Wizard

You can create new forms quickly with the Form Wizard. You can use a new form immediately after creating it or choose to edit the form using the Form editor.

The Form Wizard has two screens. The Basic screen lets you create new forms with a few simple operations. The steps for creating a form are shown in the following illustration.



If you want to create a standard form quickly, use the Basic screen. Here are the basic operations:

- Name the new form by filling in the Form Name area.
- Choose a form type from the Form Type drop-down list. Your choices are:
 - **Detail Form** an input form for entering and modifying individual records,
 - **List Form** an output form for listing records on the screen,

- **Detail Form For Printing** a form for printing individual records with one page per record,
- **List Form for Printing** a form for printing a list of records, with several records per page.
- Choose a template from the Template drop-down list. A template controls many aspects of the appearance of the form. The Template drop-down list includes a variety of templates that ship with 4th Dimension and any user-defined templates that you create with the Advanced options in the Form Wizard. For information on adding templates to this list, see the section [“Creating a Form Template” on page 220](#).
- Choose the fields for the form by dragging the desired fields in the Fields area to the Selected fields area (to its right).

As you add fields or change the form type or template, your changes are reflected in the Preview area on the right.

If you need more control over the appearance of the new form, you have two choices:

- **Use the Advanced options in the Form Wizard** Click **Advanced** to customize the new form with the Form Wizard. The Advanced options let you set the font attributes of fields and labels, the platform interface for the form, appearance of fields and field labels, form size, the form background, associate a menu, choose custom buttons, and add a sub-form.
- **Use the Form editor** Click **Edit** to continue designing the form using the Form editor.

If you are finished creating the form, you can click **Use** to switch to the User environment to use the form.

For complete information on using the Form Wizard, see the section [“Creating a New Form” on page 197](#).

The Form Editor

The Form editor is a powerful object-oriented drawing environment. Each of your forms can be displayed in a separate window and several forms can be open the same time. Objects on the form can be created or manipulated with the tools in the Form editor's Tools palette. You can set each form's properties and each form object's properties. While you are designing a form in the Form editor, you can switch to the User environment to test the form without having to first close the Form editor.

The Form editor offers many customization options that are not available in the Form Wizard, including the ability to:

- Set access privileges for the form,
- Assign properties to each object,
- Attach a method to the form and to each form object,
- Add pictures from the Picture library to the form,
- Resize and reposition each object directly,
- Add types of interface objects that are not supported by the Forms Wizard, including tab controls, drop-down lists and pop-up menus, picture buttons, combo boxes, check boxes and radio buttons, hierarchical menus and hierarchical lists, scrollable areas, plug-ins, graphs, splitters, and additional subforms.¹
- Set drag and drop properties for individual objects,
- Set data entry controls such as minimum, maximum, and default values, entry filters, and choice lists,
- Customize automatic resizing and repositioning options for each object.

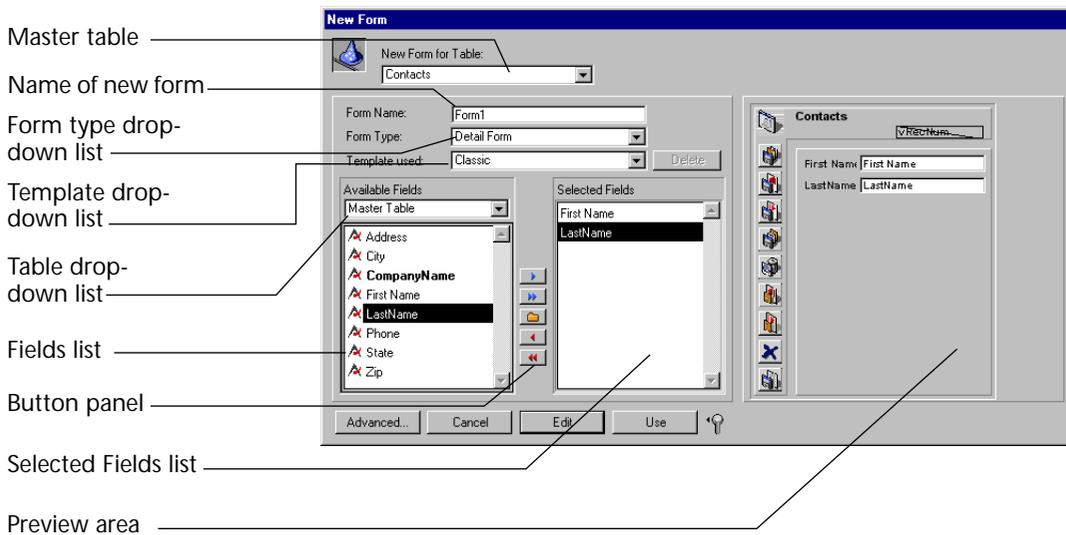
For a complete description of the Form editor, see [Chapter 4](#) and [Chapter 5](#).

1. The Advanced options in the Form Wizard lets you add only one subform.

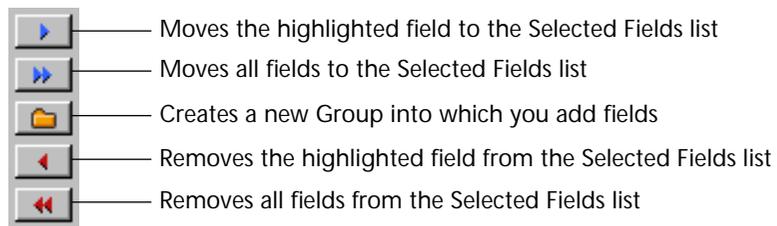
Creating a New Form

This section gives the basic steps for creating a form using the Basic screen of the Form Wizard. For information on the Advanced options of the Forms Wizard, see [“Using the Form Wizard’s Advanced Options” on page 205.](#)

- ▶ To create a new form:
 - Choose New Form from the Design menu.
 - OR
 - Highlight the table name in the Forms page of the Explorer and click New.
- 4th Dimension displays the Basic screen of the the Form Wizard.



The name of the master table is shown at the top of the screen. Its fields are listed in the Fields list. The Button panel contains shortcut tools for moving fields to and from the Selected Fields list.



- 4 If you need to change the master table, choose the master table from the New Form for Table drop-down list.
- 5 Name the form by filling in a name in the Form Name area.
You can refer to the form by name using the language.
- 6 Choose a Form Type from the Form Type drop-down list.
Your choices are:
 - Detail form a form for data entry and modification.
 - List form a form for listing records on the screen.
 - Detail form for Printing¹ a printed report with one page per record, such as an invoice.
 - List form for Printing¹ a printed report that list records.
- 7 Choose a template for the form.
The template controls several aspects of the appearance of the form, including font attributes, field label placement, the design of decorative rectangles surrounding fields, and platform interface.
4th Dimension ships with several templates and you can add use the Form Wizard to add custom templates to this list. For more information about adding custom templates, see the section [“Creating a Form Template” on page 220](#).
- 8 Select the fields you want on your form.
For complete information about selecting fields for the form, see the following section, [“Selecting Fields for the Form” on page 199](#).
- 9 If you want to edit the new form in the Form editor, click Edit.
OR
If you want to switch to the User environment to use the form, click Use.
OR
If you want to customize the new form with the Forms Wizard's advanced options, click Advanced.
For information about the Advanced options, see the section [“Using the Form Wizard's Advanced Options” on page 205](#).

1. For information on creating forms for printing, see [Chapter 6](#).

Selecting Fields for the Form

You select the fields that you want to appear in the form in the Form Wizard. You can also use the Form editor to add fields to the form after it has been created.

You can select any fields of any type, except Blob fields.

You can create forms that include fields from:

- The master table,
- A related One table.
- Any table.

The Subforms page in the Advanced options screen lets you create subforms that display fields from:

- Subtables of the master table,
- A related Many table,
- Unrelated tables.

The following sections explain how to select fields from the master table and from a related One table.

Selecting Fields from the Master Table

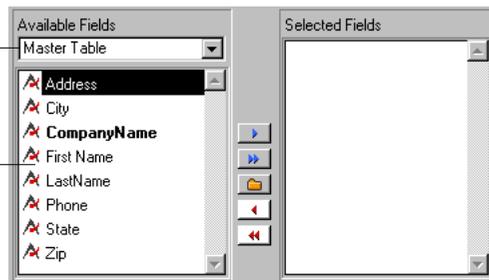
When Master Table is selected in the Table drop-down list, the Fields list displays a list of fields in the master table. Indexed fields are shown in bold.

- ▶ To select fields from the master table:
 - 1 If it is not already selected, choose Master table from the Table drop-down list.

The master table is the table to which the form belongs.

Table drop-down list

Fields in the master list



2 Drag a field from the Fields list to the Selected Fields list.

OR

Click the field in the Fields list and click the Append button .

OR

Double-click the field in the field list.

4th Dimension moves the selected field to the Selected Fields list adds the field to the Preview area (if you have expanded the Form Wizard view).

Note If you want to include all fields on the form, click the Add All button .

After you have selected fields for the form, you can reorder fields by dragging them vertically within the Selected Fields list.

3 Repeat this process for each field you want to add to the form.

To remove a field, select it and click the Remove Field button . To remove all fields, click the Remove All button .

If you are adding fields by dragging, you can insert a field in the list by dragging to the desired position in the Selected Fields list. Otherwise, 4th Dimension appends each field you add to the end of the list.

Note You can reorder fields in the Selected Fields list by dragging a field up or down.

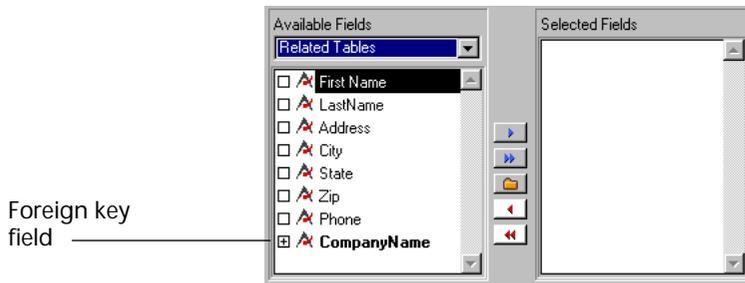
Selecting Fields from Related One Tables

You can select fields from related One tables. 4th Dimension allows you to enter values directly into related One fields.

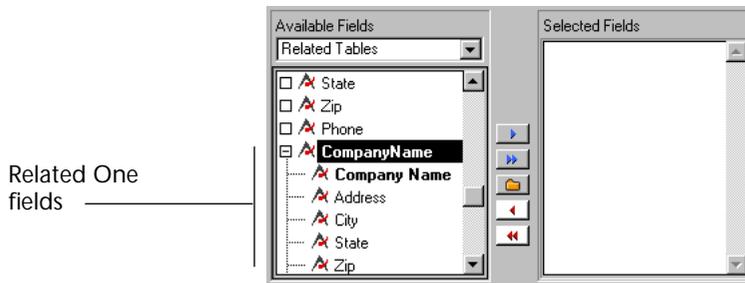
► To add fields from a related One table:

1 Choose Related Tables from the Tables drop-down list.

The Fields list changes to display a hierarchical list of fields in the master table. The foreign key fields are shown in bold and have a plus sign (on Windows) or an arrow (on Macintosh).



- 2 Expand a foreign key field to display the related One fields in the related table.



- 3 Add related One fields to the form using any of the methods used for adding fields from the Master table.

As you add related One fields, they are shown in the Preview area.

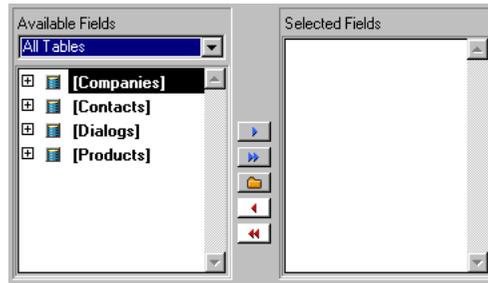
Selecting Fields From Other Tables

You can select fields from any table in the database. However, if the table is not the master table or an automatic related One table, you will need to use the language to manage data entry and display in the fields you select.

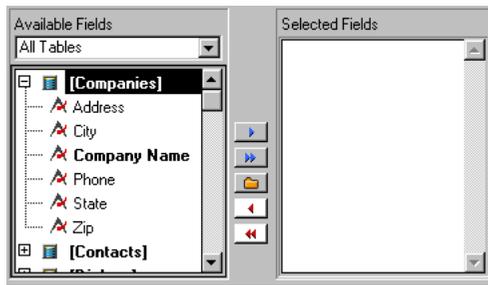
- To add fields from any table:

- 1 Choose All Tables from the Tables drop-down list.

A hierarchical list of all tables in the database appears in the Fields list.



- 2 Expand the desired table to view its fields.
The fields belonging to the table appear.



- 3 Add the desired fields to the form using any of the methods described in the previous section.
As you add fields, they appear in the Preview area.

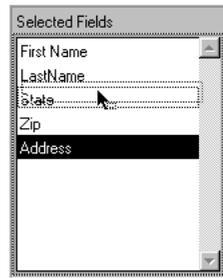
Reordering Fields

After you have added fields to the form, you can reorder them by dragging fields up or down in the Selected Fields list. Reordering a field list affects its position on the form. When you reorder fields, the Preview area shows the effects of your changes.

Note With the Form editor, you can rearrange fields by manipulating them directly.

- To reorder a field:
 - 1 Select the field to be reordered in the Selected Fields list.
 - 2 Drag the field up or down.

As you drag, the outline of the field indicates its new position. The following illustration shows the Address field being moved between the LastName and State fields.

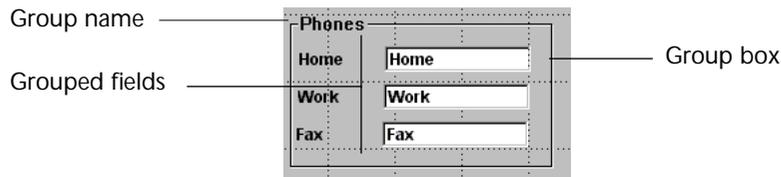


3 Drop the field at the desired location.

When you drop the field, the Selected Fields list changes to show the new field order and the Preview area reflects your changes.

Grouping Fields

When you are creating a Detail form, you can define a group of fields. A group has its own label and set of fields in the group. A group looks like this:



You can create several groups in each form.

Note With the Form editor, you can also create group objects and move fields or other objects into or out of groups.

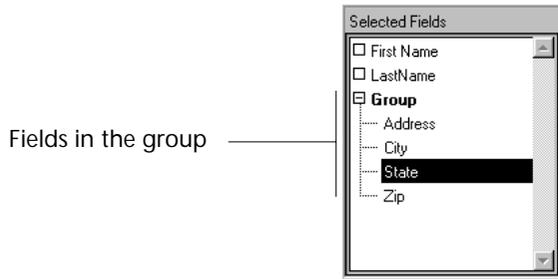
► To create a group:

- 1 Click the Group button .

A new group is created in the Selected Fields list. Its default name is “Group” and it appears as a hierarchical list.

- 2 Add fields to the group by dragging the fields to the group name.

As each field is added to the group, it appears below the group name.



When you are finished adding fields to the group, you can:

- Create another group,
- Add more ungrouped fields,
- Edit the form, use the form, or use the Advanced options.
- To create another group, click the Group button  and repeat the process of adding fields to the group.
- To add ungrouped fields, collapse the Group item in the Selected Fields list and continue adding fields.
- To edit the form, use the form, or use Advanced options, click the desired button at the bottom of the Form Wizard screen.

Renaming the Group

Unless you rename the group, the name of the group on the form will be the default name, “Group.”

- ▶ To rename the group:
 - 1 Hold down the Command key (on Macintosh) or Ctrl key (on Windows) and click the group name in the Selected Fields area. The name becomes editable.
 - 2 Type the new group name and click anywhere outside the text entry area to save the new name.



When you save the new name, the preview area reflects your changes.

- Reordering Fields in the Group** You can reorder fields in the group in the same way as ungrouped fields. Select a field to be reordered and drag up or down as described earlier in the section [“Reordering Fields” on page 202](#).
- Moving Fields from the Group** You can also move a field from a group to make it an ungrouped field or move the field to a different group.
- To make a field an ungrouped field, drag the field from the group and move it diagonally in the direction of other ungrouped fields — to the above-left or below-left.
 - To move a field to another group, make sure the target group is expanded, then drag the field to the field list in the other group. If the destination group is empty, you first need to remove the field from the Selected Field and then insert it in the group as described above. 4th Dimension updates the preview area to reflect your changes.
- Removing Fields** To remove a field from a form, highlight the field in the Selected Fields list and click the Remove button . To remove all fields from the form, click the Remove All button .

Using the Form Wizard's Advanced Options

The Form Wizard's Advanced screen also lets you create new forms with point-and-click operations but offers a wider variety of customization options.

The customization options depend on the form type that you select in the first screen of the Form Wizard. The Form Wizard supports the following form types:

- Detail forms
- List forms
- Detail forms for printing
- List forms for printing

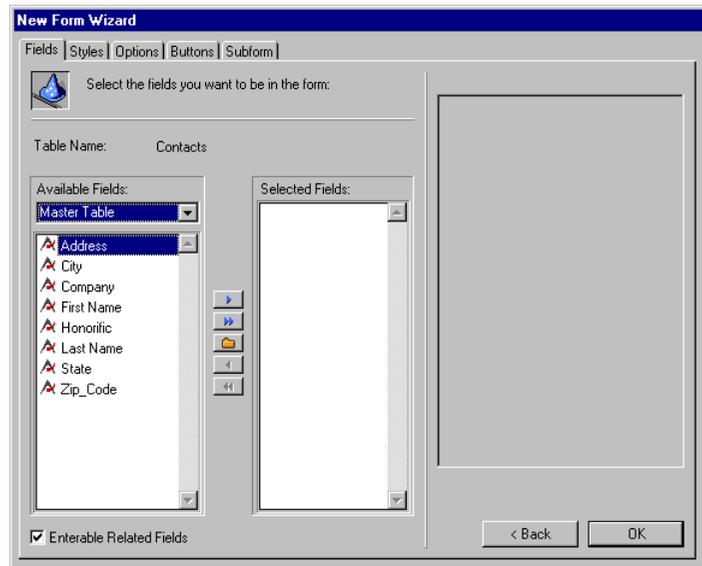
In addition, the Form Wizard lets you save your customization options as a template. The template name is added to the Template drop-down list that appears on the Basic screen of the Forms Wizard. With user-

defined templates, you can quickly create highly customized forms from the Basic screen of the Form Wizard simply by selecting the desired fields and your custom template.

If you want to go back to the Basic Screen of the Form Wizard, you can do so by clicking the <Back button.

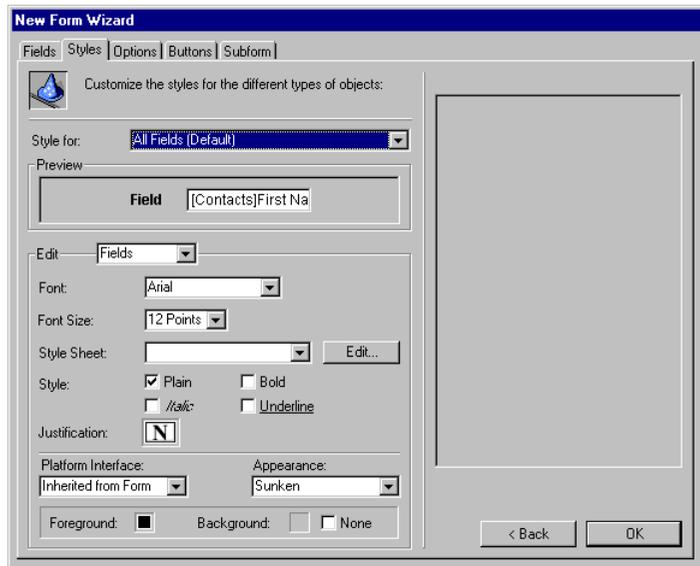
This section describes the advanced options available for creating Detail forms. For more information about the creation of List forms, refer to [Chapter 6, “Output Displays and Reports” on page 439](#). The Advanced screen contains the following pages:

- **Fields page** Choose the fields for the form. This page is similar to the Basic screen of the Form Wizard.

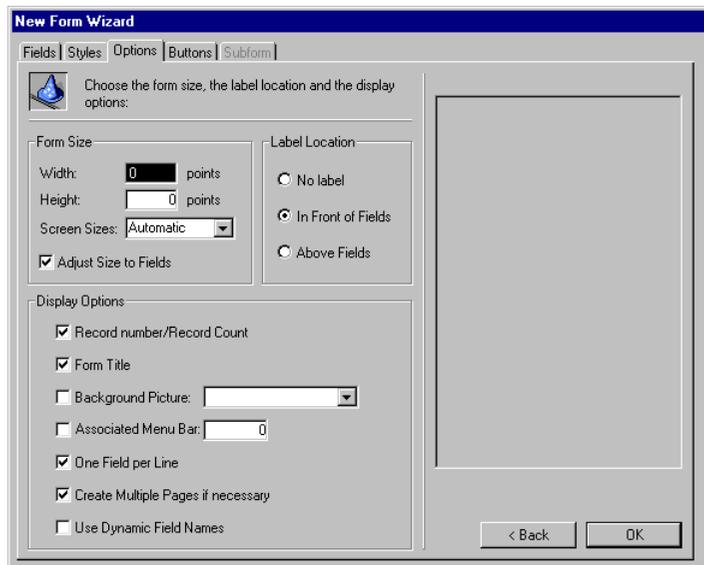


- **Styles page** Set the appearance of form objects and their labels. You can specify font attributes, foreground and background colors, platform interface, and appearance of decorative rectangles surrounding

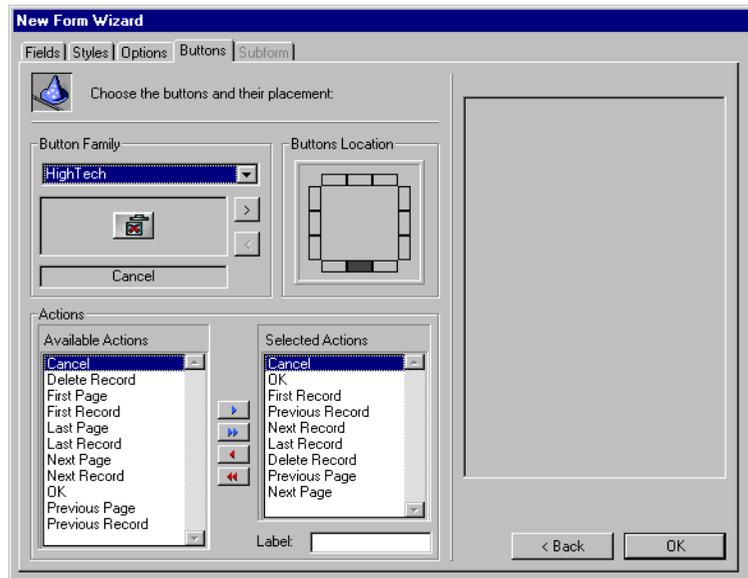
fields and field labels. You can also define or apply Style Sheets to specify font attributes.



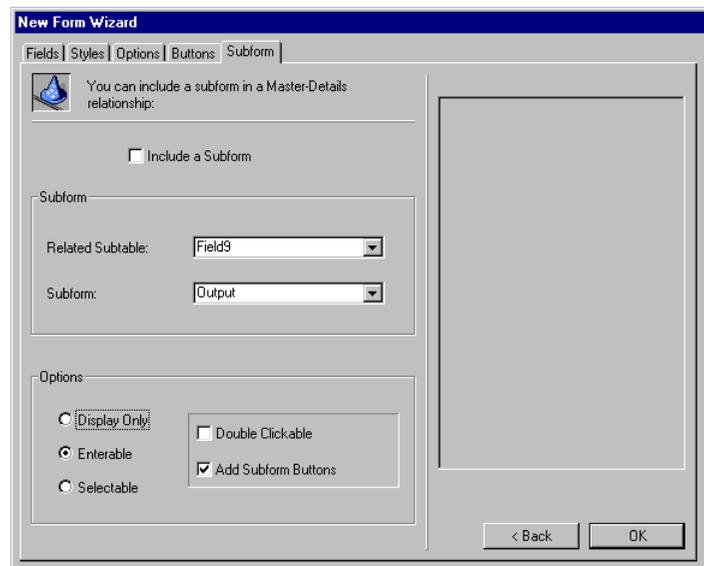
- **Options page** Set the form size, placement of field labels relative to fields, associate a background image, associate a menu, add a title and/or record count or use dynamic field names.



- **Buttons page** Choose a set of buttons to be added to the form, assign automatic button actions, set the placement of the buttons on the form, and (optionally) label the buttons.



- **Subform page** Add a subform from a related Many table or subtable to the form.



Note The Subform page is enabled only if there is a related many table related to the master table.

Adding Fields

Use the Fields page to add fields to the form.

You add fields to the form in exactly the same way as on the Basic screen of the Forms Wizard. This functionality is duplicated for users who want to skip the Basic screen and go directly to the Advanced options screen. For a complete description of the process of adding fields to the form, see the section [“Selecting Fields for the Form” on page 199](#).

If you have already added fields to the form using the Basic screen, you can modify your selections using the Fields page. For more information, see the sections [“Reordering Fields” on page 202](#) and [“Removing Fields” on page 205](#).

Customizing the Appearance of Form Objects

The Styles page of the Forms Wizard lets you customize the appearance of the following types of objects:

- Related fields
- Non-enterable fields
- Mandatory fields
- Indexed unique fields
- Group box text
- The form title
- Information: Form elements that provide information using internal variables, such as page number, record number, and records in selection. Several of the default templates add such variables automatically.
- Check boxes and radio buttons
- Default: Fields and field labels not specified by other items in the Type drop-down list.

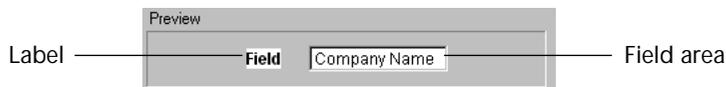
For fields, you can specify the appearance of the field and the field's label separately.

For each object type, you can customize the following properties:

- Font attributes,
 - Platform interface,
 - Appearance of decorative rectangles surrounding the object,
 - Text justification,
 - Foreground and background colors.
- ▶ To customize the appearance of the form's objects:
- 1 Click the Styles tab in the Advanced options screen.
The Styles page appears.
 - 2 Choose the type of object whose appearance you want to customize from the Style For drop-down list.



When you make your selection, the Preview area changes to show a preview of the type of object you selected. If you selected a field type or Default, the preview area shows preview images for both the field and the field label.



- 3 Choose either Labels or Fields in the Edit pop-up menu.
- 4 Choose the desired font, font size, and font style.
OR
Choose a style sheet or click Edit to create a style sheet.

For information on creating and using styles, see [“Using the Style Sheet Editor” on page 222](#).

The selected object type appears in the Style For area. The Preview area shows a preview of the object. If the object is a field, only the label or the field will be previewed, depending on whether you are modifying the object or the label.

- 5 Choose the Platform Interface for the object from the Platform Interface drop-down menu.

Your choices are:

- **Inherited from Form** The platform interface for the object is the same as the platform interface of the form. The Platform Interface of the form is set in the Form Properties window. For information on setting the Platform Interface for the form, see [“Setting the Platform Interface” on page 255](#).
- **Automatic** The platform interface is based on the platform on which the database is currently running. For more information on the Automatic option, see [“Platform Interface Settings” on page 105](#).
- **Mac OS** The object will be displayed as a Macintosh object (System 7.5 or above).
- **Windows NT 3.5.1** The object will be displayed as a Windows NT 3.5.1 object.
- **Windows 95** The object will be displayed as a Windows 95 object.
- **Platinum** The object will be displayed as a Macintosh object that uses the “Platinum” user interface guidelines.

For complete information on how the Platform Interface options affect the appearance of objects, see [“User Interface” on page 89](#).

6 Choose the Appearance of decorative rectangles surrounding the object or label.

Your choices are:

- None
- Plain
- Dotted
- Raised
- Sunken
- Double

For illustrations of the effects of these choices on various object types, see the sections, [“Fields and Field Labels” on page 221](#), [“Buttons” on page 382](#), and [“Button Actions” on page 385](#).

7 Choose the desired text justification.

Your choices are:

- **Left, Right, or Centered** alignment,
- **Default** Right-aligned numbers and left-aligned text, dates, and times.

For more information on justification options, see the section [“Using the Style Sheet Editor” on page 222](#).

- 8 Choose a foreground and background color or click the None check box for automatic background (Optional).

The foreground color is the color of text in an area. The background color is the color of the area itself. For more information, see the section [“Foreground and Background Colors” on page 307](#).

- 9 Choose another object type from the Style For drop-down list and repeat steps 3 to 8.
- 10 When you are finished, click another tab to customize another aspect of the new form.

OR

If you are finished with all Advanced pages, click OK to generate the new form.

Customizing Buttons on the Form

Detail forms use buttons that allow the user to save and cancel changes to a record, move from one record to another (first record, last record, next record, previous record), or one page to another in a multi-page form, add or delete records in a subform, or delete the current record. With the Buttons page in the Advanced options screen, you can choose a button design, choose the desired button actions, specify the position of the buttons on the form, and label each button.

Note In the Form editor, you can add, delete, or reposition individual buttons and attach a method to a button that specifies its action when clicked. For more information, see [“Buttons” on page 382](#).

Automatic Button Actions

4th Dimension provides a set of built-in button actions. When you assign a built-in button action to a button, you don't need to write a method to specify what happens when a user clicks the button.

The built-in button actions are:

- **OK** Save a new record or save changes to an existing record,
- **Cancel** Discard the new record or discard changes to an existing record,
- **Delete Record** Deletes the current record from the database,
- **Next Record, Previous Record, First Record, Last Record** Save the current record and display the desired record,
- **Next Page, Previous Page, First Page, Last Page** Display the desired page in a multi-page form.

Note When you insert a subform, 4th Dimension can automatically insert three additional subform buttons if you click the **Add Subform Button** check box in the Options area of the Subform page. The subform buttons are: **Open** (enables the user to add a new record by opening a Detail form), **Add** (enables the user to add a new record by entering data directly into the subform), and **Delete** (delete the currently selected record in the subform).

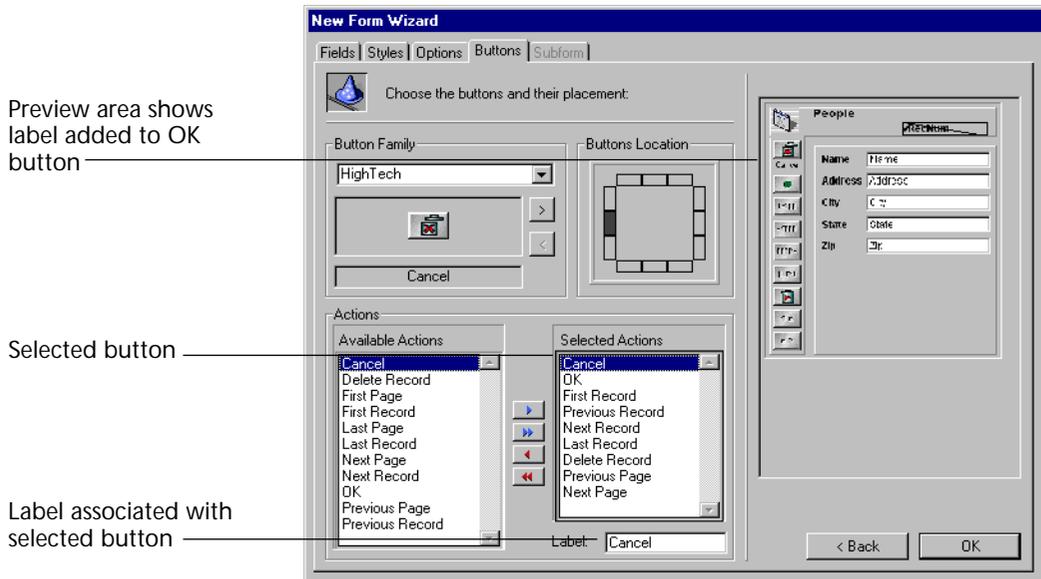
You can assign other predefined actions to buttons. These actions are available when you create a form using the Form Wizard or when you modify a form using the Form editor.

The buttons are listed in the Available Actions area of the page.



You can select and deselect automatic buttons in the same way that you can add or delete fields from the form on the Fields page. The buttons that you add to the Selected Actions area appear on the form.

If you want to label a button, highlight the button in the Selected Actions list and enter a label in the Label area. The following illustration shows a Cancel button being labelled. The OK button has already been labelled and the Preview area shows the label.

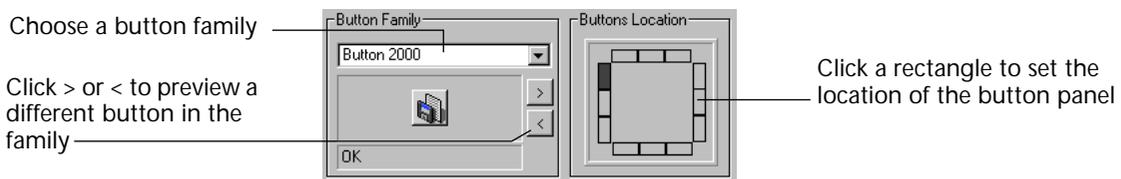


After entering the label, press Tab or click another button in the Selected Actions list. The label you entered is then displayed in the Preview area, as shown above.

Note By default, buttons belonging to graphic families (such as Cactus, Hitech, and so on) have no labels. If you choose to enter labels for those buttons, they will be displayed under each button.

Button tips are independent of button labels. If you want to assign a tip to a button, you can do so using the Object Properties palette in the Form editor. For more information, refer to [“Adding Help to a Field or Object” on page 353](#).

The Button Family and Button Location areas let you choose the style and location of the buttons.



Choose a family from the Button family drop-down list and click the > or < buttons to preview each button.

- ▶ To customize the buttons added to the form:
 - 1 Click the Buttons tab to display the buttons page in the Advanced screen.
The Buttons page appears.
 - 2 Choose a button style from the Button family drop-down list.
The Preview area shows your selected button family.
 - 3 In the Available Actions area, click each button action that you want to include on the form or click the Append button .
If you want to label a button, highlight it in the Selected Actions area and enter a label in the Label area.
 - 4 Select the location of the button palette by clicking on a rectangular area in the Button Location area.
The preview area indicates the effects of your choice.
 - 5 When you are finish adding buttons and button actions to the form, click another tab to customize another aspect of the form.
OR
Click OK to generate the form.

Setting the Form Size

The Options page of the Advanced screen lets you specify the form size. This section allows you to adjust the form size or set the form to a fixed size either by entering in its maximum width and height or by selecting a screen size. You can also combine the two settings.

The Screen Size drop-down list gives you the following choices:

- Low Resolution Windows,
- 9" Macintosh¹
- 9" Macintosh Powerbook,
- 13"
- 15" Macintosh,
- 15" Windows,
- 17",

1. Screen size of original one-piece "Classic" Macintosh.

- 20" Macintosh,
- 20" Windows.

When you enter a screen size or choose a size from the drop-down list, the preview area changes to reflect your selection. The Form Wizard will try to adjust field and object placement on the form so that all the form objects will fit in the selected screen size. If the **Create Multiple Pages if Necessary** option is selected and 4th Dimension cannot make all the fields fit in one page, it will generate multiple display pages to fit all the fields on the form. For more information about that option, refer to [“Form Display Options” on page 216](#). If the Form Wizard generates multiple pages, it places buttons, the form title, and decorative rectangles on the background page (page 0).

Adjust Size to Fields

If you check this check box, the Form Wizard will shrink the background items around the fields so that less blank space is left.

Field Label Placement

The Field Labels area in the Options screen allows you to control where a field label is placed in relation to the field. If you want labels, they can be placed either in front of or on top of the fields.

Form Display Options

The Display Options area on the Options screen lets you add several optional elements to the form and set additional options. Your choices are:

- **Record Number/Record Count** Adds a 4th Dimension variable to the form that display the current record number and the total number of records. That variable is named vRecNum. It can be edited in the Form editor as any other variable.
- **Form Title** Adds the name of the table as the title of the form above the fields.
- **Background Picture** Click the Background Picture check box and choose a background from the drop-down list to add a background to the entire form. Several interesting background patterns are shipped with 4th Dimension.

Note This option does create a background picture for a Web form. For information about how to set a background picture for a Web form, refer to [“Defining a Background Picture for Web forms” on page 313](#).

- **Associated menu bar** Enter the number of a menu bar that you want to associate with the form¹.
- **One Field per Line** Check to arrange the fields vertically. If this option is not checked, the Form Wizard will try to arrange fields in rows.
- **Create Multiple Pages if Necessary** Check to have the Form Wizard create extra pages automatically if the fields don't fit on one page. If you use this option, the Form Wizard places the appropriate objects on the background page.
- **Use Dynamic Field Names** When this option is selected, field and table names are inserted in the form as dynamic references. It ensures that the field and table labels will reflect any changes to the field or table name. Table or field names can be modified in the Structure editor or using the Table name or Field names commands. For more information see [“Inserting Dynamic Table and Field Names” on page 329](#).

Adding a Subform to the Form

When you want to use fields from a related Many table or from a sub-table, you add a subform to the form. The subform lists several records at once.

Using a subform allows you to view the related records or subrecords. You can also enter information into records and subrecords that are

1. You can also associate a menu bar with a form using the Form editor.

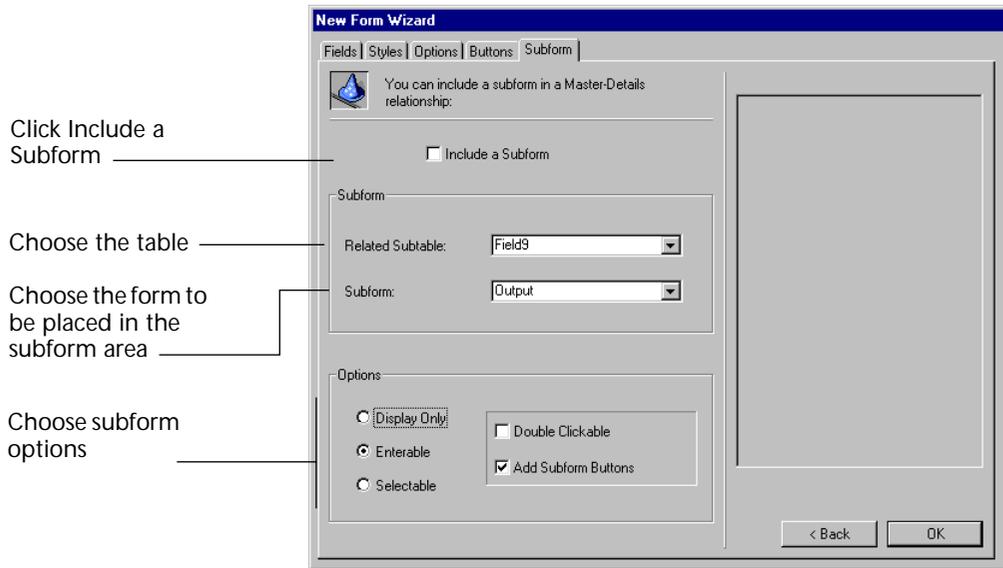
displayed in the subform. The figure below shows a Detail form with a subform during data entry.



You can display fields from a subtable, a related Many table, or an unrelated table in a subform. If you include fields from a related Many table, the relation determines which records are displayed. If you include fields from an unrelated table or from a table with a manual relation, by default the current selection of records from that table is displayed. You can also control the selection of records using a method.

You can include subfields from a subtable in a subform. The subrecords for the parent record are displayed in the subform. Only those subrecords that belong to the parent record are displayed.

The Subform page¹ of the Advanced screen lets you add a subform to the form, specify subform options, and add buttons to allow users to work with the subform.



► To add a subform to the form:

1 Click the Include a Subform check box.

If there is only one candidate related Many table or subtable, 4th Dimension adds the subform to the form. It appears in the preview area and the related Many table or subtable is shown in the Subform area.

2 If necessary, choose the desired related Many table or subtable from the drop-down list.

3 Choose the desired form to use as the subform from the drop-down list.

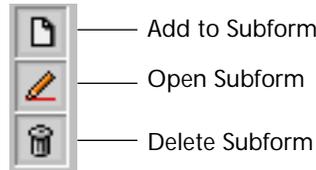
4 Choose any desired options from the Options area.

For information on subform options, see the section [“Data Entry Options for Subforms”](#) on page 431.

The Add Subform Buttons check box adds a standard subform button panel to the form. The button panel contains Add to Subform, Open

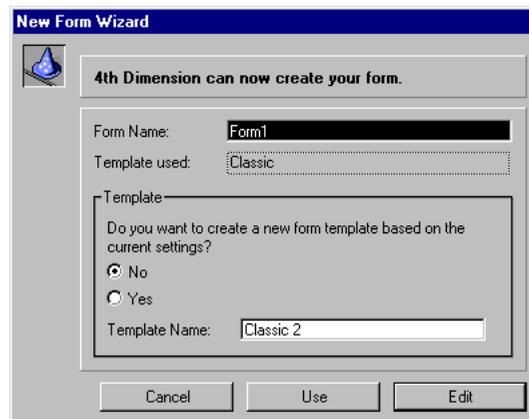
1. The Subform page is available only if the master table contains a subtable or has a related Many table.

Subform, and Delete Subform record buttons. The standard subform button panel is shown in the following illustration.



Creating the New Form

When you have finished specifying all properties of the new form, click OK on any page to create the new form. When you click OK, the following dialog box appears:



To create the new form, click either **Use** to switch to the User environment to test the form or **Edit** to open the new form in the Form editor.

Creating a Form Template

The Accept Form dialog box gives you the option of creating a new form template using the current Advanced settings. If you create a form template, its name will be added to the Form Template drop-down list in the Basic screen of the Form Wizard. The form template is saved in addition to the form itself.

- ▶ To create a form template:
 - 1 Click the **Yes** radio button in the **Template** area and enter a name in the **Template Name** area.
 - 2 Click either the **Use** or **Edit** buttons.

Edit opens the new form in the Form editor, ready for further customization. Use switches to the User environment so that you can begin using the new form.

If you click Use, you can return to the Design environment and open the form in the Form editor at any time.

Setting the Appearance of Form Objects

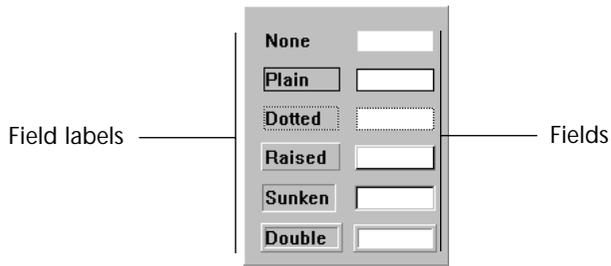
The Styles page of the Form Wizard lets you specify the platform interface, appearance, and font attributes of fields and field labels, text, and check boxes and radio buttons. This section describes the available options.

Fields and Field Labels

The Appearance drop-down list in the Styles page lets you customize the look of the fields you place on the form. Your choices are:

- **None** All objects appear with their normal graphical settings according to the selected Platform Interface and user choices.
- **Plain** The non-framable objects appear with their normal graphical settings according to the selected Platform Interface. The framable objects appear with a solid 1-pt frame.
- **Dotted** The non-framable objects appear with their normal graphical settings according to the selected Platform Interface. However, the dotted appearance overrides the selected pattern. The framable objects appear with a dotted 1-pt frame.
- **Raised** All objects (or their frames) appear with a 3D raised effect.
- **Sunken** All objects (or their frames) appear with a 3D sunken effect.
- **Double** On Macintosh, the objects or their frames appear with a double-line: two 1-pt solid lines separated by one pixel. On Windows, the objects or their frames appear with one black and one white lines offset by one pixel.

The following illustration compares the six appearance settings.



You can customize the appearance of the field and the field label separately.

After you create the form, you can set the Appearance of individual objects using the Object Properties window in the Form editor. For more information, see the section [“Changing the Appearance of Objects” on page 297](#).

Using the Style Sheet Editor

The Style Sheet editor lets you save sets of font attributes — font, font size, and style — as a named style sheet. The style sheet can then be used to specify font attributes in the Styles page of the Forms Wizard or the Font page of the Object Properties window.

Each named style sheet saves separate sets of font attributes for each platform interface supported by 4th Dimension. For example, the Macintosh platform interface could use Geneva as the font, while the Windows 95 and Windows NT 3.5.1 platform interfaces could use Arial. Similarly, the font sizes can be specified separately for each platform interface.

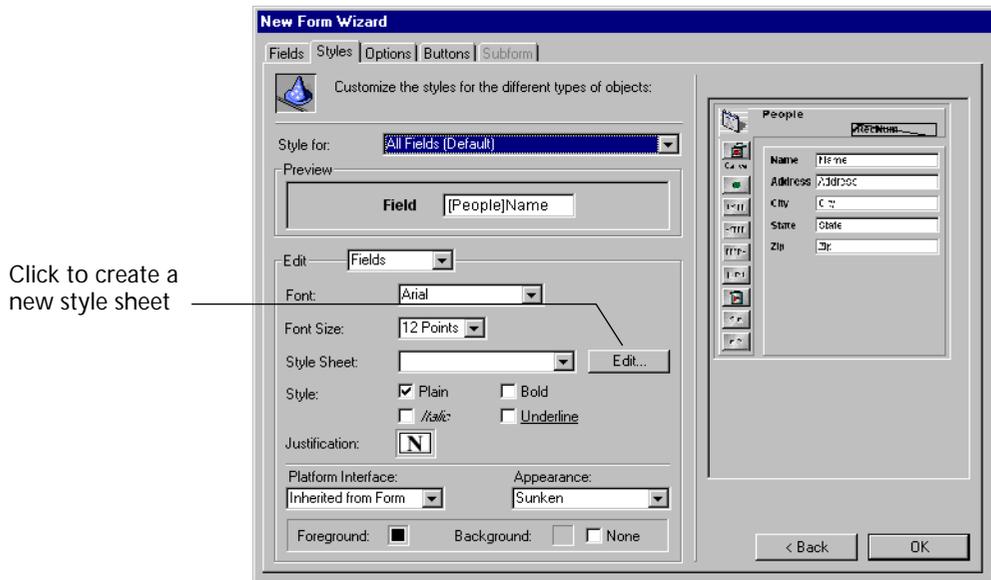
Creating a Style Sheet

You can create style sheets from several places in 4th Dimension:

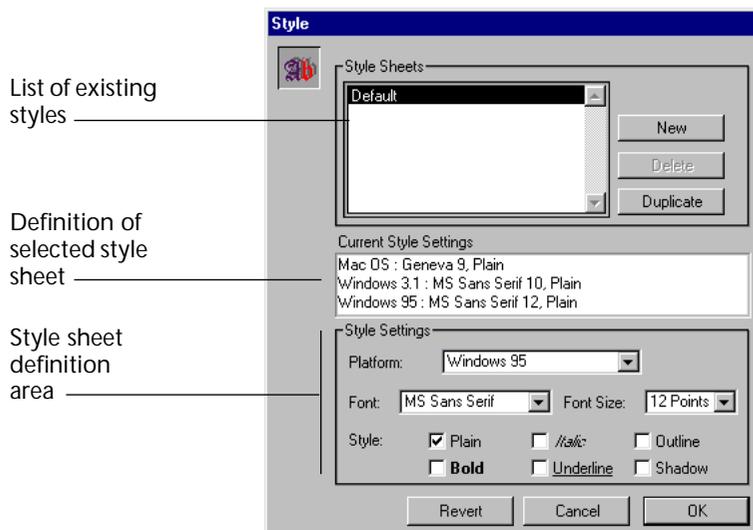
- From the User Interface page of the Database Properties dialog box. For information, see the section [“User Interface” on page 89](#).
- From the Styles page in the Advanced screen in the Form Wizard.
- From the Font page of the Object Properties window.

The dialog box used for creating style sheets is the same.

- To create a style sheet:
 - 1 Click the Style Sheet Edit button.



The Style Sheet Definition dialog box appears:



- 2 Click New.

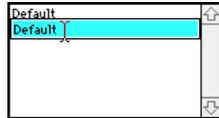
A new style sheet appears in the list of existing styles. Its default name is “Default.”

- 3 To rename the new style, hold down the Command key (on Macintosh) or the Ctrl key (on Windows) and click the new style sheet name.

OR

Double-click the style's name.

The text becomes editable.



- 4 Rename the style sheet and press Tab or click anywhere outside the entry area to save your changes.
- 5 In the Style Sheet definition area, choose a platform whose font attributes you want to define.
- 6 Choose the desired font, font size, and font style options.
The Definition area changes to reflect your changes.
- 7 Repeat steps 5 and 6 for each platform (optional).

When you are finished defining the style sheet, you can click **New** to create another style sheet or click **OK** to save the new style sheets and put away the Style Sheet Definition dialog box.

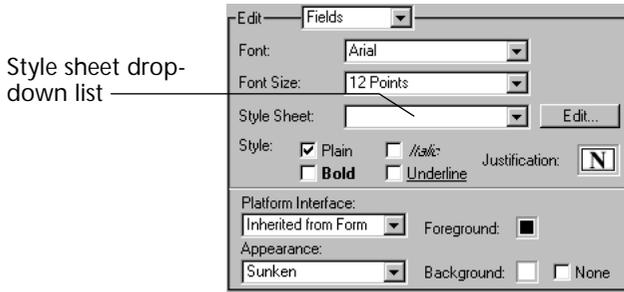
Note If you like, you can begin a new style sheet definition by duplicating an existing style sheet by clicking **Duplicate** rather than by clicking **New**. This will allow you to avoid having to respecify the properties of the new and old styles will have in common.

The **Revert** button allows you to cancel all the changes you made since you opened the **Style** dialog box. Clicking it reverts the **Style** dialog to the state it was when it was opened.

When the **Styles** page of the **Form Wizard** reappears, the new style sheet names appear in the **Style Sheet** drop-down list. You can then specify font attributes by choosing a style sheet rather than by making font, font size, and style selections.

Applying a Style Sheet

To apply a style sheet to an object, choose the name of the style sheet from the Style Sheet drop-down list.



Your selection sets the font, font size, and font style attributes for the currently selected label or object.

Setting the Current Input and Output Forms

Each table has one current input form and one current output form. The input form is used for entering and modifying records, and the output form is used to list records. Usually, you use a Detail form for input and a List form for output.

You can change which form to use for input and output at any time. You can change these designations in both the Design and User environments as well as using the commands INPUT FORM and OUTPUT FORM.

► To designate input and output forms:

- 1 Choose Edit Form from the Design menu.

OR

If the Explorer is already displayed, click the Forms tab to display the Forms page.

4th Dimension displays the Forms page of the Explorer.

- 2 If the Explorer is not already displaying the Preview area, click the Preview area icon.
- 3 Select the table whose form you want to set and expand it to view the forms for that table.
- 4 Select the form you want to use for input.
The Preview area shows the form.

- 5 Click the **Input Form** check box below the Preview area.
The **I** character (for Input) is then displayed next to the form's name. This indicates that the form is the current input form.
 - 6 Select the form you want to use for output.
Its preview appears in the Preview area.
 - 7 Click the **Output Form** check box.
The **O** character (for Output) is then displayed next to the form's name. This indicates that the form is the current output form.
- You can also designate the same form as the Input and Output form. In this case the character **B** (for Both) will be displayed next to it.

Deleting a Form

You can delete any form that is not currently designated as an input form or an output form (or both). The Delete button is disabled when you select the current input or output form.

- ▶ To delete a form:
 - 1 Choose **Edit Form** from the **Design** menu.
OR
If the Explorer is already displayed, click the **Forms** tab to display the **Forms** page.
4th Dimension displays the **Forms** page of the Explorer.
 - 2 Select the table that contains the form you want to delete and expand it to view the forms for that table.
 - 3 Select the form you want to delete.
When you select a form that is not the current input or output form, the **Delete** button becomes active.
 - 4 Click the **Delete** button.
4th Dimension asks you to confirm the deletion.
 - 5 Click the **OK** button.
4th Dimension deletes the form.

Renaming a Form

You rename a form in the Explorer.

► To rename a form:

- 1 Choose Edit Form from the Design menu.

OR

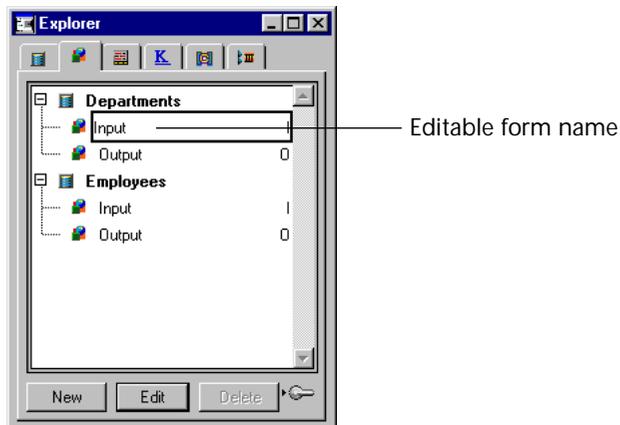
If the Explorer is already displayed, click the Forms tab to display the Forms page.

4th Dimension displays the Forms page of the Explorer.

- 2 Select the table that contains the form you want to rename and expand it to view the forms for that table.

- 3 Hold down the Command key (on Macintosh) or the Ctrl key (on Windows) and click the form name you want to rename.

The name becomes editable.



- 4 Type a new name.
- 5 Press Tab or click anywhere outside the entry area to save your changes.

4th Dimension changes the name of the form.

Note Changing a form name can invalidate any methods or formulas that use the previous form name. Each of these items has to be updated in order to function.

4D Server The form name is changed on the server when the user clicks outside the entry area to save the new name. If more than one user is modify-

ing the form name at the same time, the final form name will be the name specified by the last user to save the name.

You may want to specify a form owner so that only certain users can change the form's name. For more information about specifying owner privileges for a form, refer to the section [“Setting Form Access” on page 253](#).

4

Form Editor Basics

When you create a new form with the Form Wizard, you can choose many customization options. Using templates, you can control the font, font size, and style of text, control the appearance of fields and field labels, and add a set of automatic buttons.

This is only the beginning though, since 4th Dimension provides a full-featured Form editor that allows you to modify your form until you achieve the effect that you want. With the Form editor, you can create and delete objects, manipulate objects directly, and set form and object properties.

This chapter provides an introduction to the Form editor. It covers:

- The Form editor window,
- The Form editor Tools palette,
- The Form editor menus,
- Tools for displaying and editing form and object properties.

The following operations are explained in detail:

- Opening a form in the Form editor,
- Setting form properties,
- Creating and managing objects,
- Moving and resizing objects,
- Grouping and ungrouping objects,
- Aligning objects,
- Copying objects,
- Layering objects,

- Creating text areas on a form,
 - Adding a picture from the Picture library to the form,
 - Creating a multi-page form,
 - Changing the data entry order of the fields.
- [Chapter 5](#) covers fields and other active objects in detail.

4D Server Object locking occurs when two or more users attempt to modify the same form simultaneously. If a user is modifying a form in the Design environment, the form is locked. Other users cannot modify that same form until the first user frees the form by closing it.

Using The Form Editor

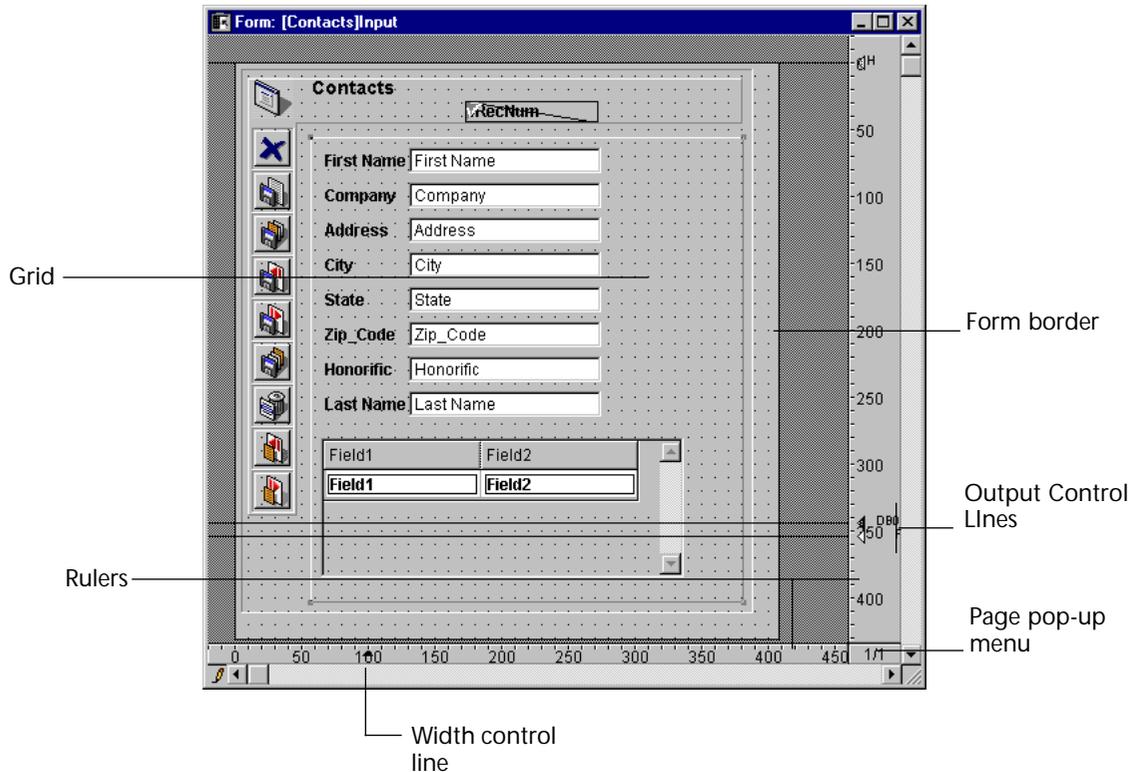
This section provides an overview of the Form editor and describes the tools available for editing forms. Subsequent sections describe in detail how to work with the Form editor's tools.

This section introduces the four components of the Form editor:

- The Form editor window,
- The Tools palette,
- The Form editor menus,
- Palettes for editing and displaying form and object properties.

Form Editor Window 4th Dimension's Form editor is a powerful object-oriented graphics editor that includes special features for working with fields and

interface objects. The following illustration shows an input form in the Form editor window.



The Form editor displays each form in its own window. You can have several forms open at the same time. The rulers on the side and bottom help you position objects in the form. You can change the units the ruler uses so that it measures in inches, centimeters, or pixels.

Note The output control lines are relevant only for output forms. They control the information that is listed and define header and footer areas. The label width triangle on the horizontal ruler controls the width of a label when you create a form for printing mailing labels. For information on working with output control lines, see the section [“Output Control Lines” on page 442](#).

The Form editor menus added to the 4th Dimension menu bar provide menu commands that allow you to change the data entry order of the fields, control interface elements and form pages, manage objects, and change the font and style for displaying information.

In addition to those menus, the Form editor also includes contextual menus that allow you to quickly access numerous actions. For more information, see the section [“Form Editor Menus” on page 238](#).

Showing/Hiding Elements in the Form Editor

You can show or hide most interface elements in the Form editor. This feature allows you to show only the elements that you need to create or view in a form or only the tools that you want to use. This option is always applied to the Form editor’s current window.

For example, it is useful to show the output control lines when you are working on an output form.

- To show or hide an element in the Form Editor:

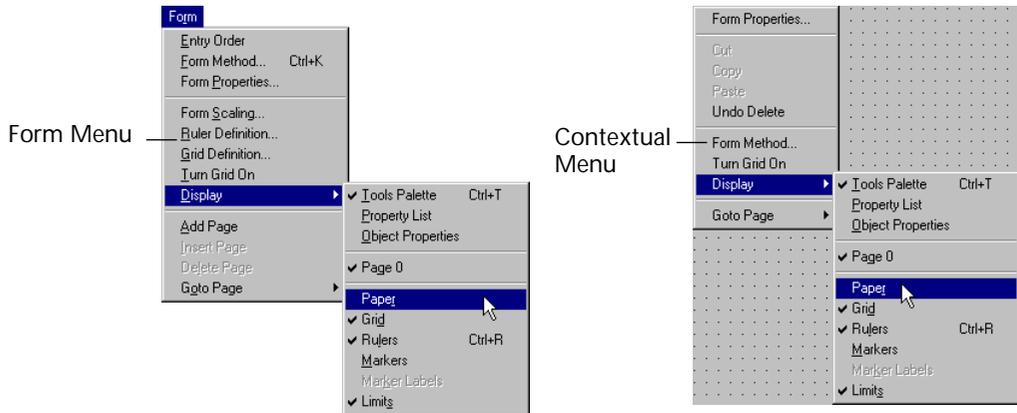
- 1 Choose Display from the Form menu.

OR

Use the Display command in the contextual menu that appears in the Form Editor’s window:

- On Windows, click with the right mouse button (without clicking on an object).
- On MacOS, Control+click (without clicking on an object).

A hierarchical submenu appears listing all the elements that you can show or hide:



A check mark placed next to the element indicates that it will be shown. To hide an element, select the element so that the check mark disappears.

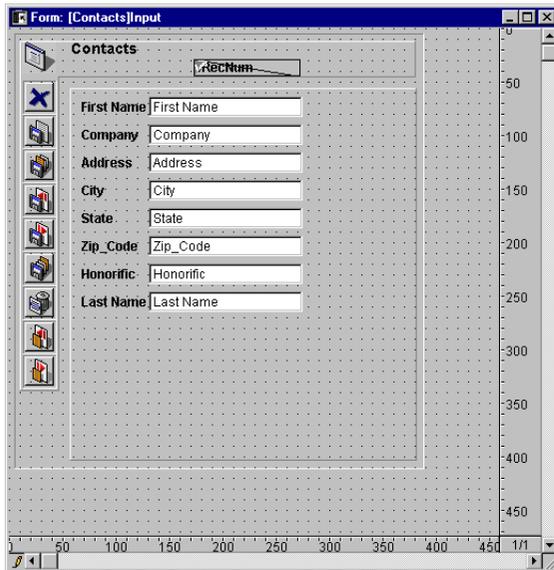
2 Select an element that you want to show or hide.

Here is a description of the commands in this menu:

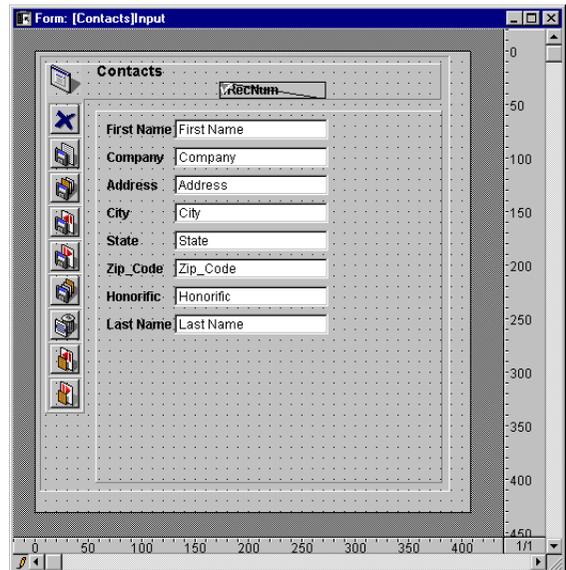
- **Tools Palette:** shows or hides the Form editor's Tools palette. This palette has been modified in version 6.5, it regroups the elements in the Tools and Object palettes in previous versions of 4th Dimension (see the section "The Tools Palette" on page 235).
- **Property List:** shows or hides the Property List. This list is described in the section "[Comparing the Property List to the Form/Object Property Windows](#)" on page 248. The Property List is an alternative to the Object Properties window. You can toggle between the Property List and the Object Properties window; you cannot display both simultaneously. For more information on both the Object Properties window and the Property List, refer to "[Displaying and Setting Form and Object Properties](#)" on page 244.
- **Object Properties:** shows or hides the Object Properties window. This window can be replaced by the Property List (see above).
- **Page 0:** shows or hides the objects from page 0 in the form's current page. This option allows you to distinguish between the objects on the form's current page and those on page 0. For more information about pages in forms, refer to the section "[Creating a Multi-page Form](#)" on page 314.
- **Paper:** shows or hides the borders of the printing page, which are shown as gray lines. This option can have no apparent effect when the Limits (see below) option is selected. If the size of the form is smaller than the printing page, the page's borders are shown outside of the form's viewing area and therefore do not appear.
- **Grid:** shows or hides the grid that is used as a guide when aligning objects. In previous versions of 4th Dimension, this grid was "invisible" and its appearance was related to the ruler unit that was chosen. The grid that appears in the form's background is independent from of the ruler; however, you can still use it to align objects on the form. You can define the grid's unit in the Grid Definition dialog box, which you can access by choosing Grid Definition in the Form menu. For more information on the grid, refer to "[Using the Alignment Grid](#)" on page 284.
- **Rulers:** shows or hides the rulers in the Form Editor's window as well as the pop-up menu that allows you to navigate the form's pages.

- **Markers:** shows or hides the output control lines and associated markers that show the limits of the form's different areas.
- **Marker Labels:** shows or hides the marker labels, available only when the output control lines are displayed. For more information, please refer to the [paragraph "Moving Output Control Lines", page 445.](#)
- **Limits:** shows or hides the form's limits. When this option is selected, the form is displayed in the Form editor as it appears in User mode.

Limits are not displayed



Limits are displayed

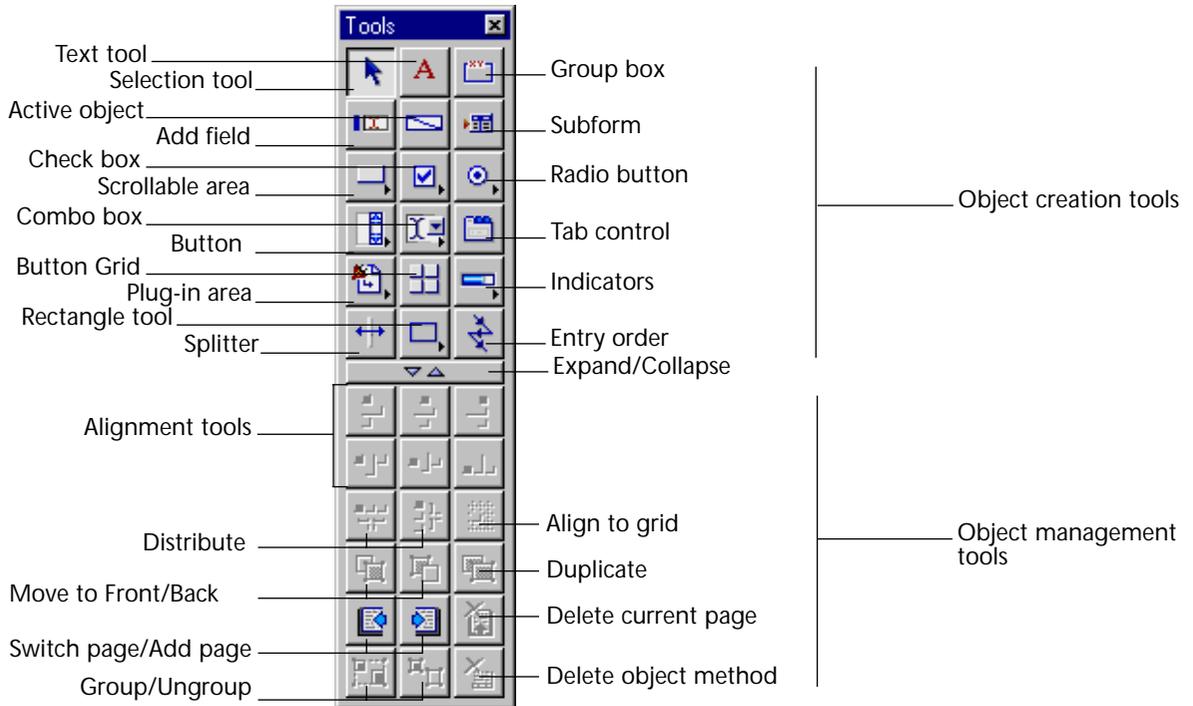


Note The Size Based on, Hor margin and Vert margin settings of the form properties affect the form's limits. When using these settings, the limits are based on the objects in the form. When you modify the size of an object that is located next to the form's border, it is modified to reflect that change. For more information on form properties, refer to ["Setting Form Properties" on page 251.](#)

The Tools Palette

The Tools palette provides a collection of tools for creating and manipulating form objects.

The following illustration shows the Tools palette.



The upper area of the Tools palette contains object creation tools. The lower area of the Tools palette contains object management tools. You can hide the lower area of the palette by clicking the  button located in the middle of the Tools palette. When you hide the object management area in the Tools palette, it appears as follows:



To display the lower part of the Tools palette, click the  button.

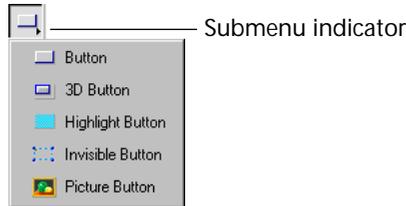
Object Creation tools You can create an object by

- dragging an object from the Tools palette to the form, or
- selecting a tool and drawing the object's shape on the form.

Once the object is created, it is selected and assigned default values. Depending on the display settings you selected, either the Property list or the Object Properties window is also displayed. For more information on object types and properties, refer to [“Types of Active Objects” on page 381](#).

After creating an object, you can modify its type and/or properties in the Object Properties window or in the Property List.

Some buttons in the Tools palette are drop-down lists that contain specific object types. These buttons are indicated by a small triangle:



Make your selection from the drop-down list associated with such a button.

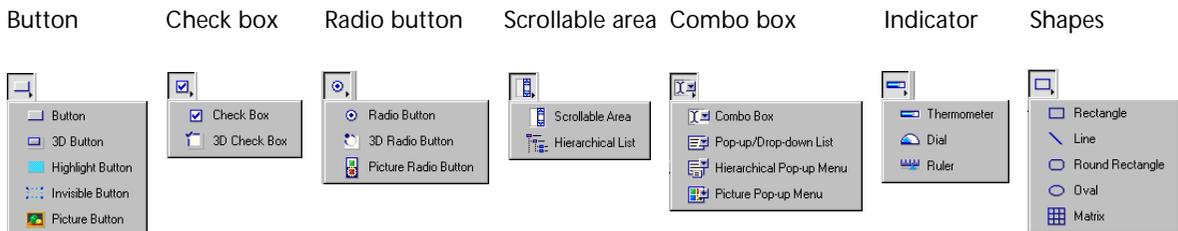
- ▶ To select a variation of an object type:
 - Click the object button and do not release the mouse button (long click) until the drop-down list appears.

OR

On Windows, right-click the object button.

On Mac OS, click the object button while pressing the Control key.

A menu is then displayed, which allows you to select an object. The menus are shown below:



Note Depending on the plug-ins that are installed in your database, the plug-in area may also be associated with a drop-down list.

Here is a brief description of the tools in the Tools palette:

- **Selection tool** Used to select, move, and resize objects in the form. For more information, see the section [“Managing Form Objects” on page 265](#).
- **Text tool** Used to create or edit text on the form for labels, titles, instructions, and so on. For more information, see the section [“Creating and Editing Text Areas” on page 299](#).
- **Group Box tool** Used to create a box with a built-in label in the upper left corner.
- **Add Field tool** Used to add a field to a form. For more information, see the section [“Adding Fields to a Form” on page 326](#).
- **Active Object tool** Used to create variables or (optionally) other types of active objects. For more information, refer to [“Enterable and Non-enterable Variables” on page 382](#).
- **Subform tool** Can be used to create an area that displays multiple records from another table or a subtable. For more information, see the section [“Adding a Subform to the Form” on page 428](#).
- **Button, Check box, Radio button, Scrollable area, Drop-down list, Tab control, Plug-in area, Button grid, Indicator, Splitter tools** Used to create active objects. For more information, refer to [“Active Objects on a Form” on page 372](#).
- **Graphic object tools** Used to create graphical objects such as circles, squares, rectangles, rounded rectangles, and matrixes. For more information, refer to [“Creating Objects” on page 276](#). For more information on how to create and use a matrix, refer to [“Duplicating on a Matrix” on page 419](#).
- **Entry order tool** Used to edit the entry order of the current form. When you click this tool, 4th Dimension displays the current entry order for the form, allowing you to modify it. For more information on the entry order, refer to [“Data Entry Order” on page 320](#).

Object Management Tools

The following is a brief description of the Object Management tools in the Tools palette:

- **Alignment and distribution tools** These tools are used to align and distribute objects in the form. These tools may be disabled, depending on the objects that are selected (i.e., unless at least two objects are selected, these tools are not meaningful). For more information, refer to [“Aligning Objects” on page 280](#).
- **Align to grid** Used to align objects on the grid that is defined for the form. For more information, refer to [“Using the Alignment Grid” on page 284](#).
- **Layering tools** Used to move objects to the front or back when objects overlap. For more information, see the section [“Layering Objects” on page 293](#).
- **Duplication tool** Used to duplicate the selected objects. For more information, refer to [“Duplicating Objects” on page 290](#).
- **Page tools** Used to move to the background page, next or previous display pages, or to add additional display pages to the form. For more information, see the section [“Creating a Multi-page Form” on page 314](#).
- **Grouping/Ungrouping tools** Used to group or ungroup a selection of objects. For more information, refer to [“Grouping Objects” on page 278](#).
- **Object method deletion tool** Used to delete the object method assigned to the selected object. For more information, refer to chapter 7, [“Creating Methods” on page 479](#).

Form Editor Menus When a Form editor window is the active window, the following menus are added to the menubar:

- Form,
- Object,
- Font,
- Style.

In addition to the standard menus, the Form editor also includes contextual menus that you can select at any time.

The Form Menu

Use the Form menu to organize form elements. Some of its commands display a hierarchical submenu. The figure below shows the Form menu.



The following is a description of the menu commands in the Form menu, with a brief description of their use:

- **Entry Order** Used to create a custom entry order for data entry objects in an input form. When the entry order mode is selected, a check mark is displayed next to that menu command. For more information, see the section [“Data Entry Order” on page 320](#).
- **Form Method** Selecting this menu command opens the form’s method in the Method editor. If no method assigned to the form, the New Method dialog box is displayed. For more information, refer to chapter 7, [“Creating Methods” on page 479](#).
- **Form Properties** Set or modify form properties. For more information, see the section [“Setting Form Properties” on page 251](#).
- **Form scaling** Enlarge or reduce all form objects by a specified factor. For more information, see the section [“Scaling a Form” on page 296](#).
- **Ruler definition** Used to set the scale of the form rulers. For more information, see the section [“Using the Rulers” on page 275](#).
- **Grid definition** Used to define the scale of the form grid to which objects are aligned. For more information, see the section [“Aligning Objects” on page 280](#).
- **Turn Grid On/Off** Used to turn on or off the grid to which objects can be aligned. For more information, see the section [“Aligning Objects” on page 280](#).
- **Display** Used to display or hide interface elements such as palettes, rulers, control lines, and so on. The commands of this submenu are

described in the section [“Showing/Hiding Elements in the Form Editor” on page 232](#).

- **Add Page** Selecting this menu command adds a page after the last page.
- **Insert Page** Selecting this menu command adds a page before the current page.
- **Delete Page** Selecting this menu command deletes the current page.
- **Goto Page** Selecting this menu command displays a hierarchical submenu that lets you choose between the form’s existing pages.

The Object Menu

Use the **Object** menu to modify and manipulate form objects. Some **Object** menu commands display a hierarchical menu of choices. The following illustration shows the **Object** menu.



Here is a description of the menu commands in the **Object** menu, with a brief description of their use.

- **Line Width** Displays a hierarchical menu of line width choices for lines and borders. For more information, see the section [“Line Widths” on page 304](#).
- **Fill** Displays a hierarchical menu of fill-pattern choices for objects. For more information, see the section [“Fill Patterns” on page 305](#).

- **Border** Displays a hierarchical menu of border-pattern choices for objects. For more information, see the section [“Border Patterns” on page 306](#).
- **Color** Displays a hierarchical menu of color choices for objects. For more information, see the section [“Foreground and Background Colors” on page 307](#).
- **Move to Front** Used to move an object in front of all other objects. For more information, see the section [“Layering Objects” on page 293](#).
- **Move to Back** Used to move an object in back of all other objects. For more information, see the section [“Layering Objects” on page 293](#).
- **Up One Level** Used to move the object selection up one level to the front.
- **Down One Level** Used to move the object selection down one level to the background.
- **Group** Used to combine multiple objects in the form into groups that you can manipulate as a single object. For more information, see the section [“Grouping Objects” on page 278](#).
- **Ungroup** Used to separate grouped objects into individual objects. For more information, see the section [“Grouping Objects” on page 278](#).
- **Align to Grid** Used to align an object to an invisible grid in the form. For more information, see the section [“Aligning Objects” on page 280](#).
- **Align** Selecting this menu command displays a submenu that allows you to choose between several alignment and distribution types. By selecting the Alignment submenu item, you can also display the alignment wizard. For more information about the use of the Alignment Assistant, refer to [“Aligning Objects” on page 280](#) and [“Distributing Objects” on page 286](#).
- **Duplicate** Used to duplicate objects. For more information, see the section [“Duplicating Objects” on page 290](#).
- **Duplicate on Matrix** Selecting this menu command duplicates an object N-1 times, using a matrix object that contains N cells. This menu command is enabled only if an object and a matrix are selected and the object is contained in the upper left cell of the matrix. For more information, refer to [“Duplicating on a Matrix” on page 419](#).

- **Duplicate many** Selecting this menu command displays the Duplicate many dialog. This dialog allows you to duplicate the same object several times and automatically assign each instance a numbered name. For more information, refer to [“Duplicating Objects” on page 290](#).
- **Show Format, Show Resource, Show Name** Controls what is displayed on the form for objects whose text is actually stored in resources. Show Resource shows the STR# resource in which the text of the object is actually stored. These menu commands also allow you to display the references for field and table labels if they have been inserted using dynamic references. For more information, refer to section [“Working with Text Areas” on page 299](#) and section [“Inserting Dynamic Table and Field Names” on page 329](#).
- **Object Method** Selecting this menu command opens the object’s method in the Method editor. If there is no object method assigned to the object and you haven’t chosen a default Method editor in Database Properties, 4th Dimension displays the New Method dialog box.
- **Clear Object Method** Used to remove an object method from a selected object. To remove a method, select the object and choose Clear Object Method from the Object menu.
- **Insert an OLE Object** This command allows you to create a plug-in area that is designed to become an OLE area. The use of OLE areas is described in a separate manual named “OLE Areas”.

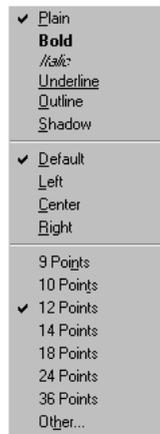
The Font Menu

Use the Font menu to apply a font to objects on the form and to specify a default font for text objects that you subsequently add to the form. The appearance of the Font menu depends on the screen fonts that you have installed on your system.

For more information, see the section [“Changing the Appearance of Objects” on page 297](#).

The Style Menu

Use the Style menu to apply a text style, alignment, and font size to objects in the form or to specify these attributes as defaults for objects that you subsequently add to the form.



For more information, see the section [“Changing the Appearance of Objects” on page 297](#).

You can set text attributes using either the **Font** and **Style** menus or the **Fonts** page of the **Object Properties** window.

Contextual menus

You can use contextual menus in the **Structure** window. To use a contextual menu: On Windows, click on an object or area with the right mouse button. On Mac OS, **Control+click** on an object or area.

With the exception of standard menu commands such as **Copy** and **Paste**, commands in contextual menus vary according to the current editor as well as the object on which you clicked:

- If you clicked a form object, the contents of the contextual menu are object-related: **Object Properties**, **Object Method**, **Color**, **Alignment**, **Border Line Style**, **Automatic Size**, **Automatic Action** (depending on the object’s type) and **Level**. Depending on the object’s type, additional commands may be displayed. For instance, **Attached Field** is displayed for a field, **List** for a hierarchical list.
- If you click outside any object, the contextual menu’s items apply to the form: **Form Properties**, **Form Method**, **Turn Grid On**, **Display** (for more information on this menu command, refer to [“Showing/Hiding Elements in the Form Editor” on page 232](#)) and **Goto Page**.

Displaying and Setting Form and Object Properties

Both forms and form objects have properties that control access to the form, the appearance of the form, and the behavior of the form when it is used. Form properties include, for example, the form's name, its menu bar, and its platform interface. Object Properties include, for example, an object's name, its dimensions, its background color, and its font.

This section describes how to display and modify form and object properties. For more information on form properties, refer to [“Setting Form Properties” on page 251](#). For more information on object properties, refer to [“Managing Form Objects” on page 265](#) and to chapter 5, [“Working with Fields and Active Objects” on page 325](#).

When you want to display or modify form or object properties, 4th Dimension lets you choose between two tools:

- For form properties, you can choose between using the Form Properties window and the Property List.
- For object properties, you can choose between the Object Properties window and the Property List.

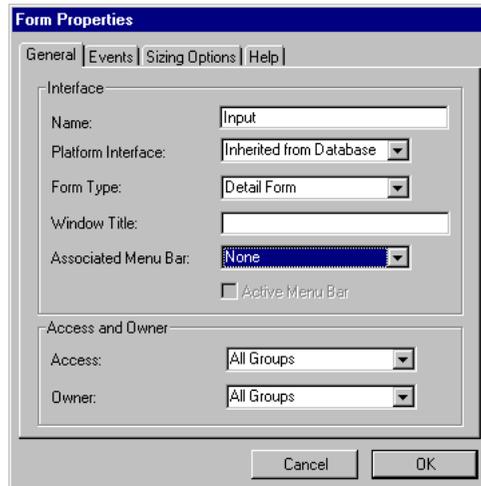
For more information on those tools, refer to [“Comparing the Property List to the Form/Object Property Windows” on page 248](#).

Form Properties

You can modify the current form's properties at any time. Modifications of the from properties can be done in both the Property List and the Form properties window.

- ▶ To display form properties in the form properties window:
 - 1 Choose Form Properties from the Form menu.
OR
Choose Form Properties from the contextual menu that appears when you right-click (on Windows) an empty area or press the Ctrl key while clicking an empty area (on Mac OS).

The Form Properties window consists of four pages, each of which can be selected using tab controls.



► To display the form properties in the Property list:

1 Select Property List from the Display submenu in the Form menu.

OR

Select Property list from the Display submenu of the Form contextual menu.

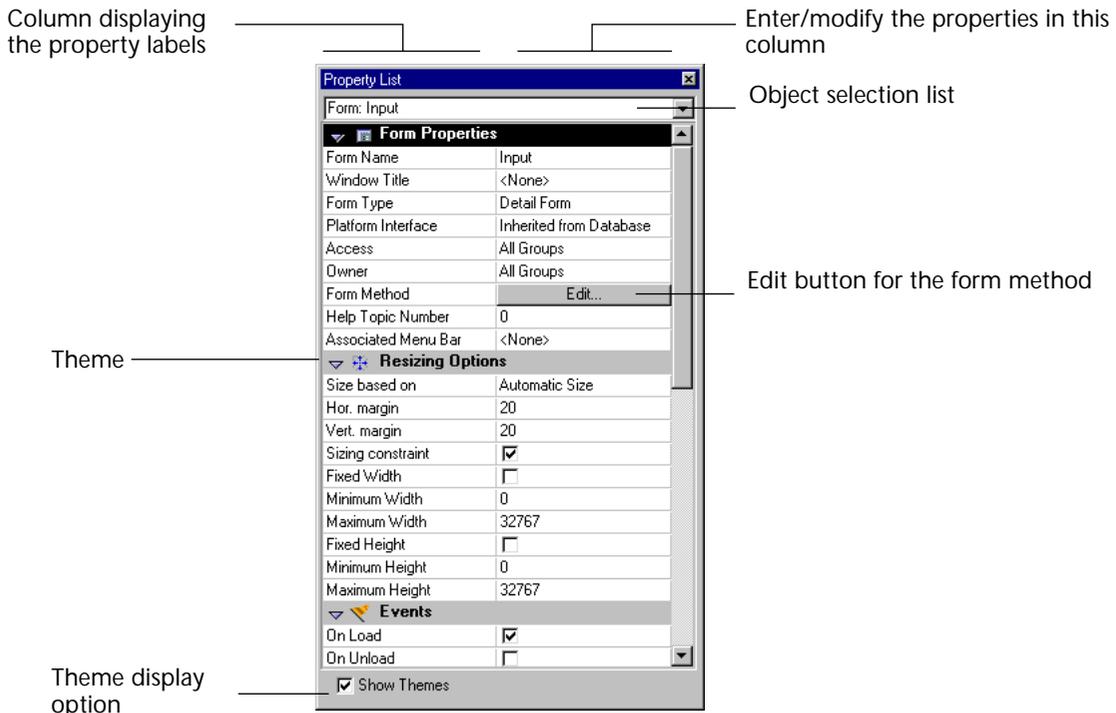
Note To toggle the display of the Property List when it is already selected in the Display submenu, press Command +Shift+Space Bar (on Mac OS) or Ctrl+Shift+Space Bar (on Windows).

2 Click an empty area of the form.

OR

Select Form: FormName from the object list located at the top of the Property List.

The Property List displays the current form's properties:



Note You can expand or collapse each theme by clicking the triangle located on the left of the theme name. It allows you to expand only the themes in which you want to work. You can also choose not to display the theme labels by deselecting the **Show themes** check box. If you deselect this option, properties will be displayed in alphabetical order.

Object Properties

When a form is opened in the Form editor, you can display and modify the properties of any object. You can modify object properties using either the Property List or the Object Properties window.

► To display the object properties in the Object Properties window:

1 Select Object Properties from the Form menu.

OR

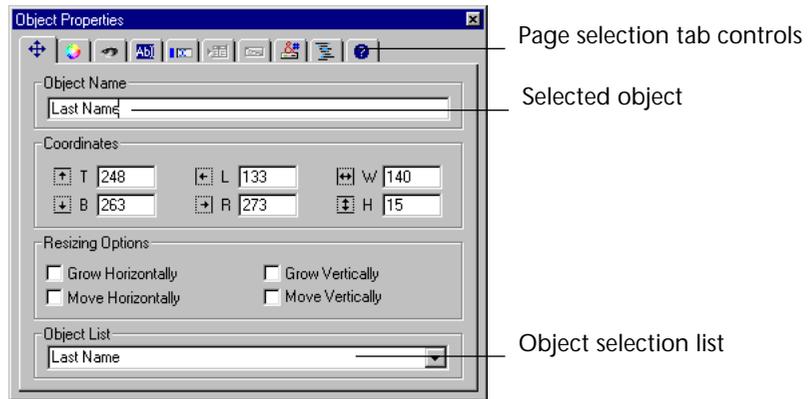
Select Form Properties from the contextual menu that appears when you right-click (on Windows) the object or press the Ctrl key while clicking the object (on Mac OS).

Note To toggle the display of the Object Properties Window when it is already selected in the Display submenu, press **Command +Shift+Space Bar** (on Mac OS) or **Ctrl+Shift+Space Bar** (on Windows).

2 Click the object whose properties you want to display.

OR

Select the object's name from the Object Selection list located at the bottom of the window.



► To display object properties in the Property list:

1 Choose Property List from the Display submenu in the Form menu.

OR

Choose Property list from the Display submenu of the form contextual menu.

Note To toggle the display of the Property List when it is already selected in the Display submenu, press **Command +Shift+Space Bar** (on Mac OS) or **Ctrl+Shift+Space Bar** (on Windows).

2 Click the object in the form.

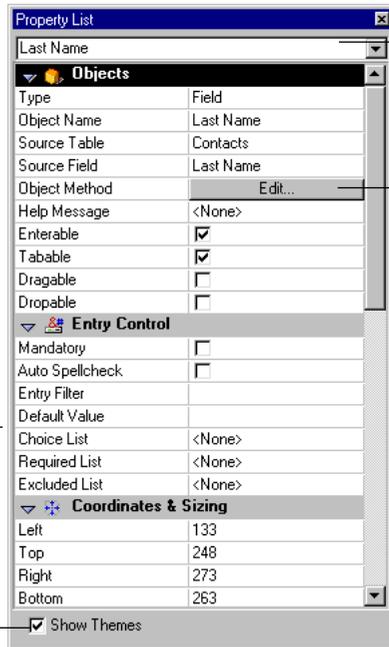
OR

Select the object name from the Object Selection List located at the top of the Property List.

The Property List displays the current object's properties:

Column displaying the property labels

Enter/modify the properties in this column



Object selection list

Edit button for the object method

Properties of the Entry Control theme

Theme display option

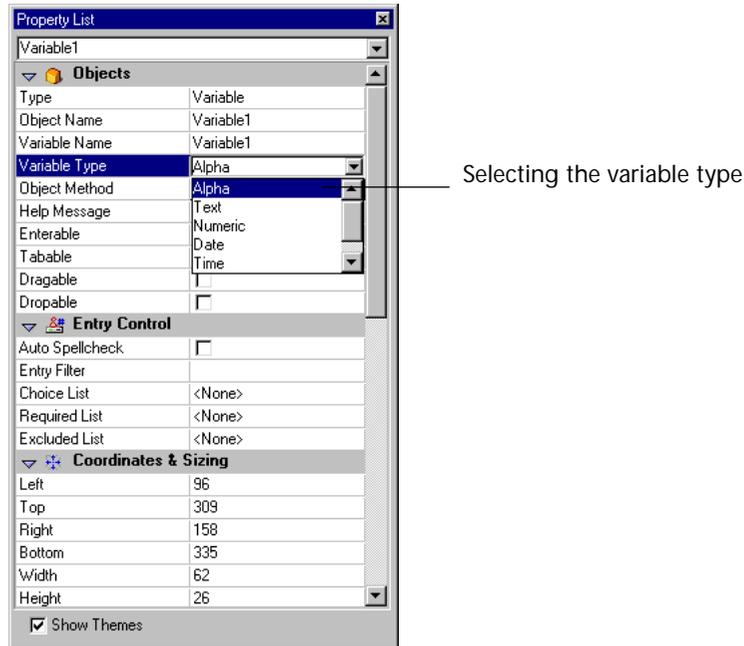
Comparing the Property List to the Form/Object Property Windows

Although the theme logic in the Property List is different from the logic used in the Object/Form Properties window (tab controls in the latter do not match the themes in the Property List), both types of windows offer the same type of features. The Property List differs from the Object/Form properties window in the following ways.

- The contents of the Property List depend on the object type or form that is selected. In other words, themes and properties are appropriate for the currently selected object.

In addition to the selective display of properties based on the object type, the Property List is dynamically updated when you modify an object's or a form's properties. For example if you set the Focusable property for an object, the Tableable property is displayed and becomes selectable.

Finally, the Property List allows you to set the type of the form variables. This allows you to modify the defaults.



- As is described above, the Property List displays the properties of either the currently selected object or the current form. This allows you to switch between the form and object properties without having to switch tools. In addition, the Property List provides direct access to the form method.
- The Property List manages objects within “families.” This allows you to change an object’s type and still maintain coherent data.
- The Property List does not allow you to edit style sheets or tips. These two object properties can be accessed only in the Object Properties window.
- To quickly navigate in the Property List, you can use the following keyboard shortcuts:
 - Arrow keys Used to go from one cell to another.
 - PgUp and PgDn Used to select the first or last visible cell of the property list.
 - Home and End Used to select the first or last cell of the property list.

- **Ctrl + click (Windows) or Command + Click (Mac OS)** on a theme label Used to Collapse/Expand every theme in the list.
- **Ctrl + click (Windows) or Command + Click (Mac OS)** on an event Used to select/Deselect every event in the list.

Opening a Form in the Form Editor

You can edit an existing form in the Form editor at any time.

► To open a form in the Form editor:

1 Choose Edit Form from the Design menu.

OR

Press **Ctrl+L** (on Windows) or **Command-L** (on Macintosh).

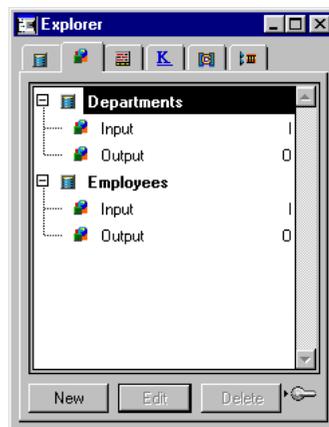
OR

Hold down the **Ctrl** key (on Windows) or the **Command** key (on Macintosh) and double-click the table title in the Structure window whose forms you want to open.

OR

In the structure window, click the table name using the right mouse button (on Windows) or press the **Control** key while clicking the table name (on Mac OS), then select **Show Table forms** from the contextual menu.

4th Dimension displays the Forms page of the Explorer. You can expand any of the table names to display the forms associated with it.



If you have double-clicked a table's name or used the contextual menu of the Structure window, that table is already selected.

- 2 If necessary, expand the table name that contains the form you want to modify.
 - 3 Select the name of the form you want to modify and click the Edit button.
OR
Double-click the name of the form.
- 4th Dimension displays the form in a Form editor window.

4D Server Object locking occurs when two or more users attempt to modify the same form simultaneously. If a user is modifying a form in the Design environment, the form is locked. Other users cannot modify that same form until the first user frees the form by closing it.

Setting Form Properties

Form properties can be set in the Property list or in the Form properties window. For more information on those tools, refer to [“Displaying and Setting Form and Object Properties” on page 244](#).

Naming the Form

You can give the form a name in the Form Properties window. You can also rename a form using the Explorer. You use the names of forms when you are establishing default input and output forms for a table and in commands that accept a form name as a parameter, such as INPUT FORM and OUTPUT FORM.

You cannot use the same name for more than one form per table. This will confuse 4th Dimension when you try to refer to a form by name. You can, however, use the same name in different tables. For example, you can name all your input forms “Input” and all your output forms “Output.”

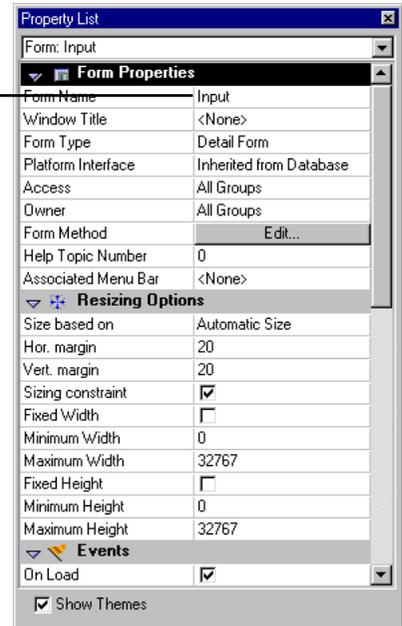
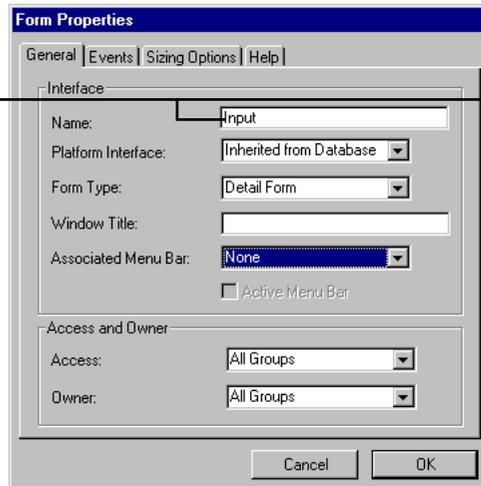
- ▶ To name a form using the Form Properties window:
 - 1 With the desired form in the frontmost window, choose Form Properties or Property List from the Form menu.

These two choices are described in “[Displaying and Setting Form and Object Properties](#)” on page 244.

Form Properties Window

Property List

Name entry areas



- 2 Enter a name in the Name area.
 - 3 If you have used the Form properties window, click OK to put it away.
- To rename a form using the Explorer:
- 1 Click the Forms tab to display the Forms page.
A hierarchical list of tables and forms appears.
 - 2 Expand the table containing the form you would like to rename.
 - 3 Hold down the Ctrl key (on Windows) or the Command key (on Macintosh) and click on the form name.
The form name becomes editable.
 - 4 Replace the current form name with the new name.
 - 5 Press Tab or click anywhere outside the entry area to save the new name.

If you rename a form that is referred to elsewhere in the database (such as in methods), update the references to the form.

Setting Form Access You can control access to a form by setting Access and Owner privileges for groups of users. A single group can be assigned for each privilege using the Access and Owner drop-down lists. For information about creating a password access system with users and groups, see [Chapter 9, “Managing Password Access” on page 555](#).

The Access drop-down list controls which group can use the form in the User environment or in custom applications. If a user that is not in this group attempts use the form, 4th Dimension displays a message saying that the user’s password does not allow him or her use the form.

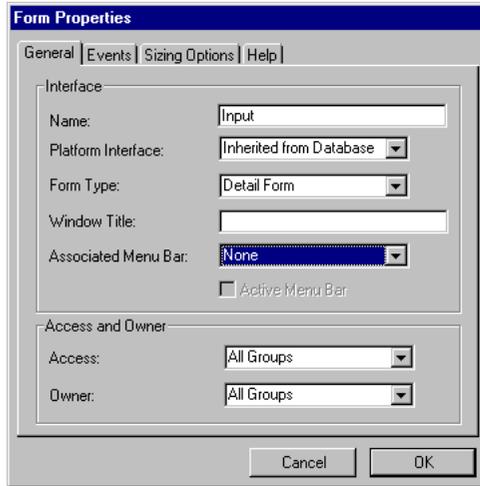
The Owner drop-down list controls which group can edit the form in the Design environment. If a user who is not in this group attempts to edit the form in the Design environment, 4th Dimension displays a message saying that the user does not have the access privilege to edit the form.

Users who are assigned to both groups can use the form in both the User and Design environments and in custom applications.

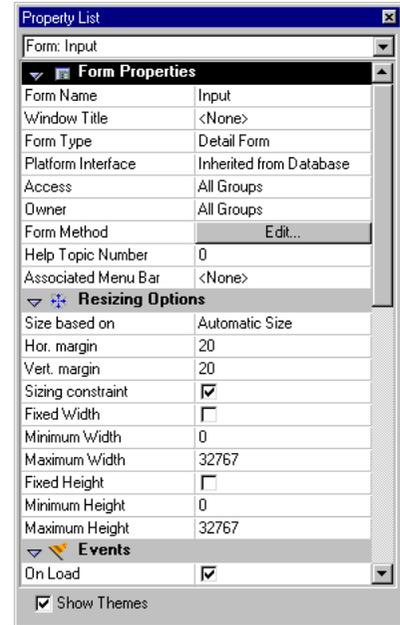
- ▶ To set access privileges for a form:
 - 1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in “[Displaying and Setting Form and Object Properties](#)” on page 244.

Form Properties Window



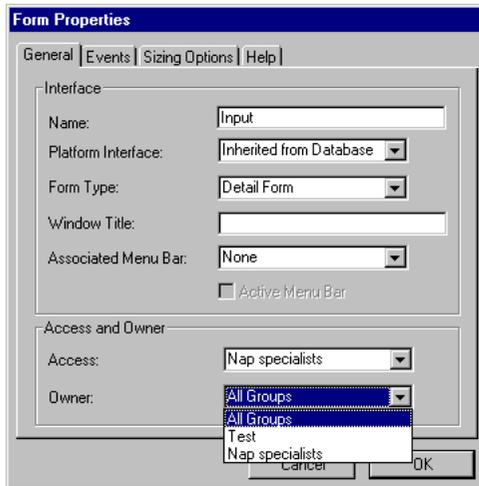
Property List



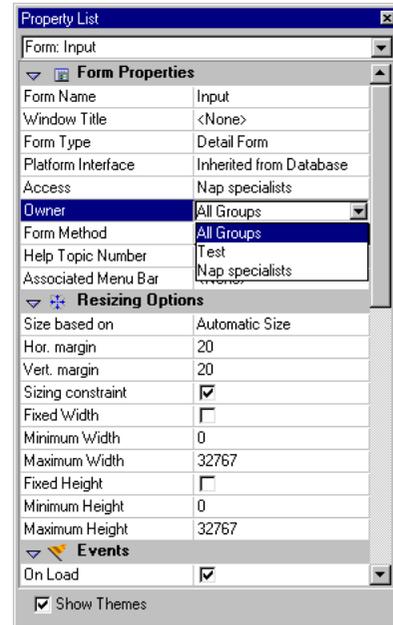
- 2 Use the Access and Owner drop-down lists to make the desired access privileges assignments.

The names of existing groups are displayed in each drop-down list.

Form Properties Window



Property List



- 3 If you have used the Form properties window, click OK to put away the Form Properties window.

Setting the Platform Interface

You can set the platform interface for the entire database (in the [Database Properties](#) dialog box) and for individual forms in the Form Properties window. At the form level, your choices are:

- **Inherited from Database** Use the platform interface selection you made in the Database Properties dialog box.
- **Automatic** Use the platform interface for the platform on which the database is currently running. That is, if the database is deployed on Macintosh, the form will be displayed using the Macintosh platform interface, and so forth.
- **Mac OS, Windows NT 3.5.1, Windows 95, or Platinum** Choose a specific platform interface that will be used regardless of which platform on which the database is actually running.

The default platform interface setting for a form is **Inherited From Database**.

► To set a platform interface:

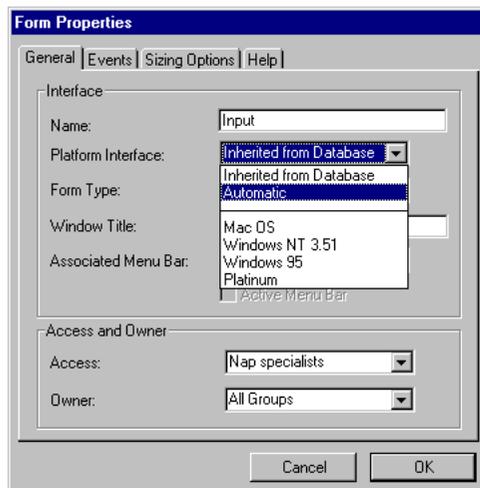
- 1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in [“Displaying and Setting Form and Object Properties”](#) on page 244.

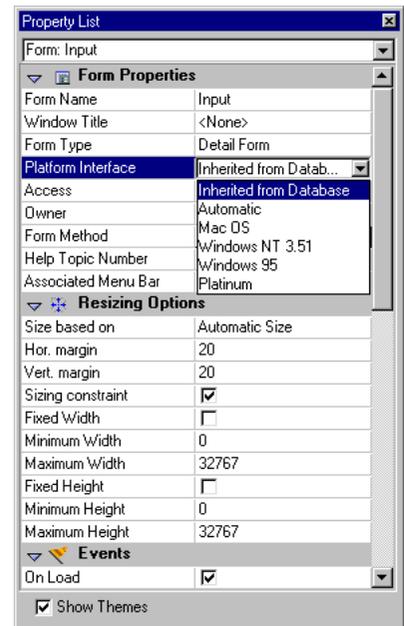
- 2 Choose the desired platform interface from the Platform Interface drop-down list.

Names of platforms are displayed in each drop-down list.

Form Properties Window



Property List

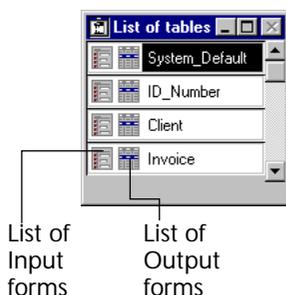


- 3 If you have used the Form properties window, click OK to put it away.

Choosing a Form Type

You can set the form type (detail form, list form, detail form for printing, or list form for printing) for a specific form.

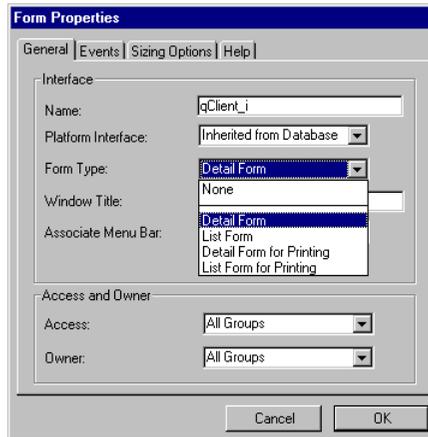
This property allows you to restrict the number of forms displayed in the current Input and Output form selection lists (the List of tables window in the User environment). Only forms whose type corresponds to a list are displayed.



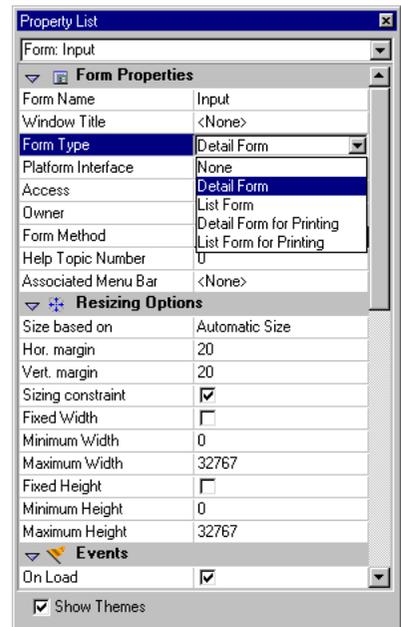
- ▶ To modify the form type:
 - 1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in [“Displaying and Setting Form and Object Properties”](#) on page 244.
 - 2 Select the desired form type, from the Form Type drop-down list.

Form Properties Window



Property List



- 3 If you have used the Form properties window, click OK.

When the form type is “None,” it is displayed in both the Output and Input form menus in the List of Tables window. By default, the form type assigned to forms that come from databases created with an earlier version of 4th Dimension that are then opened with 4th Dimension 6.5 is “None.”

Setting the Default Window Title

The default window title is used when the form is opened using the Open window and Open form window functions in custom applications. The default window title appears in the Title bar of the window. To set the default window title, enter it in the Window Title entry area.

You can use dynamic references to define the window title. The reference is resolved at the time the INPUT FORM command is called (when the * parameter is passed and the call to that command follows a call to Open window) and Open form window.

The following types of dynamic references can be inserted in a window's title:

- a STR# resource reference: the syntax to apply is “:16000,2” where 16000 is the resource number and 2 is its element.
- a table or field label: the syntax to apply is <?[TableNum]FieldNum> or <?[TableName]FieldName>. For more information, refer to [“Inserting Dynamic Table and Field Names” on page 329](#).
- a variable or a field: the syntax to apply is <VariableName> or <[TableName]FieldName>. The current value of the field or variable will be displayed in the window title.

Note The number of characters for a window title is limited to 31 characters.

Assigning a Menu Bar to a Form

When you create a custom database, you create custom menus. Custom menus allow you to add menu commands for automating specific tasks in the database, such as, for example, creating a report.

Custom menus are created in the Menu bar editor. Each menu bar that you create includes at least one menu and is assigned a unique ID number. For more information on creating menu bars, menus and menu commands, refer to chapter 8, [“Creating Custom Menus” on page 531](#).

When in User environment, a menu bar that is assigned to a form is added to the right of the current environment menu bar. A form menu bar is added only when the form is used for data entry.

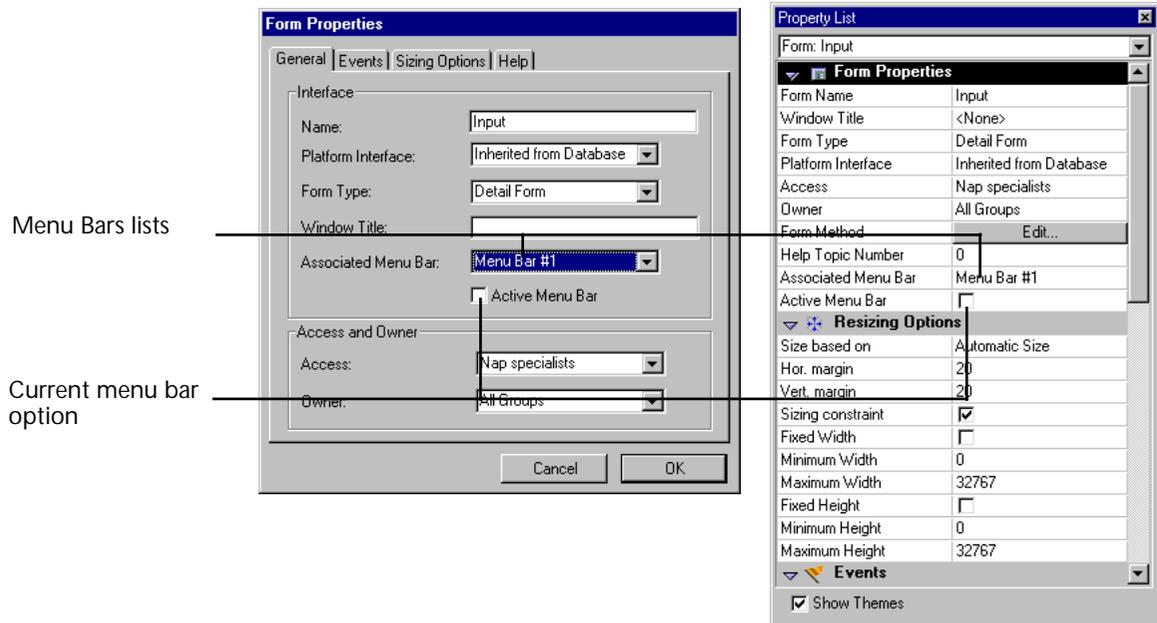
In custom menus, a menu bar that is assigned to a form is added to the right of the current menu bar. The custom menu bar menu bar disables other menus unless the Active Menu Bar option is selected. If the custom menu bar is identical to the current menu bar, it is not added.

- ▶ To assign a menu bar to a form:
 - 1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in “[Displaying and Setting Form and Object Properties](#)” on page 244.

Form Properties Window

Property List



2 Select a menu bar from the Associated menu bar List.

When the current form is displayed in User environment, the selected menu bar will be displayed to the right of the current menu bar.

3 If you want to use that form in a custom application, select the Active Menu Bar option.

This option tells 4th Dimension not to disable the current menu bar. If this option is not selected, 4th Dimension disables the current menu bar and only permits access to the form’s menu bar.

4 If you have used the Form properties window, click OK to put it away.

Setting Sizing and Resizing Options

When an input form is displayed in a custom application, you ordinarily open the form using the Open window or Open form window functions. Open window lets you specify the top, left, bottom, and right coordinates of the window as well as the window type. If you do not use any resizing options, the user's ability to resize the window depends on the window type.

Open form window creates a new window using the sizing and resizing options that are set in the Form editor.

The sizing and resizing options available in the Form Properties window give you greater control over sizing and resizing of the window.

Size Options

Size Options gives you control over the initial size of the window. Your choices are:

- **Automatic size** The size calculated by 4th Dimension based on the size necessary to display all objects on the form.
- **Set size** The size you enter in the width and height entry areas.
- **Size based on selected form object** The size based on a selected form object. 4th Dimension uses the smallest size necessary to display the selected object. For example, if you choose an object that is placed at the bottom-right corner of the area to be displayed, 4th Dimension will open a window just large enough to include the object.

Choose this option if you need to place some active objects in an off-screen area (i.e., outside the bounding rectangle). When you use this option, these objects will not affect the size of the window.

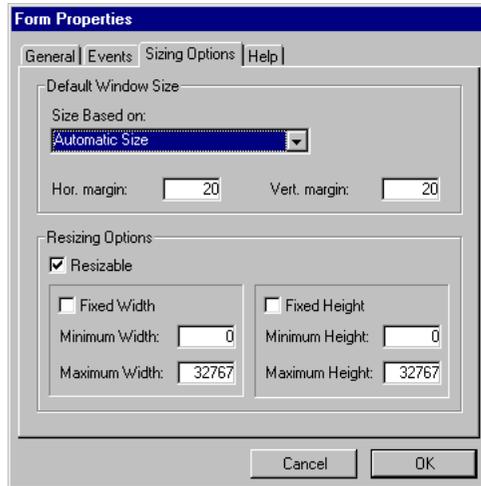
When you select either automatic size or size based on a selected object, the Width and Height areas change to Horizontal Margin and Vertical Margin, respectively. You can then enter a margin (in pixels) that defines a border area so that the edges of the object in the bottom-right corner of the form is not flush against the edge of the window.

- ▶ To set the sizing options:
 - 1 Display the form properties in the Form Properties window or in the Property List.

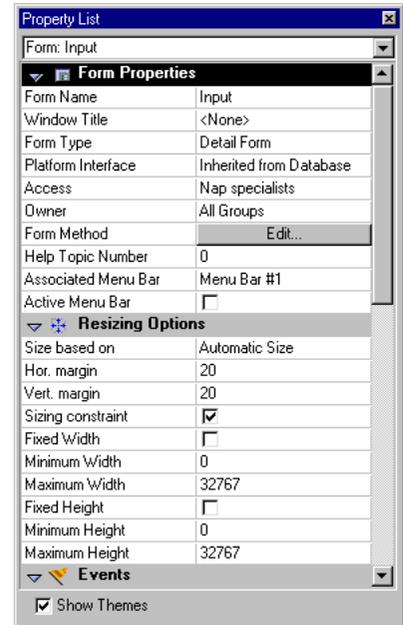
These two choices are described in [“Displaying and Setting Form and Object Properties” on page 244.](#)

- 2 In the Form Properties Window, click the Sizing Options tab.
 OR
 In the Property List, expand the Resizing Options theme.
 The Sizing options are displayed.

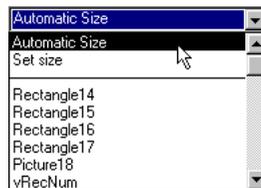
Form Properties Window



Property List



In the Default Window Size area, the Size Based On drop-down list controls the initial size of the window.



Names of objects on the current form

The two options above the dividing line let you choose either Automatic or Set Size. If you choose Set Size, you need to enter the desired Width and Height into the appropriate entry areas.

All the objects on the current form are listed below the dividing line in the drop-down list. To size the window based on a form object, choose the desired form object from the drop-down list.

- 3 Choose the desired sizing option from the drop-down list.
If you did not choose Set Size, the Width and Height areas change to Horizontal Margin and Vertical margin.
- 4 If you chose Set Size, enter the desired width and height in the entry areas.
OR
If you chose either Automatic Size or a form object, enter the desired horizontal and vertical margins (in pixels) in the appropriate entry areas.

Resizing Options

The resizing options in Form Properties let you make windows of any type resizable in custom applications and let you set the minimum and maximum sizes. Setting the minimum size is a way of preventing users from resizing a form so that the button panel or other essential objects are no longer visible.

- To make the window resizable:
 - 1 Click the Resizable check box.
When you click Resizable, the Fixed Width and Fixed Height check-boxes are enabled and the minimum and maximum entry areas become enterable (in the Property List, those areas are displayed only if the Resizable option is selected).
 - 2 Click either Fixed Width or Fixed Height to prevent resizing in either direction (Optional).
For example, you may have a column of buttons and you want to prevent the user from resizing the window so that some of the buttons are no longer visible. You would choose Fixed Height.
 - 3 Enter a minimum value in either dimension to prevent the user from making the window too small (Optional).
OR
Enter a maximum value to prevent the user from making the window too big (Optional).
You would use this option to prevent the user from hiding necessary entry areas or controls.
 - 4 If you have used the Form properties window, click OK to put it away.

Form Events

You can write a form method using the Method editor. If you do so, you should indicate which form events should be executed. You do this in the Events page of the Form Properties window or in the Events

theme of the Property List. When the form is used, only the events that you select will actually occur.

For information on the available events, see the section [“Form and Object Events” on page 487](#).

Your database will run faster if you deselect superfluous events.

► To activate events for the form:

1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in [“Displaying and Setting Form and Object Properties” on page 244](#).

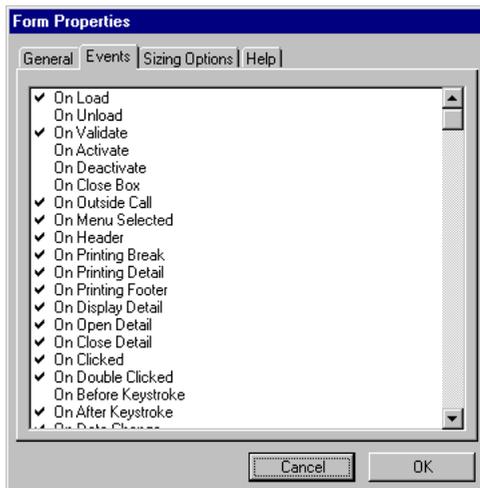
2 In the Form Properties Window click the Events tab.

OR

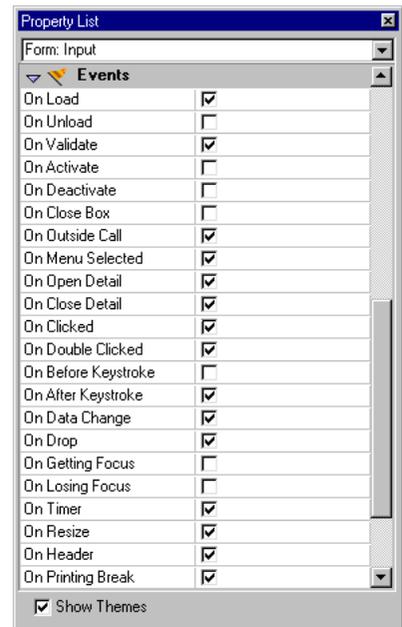
In the Property List, expand the Events theme.

The list of events is displayed.

Form Properties Window



Property List



A check box next to the event indicates that the event will occur when the form is used.

3 Select only the events that are needed.

To select or deselect all events, hold down Ctrl (Command on Macintosh) and click an event.

4 When you are finished setting Form properties, click OK.

Contextual On-line Help

4th Dimension allows you to associate a custom on-line help file to each database. The creation of help files is described in [Appendix B on page 641](#).

Help files can be contextual, which means that they can display information related to the context from which they were called. To do so, you can associate a precise section of this help file with each of the database's forms.

► To associate a section number with a form:

1 Display the form properties in the Form Properties window or in the Property List.

These two choices are described in “[Displaying and Setting Form and Object Properties](#)” on page 244.

2 In the Form Properties window, click the Help tab.

OR

In the Property List, expand the Form Properties theme.

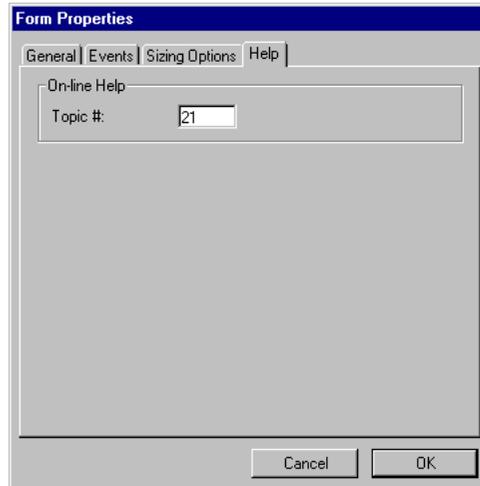
3 Enter the help section number you want to associate with the form in the Help Topic Number area.

4 If you have used the Form properties window, click OK.

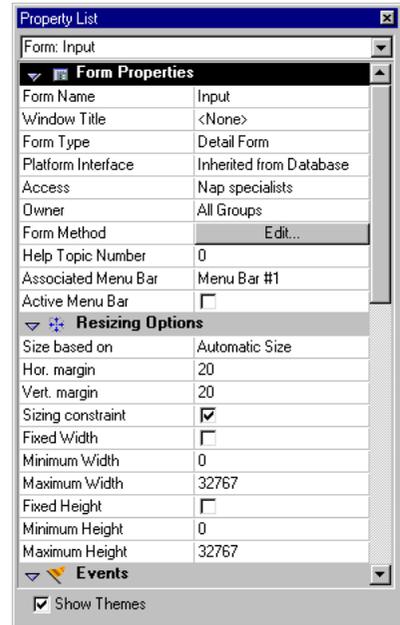
You can repeat this procedure for each form that needs to be assigned a help file.

Make sure you assign help topic numbers that match numbers defined in the help file. For more information on this point, refer to [Appendix B on page 641](#).

Form Properties Window



Property List



Managing Form Objects

You customize a form by creating and manipulating objects in the form. You use the Form editor to draw the objects, modify them, arrange them, set their properties, or delete them.

You can select and modify any object in a form, including the fields, buttons, and graphic objects created by the Form Wizard.

Selecting Objects

Before you can perform any operation on an object (such as changing a line width or font), you need to select the object that you want to modify.

When the Form Wizard creates a form, it creates one or more display pages and a background page. The Form Wizard may place buttons, the form title, and decorative rectangles on the background page. Objects on the background page are selectable only from the background page. If you have difficulty selecting an object that was created by the Form Wizard, switch to the background page and try again. You can also choose to hide any element located on the background page by

deselecting Page 0 from the Display submenu of the Form menu (or the Form editor contextual menu).

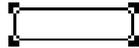
- ▶ To select an object using the Tools palette:

- 1 Click the Arrow tool  in the Tools palette.

When you move the pointer into the form area, it becomes a standard arrow-shaped pointer.

- 2 Click the object you want to select.

Resizing handles identify the selected object.



- ▶ To select an object using the Object Properties window or the Property List:

- Choose the object's name from the Object List drop down list located at the bottom of the Object Properties window.

OR

Choose the object's name from the Object List drop down list located at the top of the Property List.

Using these two methods, you can select an object that is hidden by other objects or located outside the visible area of the current window.

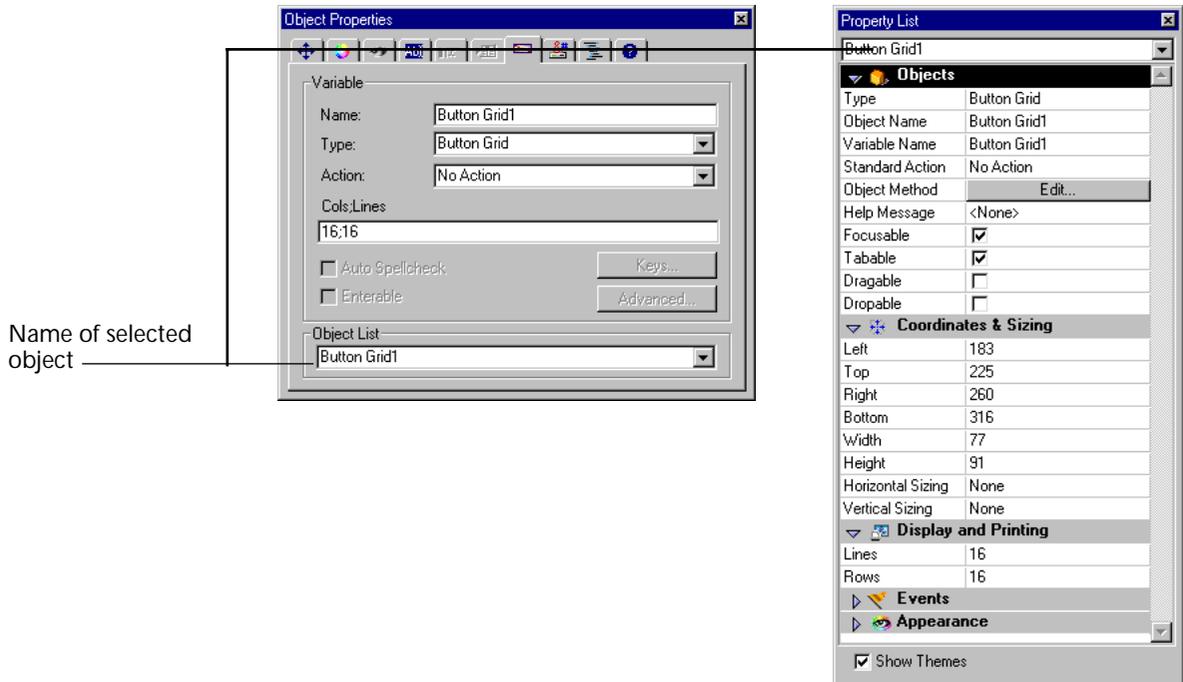
Viewing or Modifying Object Properties

You view or modify an object's properties using the Object Properties window or the Property List (for more information about these two methods, refer to [“Displaying and Setting Form and Object Properties” on page 244](#)). To select the window you want to work with, select it from the Display submenu of either the Form menu or the Form editor contextual menu. To hide or show the selected window, double-click the object or press Ctrl+Shift+Space (Command–Shift–Space on Macintosh).

When either window is on screen, click the object or select it from the Object List drop-down list.

Form Properties Window

Property List



The Object List drop-down list contains the names of all the objects on the current page. You can select an object and view its properties by choosing its name from this list.

Viewing or Modifying an Object's Method

To view the object's method or create a new method for the object hold down the Alt key (on Windows) or Option key (on Macintosh) and click the object¹. For more information about object methods, see the section [“Using Object Methods with Fields and Objects” on page 422](#) and [“Object Methods” on page 480](#).

To deselect an object, click outside the object's boundary or Shift+click the object.

1. Only active objects can have methods, but all objects have properties. If an object cannot have a method, the Events tab in the Object Properties window is dimmed and the Property List contains no Object Method line.

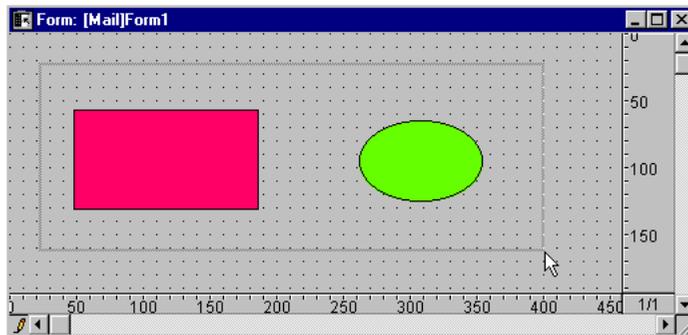
Selecting Multiple Objects

You may want to perform the same operation on more than one form object — for example, to move the objects, align them, or change their appearance. 4th Dimension lets you select several objects at the same time. There are four ways to select multiple objects:

- Choose **Select All** from the **Edit** menu to select all the objects.
- Hold down the **Shift** key and click the objects you want to select.
- Start at a location outside the group of objects you want to select and drag a marquee (sometimes called a selection rectangle) around the objects.
- Hold down the **Alt** key (on Windows) or the **Option** key (on Macintosh) and draw a marquee. Any object that is completely enclosed by the marquee is selected.

You draw a marquee with the arrow pointer. A marquee defines a rectangular region that select objects it surrounds or touches.

To select objects by drawing a marquee around them, you must press the mouse button down and start dragging *in an area that contains no objects*. When you release the mouse button, if any part of an object lies within the boundaries of the selection rectangle, that object is selected. The figure below shows a marquee being drawn to select two objects.



To deselect an object that is part of a set of selected objects, hold down the **Shift** key and click the object. The other objects remain selected. To deselect all the selected objects, click outside the boundaries of all the objects.

Moving Objects

You can move any graphic or active object in the form including fields and objects created with a template.

When moving an object, you have the following options:

- Move the object by dragging it.
- Move the object one pixel at a time using the arrow keys.
- Move the object in the user-defined increments established in the Grid Definition dialog box.
- Use the Coordinates page of the Object Properties window (described in the section [“Resizing Objects” on page 270](#)).

► To move an object by dragging:

- 1 Select the object or multiple objects that you want to move.
- 2 Move the pointer over the selected object or one of the objects in a selected set of objects and drag to the new location.

4th Dimension displays markers that show the location of the object’s boundaries in the rulers so that you can place the object exactly where you want it. As you begin dragging the selected object, its handles disappear.

Be careful not to drag a handle. Dragging a handle resizes the object.

- 3 Release the mouse button to complete the move.

► To move an object one pixel at a time:

- 1 Select the object or objects you want to move.
- 2 Use the arrow keys on the keyboard to move the object.

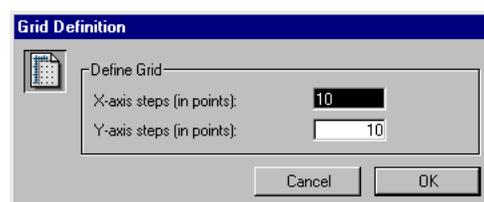
Each time you press an arrow key, the object moves one pixel in the direction of the arrow.

You can also use the arrow keys to move an object x pixels at a time, when x is a value set in the Define Grid dialog box.

► To move an object in user-defined increments:

- 1 Choose Grid Definition from the Form menu.

The Grid Definition dialog box appears.



- 2 Enter the desired values in the X-axis and/or Y-axis entry areas and click OK.
- 3 Choose Turn Grid On from the Form menu or from the Form editor contextual menu.
- 4 Hold down the Shift key and use the arrow keys to move the object in the increments you specified.

Resizing Objects

You can change the size of any object that appears on the form. 4th Dimension lets you stretch or shrink objects on the form.

When resizing objects, you have the following four options:

- Resize an object by dragging a resizing handle.
- Resize an object one pixel at a time by using the Ctrl key (on Windows) or the Command key (on Macintosh) and the arrow keys.
- Resize an object in user-defined increments using settings established in the Grid Definition dialog box.

► To resize an object by dragging:

- 1 Select the object you want to resize.
- 2 Move the pointer over one of the four handles that appear on the selected object.

The pointer changes into a multi-directional arrow  and the handles disappear.

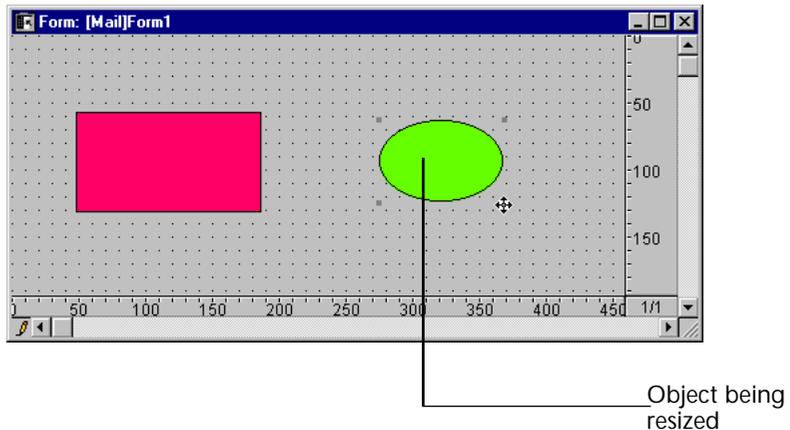
- 3 Drag the handle toward the center of the object to shrink it.
OR

Drag the handle away from the object's center to enlarge it.

4th Dimension resizes the object.

As you drag the handle, the corner of the object opposite the dragging handle remains stationary.

The following illustration shows an object being resized.



Note If you press **Shift** and then drag the handle, the movement is constrained. Lines can then be only vertical, 45°, or horizontal, rectangles can be only square, and ovals can be only circular.

► To resize an object one pixel at a time:

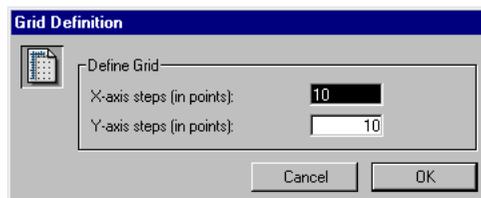
- 1 Select the object you want to resize.
- 2 Hold down the **Ctrl** key (on Windows) or **Command** key (on Macintosh) and use the arrow keys to resize the object.

Pressing the up or down arrow keys resizes the object's height while pressing the left or right arrow keys resizes the object's width.

► To resize an object in user-defined increments:

- 1 Choose **Grid Definition** from the **Form** menu.

The **Grid Definition** dialog box appears.

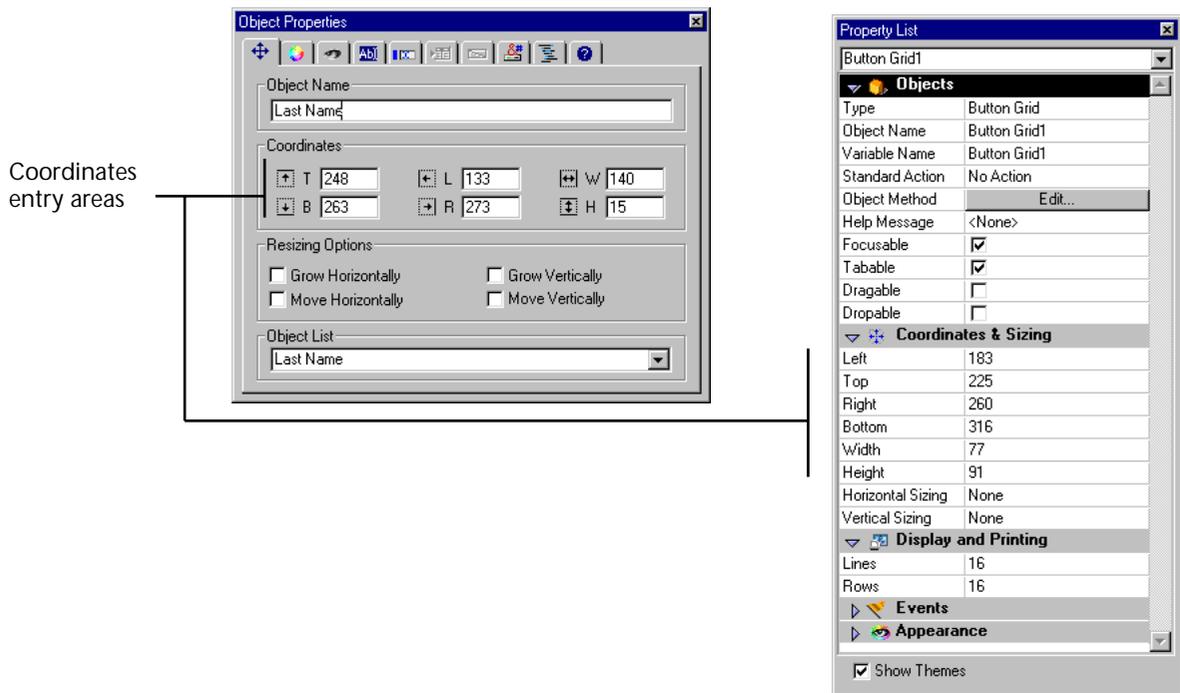


- 2 Enter the desired increments in the X-axis and/or Y-axis entry areas and click **OK**.
- 3 Choose **Turn Grid On** from the **Form** menu or from the **Form editor** contextual menu.

- 4 Select the objects you want to resize.
 - 5 Hold down the Shift+Ctrl keys (Shift+Command key on Macintosh) and use the mouse or the arrow key to resize the objects in user-defined increments.
- To resize an object by entering coordinates:
- 1 Display the Object Properties window or the Property List and click the object you want to resize.
OR
Double-click the object.
 - 2 In the Object Properties window, click the Coordinates tab.
OR
In the Property List, expand the Coordinates and Sizing theme.
The coordinates of the object appear. They are displayed in pixels, centimeters, or inches (depending on the ruler units). The upper-left coordinates of the form area are 0,0.

Form Properties Window

Property List



- 3 Enter new values in the coordinate entry areas.

4th Dimension moves the boundaries of the object to the positions you entered. Depending on the values you use, the object may be resized or moved (or both).

Automatic Resizing and Repositioning

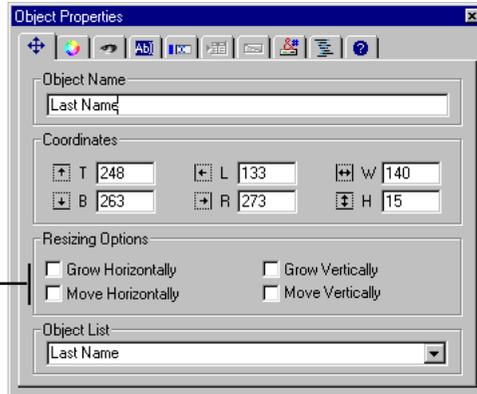
When the user resizes the window in which form is displayed, the objects that are included in that window can be resized or moved.

Automatic resizing works when a user resizes a window that displays a form. Automatic resizing causes an object to grow as the form is enlarged (or become smaller as the enlarged window is reduced). For example, if you use a rectangle that encloses the fields on an entry form, automatic resizing causes the rectangle to grow to the edges of the window as the user enlarges the window.

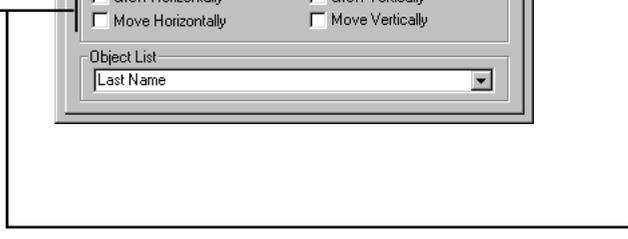
You can also enable automatic repositioning. Automatic repositioning moves an object either horizontally or vertically as the form is resized. When automatic repositioning is on, 4th Dimension tries to keep the object in view as the user reduces the size of the window. For example, if the user resizes a row of buttons so that some of the buttons become obscured, automatic repositioning tries to move the buttons either horizontally or vertically, so that they remain in view.

You enable automatic resizing or repositioning in the Coordinates page of the Object Properties window or in the Coordinates and Sizing theme of the Property List.

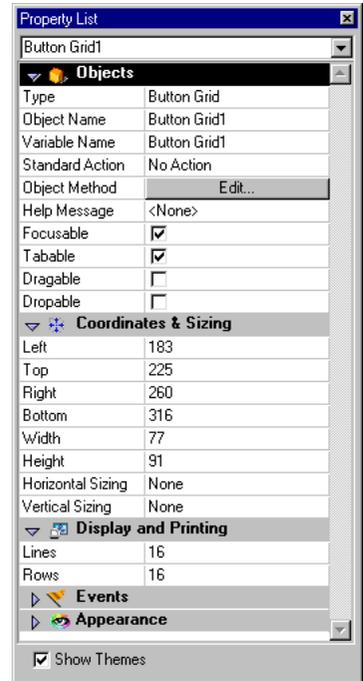
Object Properties Window



Resizing and repositioning options



Property List



- In the Object Properties window, the resizing and positioning options are defined using four check boxes.
- In the Property List, there are two lines, Horizontal sizing and Vertical sizing, for which you can assign three properties (None, Grow, and Move).

Here is the effect of these options:

Object Properties Window	Property List Option	Result
Grow horizontally ¹	Horizontal Sizing: Grow	When the user resizes the width of the window, 4 th Dimension applies the same percentage to the object's width.
Move horizontally	Horizontal Sizing: Move	When the user resizes the width of the window, 4 th Dimension moves the object left or right the same amount as the width increase.
Grow vertically	Vertical Sizing: Grow	When the user resizes the height of the window, 4 th Dimension applies the same percentage to the object's height.
Move vertically	Vertical Sizing: Move	When the user resizes the height of the window, 4 th Dimension moves the object up or down the same amount as the height change.

1. The options Grow horizontally and Move e horizontally are mutually exclusive. The options Grow vertically and Move vertically are mutually exclusive.

The repositioning options enable the object to move in the specified direction to try to remain visible.

Using the Rulers

The Form editor rulers extend along the height and width of the form. A dotted grid appears in the Form editor whenever you have rulers showing. This feature helps you position objects precisely.

You can hide the rulers to increase your working area in the Form editor window. You can display the rulers again when necessary. To hide or display rulers, choose Rulers from the Display submenu in the Form menu or the Form editor contextual menu.

The rulers contain markers that show the position of the pointer when creating or resizing an object. While you are moving the object, the markers change to show the top and bottom and left and right sides of the object.

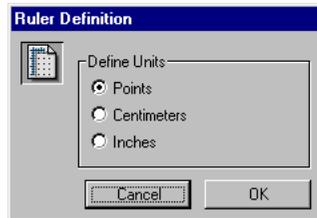
The object markers allow you to align other objects to the same position on the rulers.

You can change the units the rulers use to suit your preference.

► To define ruler units:

1 Choose Define Ruler Units from the Form menu.

4th Dimension displays the Ruler Definition dialog box, shown below.



2 Click the measurement scale you want to use.

- Click Points to display rulers that provide measurement in points. One point is equal to the width of one pixel. There are 72 points in an inch.
- Click Centimeters to display metric scale rulers.
- Click Inches to display rulers that use feet and inches.

3 Click OK.

4th Dimension changes the measurement units to the scale you have selected. The objects' coordinates will also use the same units. For information about the Object Coordinates dialog box, see the section [“Resizing Objects” on page 270](#).

Creating Objects

You can use the Tools palette to add objects to a form. The object creation tools are located in the upper part of the Tools palette:



The object type is selected by clicking the appropriate tool in the Tools palette. Some buttons display a small triangle located in the lower right corner of the tool icon. This indicates that the corresponding object type includes several variations within that type. To display the variations for a type, click the tool's icon and hold down the mouse button until variations are displayed in a menu. For a faster display,

you can also right-click the tool's icon (on Windows) or press the Ctrl key while clicking the tool's icon (on Mac OS). If you select a type from the menu, it becomes the current type and the tool's icon is updated to reflect that change.

For more information on the Tools palette and its tools, refer to [“The Tools Palette” on page 235](#).

You can create any type of object (graphic or active) by drawing it or by dragging the tool from the Tools palette to the form.

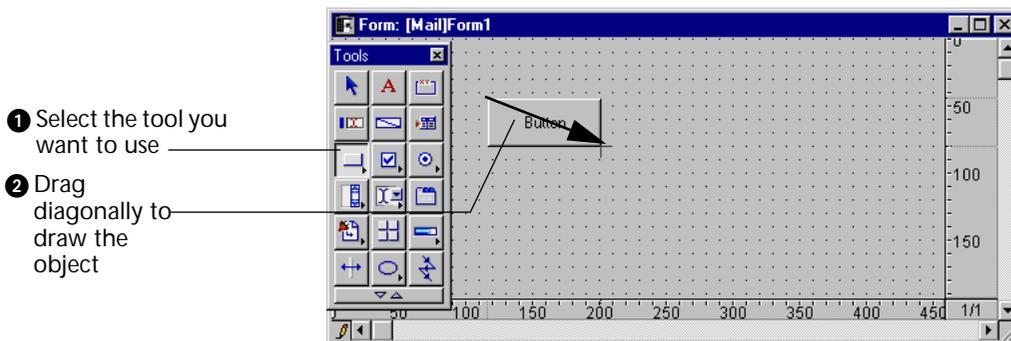
► To create an object by drawing:

- 1 Select the type or subtype of object you want to draw by clicking its tool in the palette.

The pointer becomes a crosshair when it is in the over the area in which you can draw the object. The pointer becomes an arrow when it is positioned over a menu, ruler, or the palette so that you can use it to select.

- 2 Drag to create an area for the object.

For two-dimensional objects (ovals, rectangles, fields, grids, text areas, active objects, and subform areas), drag diagonally.



Text areas include a text sample.

Note Hold down the **Shift** key as you draw to constrain the object to a regular shape. Lines are constrained to horizontal, 45°, or vertical, rectangles are constrained to squares, and ovals are constrained to circles.

- 3 When you have finished drawing the object, release the mouse button.

4th Dimension creates the object, assigns it a name and makes it the currently selected object. The Arrow tool is automatically selected and the pointer becomes an arrow, unless the last tool you have used is the Text tool.

If you created a field or active object, 4th Dimension also displays the Object Properties window or the Property List. You can use it to specify additional information.

- ▶ To create an object by drag and drop:
 - 1 If the desired object does not have a drop-down list associated with it, select the object by clicking its tool in the palette and drag it to the location on the form where you want the object to be.
If the desired object is in a drop-down list, first select the object from the drop-down list and then drag the object from the palette to the form.

A dotted line indicates the location of the cursor.

- 2 When you reach the location where you want the object to be, release the mouse button.

4th Dimension creates the object, assigns it a name, and makes it the currently selected object. The Arrow tool is automatically selected and the pointer becomes an arrow, unless the last tool you have used is the Text tool.

If you created a field or active object, 4th Dimension also displays the Object Properties window or the Property List. You can use it to specify additional information.

Deleting an object

To delete an object, select that object and press the Backspace key (Delete key on Macintosh).

Grouping Objects

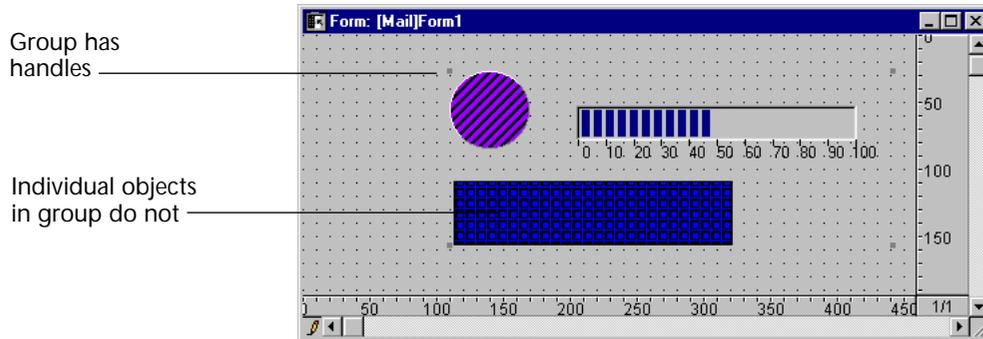
4th Dimension lets you group objects so that you can select, move, and modify the group as a single object.

Objects that are grouped retain their position in relation to each other. You would typically group a field and its label, an invisible button and its icon, and so forth.

When you resize a group, all the objects in the group are resized proportionally (except text areas, which are resized in steps according to their font sizes).

Grouping is also used for grouped scrollable areas. For more information, refer to the *4th Dimension Language Reference*.

Groups can be part of other groups. The following illustration shows grouped objects.



You can ungroup a group of objects to treat them as individual objects again.

An active object that has been grouped must be ungrouped before you can access its properties or method.

Grouping affects objects only in the Form editor. In the User environment, all grouped objects except for scrollable areas act as if they were ungrouped.

► To group objects:

- 1 Select the objects that you want to group.
- 2 Choose Group from the Object menu.

OR

Press Ctrl+G key (on Windows) or Command-G (on Macintosh).

4th Dimension marks the boundary of the newly grouped objects with handles. No handles mark the boundary of any of the individual objects within the group.

Now, when you modify the grouped object, you change all the objects that make up the group.

► To ungroup an object:

- 1 Select the grouped object that you want to ungroup.

2 Choose Ungroup from the Object menu.

OR

Press Ctr+H (on Windows) or Command-H (on Macintosh).

If Ungroup is dimmed, this means that the object is already separated into its simplest form.

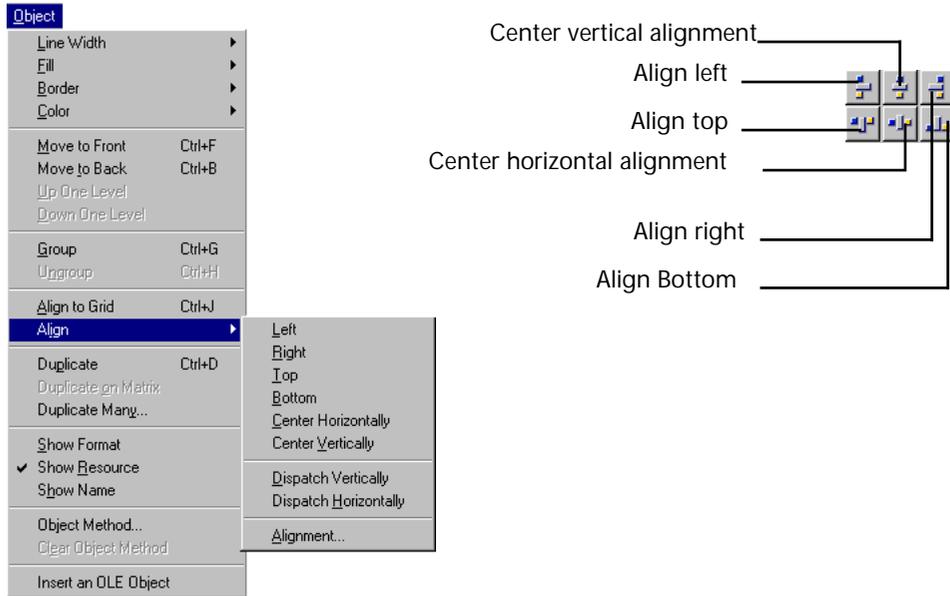
4th Dimension marks the boundaries of the individual objects with handles.

Aligning Objects

The Form editor's alignment tools and grid tools let you align objects to each other or to an invisible grid on the form.

- When you align one object to another, you can align it to the top, bottom, side, or horizontal or vertical center of the other object. You can directly align a selection of objects using the alignment tools or apply more advanced alignment settings using the Alignment Assistant. The latter option allows you, for example, to define the object will be used as the position reference and to preview the alignment in the form before applying it.
- When you use the invisible grid, you can align an object or collection of objects to a grid on the form.

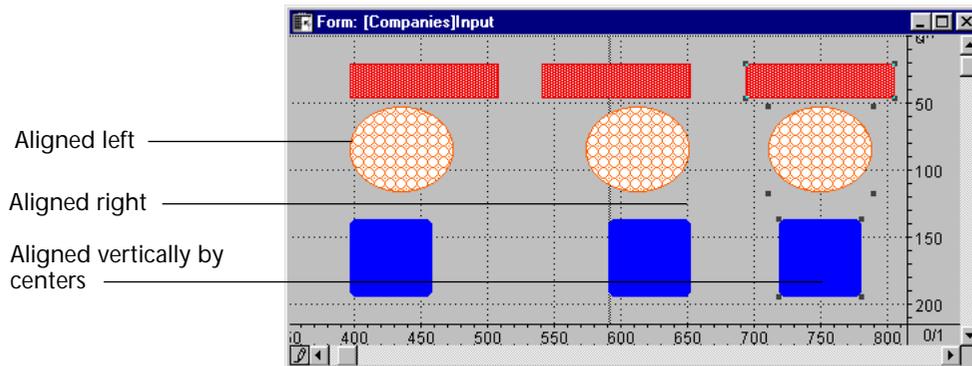
Using the Instantaneous Alignment Tools The alignment tools in the Tools palette and in the Align submenu of the Object menu allow you to quickly align selected objects.



The alignment icons in the Tools palette illustrate each alignment type.

When 4th Dimension aligns objects, it leaves one selected object in place and aligns the remaining objects to that one. This object is the “anchor.” It uses the object that is the furthest in the alignment’s direction as the anchor and aligns the other objects to that object. For instance, if you want to perform a right alignment on a set of objects, the rightmost object will be used as the anchor.

The figure below shows aligned objects.



- ▶ To align a set of objects:
 - 1 Select the objects you want to align.
For more information on how to select objects, refer to [“Selecting Multiple Objects” on page 268](#).
 - 2 Click the alignment tool in the Tools palette that corresponds to the alignment you want.
OR
Choose an alignment menu command from the Align submenu in the Object menu.
OR
Choose an alignment menu command from the Align submenu in the contextual menu that is displayed when you right-click one object in the selection (on Windows) or when you press the Control key while clicking an object in the selection.

4th Dimension aligns the selected objects according to the alignment you selected.

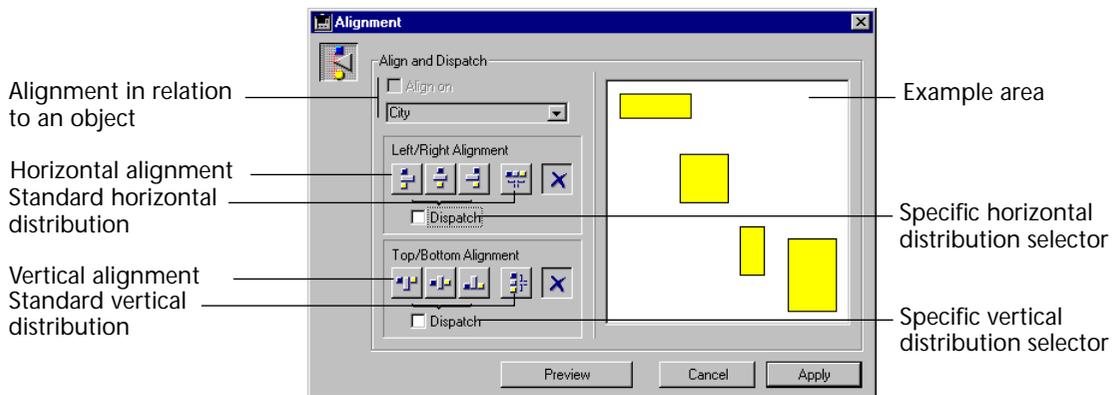
Using the Alignment Assistant

The Alignment Assistant allows you to perform any type of alignment and/or distribution of objects.

- ▶ To align objects using the Alignment Assistant:
 - 1 Select the objects you want to align.
For more information on how to select objects, refer to [“Selecting Multiple Objects” on page 268](#).
 - 2 Choose the Alignment command from the Align submenu in the Object menu.
OR

Choose the Alignment command from the Align submenu in the Form editor contextual submenu.

The Alignment dialog box is displayed:



- 3 In the Left/Right alignment and/or Top/Bottom alignment areas, click the Alignment icon that corresponds to the alignment you want to perform.

The example area displays the results of your selection.

- 4 To perform an alignment that uses the standard anchor scheme, click Preview or Apply.

In this case 4th Dimension will perform an alignment using the standard anchor as described in [“Using the Instantaneous Alignment Tools” on page 281](#).

OR

To align objects to a specific object, select the **Align on** option and select the object to which you want the other objects to be aligned from the object list.

In this case, the position of the reference object will not be altered.

You can preview the results of the alignment by clicking the Preview button. The objects are then aligned in the Form editor but since the Alignment dialog does not go away, you can still cancel or apply the alignment.

Note The Alignment Assistant allows you to align and distribute objects in one operation. For more information on how to distribute objects, refer to [“Distributing Objects” on page 286](#).

Using the Alignment Grid

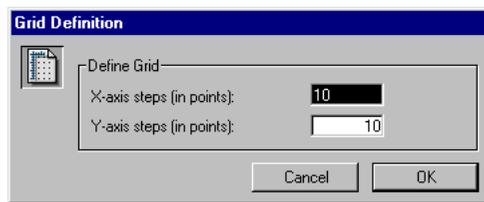
The Form editor provides a grid of horizontal and vertical lines that help you place and align objects in a form. You can do the following:

- Specify the dimensions of the grid,
- Hide or show the grid,
- Turn the grid on (in this case, objects are automatically aligned to that grid when they are created or moved),
- Align a selection of objects to the grid, even if it is disabled.

► To define the grid:

1 Choose Define Grid from the Form menu.

4th Dimension displays the Grid Definition dialog box, shown below.



2 Enter a number of points (72 points to an inch) for each step on the x-axis and y-axis.

Grids are always defined in points. In effect, you are defining rectangles that are *x* points wide and *y* points high. For example, you might specify 10 points for x-axis steps and 20 points for y-axis steps.

If you want the x-axis and y-axis steps to be equal, enter identical point values in the boxes.

3 Click the OK button to accept the grid settings.

OR

Click the Cancel button to cancel the settings.

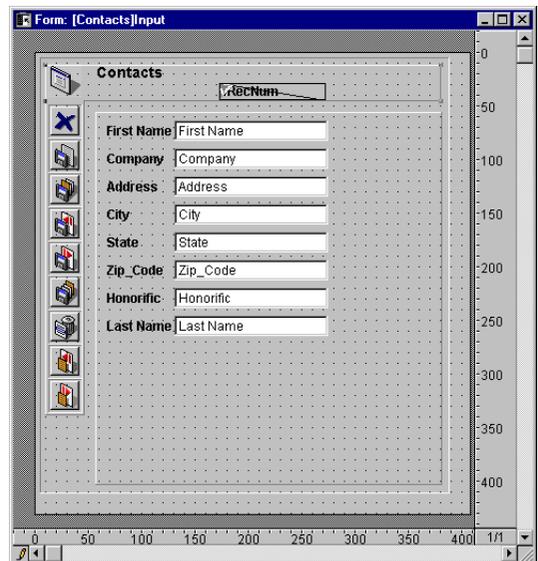
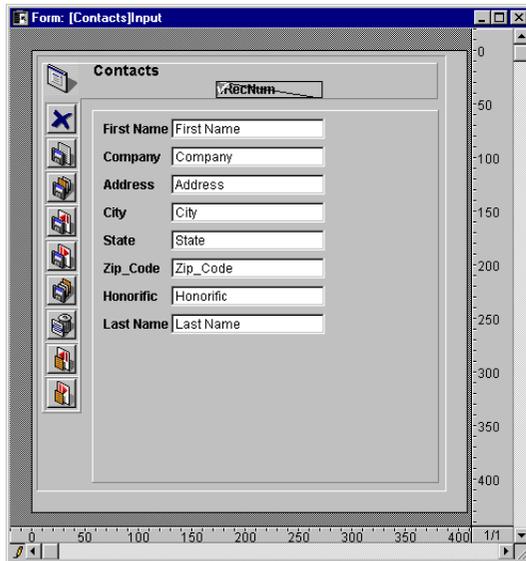
► To Hide/Display the grid:

1 Select Grid From the Display submenu in the Form menu or in the Form editor contextual menu.

For more information about this point, refer to [“Showing/Hiding Elements in the Form Editor”](#) on page 232.

Grid not displayed

Grid displayed



Note The display status of the grid is independent from its activation status. It can be displayed but inactive or vice-versa.

► To turn the grid on

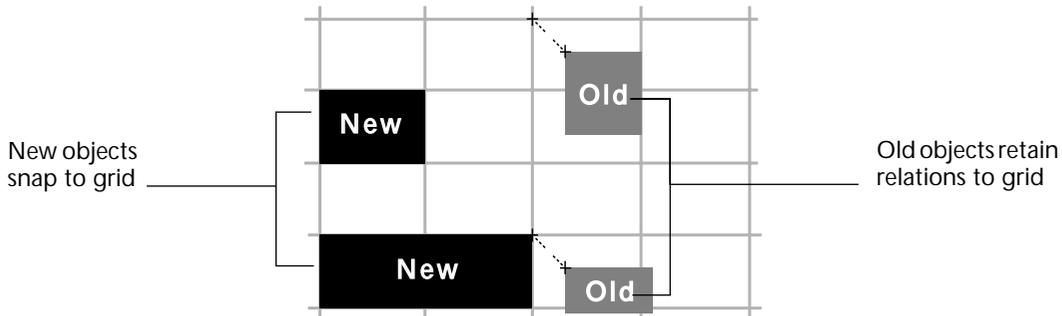
1 Choose Turn Grid On from the Form menu.

When the grid is on, there is a check mark next to the Turn Grid On menu item. If you subsequently move the objects with the mouse, they snap to align with intersections of the invisible grid.

If you add an object to the form while the grid is on, it aligns to the invisible grid as you draw it. Each corner of the object's defining rectangle jumps to the nearest intersection of the grid.

Existing form objects are not aligned to the grid when it is turned on. Instead, 4th Dimension remembers each object's relation to the nearest intersection. If you then move or resize the object, 4th Dimension aligns the object to the nearest intersection of the grid using the original relation. To align these objects to the new grid, you use the Align to Grid menu item.

The figure below depicts the invisible grid used to align objects.



- ▶ To align existing objects to the grid:
 - 1 With the grid turned on, select the object or objects you want to align to the grid.
 - 2 Choose the Align to Grid from the Object menu.

OR

Click the Align to Grid icon  .

4th Dimension aligns the upper-left corner of the object or of each object, to the nearest point on the invisible grid.

Distributing Objects You can distribute objects so that they are aligned and have an equal amount of space between them.

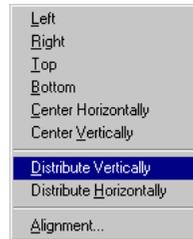
You can distribute objects using either the Distribute tools in the Tools palette or the Alignment Assistant. The latter allows you to align and distribute objects in one operation.

- ▶ To distribute objects with equal spacing:
 - 1 Select three or more objects and click the desired Distribute tool. For more information on how to select objects, refer to [“Selecting Objects” on page 265](#).
 - 2 Click the distribution tool that matches your needs.

Distribute horizontally   Distribute vertically

- OR
- Select a distribution menu command from the Align submenu in the Object menu.
- OR

Select a distribution menu command from the Align submenu in the contextual menu that is displayed when you right-click one object of the selection (on Windows) or when you press the Control key while clicking an object of the selection.



Distribute horizontally and distribute vertically menu commands

4th Dimension distributes the objects accordingly. Objects are distributed using the distance to their centers and the largest distance between two consecutive objects is used as a reference.

► To distribute objects using the Alignment dialog box:

1 Select the objects you want to distribute.

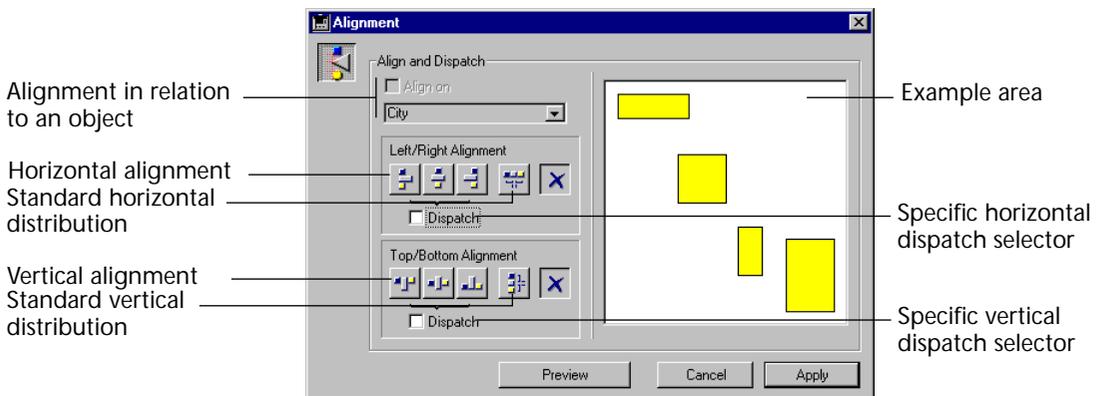
For more information on how to select objects, refer to [“Selecting Multiple Objects”](#) on page 268.

2 Choose the Alignment command from the Align submenu in the Object menu.

OR

Choose the Alignment command from the Align submenu in the form editor contextual submenu.

The Alignment Assistant appears:



3 In the Left/Right alignment and/or Top/Bottom alignment areas, click the standard distribution icon.

The example area displays the results of your selection.

- 4 To perform a distribution that uses the standard scheme, click Preview or Apply.

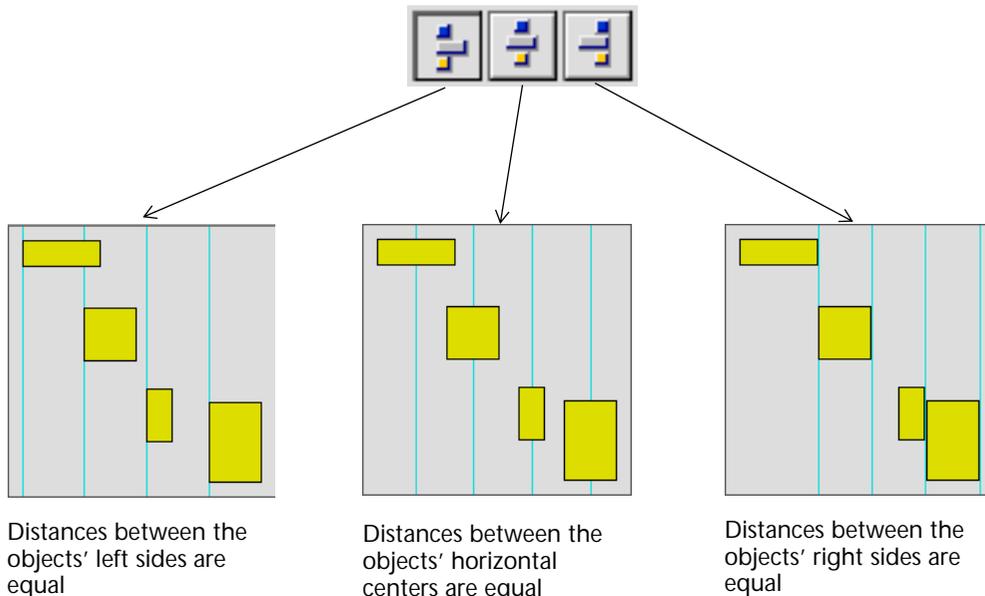
In this case 4th Dimension will perform a standard distribution, as described earlier in this section.

OR

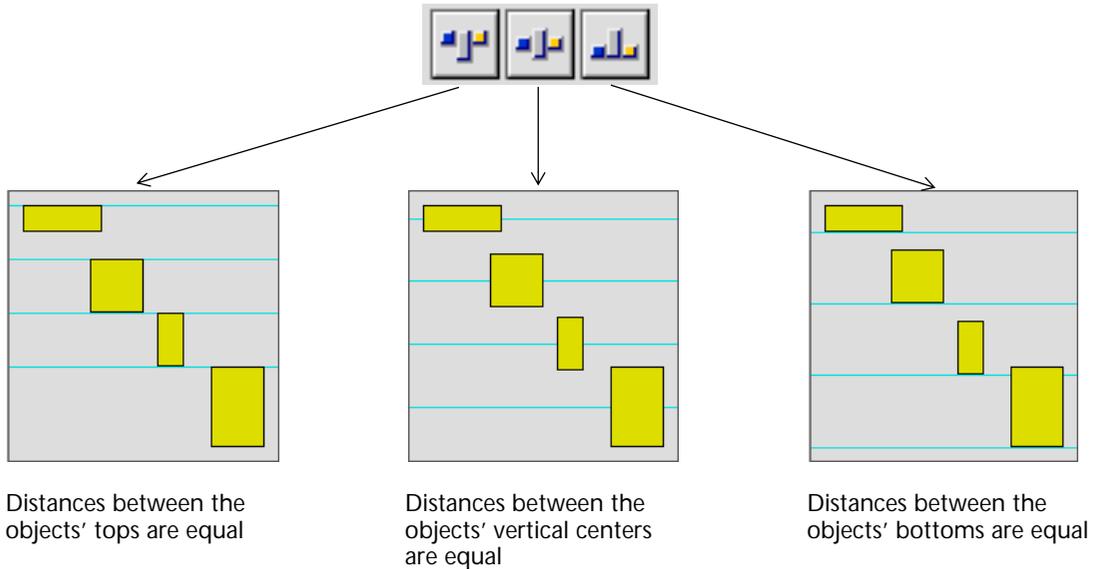
To execute a specific distribution, select the Dispatch option (for example if you want to distribute the objects based on the distance to their right side).

This option acts like a switch. If the Dispatch check box is selected, the icons located above it perform a different function. They now allow you to specify whether the selected objects will be distributed evenly with respect to their left/top sides, their centers, or their right/bottom sides. The effects of each icon are shown in the Preview area and are illustrated below.

- **Horizontal Distribution** For horizontal distribution (i.e., from left to right): the icons have the following meaning.



- Vertical Distribution** For vertical distribution (i.e, from top to bottom), the icons have the following meanings.



The Alignment Assistant allows you to align and distribute objects in one operation. For example, suppose you have a group of field labels that are arranged vertically. You would like to left align the labels and distribute them evenly in the vertical dimension. The following specification does this:

The screenshot shows the 'Alignment' dialog box with the following settings:

- Align on:** Text22
- Left/Right Alignment:** Left alignment icon is selected.
- Top/Bottom Alignment:** Vertical distribution from centers icon is selected.
- Dispatch:** Checked for both alignment sections.

The preview area on the right shows four yellow rectangles. The first three are left-aligned, and the fourth is shorter. They are vertically distributed such that their centers are aligned to the same horizontal level. Labels with lines point to the 'Left alignment' and 'Vertical distribution from centers' options, and another label points to the preview area.

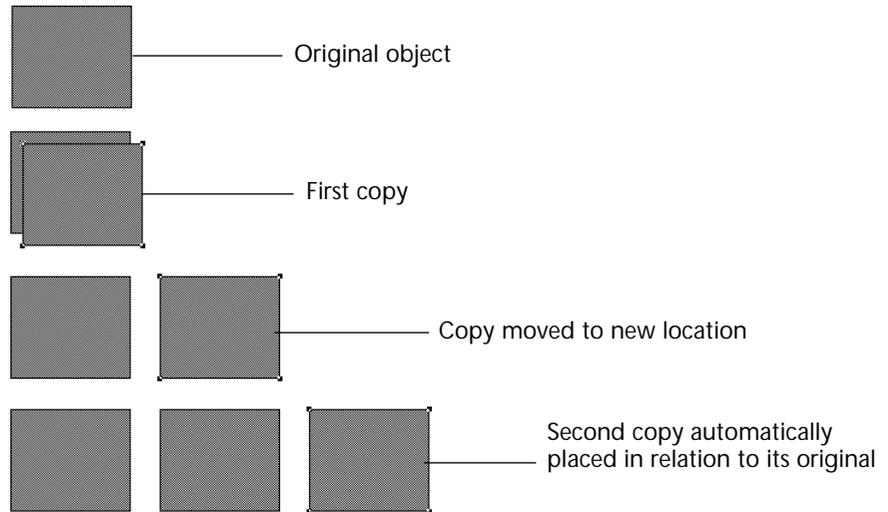
Duplicating Objects You can duplicate any object in the form, including active objects. Copies of active objects retain all the properties of the original, including name, type, automatic action, display format, and object method.

You can duplicate an object directly using the Duplicate tool in the Tools palette or use the Duplicate Many dialog box to duplicate an object more than once. Also, using that dialog, you can set the distance between two copies.

When duplicating a variable, you can use specific duplication features that allow you to include an automatic number in the copies' names. For more information on this point, refer to [“Duplicating on a Matrix” on page 419](#).

- ▶ To duplicate an object:
 - 1 Select the object or objects that you want to duplicate.
 - 2 Choose Duplicate from the Object menu.
OR
Click the Duplicate icon  in the Tools palette.
OR
Press the keys Ctrl+D (on Windows) or Command+D (on Mac OS).
4th Dimension creates a copy of each selected object and places the copy in front and slightly to the side of the original.
 - 3 Move the copy (or copies) to the desired location.
If you choose the Duplicate menu item again, 4th Dimension creates another copy of each object and moves it the exact same distance and direction from the first copy.
If you need to distribute copies of the object along a line, you should use the following procedure. Duplicate the original object, move the copy to another location in the form, and then duplicate the copy. The second copy is automatically placed in the same relation to the first copy as the first copy was in relation to the original object. Subsequent copies are also placed in the same relation to their originals.

The figure below shows how this relative placement of copies works.



► To duplicate multiple objects using the Duplicate Many dialog:

- 1 Select the object(s) to duplicate.
- 2 Choose Duplicate Many... from the Object menu.

The Duplicate Many dialog box appears:

- 3 In the upper area, enter the number of columns and lines (rows) of objects you want to get.
 - For example, if you want three columns and two lines of objects, enter 3 in the Column(s) area and 2 in the Line(s) area.
 - If you want three horizontal new copies of an object, enter 4 in the Column(s) area and leave the default value, 1, in the Line(s) area.
- 4 For lines and columns, define the offset that you wish to leave between each copy.

The value must be expressed in points. It will be applied to r each copy, or copies, in relation to the original object.

For example, if you want to leave a vertical offset of 20 points between each object and the height of the source object is 50 points, enter 70 in the column's "Offset" area.

- 5 If you wish to create a matrix of variables, select the "Number Variables" option (otherwise go to step 7).

This option is active only if the selected object is a variable. For more information on that option, refer to "[Duplicating on a Matrix](#)" on [page 419](#).

- 6 Select the direction in which the variables are to be numbered, either by line(s) or by column(s).

- 7 Click the OK button.

The number of columns and lines of the specified object(s) is created.

Copying Objects on a Form

You can copy all, some, or one of the objects on a form using the Copy menu item in the Edit menu. You can use the copied objects in another form in the database or in another database.

Like objects duplicated using the Duplicate command, each object copied using the Copy command is a complete copy of the original object and retains all the properties of the original. Copies of active objects such as fields and buttons retain all the properties of the original including name, type, action, display format, and the method associated with the object.

Objects copied using the Copy command are copied to the Clipboard while objects duplicated with the Duplicate command are reproduced on the current form page.

You can copy objects and save them in the Scrapbook for later use. For example, suppose you create several custom buttons. You can use the same buttons with their corresponding actions in any form in any database simply by copying them to the Scrapbook and then pasting them into another form.

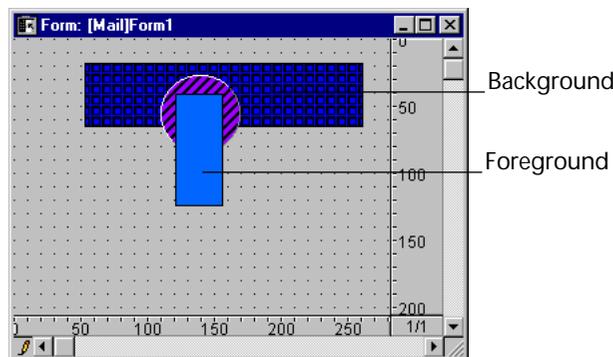
Note A method that is copied with an object and pasted in a different form may lose its meaning unless it is updated. For example, if the method refers to a specific field and you paste the object in a database that does not have that field, the method loses its meaning.

- ▶ To copy all the objects in a form and paste them into a new form:
 - 1 Choose Select All from the Edit menu.
4th Dimension selects every object on the current form page.
 - 2 Choose Copy from the Edit menu.
4th Dimension places a copy of the form on the Clipboard.
 - 3 Open a blank page or create a new form using the Form Wizard.
For information on creating a form, see [Chapter 3](#).
 - 4 Choose Paste from the Edit menu.
The new form contains copies of all objects in the previous form.

Note If the form is to be used as an output form, you may need to adjust the output control lines. The output control lines are not associated with objects, so they must be repositioned on the new form. For information about working with output control lines, see [“Moving Output Control Lines” on page 445](#).

Layering Objects

You will sometimes have to rearrange objects that are obstructing your view of other objects in the form. For example, you may have a graphic that you want to appear behind the fields in a form. 4th Dimension provides two menu items, Move to Back and Move to Front, that let you “layer” objects on the form. The figure below shows objects in front of and behind other objects.



- ▶ To move an object to the back:
 - 1 Select the object or objects that you want to move to the back.
 - 2 Click the Move to Back tool  in the Tools palette.
OR
Choose Move to Back from the Object menu.

OR

Press Ctrl+B (on Windows) or Command-B (on Macintosh).

OR

Choose Move to Back from the Level submenu in the contextual menu. This menu is displayed by clicking the object using the right button of the mouse (on Windows) or by clicking the object while pressing the Control key (on Mac OS).

4th Dimension moves the selected object or objects behind all the other form objects.

Note When you move an object to the back, it may be hidden by objects in front of it. To see the object, select the object in front and send it to the back.

► To move an object one level to the back:

1 Select the object or objects that you want to move to the back.

2 Choose Down One Level from the Object menu.

OR

Choose Move Down One Level from the Level submenu in the contextual menu. This menu is displayed by clicking the object using the right button of the mouse (on Windows) or by clicking the object while pressing the Control key (on Mac OS).

4th Dimension moves the selected objects down one level.

► To move an object to the front:

1 Select the object or objects that you want to move to the front.

2 Click the Move to Front tool  in the Tools palette.

OR

Choose Move to Front from the Object menu.

OR

Press Ctrl+F (on Windows) or Command-F (on Macintosh).

OR

Choose Move to Front from the Level submenu in the contextual menu. This menu is displayed by clicking the object using the right button of the mouse (on Windows) or by clicking the object while pressing the Control key (on Mac OS).

4th Dimension moves the selected object or objects in front of all the other objects.

- ▶ To move an object one level to the front:
 - 1 Select the object or objects that you want to move to the front.
 - 2 Choose **Up One Level** from the **Object** menu.

OR

Choose **Move Up One Level** from the **Level** submenu in the contextual menu. This menu is displayed by clicking the object using the right button of the mouse (on Windows) or by clicking the object while pressing the **Control** key (on Mac OS).

4th Dimension moves the selected objects up one level.

Deleting Objects

You can delete any object on a form. If you want, you can place a copy of the deleted object on the Clipboard. Objects placed on the Clipboard can later be pasted to new locations in the form. If the **Object Properties** window is open while you try to delete an object, the deletion will affect the contents of the **Object Properties** window.

- ▶ To delete an object:
 - 1 Select the object or objects you want to delete.
 - 2 Choose **Clear** from the **Edit** menu.

OR

Press the **Backspace** key (on Windows) or the **Delete** key (on Macintosh) on your keyboard.

4th Dimension deletes the selected object or objects.

To cut an object to the Clipboard, choose **Cut** from the **Edit** menu. 4th Dimension removes the selected object or objects and places a copy on the Clipboard. The **Cut** command works even if the **Object Properties** window is open.

If you change your mind, choose **Undo** from the **Edit** menu or from the editor's contextual menu before performing another action. 4th Dimension restores the deleted object or objects.

Optimizing the Appearance of Text and Picture Objects

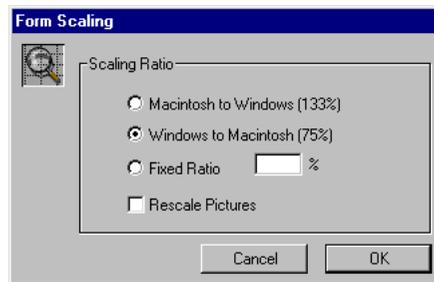
You can resize static text areas and pictures for optimal display. A static text object will be resized to fit the text. A picture object will be adjusted to display the picture without horizontal or vertical distortion. **Ctrl+click** (on Windows) or **Command-click** (on Macintosh) the bottom right corner of the object to resize the object.

Scaling a Form

The Form editor includes a feature for rescaling form objects so that they look good when a database is transported to another platform.

Form objects created on Macintosh will look smaller when viewed on Windows, and vice versa — even though the objects are actually the same size. This is because the Windows screen resolution is about 25% greater than the Macintosh resolution. For instance, 12-point text on a Macintosh will appear as 9-point text on Windows. If the font size is just large enough on Macintosh, it may be too small on Windows. Conversely, if a font size on Windows is adequate, it may be too large on Macintosh.

To compensate for screen resolution differences, you need to rescale objects. With the Scale item on the Form menu you can proportionally resize all the form objects in one operation. When you choose Scale, the Form Scaling dialog box appears.



You can choose among the following options:

- **Macintosh to Windows platform (133%)** This option is the default option when you use 4th Dimension on Macintosh. Use this option when you want to resize a form that was created according to the Macintosh screen resolution so it will look like it was created according to the Windows screen resolution. To do so, the program increases the size of all the form objects by one-third. For instance, 9-point text will become 12-point text.
- **Windows to Macintosh platform (75%)** This option is the default option when you use 4th Dimension on Windows. Use this option when you want to resize a form that was created according to the Windows screen resolution so it will look like it was created according to the Macintosh screen resolution. To do so, the program decreases the size of all the form objects by one-quarter. For instance, 12-point text will become 9-point text.

- **Fixed Ratio scaling** This option lets you resize a form using the percentage you type in the “%” enterable area. With this option you can resize a form so it will look good on any unusual screen resolution you may encounter on either the Macintosh or Windows platform. You can also use this option to change the size of all the form’s objects for the platform you are using. For example, if you want to double the size of all objects, enter 200%; if you want to halve the size, enter 50%.
- **Rescale pictures** This option is not selected by default. Usually, decreasing or increasing the size of bitmapped pictures does not provide good results from a cosmetic point of view. For this reason, the program, does not resize any static pictures in a form unless you select this option. Instead it moves them to their new “center relative” positions. If you know that rescaling bitmaps will produce pleasing results or if you use non-bitmapped pictures, you may chose to rescale the pictures.

When you have selected your options, click **OK** to resize the form, otherwise click **Cancel**. If you resize a form by mistake or with the wrong option, choose **Undo** from the **Edit** menu to recover the form as it was before the rescaling.

Changing the Appearance of Objects

You can change the appearance of any object in a form. Each object has its own platform interface and appearance setting.

For any object that uses text (a field, a text area, a button, and so forth), you can change the following attributes:

- Platform interface,
- Appearance,
- Font,
- Style,
- Size,
- Alignment within the object’s area.

For any object that uses lines, fill patterns, or colors, you can change the following attributes:

- Line width,

- Fill pattern,
- Border pattern,
- Foreground and background color.

Platform Interface and Border Line Style

You can set the platform interface and appearance on an object-by-object basis. For platform interface, you have the following choices:

- **Inherited from Form** The platform interface for the object is the same as the platform interface of the form. The platform interface of the form is set in the Form Properties window.
- **Automatic** The platform interface is based on the platform on which the database is currently running.
- **Mac OS** The object will be displayed as a Macintosh object.
- **Windows NT 3.5.1** The object will be displayed as a Windows NT 3.5.1 object.
- **Windows 95** The object will be displayed as a Windows 95 object.
- **Platinum** The object will be displayed as a Macintosh object using the Copland user interface guidelines.

For information on setting the Platform Interface for the form, see [“Setting the Platform Interface” on page 255](#). For information on each platform interface option, see [“User Interface” on page 89](#).

For Appearance, you have the following choices:

- None,
- Plain,
- Dotted,
- Raised,
- Sunken,
- Double.

For illustrations of the effects of these choices on various object types, see the sections, [“Fields and Field Labels” on page 221](#), [“Buttons” on page 382](#) and [“Button Actions” on page 385](#).

Platform interface and appearance are set using the Display page of the Object Properties window.

► To set Platform Interface or Appearance:

1 Double-click the object.

Depending on your display options, the Object Properties window or the Property List for the object appears.

2 In the Object Properties window, click the Display tab.

OR

In the Property List, expand the Appearance theme.

3 Choose the desired platform interface and border line style from the drop-down lists.

Working with Text Areas

You can make the following changes to text areas.

- Establish default settings for font attributes,
- Create text areas and add or edit text,
- Setting the text area's platform interface, appearance, and font attributes.

Creating and Editing Text Areas

Use text areas to provide labels, titles, and descriptions in your form.

Text you create in a text area is different from the text contained in a Text field. A Text field contains data stored in the database. The contents can be different in each record. A text area is a graphic object; it is not active. Text in a text area remains the same whenever the form is displayed.

There are three exceptions to this rule:

- You can embed field names or variable names in text areas. When the text area is displayed or printed, the values of the fields or variables from the current record are substituted. Use embedded fields and variables to create mail-merge documents and in report headers and footers. For complete details, see the section [“Creating Mail-Merge Documents” on page 456](#).
- You can integrate dynamic table or field references to text areas. When you place dynamic labels in your forms, they are automatically updated throughout the database when you modify a table's or a field's

name using the Structure window or the Table name and Field name commands.

To insert a dynamic table name: <?[TableName]> or <?[2]> (the table's creation order number, meaning the second table created).

To insert a dynamic field name: <?[TableName]FieldName> or <?[2]3> (the table's and field's creation order number), or even <?3> (the field's creation order number) for the current table's field.

Please note that table and field numbers correspond to their creation order. You can therefore add or rename tables and fields without modifying the dynamic reference system. The actual contents of the text area can be displayed using the Show Name and Show Format menu commands from the Object menu.

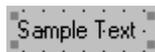
Note You can automatically insert dynamic references using the Options page of the Form Wizard. For more information about this point, refer to [“Form Display Options” on page 216](#).

- You can assign a STR# resource to a Text area. Use the format *:STR# ID, line ID* in place of static text. If, for example, you enter *:20000,10*, the reference will be replaced by the text whose ID is 10 from the resource whose ID is 20000. When you choose **Show Resource** from the **Object** menu, the text in the resource appears in the Form editor.

The Form Wizard automatically creates text areas that contain field labels for the fields and, optionally, a title for the form. You can modify or create these labels as you would modify or create any other text area you add to the form.

As you draw a text area, it snaps to a size that reflects the font size.

After you create a text area, the text box displays a sample text that is selected.



Type text in the text area. When the text you are typing reaches the edge of the text area, 4th Dimension automatically wraps the text to the next line in the area.

Note You can tell 4th Dimension to resize a text area to fit its contents. Ctrl+click (on Windows) or Command-click (on Macintosh) the bottom right corner of the object to resize the object.

If you enter more text than the area can display, the text is not visible until you resize the text area.

► To modify text in a text area:

- 1 Click the Text Area tool .
- 2 Select the text you want to modify or click to insert an insertion point. Use the standard text editing operations to edit the text.

Default Settings for Text Objects

When you create any object, 4th Dimension uses the default settings for these features. You can establish new default settings at any time.

For example, if you establish a new default font, 4th Dimension uses that font for any object you subsequently create that displays text.

You can change the settings for a selected object without changing the default settings. 4th Dimension changes the appearance of the selected object, but the change affects only that object. 4th Dimension continues to use the default settings for objects created in the form.

For example, if you change the font for one text area, the change affects only that area, not subsequently created fields or text areas.

This section gives the basic steps for establishing default settings and for changing the settings for selected objects. It then describes in detail each option for the appearance of objects in a form.

► To establish default settings:

- 1 Make sure that no object in the form is selected.
- 2 Use the items in the Font, Style, and Object menus to choose the font, size, style, text justification, line width, fill pattern, border pattern, or color.

These settings will be used as the default settings for any new objects that use text that you create.

The following sections describe how to change these attributes for selected objects.

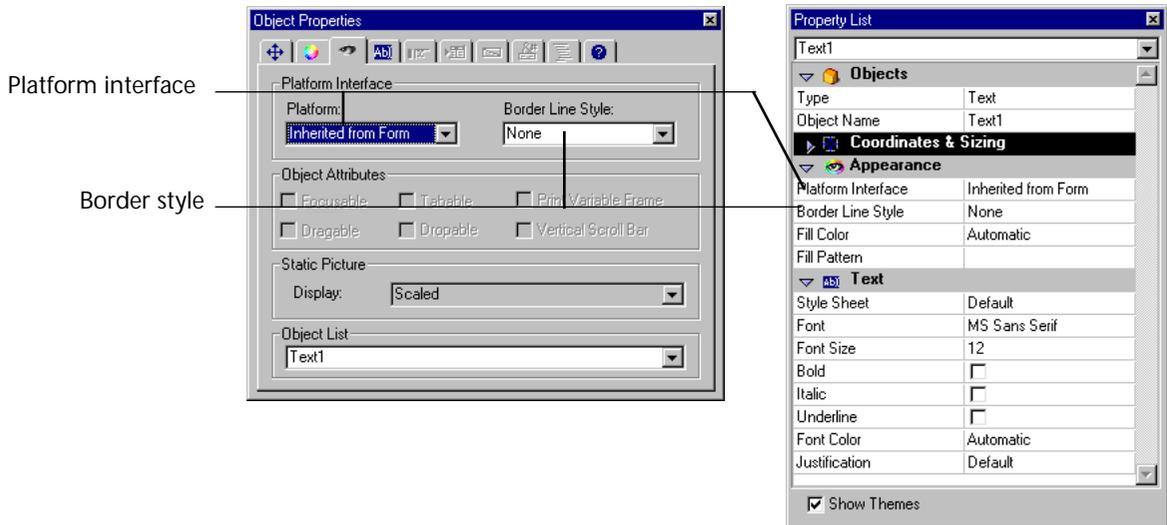
Setting a Text Object's Platform Interface and Appearance

You can set the platform interface, appearance, color, border, and fill pattern of a text object.

- ▶ To set the text object's appearance:
 - 1 Double-click the text object to display its properties in either the Object Properties window or the Property List.
The choice between these two palettes is explained in [“Displaying and Setting Form and Object Properties” on page 244.](#)
 - 2 In the Property List, expand the Appearance theme.
OR
In the Object Properties window, click the Display tab control.
The appearance properties for the object are displayed:

Form Properties Window

Property List



- 3 Use the drop-down lists to set the desired appearance of the text object.

For information on these options, see the section [“Platform Interface and Border Line Style” on page 298.](#)

Note You can also set the border line style using the Form editor contextual menu.

Setting Text Attributes

You can set text attributes for text objects using either the Object Properties window or the Font and Style menus.

- ▶ To set text attributes using the Font and Style menus:
 - 1 Select the object or objects whose text attributes you want to change.
 - 2 Make the appropriate selections from the Font and Style menus.
- ▶ To set text attributes using the Object Properties window or the Property List:
 - 1 Double-click the picture to display its properties in either the Object Properties window or the Property List.

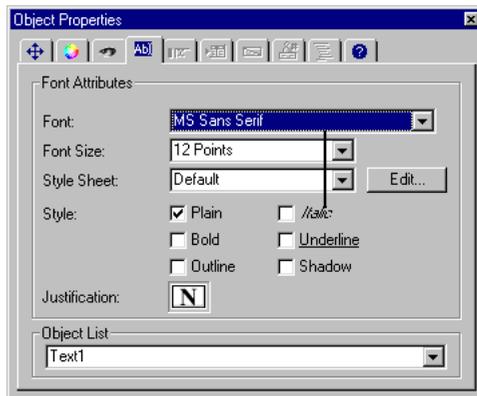
The choice between these two windows is explained in [“Displaying and Setting Form and Object Properties”](#) on page 244.
 - 2 In the Object Properties window, Click the Font tab to view the current text attributes.

OR

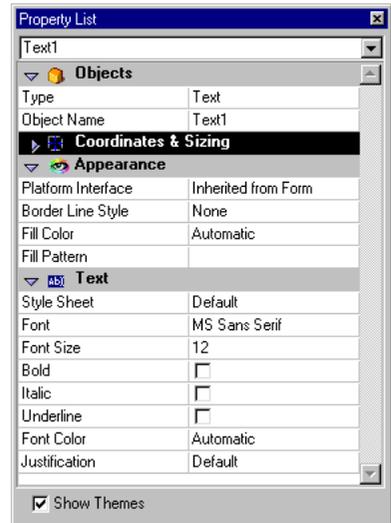
In the Property List, expand the Text theme.

The text properties are displayed:

Object Properties Window



Property List



- 3 Make the appropriate font, font size, and font style selections.

OR

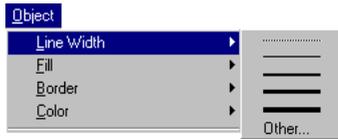
Choose a style sheet from the Style Sheet drop-down list.
- 4 Choose a justification.
- 5 To set text attributes for another text object, choose the desired object from the Object List or click another object on the form.

The settings in the Font page change to reflect the settings of the new object. If you select several text objects, the Object List area changes to “Selected Objects.”

Line Widths

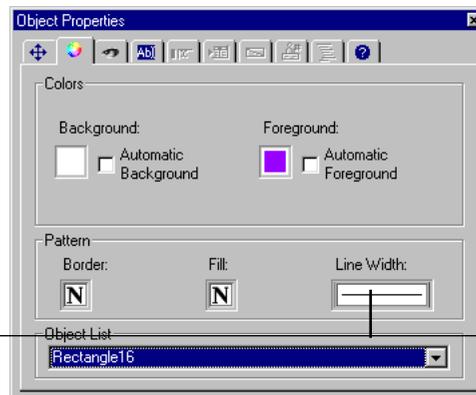
4th Dimension lets you specify different widths for lines and objects that have lines such as ovals, grids, and rectangles.

You can specify line widths using either the Line Width menu item on the Objects menu, the Colors page of the Object Properties window or the Appearance theme of the Property List.

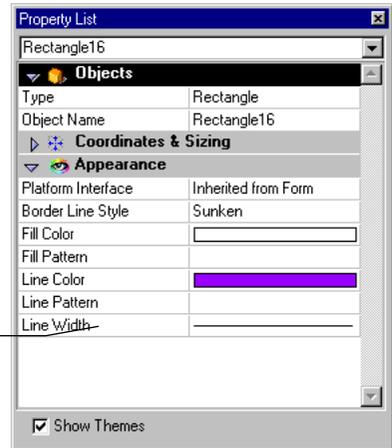


Object Properties Window

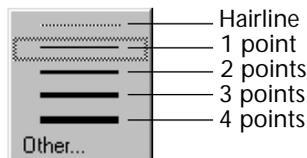
Property List



Line width menu



Choose one of the line widths displayed in the Line Width menu of either the Object menu or the Colors page.



If you choose **Other**, 4th Dimension displays a dialog box in which you can specify any line width up to 72 points.

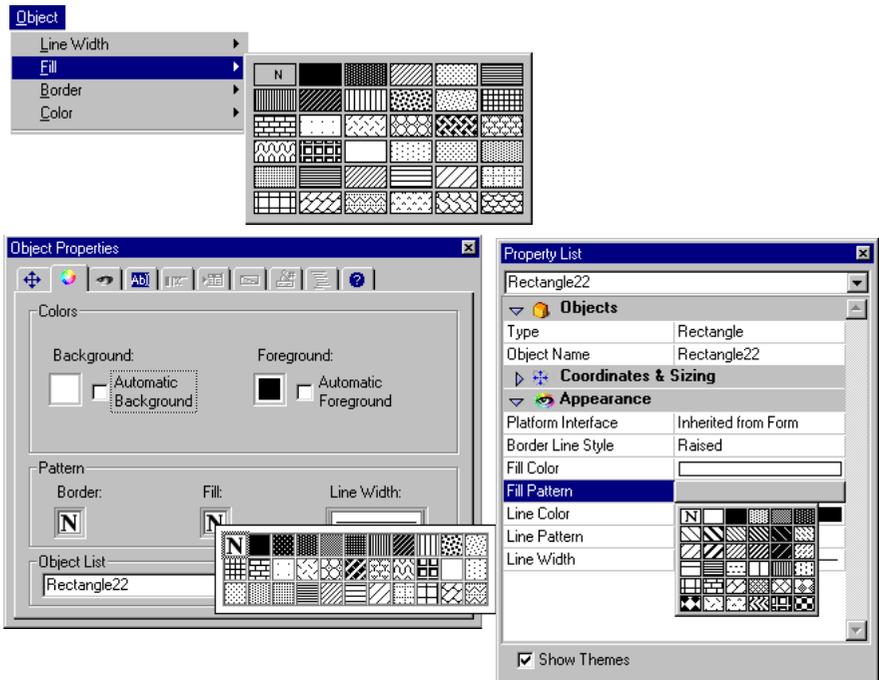


Fill Patterns

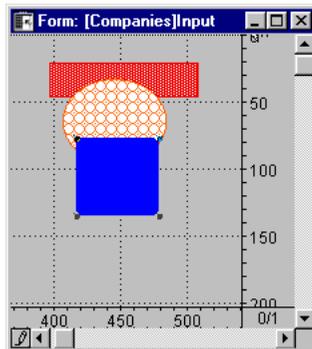
You can apply a fill pattern to any two-dimensional graphic object in the form such as an oval, a rectangle, a line, a grid object, and the enclosed area of a text object.

You can choose the fill pattern using either the **Fill** menu item in the **Object** menu or the **Fill Pattern** pop-up menu in the **Colors** page of the **Object Properties** window.

Select an object and choose one of the patterns displayed in either **Fill** menu item or from the **Property List**.



The figure below shows examples of areas filled with patterns.

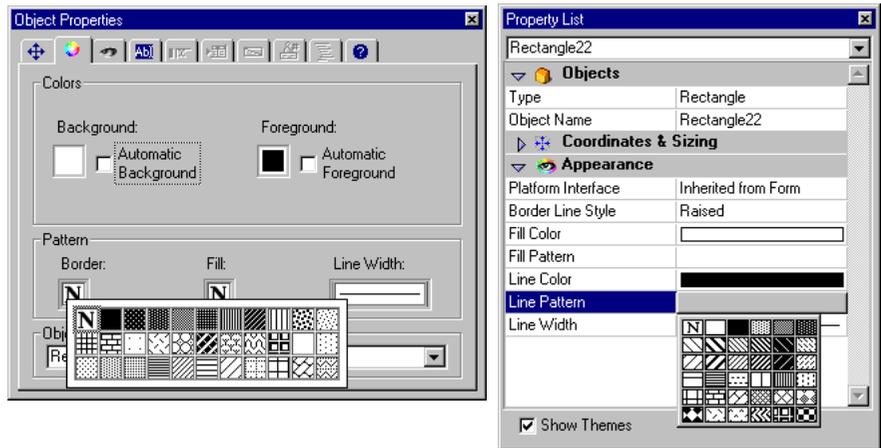
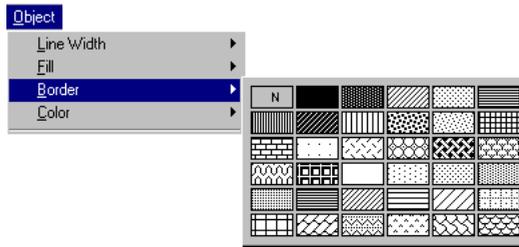


Border Patterns

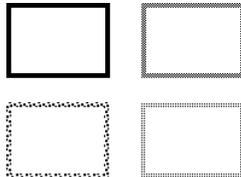
You can set patterns for the borders of any object in the form that has a border — such as an oval, a rectangle, and a grid object. The border patterns available are the same as the fill patterns. The appearance of the border also depends on the line width you have specified for the border.

You can set the border pattern using either the **Border** item of the **Object** menu, the **Border** menu in the **Colors** page of the **Object Properties** window or the **Property List**.

Choose one of the patterns displayed in the either Border menu item.



The following illustration shows examples of borders.



Foreground and Background Colors

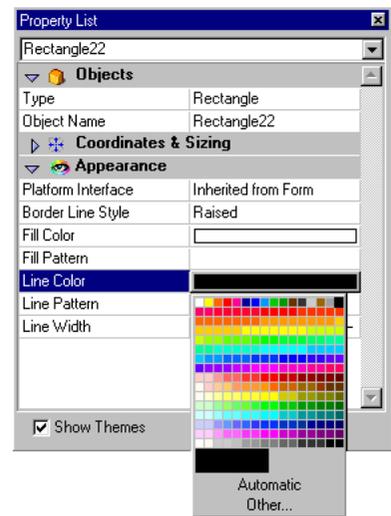
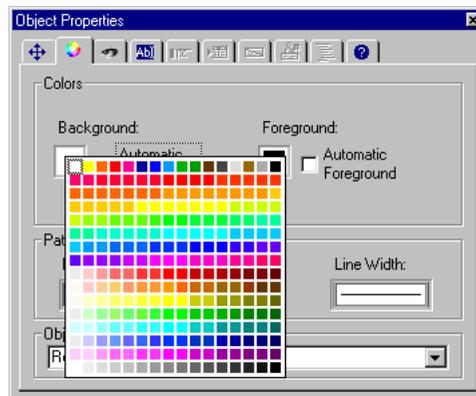
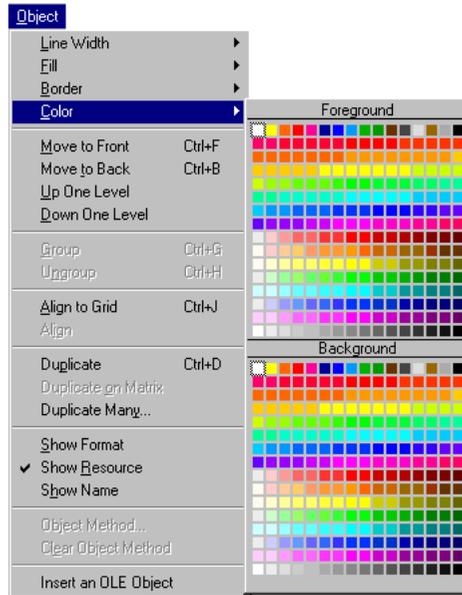
4th Dimension lets you add colors to objects for display on a color monitor or (if your printer supports color) for color printing.

Note Colors appear black and white on a black and white monitor. They appear as shades of gray on a gray-scale monitor. The color palettes display shades on a gray scale monitor.

You can specify different colors for foreground pixels (pixels that appear black on a black-and-white monitor) and background pixels (pixels that appear white on a black-and-white monitor). On a color

display, the mixing of foreground and background colors allows you to create custom tints and shades.

You can set foreground and background colors using either the Colors page of the Object Properties window, the Color item in the Object menu or the Property List.



Note If you select the **Automatic** option for foreground and background colors, the selected colors will be the colors selected in your OS.

You can select the **Automatic** option by selecting **Foreground** or **Background** in the **Color** submenu.

The **Property List** allows you to use the system color chooser to define a color by selecting **Other**.

If your monitor supports 16 colors, choose the colors from the first 16 colors on each palette. If your monitor supports 256 colors (or more), any colors you choose will display properly.

Placing a Picture from the Picture Library

You can insert static pictures in your forms using two methods:

- by pasting a picture,
- by dragging a picture from the Picture library.

The Picture library stores images that you can use as graphic elements on forms, as picture menu items, as picture buttons, as small icons in lists, and as custom toolbar icons. When you want to place a background graphic on a form, you should add it to the Picture library and then place the picture on the form. If you use a picture in the Picture library on more than one form, it is stored only once. Also, if you update a picture in the Picture library, references to it will be updated automatically throughout the database.

For more information on how to use the picture library refer to [Chapter 11](#).

If you place a picture on the background page of a multi-page form, it will appear automatically as a background element on all pages. Therefore, your database will run faster than if the picture was pasted into each page.

- ▶ To place a picture on a form:

- 1 Open the form to which you want to place the picture.

If necessary, navigate to the page on which you want to place the picture. For information on navigation, see the section [“Moving from Page to Page” on page 316](#).

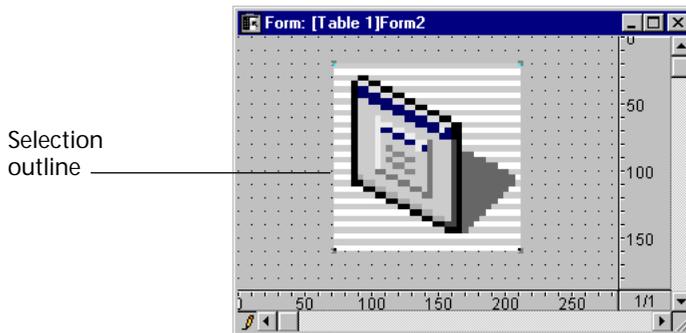
- 2 Open the Picture library and click on the name of the desired picture.
OR
If the picture you want to insert is on the Clipboard, choose Paste from the Edit menu and then go to step 6.
- 3 Display the pictures as a list in the picture library.
- 4 Click on the picture and drag it from the Picture library to the form.

Note If the picture you drag is defined as a table of thumbnails, it will automatically be inserted as a picture button or picture menu. If you want to insert it as a static picture, press the Alt key (on Windows) or the Option key (on Mac OS) when dragging the picture.

- 5 When it is at the desired location, release the mouse button.
- 6 Reposition the picture as desired and set its properties.
The picture has a set of object properties, just as any object on the form. If you like, you can modify those properties. This is described in the following sections.

Modifying the Background of the Picture

You can change the picture's background to Transparent so that it takes the form's background color.



- To modify the picture's background:
 - 1 Double-click the picture to display its properties in either the Object Properties window or the Property List.
The choice between these two palettes is explained in [“Displaying and Setting Form and Object Properties” on page 244](#).
 - 2 In the Property List, expand the Appearance theme.
OR
In the Object Properties window, click the Colors tab.

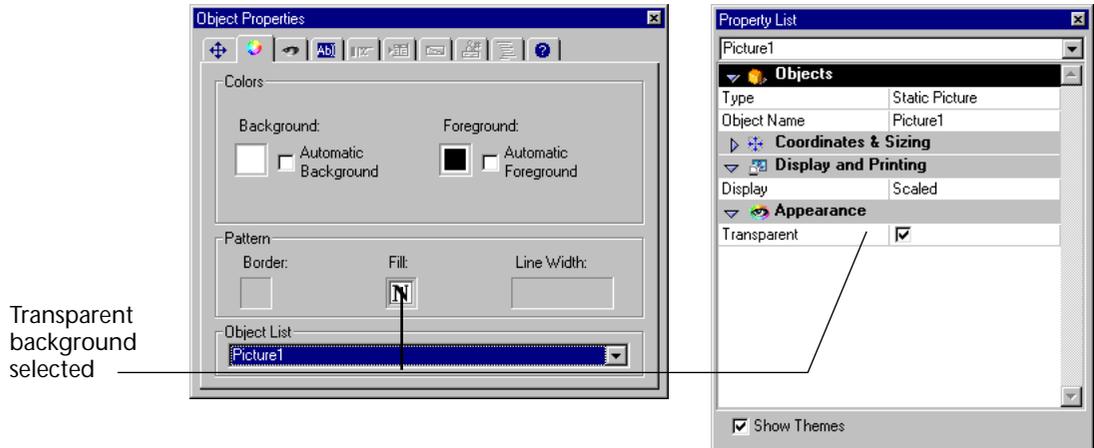
3 In the Object Properties window, select the N character in the Fill picture menu.

OR

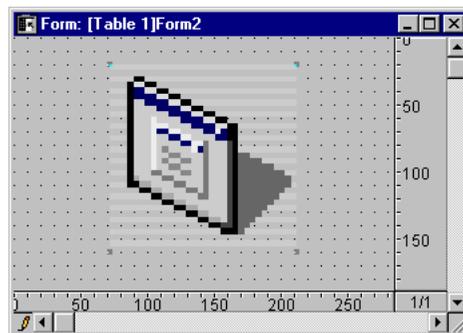
In the Property List, check the Transparent option.

Object Properties Window

Property List



The background of the picture becomes transparent:



Setting the Display Mode for a Static Picture

You can set the display mode of a picture that is placed in a form.

► To set the display mode for a static picture:

- 1 Double-click the picture to display its properties in either the Object Properties window or the Property List.

The choice between these two palettes is explained in [“Displaying and Setting Form and Object Properties” on page 244](#).

2 In the Property List, expand the Display and Printing theme.

OR

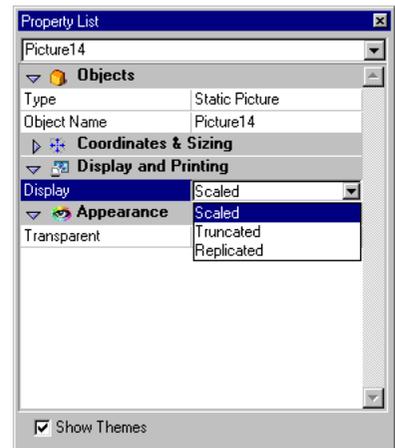
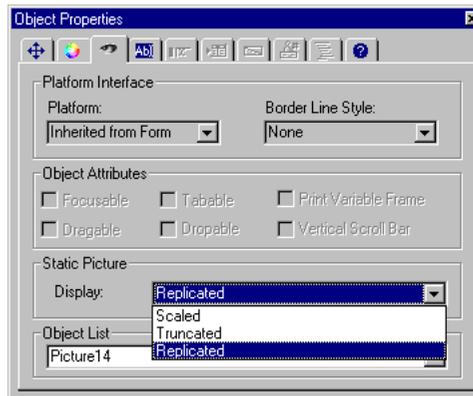
In the Object Properties window, click the Display tab.

3 Choose a display option from the Display drop-down list.

These options are described below.

Object Properties Window

Property List



- **Scaled (default mode)** When the picture object is resized, the picture is resized so that the entire picture remains visible.
- **Truncated** When the picture object is resized, the picture keeps its proportions and only its boundaries change. The picture always stays in the centre of the picture object. If the picture object is reduced to a smaller size than the picture, the picture is truncated.
- **Replicated** When the size of the picture object is increased, the picture is replicated as many times as necessary to fill the new area. This mode is recommended for background pictures since it does not require large amounts of memory. The Form wizard uses that option when selecting the Background picture option (for more information, refer to [“Form Display Options” on page 216](#)). If the picture size is reduced to a smaller size than the original picture size, the picture is truncated (not centered).

Defining a Background Picture for Web forms

You can insert a static picture in a form designed to be published on the Web and use that picture as a background replicated picture for browsers. To do so, the picture's settings must comply with the following conditions:

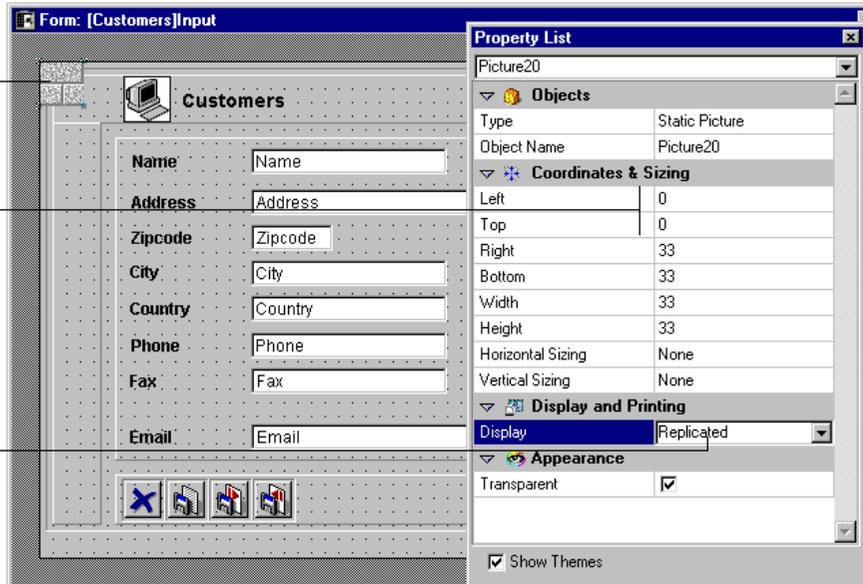
- The picture must be located in the upper left corner of the form (coordinates (0,0,x,x)).
- The Replicated option has to be assigned to the Display property of the picture.

Form editor

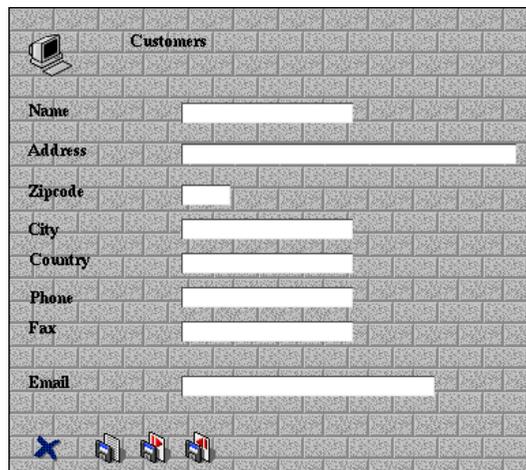
Picture inserted

Coordinates

Display



Browser



Dissociating a form Picture from its Library Source

When you insert a picture that comes from the Picture Library, you actually insert a reference to a picture. If the picture is modified in the Picture Library, each instance will be modified accordingly.

You may want to dissociate a picture inserted in a form from its source in the Picture library.

- ▶ To disassociate a picture from its source in the picture library.
 - 1 Display the picture's properties in the Property List.
 - 2 If necessary, expand the Objects theme.
 - 3 Click the Type line.

The Property List displays two types: Library Picture and Static Picture.
 - 4 Select Static picture.

The picture is then treated as if it had been pasted from the Clipboard.

Creating a Multi-page Form

You can create multiple pages for an input form. If you have more fields than will fit on one screen, you may want to create additional pages to display them. Multiple pages allow you to do the following:

- Place the most important information on the first page and less important information on other pages,
- Organize each topic on its own page,
- Reduce or eliminate scrolling during data entry,
- Provide space around the form elements for an attractive screen design.

Note Multiple pages are a convenience used for input forms only. They are not for printed output. When a multi-page form is printed, only the first page is printed.

There are no restrictions on the number of pages a form can have. The same field can appear any number of times in a form and on as many pages as you want. However, the more pages you have in a form, the longer it will take to display the form.

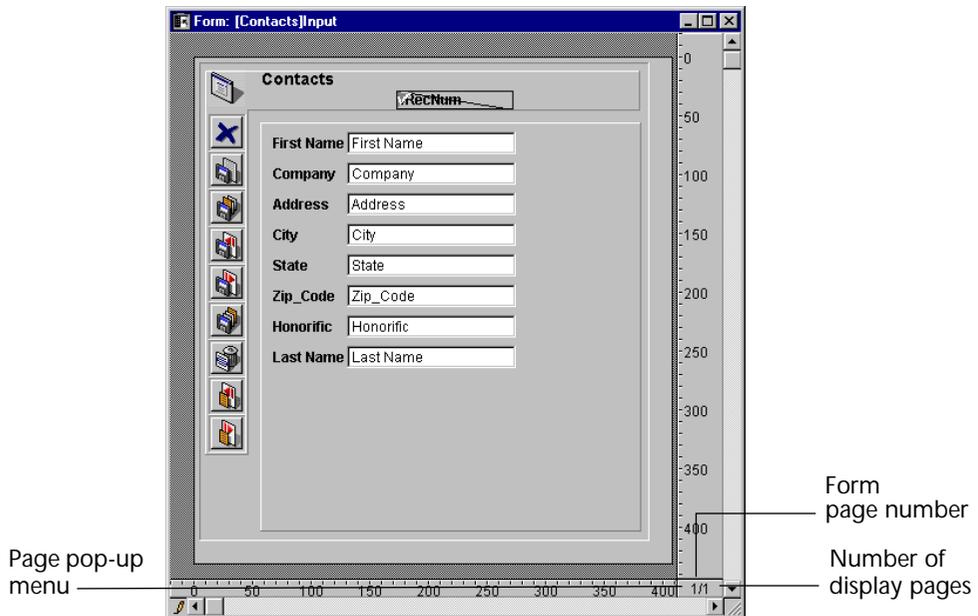
A multi-page form has both a background page and several display pages. Objects that are placed on the background page are visible on all

display pages, but can be selected and edited only on the background page. In multi-page forms, you should put your button palette on the background page. You also need to include one or more objects on the background page that provide page navigation tools for the user. For information on adding page navigation tools, see the section [“Adding Page Navigation Controls”](#) on page 319.

This section tells you how to add and delete pages, how to add objects to the background page, how to move from page to page, and how to add fields to a new page.

Adding a Display Page to a Form

Every form has at least one display page¹ and a background page. The current page number appears in a box in the lower-right corner of the form window when rulers are displayed. This corner also includes a pop-up menu that allows you to select the displayed page. The background page is numbered zero (0).



1. The Options page of the Form Wizard contains an option that instructs the Form Wizard to create a multi-page form automatically if the fields you select don't fit on one page. If you selected this option, your form may initially have more than one display page. When the Form Wizard creates more than one display page, it puts buttons, variables, the form title, and decorative rectangles on the background page.

- ▶ To add a display page:
 - 1 Move to the last page of the form, then click the Next Page icon  in the Tools palette.

4th Dimension displays a dialog box asking if you want to add a page. Click OK to create the new page.

OR

Choose Add Page from the Form menu.

4th Dimension creates a new page.

A new, blank display page appears in the Form editor window. The page indicator box in the lower-right corner of the window displays the number of the page you are viewing.

You can now add fields and other form elements to the new page.
- ▶ To insert a display page before the current page:
 - 1 Choose Insert Page from the Form menu.

4th Dimension inserts a new page before the current page and displays it on screen.

Moving from Page to Page

When you want to display the background page or move to another display page, you can either use the page navigation tools in the Tools palette or the page pop-up menu in the Form editor window.

- ▶ To display the background page (page 0):
 - Move to the first page of the form and click the Previous Page icon  in the Tools palette.

OR

Use the Page pop-up menu to move to page 0.

OR

Select 0 from the Goto Page submenu in the Form menu or in the Form editor contextual menu.

4th Dimension displays the background page. The page number of the background page is zero (0). Objects located in the background page are displayed in each page. You can place any object type in the background page.

Note There should be no confusion between using the Page 0 menu item from the Display submenu and actually making the background page the current edited picture. Selecting the Page 0 menu item from the Display submenu only displays the items of the background page. For

more information, refer to [“Showing/Hiding Elements in the Form Editor” on page 232.](#)

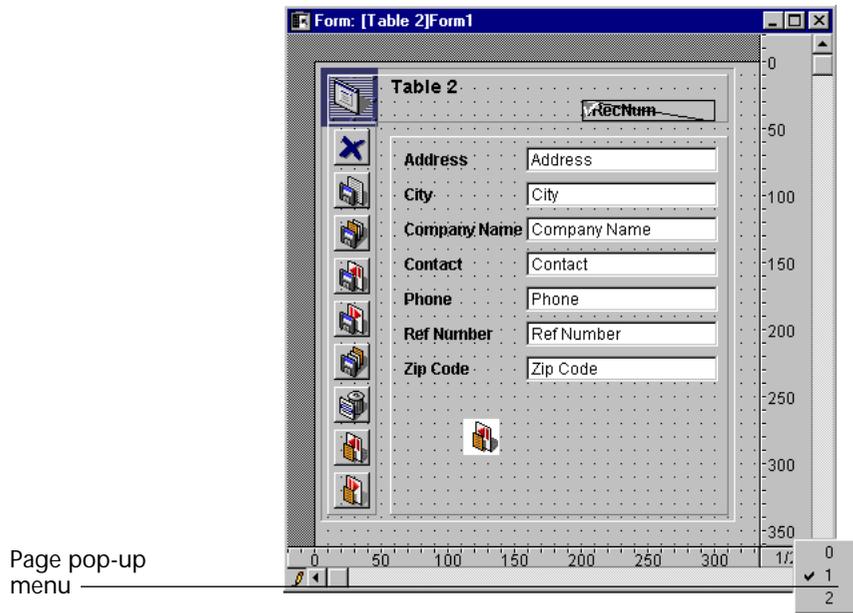
If you want to use a graphic as a background image, add it to the Picture library and then place it on the background page. For information on the using the Picture library, refer to [Chapter 11.](#)

Note You can also directly paste the picture in the form.

- ▶ To use the page navigation tools:
 - To move to the next page, click the Next Page icon  in the Tools palette.
 - To move to the previous page, click the Previous Page icon  in the Tools palette.
- 4th Dimension displays the page immediately following or prior to the current page.

If you click **Previous Page** while viewing the first page of the form, the background page appears. If you click **Previous Page** while viewing the background page, nothing happens. If you click **Next Page** while viewing the last page of the form, 4th Dimension asks if you want to create another page for the form.

- ▶ To display any page:
 - 1 Hold down the mouse button on the page indicator at the bottom-right corner of a Form editor window.



OR

- Display the Goto Page submenu in the form editor contextual menu.
- 2 Choose the desired page number.

Deleting a Page

You can delete unwanted display pages from a multi-page form. Any fields or other objects on the deleted pages will be deleted as well. The remaining pages are renumbered. You cannot delete the first page or the background page in a form that consists only of those two pages.

- ▶ To delete a page from the form:
 - 1 Use either the page navigation tools or the page pop-up menu to display the page you want to delete.

- 2 Click the Delete Page tool in the Tools palette .

OR

Choose Delete Page from the Form menu.

A dialog box appears asking if you are certain that you want to delete the page from the form.

- 3 Click OK.

4th Dimension removes the page and any objects on the page from the form.

Adding Fields to a Blank Page

When you add a new page to a form, it is blank. You can add fields in the following ways:

- Use the Add Field tool to place each field.
- Copy or duplicate fields from other pages or from other forms, paste them onto the new page, and change the copied fields' properties.
- Drag fields from the Tables page of the Explorer onto the form.

Adding Page Navigation Controls

When you create a multi-page form, you need to provide a way for users to move from one page to another. 4th Dimension provides three ways that you can use to add navigation tools:

- **Tab control** The tab control object gives users random access to individual pages. You place the tab control on the background page of the form and use its properties to provide page navigation controls.
- **Automatic buttons** You can add automatic page navigation buttons to the form — First Page, Last Page, Previous Page, and Next Page. These buttons should be placed on the background page.
- **Object methods** In addition, the language includes the GOTO PAGE command. You can use this command as part of an object method to create custom navigation controls using any suitable object type. For example, you can choose to use a picture button or pop-up menu to provide page navigation controls.

Adding Page Navigation Buttons

You can include page navigation buttons when you generate the form using the Buttons page of the Form Wizard. After the form is generated, open it in the Form editor and add the necessary pages. If you need to add the page navigation buttons after the form is created, you can do so using either the Active object tool in the Tools palette or a button tool in the Objects palette. For more information, see [“Creating an Active Object” on page 374](#).

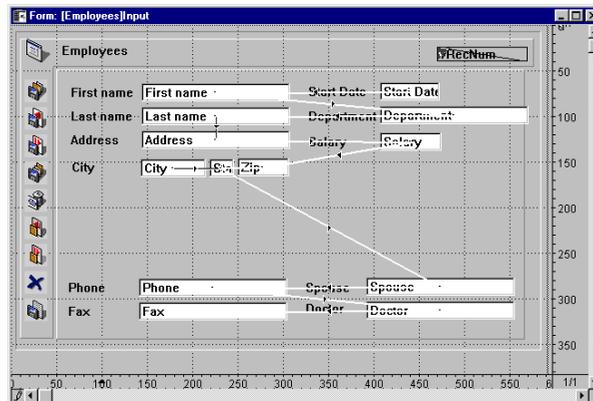
Using a Tab Control

The tab control provides a visual indication of the current page and the remaining pages. For information on creating and activating a tab control, see the section [“Tab Controls” on page 407](#).

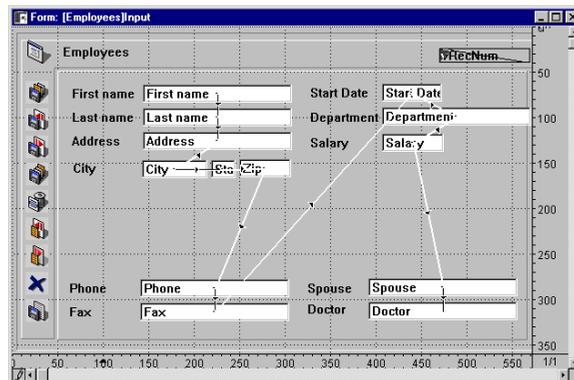
Data Entry Order

The data entry order is the order in which fields, subforms, and other tabable objects are selected as you tab through an input form. If you don't specify a custom entry order, 4th Dimension always selects the upper-left object first and subsequently moves to the right and down. If two objects are exactly the same distance from the top, 4th Dimension selects the leftmost object first. If an object is even one pixel higher than an object to the left, the slightly higher object will be selected first.

In some forms, a custom data entry order is needed. For example, the following figure shows fields from an employee database. The fields are placed in groups. However, the standard data entry order forces the user to enter the information in an awkward manner.



The custom data entry order allows you to enter the information in a more logical order.



Viewing and Changing the Data Entry Order

The Entry Order menu item lets you view the current entry order of all fields in a form and allows you to create a custom entry order.

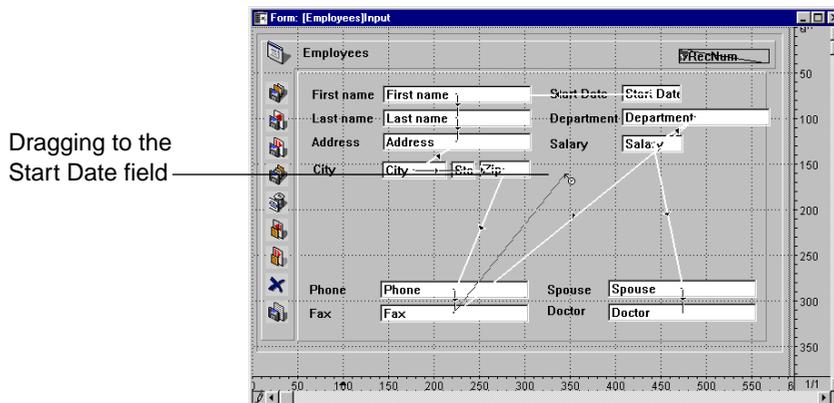
- ▶ To view or change the entry order:

- 1 Choose Entry Order from the Form menu.

The pointer turns into an entry order pointer , and 4th Dimension draws a line in the form showing the order in which it selects objects during data entry.

Viewing and changing the data entry order are the only actions you can perform until you click any tool in the Tools palette.

- 2 To change the data entry order, position the pointer on an object in the form and drag to the object you want next in the data entry order. 4th Dimension adjusts the data entry order accordingly.



- 3 Repeat step 2 as many times as necessary to set the data entry order you want.
- 4 When you are satisfied with the data entry order, click any unselected tool in the Tools palette or choose Entry Order from the Form menu. 4th Dimension returns to normal operation of the Form editor.

Setting the First Object in the Data Entry Order

All enterable objects are part of the data entry order. To define the first object of the entry order, the Entry Order mode must be disabled.

- ▶ To establish one of the objects as the first in the data entry order:

- 1 Select the object you want to be first in the entry order.

- 2 Choose Move to Back from the Object menu.
OR
Click the Move to Back tool in the Tools palette.
OR
Select Move to Back from the object contextual menu.
The object will be placed in back of any other form element. This is temporary.
- 3 Choose Entry Order from the Form menu.
The selected object becomes the first object in the entry order and the object that was first becomes second. You can now drag from object to object in order to create the data entry order you want.
- 4 When you have finished, click any unselected tool in the Tools palette.
The Form editor returns to normal operation.
The object you selected to make first in the entry order has returned to its previous location (it is no longer moved to the back of the form).

Using a Data Entry Group

While you are changing the data entry order, you can select a group of objects in a form so that the standard data entry order applies to the objects within the group. This allows you to easily set the data entry order on forms in which fields are separated into groups or columns.

- ▶ To create a data entry group:
 - 1 Choose Entry Order from the Form menu.
 - 2 Draw a marquee around the objects you want to group for data entry.
When you release the mouse button, the objects enclosed or touched by the rectangle follow the standard data entry order. The data entry order for the remaining objects adjusts as necessary.

Restoring the Standard Data Entry Order

You can restore the standard data entry order at any time.

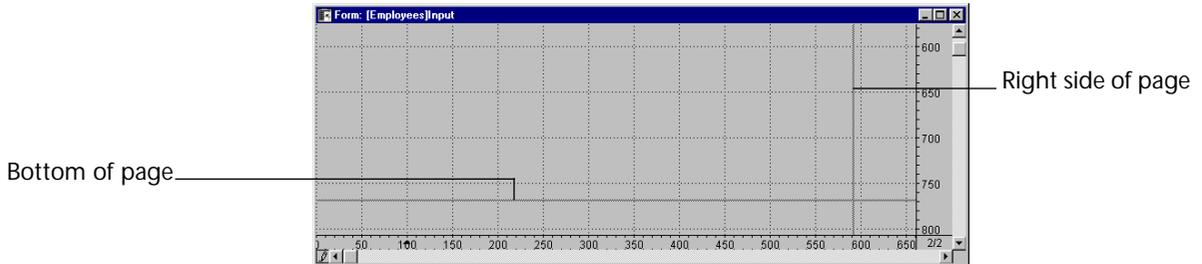
- 1 Choose Entry Order from the Form menu.
- 2 Draw a marquee around all the objects on the form.
When you release the mouse button, the objects enclosed or touched by the rectangle take the standard data entry order.

Viewing and Printing Forms

Each form has an area of about 1245 square feet. You scroll to bring hidden portions of the form into view. For viewing on screen, your form design can use this entire area. You can scroll to view any element you place in the form.

For printing, form elements must fit within a single page width, but may be several pages in length. The actual size of a page depends on your printing device, the paper it is using, and the specifications you enter in the Page Setup dialog box. 4th Dimension displays page border lines in the Form editor. These lines indicate the page limits. The page border lines respond to any page setup changes. The page setup specifications are stored with the form when it is closed. The form's limits can be displayed or hidden, for more information refer to [“Showing/Hiding Elements in the Form Editor” on page 232](#).

The figure below shows the page border lines.



Saving Forms

It is a good idea to save any changes you make to a form, especially when using 4D Server with multiple users. You can save a form by closing or saving it. You can close a form by clicking its Control-menu box or by choosing Close Form: **Name** from the File menu.

To save a form without closing it, choose Save Form: **Name** from the File menu.

Once a form has been saved, you can continue to work on it. If you make a mistake or do not like the changes you have made, you can revert to the last saved version of the form. This makes the form appear exactly as it did the last time that it was saved. To do this, choose Revert to Saved from the File menu.

4D Server When a form is saved in the Design environment, users are able to see your changes the next time they open the form.

5

Working with Fields and Active Objects

4th Dimension allows you to customize data entry forms so that your interface includes exactly those features that you need. You can add interface elements such as picture buttons, tab controls, drop-down lists, combo boxes, and hierarchical lists to your data entry forms. You can also implement drag and drop operations. You can use triggers, form methods, or object methods to enforce business rules during data entry.

This chapter tells you how to do the following:

- Place fields and other active objects in the form,
- Set display formats and data entry filters,
- Use data entry constraints such as maximum, minimum, default, or required values,
- Write form or object methods,
- Add interface objects such as buttons, pop-up menus or drop-down lists, combo boxes, scrollable areas, splitters and tab controls,
- Add subforms to forms,
- Attach custom menus to forms.

Active Objects Defined

An active object is anything on a form that performs a database task or an interface function. There are many kinds of active objects. Fields are considered active objects. Other active objects — enterable objects (variables), combo boxes, drop-down lists, picture buttons, and so on — store data temporarily in memory or perform some action such as opening a dialog box, printing a report, or starting a background process.

In some cases, you can specify the active object's action by making selections in the Object Properties window. For example, you can use built-in automatic button actions to specify the action of a button. In other cases, you specify the object's action by writing a method that is automatically attached to the object.

There are also instances in which you will want to manage active objects at a higher level. For example, data validation tasks can be handled by the trigger that runs automatically when a record is saved. The trigger can examine the values in each field for possible violations of business rules.

Adding Fields to a Form

You can add or delete fields from a form at any time. For example, you may decide to add fields to a form when the following occurs:

- You discover you need a field you did not choose in the Form Wizard.
- You add a field to the database structure and need to add it to a form so that you can use it.

When you place a field in a form, you can immediately set its properties. You can add a field to a form using either the Tools palette or the Explorer.

Note You can also add a field by duplicating an existing field and then modifying the duplicate's properties.

- ▶ To add a field to a form using the Tools palette:
 - 1 If the field does not exist in the database, use the Structure editor to create the field.

For information on adding a field to a table, see the section [“Creating Fields and Setting Field Properties”](#) on page 131.

- Open the form to which you want to add the field.

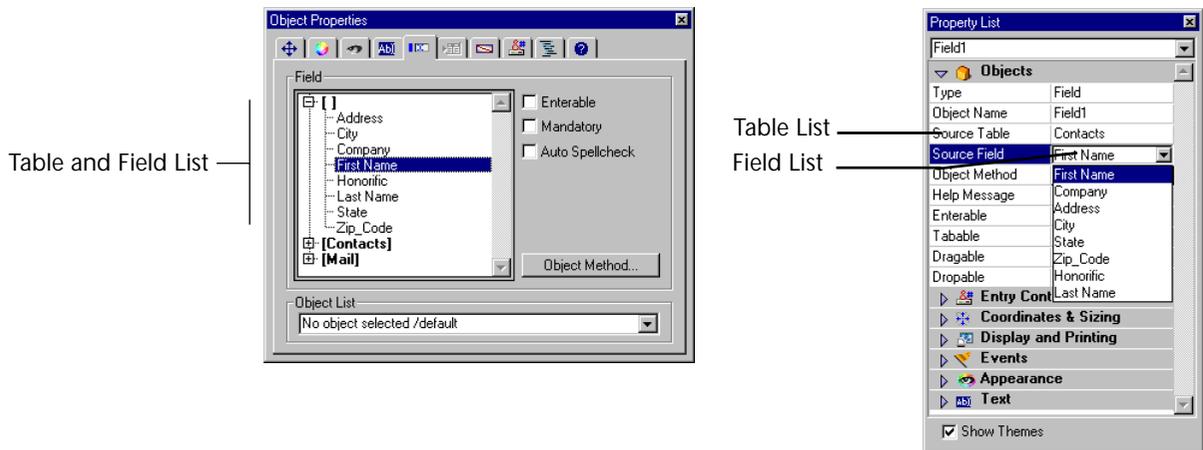
For information on opening a form, see the section [“Opening a Form in the Form Editor”](#) on page 250.

- Click the Add Field tool  in the Tools palette.
- Either drag and drop the tool or select the tool by clicking it and draw the field area.

For more information about how to create an object refer to [“Creating an Active Object”](#) on page 374.

4th Dimension automatically displays the properties of the new field in either the Object Properties window or the Property List (depending on your current settings).

- Select the field you want to insert from the table/field list.



Note Fields from the master table appear twice, once at the top of the list of fields (beneath the empty brackets), and once in the table name’s expanded list. If you select the field from the top list, 4th Dimension takes this to mean “field01, in this position in the master table.” If you select the field from the table name’s expanded list, 4th Dimension takes this to mean “this field from this table.” For example, if you select the third field from the top list and then copy and paste the field into another table’s form, in the new form it would become the third field in the new master table. If you select the field from the table

name's expanded list and then copy and paste the field into another table's form, it would remain the same field from the same table.

- 6 If desired, select the specific properties you want to assign the field. After creating the field, you generally need to set additional properties. You can set data entry controls, write help text, attach a method, set resizing or repositioning options, set platform interface, font, or appearance options.

For information on field attributes, see the sections [“Field Attributes” on page 140](#) and [“Setting the Enterable and Mandatory Attributes” on page 332](#).

The new field appears in the form where you placed it. The field area displays the name of the field you selected.

- To add a field using the Explorer:

- 1 Open the form to which you want to add the field.

For information on opening a form, see the section [“Opening a Form in the Form Editor” on page 250](#).

- 2 Open the Explorer and click the Tables tab to display the hierarchical list of tables and fields.
- 3 Expand the table that contains the field you wish to add.

The fields belonging to that table appear.

- 4 Drag the desired field from the Explorer to the form.

4th Dimension automatically displays the field properties in either the Object Properties window or the Property List, depending on the display settings that you selected.

After you place a field in a form, you can modify it as you would any other form object. You can resize it, change the font, choose colors for display on a color monitor, and so on. You can return to the Object Properties window or the Property List to modify the field's properties at any time.

Modifying a Field in a Form

As for any active object, you can set the field properties in the Object Properties window or in the Property List. For more information about this possibility, refer to [“Displaying and Setting Form and Object Properties” on page 244](#).

- To modify a field's properties:

- 1 Select the field you want to work with and double-click it.

Depending on the current Display settings, either the Object Properties window or the Property List appears.

You can use the Object Properties window or the Property List to change any setting. You can even select a new field in the form to display its properties and you can select a set of fields to modify their common properties.

2 Make any changes you like.

Your changes take effect immediately.

Note If the field has been grouped with another object, you must ungroup it before you can display the Object Properties window.

Changing a Field into a Variable and Vice-Versa

The Property List allows you to transform every object type (active or not) into another object type. You can also transform a field into a variable and vice-versa. This is useful when, after inserting a field in a form, you want to change that field into a variable because you don't need to store the value. When 4th Dimension changes an object into another object, it keeps the original properties of the object (coordinates, object method, appearance, color and so on).

The data type assigned to a field will be kept for the variable: a picture field will be converted to a picture variable.

To change a variable into a field or a field into a variable, select the object and select Field or Variable from the Type drop-down list in the Property List. The Property List is then updated to display the properties for that new object type. The object name, object method, and its properties (size, enterable and so on) remain identical.

Note When you change a variable into a field, 4th Dimension assigns the first field in the first table to the object by default. You can manually define the table and field in Source table and Source field.

Inserting Dynamic Table and Field Names

You can insert dynamic table and field names into your forms (as static text). When you place dynamic table or field names in your forms, they are automatically updated throughout your database:

- either when you modify the table or field name in the Structure window, or
- when the 4th Dimension commands Table name or Field name (in the "Structure Access" theme) are called.

This feature is particularly useful when you want to update the table or field names in the forms on-the-fly.

- ▶ To insert a dynamic table or field name in a form:
 - 1 In a static text area, enter the following reference:
 - to insert a dynamic table name: `<?[TableName]>` or `<?[2]>` (the table's creation order number, meaning the second table created).
 - to insert a dynamic field name: `<?[TableName]FieldName>` or `<?[2]3>` (the table's and field's creation order number), or even `<?3>` (the field's creation order number) for the current table's field.

Please note that table and field numbers correspond to their creation order. You can add or rename tables and fields without modifying the dynamic reference system.

- 2 Click outside of the text area.

The current field's or table's name appears as it has been defined in the Structure window.



You can view the “actual” contents of a static area in the Form editor by clicking on the area or by selecting Show Format or Show Name from the Objects menu.

In the User and Custom Menus environments, a table's or field's name can be modified on-the-fly by using the Table name and Field name commands. In this case, the table and field name references will display the values defined by these commands.

Note Dynamic field names is available as an option in the Options page of the Form Wizard. For more information, refer to [“Form Display Options” on page 216](#).

Data Entry Controls

You can establish data entry controls for fields and enterable objects at the form level. Data entry controls restrict what the user can enter into the field or enterable object on a particular form. You can do the following:

- Set the Enterable or Mandatory attributes,
- Attach a choice list,
- Establish a list of required or excluded values,
- Set an entry filter that defines allowable characters,
- Set maximum and minimum allowable values,
- Set default values,
- Write an object method.

You can also establish data entry controls at a higher level. You have the following options:

- **Field properties** The Field Properties window (accessed from the Structure editor window) lets you set field attributes at the table level. Field properties are enforced throughout the database. In some cases, you have the option of setting a particular attribute at either the table or form level. You can set the following attributes at the table level:
 - **Mandatory** Set the Mandatory attribute for a field that is required for all records.
 - **Display Only** Set the Display Only attribute for calculated fields or other fields that are not enterable.
 - **Can't Modify** Set the Can't Modify attribute for fields that accept an initial entry but should not be changed after the record is first saved.
 - **Indexed** Set the Indexed attribute for fields on which you will be searching and sorting frequently. Also, use the Indexed attribute for primary and foreign key fields.
 - **Unique** Set the Unique attribute for the field that must be used to uniquely identify the record.
 - **Choice list** You also can associate a choice list with the field. When the choice list is assigned at the table level, it is used on all entry forms and in the Query editor.

- **Relation properties** The Relation properties window contains the Deletion control option that lets you set and enforce referential integrity. You can
 - Prevent 4th Dimension from deleting a related One record if there are related Many records,
 - Automatically delete the related Many records if the related One record is deleted,
 - Allow the user to delete a related One record even if related Many records exist (turn off referential integrity).
- **Triggers** You can create triggers that manage the process of loading, saving, and deleting records. Triggers run if a record is loaded, saved, or deleted programmatically, using any form, or during data imports and exports. With triggers, you can enforce complex business rules in a very comprehensive and systematic way. For information on using triggers, see the section [“Using Triggers” on page 487](#) and the section “Triggers” in the *4th Dimension Language Reference*.
- **Form methods** You can create form methods that manage the use of the form.

The following sections review the data entry control options that you have at the form level. Use these options in conjunction with data entry controls that are available at the database and table levels.

Setting the Enterable and Mandatory Attributes

The Enterable and Mandatory attributes are similar to the field attributes you set in the Structure editor. If you want these attributes to be different on a particular form, you can change them here. These attributes can be set on the Field page of the Object Properties window as well as in the Property List.

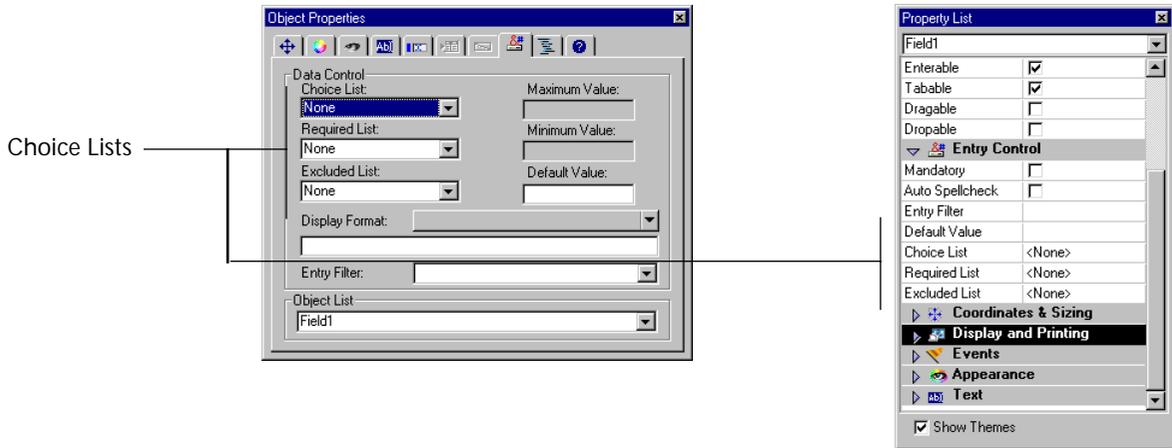
These attributes do not override the field attributes set in the Structure editor. If a field already has the Display Only attribute assigned in the Structure editor, you cannot make it enterable with the Enterable form attribute. If a field already has the Mandatory attribute assigned in the Structure editor, you cannot make it non-mandatory by deselecting the Mandatory form attribute. The Enterable and Mandatory check boxes do not necessarily reflect the attribute settings in the Structure editor.

-
- The Enterable Attribute** Every field is enterable by default. If you want to make a field non-enterable for that form, you can deselect the Enterable check box either in the Object Properties window or in the Property List.
- A field from a related table may not be enterable if you deselected the Enterable Related Fields check box in the Form Wizard. You can make the related field enterable by selecting the Enterable check box.
- For enterable objects, the Enterable check box is checked. You can make an enterable object non-enterable by changing the definition of the object from enterable to non-enterable by choosing non-enterable from the Type drop-down list or by unchecking the Enterable check box. For information about enterable objects, see the section [“Enterable and Non-enterable Variables” on page 382](#).
-
- Note* The contents of the Property List are contextual. When the Enterable attribute is deselected in the Property List, properties that are related to entry control (Mandatory, Tabable, Entry filter and so on) disappear from the list.
-
- The Mandatory Attribute** No field or enterable object is mandatory by default. To make the field mandatory for all forms, set the Mandatory attribute in the Field Properties window in the Structure editor. If you want to make a field or enterable object mandatory for a particular form, you can select the Mandatory check box in either the Object Properties window or the Property List.
- Selecting the Mandatory check box makes a field or enterable object mandatory for that form. 4th Dimension does not accept a record if the field or object does not contain a value. For information about enterable objects, see the section [“Enterable and Non-enterable Variables” on page 382](#).
- Using Choice Lists** You can assign a choice list to a field at either the table or form level. If you want to assign the choice list at the table level, use the Field Properties window in the Structure editor. The choice list will then be available on all forms and in the Query editor in the User environment and in custom applications (assuming you use the Query editor in the custom application).
- You can also attach a choice list at the form level. The list can serve as a choice list for that form, as a list of required entries, or as a list of excluded entries. Choice lists can be associated with a field at the form

level by either using the Data Control page of the Object Properties window or the Property List.

Object Properties window

Property List



Assigning a list at the form level gives you the freedom to vary data entry constraints from form to form. For example, a field in one input form that is used only by managers can display a comprehensive choice list while the same field in the form that is seen only by drudges has fewer choices.

If a field is already assigned a choice list at the table level, you can override the choice list at the form level. If you assign a different list to the field at the form level, then that list is used for that form only.

Before you can assign a choice list, you must have created the list in the List editor. For more information about creating choice lists, see [Chapter 10, “Creating Lists” on page 583](#).

Choice Lists

Assigning a choice list to a field with the Choice List drop-down list causes 4th Dimension to display the choice list during data entry. The choice list appears when the field or enterable object is selected in the User environment or in custom applications. The user can then select an entry from the list. The user can overwrite the entry chosen from the choice list by typing (unless the list is also a required list).

Required Lists

A Required choice list to limit the valid entries to the items on the list. For example, you may want to require a list of job titles so that valid entries are restricted to titles that have been approved by management.

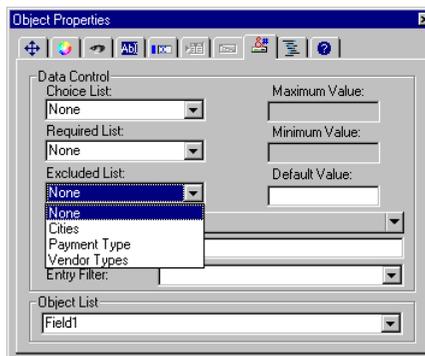
Note Making a list required does not automatically display the list when the field is selected. If you want to display the required list, assign the same list with the Choice List drop-down list.

Excluded Lists

An Excluded choice list prevents the items on the list from being entered. For example, for a field on an input form used only by drudges, you may want to attach a list of choices that can only be authorized by a manager.

- ▶ To assign a choice list to a field or enterable object:
 - 1 Display either the Object Properties window or the Property List. For more information on how to do this, refer to [“Displaying and Setting Form and Object Properties”](#) on page 244.
 - 2 Choose the desired list from the appropriate drop-down list in the Data Control page of the Object Properties window or in the Entry control theme of the Property List.

Object Properties window



Property List



4th Dimension assigns the list you have selected to the field or enterable object.

Using Entry Filters

An entry filter controls exactly what the user can type during data entry. Unlike the data entry controls discussed earlier in this section,

entry filters operate on a character-by-character basis. For example, if a part number always consists of two letters followed by three digits, you can use an entry filter to restrict the user to that pattern. You can even control the particular letters and numbers.

An entry filter operates only during data entry. It has no effect on data display after the user deselects the field. Usually, you use entry filters and display formats together. The filter constrains data entry and the format ensures proper display of the value after data entry. For complete information about display formats, see the section [“Data Entry Controls” on page 331](#).

During data entry, an entry filter evaluates each character as it is typed. If the user attempts to type an invalid character (a number instead of a letter, for example), 4th Dimension simply does not accept it. The null character remains unchanged until the user types a valid character.

Entry filters can also be used to display required formatting characters so that the user need not enter them. For example, an American telephone number consists of a three-digit area code, followed by a seven-digit number that is broken up into two groups of three and four digits, respectively. A display format can be used to enclose the area code in parentheses and display a dash after the third digit of the telephone number. When such a format is used, the user does not need to enter the parenthesis or the dash.

An Introduction to Entry Filter Codes

Entry filter codes usually start with an ampersand (&). This character tells 4th Dimension to use what follows as an entry filter. If the code starts with a tilde (~), it means the same thing as “&” except that any letter is automatically made uppercase.

The & is usually followed with an “A”, an “a”, or a “9”, meaning allow only uppercase letters (A), allow lowercase and uppercase letters (a), or allow only numbers (9). For example, &9 allows only numbers and &A allows only capital letters.

The number sign (#) tells how many digits or characters is allowed by the code. If the code uses no number signs, the filter allows as many digits or characters as you want. For example, &9 allows as many digits as is entered. The filter &9## allows only two digits.

The exclamation point (!) is sometimes used to change which character will appear on screen to indicate the number of characters the user can enter. Without an !, 4th Dimension displays an underscore for each

digit or character the user can enter. For example, !?&9## displays question marks in both of the places the user will type and it allows only numbers and only two digits.

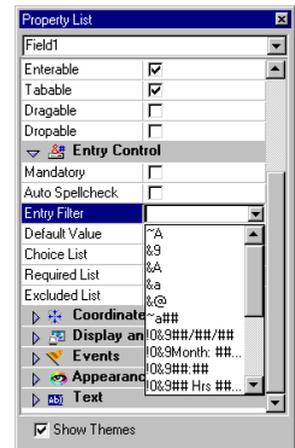
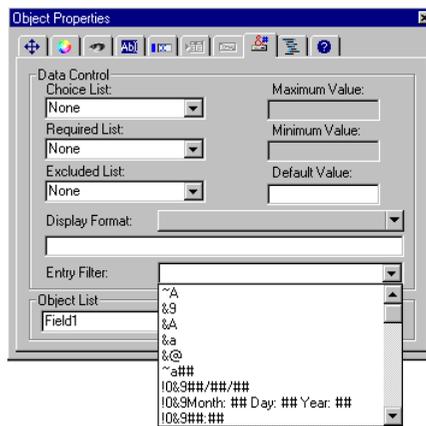
For information about creating entry filters, see the section [“Entry Filter Codes”](#) on page 340.

Choosing an Entry Filter You create the entry filter by choosing a built-in or custom filter from the Entry Filter drop-down list or by typing an entry filter code into the Entry Filter Display area. The Entry Filter drop-down list contains filters for date, time, and alpha fields. The names of any custom filters you create are added to the entry filter drop-down list. For information on creating custom filters, see the section [“Creating Custom Display Formats and Entry Filters”](#) on page 344. Most often you will find a suitable entry filter in the drop-down list.

The figure below shows an entry filter being chosen from the drop-down list.

Object Properties window

Property List



Here is a table that explains each of the entry filter choices in the Entry Filter drop-down list.

Entry Filter	Explanation
~A	Allow any letters, but convert to uppercase.
&9	Allow only numbers
&A	Allow only capital letters

Entry Filter	Explanation
&a	Allow only letters (uppercase and lowercase)
&@	Allow only alphanumeric characters. No special characters.
~a##	State name abbreviation (e.g., CA). Allow any two letters, but convert to uppercase.
!0&9##/##/##	Standard date entry format. Display zeros in entry spaces. Allow any numbers.
!0\$9 Month: ## Day: ## Year: ##	Custom date entry format. Display zeros in entry spaces. Allow any number. Two entries after each word.
!0&9##:##	Time entry format. Limited to hours and minutes. Display zeros in entry spaces. Allow any four numbers, separated by a colon.
!0&9## Hrs ## Mins ## Secs	Time entry format. Display zeros in entry spaces. Allow any two numbers before each word.
!0&9Hrs: ## Mins: ## Secs: ##	Time entry format. Display zeros in entry spaces. Allow any two numbers after each word.
!0&9###-####	Local telephone number format. Display zeros in entry spaces. Allow any number. Three entries, hyphen, four entries.
!_&9(###)!0###-####	Long distance telephone number. Display underscores in first three entry spaces, zeros in remainder.
!0&9###-###-####	Long distance telephone number. Display zeros in entry spaces. Allow any number. Three entries, hyphen, three entries, hyphen, four entries.
!0&9###-##-####	Social Security number. Display zeros in entry spaces. Allow any number.
~"A-Z;0-9; ;,; ;-"	Uppercase letters and punctuation. Allow only capital letters, numbers, space, comma, period, and hyphen.
&"a-z;0-9; ;,; ;-"	Upper and lowercase letters and punctuation. Allow only lowercase letters, numbers, space, comma, period, and hyphen.
&"0-9; ;-"	Numbers. Allow only numbers, decimal point, hyphen (minus sign).

You can modify an entry filter after you choose it from the drop-down list. For example, if you want to use a filter that allows upper and lowercase letters, but also need to allow the wildcard character (@), you could choose the filter:

```
&"a-z;0-9; ;,.;-"
```

and change it to:

```
&"a-z;0-9; ;,.;-;@"
```

For more information about modifying entry filters, see the section [“Entry Filter Codes” on page 340](#).

Using Entry Filters and Display Formats Together

You often create a matching display format when you use an entry filter. An entry filter operates only during data entry. It has no effect on how the data is displayed after you tab out of the field.

For example, if you use the Social Security number entry filter (&9###-##-####), you should also choose the matching Social Security number display format (###-##-####). Without the display format, only the numbers, not the hyphens, are displayed in the field. Display formats can be used in both input and output forms and in quick reports.

Here are some suggested entry filters and matching display formats for common types of fields.

Field Type	Entry Filter	Display Format
State	~a##	(none needed)
Zip Code (standard)	&9#####	(none needed)
Zip Code (extended)	&9#####-####	#####-####
Phone number	&9###-#### &9(###) ###-#### &9 ###-###-####	###-#### (###) ###-#### ###-###-####
Social Security number	&9###-##-####	###-##-####
Date	!0&9##/##/## &9Month: ## Day: ## Year: ##	(Any Date Format)
Time	!0&9##:## !0&9##Hrs##Mins##Secs !0&9Hrs:##Mins:##Secs:##	(Any Time Format)

You can use display formats on input forms, output forms, and quick reports. For information about using display formats in quick reports, refer to the chapter on quick reports in the *4th Dimension User Reference*.

Entry Filter Codes

Often, you create an entry filter simply by choosing it from the entry filter drop-down list. If you need a filter for a type of field not covered by the choices in the drop-down menu, you can create a filter or modify an existing one.

This section describes how to write the code for an entry filter.

An entry filter code has three parts, in this order:

initiator "argument" placeholders

The *initiator* informs 4th Dimension that the subsequent argument is to be used as a filter during data entry in the field. The *argument* defines the allowable characters. The *placeholders* define the places available for the characters.

For example, the following entry filter allows only the letters “a”, “b”, “c”, or “g” to be entered in two places:

```
&"a;b;c:g"##
```

In this example, the ampersand (&) is the initiator; the “a;b;c:g” is the argument; and the number signs (#) are the placeholders. The filter can be read as, “Allow the letters ‘a’, ‘b’, ‘c’, or ‘g’ in two places.” Thus the user may enter “ag”, “gc”, “ba”, “ab”, “aa”, “ac”, or any other combination of the four allowed characters.

Entry filters can be combined. The following entry filter allows only the letters “a”, “b”, “c”, or “g” to be entered in two places, followed by the numbers 1, 3, or 8 in one place:

```
&"a;b;c:g"##&"1;3;8"#
```

The user must use two of the allowed letters, followed by one of the allowed numbers.

Characters that Initiate a Filter

Two characters initiate a filter: the ampersand (&) and the tilde (~). These characters instruct 4th Dimension to use the argument that follows immediately as the filter for the subsequent placeholders.

In addition, the tilde (~) also instructs 4th Dimension to make any letters uppercase. It does not prevent a lowercase letter from being typed; it simply changes it to an uppercase letter.

The following entry filters are equivalent in their effects:

`&"P"#`

`~"p"#`

The difference between them is that the filter initiated with the ampersand (&) does not accept a lowercase p. The filter initiated with the tilde (~) accepts the lowercase p but converts it to uppercase.

Because no letters are involved, the following entry filters are equivalent:

`&"1;5;8"#`

`~"1;5;8"#`

Arguments

A filter argument follows the initiator and defines the characters that are allowed in the subsequent placeholders. To create a filter argument, surround the allowable characters with quotation marks.

Arguments are made up of lowercase letters, uppercase letters, numbers, punctuation marks, and special characters (!@#\$%^&*(){}[]";?><.,/'~). If you use a lowercase letter in the argument, only the lowercase form of the letter can be typed by the user. If you use an uppercase letter in the argument, only the uppercase form of the letter can be typed by the user.

An argument may be a single character (a letter or a number), for example, "j", "J", or "6".

An argument may be a set of characters separated by semicolons, for example, "a;r;t" or "1;5".

An argument may include ranges of characters. A range is defined by the first character, a hyphen, and the last character. Examples are, "a-c" and "1-5". The "a-c" argument is equivalent to "a;b;c", and "1-5" is equivalent to "1;2;3;4;5".

An argument may include single letters, single numbers, and one or more ranges, for example, "a;m-z;3;5-9".

The following table shows useful shorthand versions of arguments. They are used in filters without quotation marks.

Character	Meaning	Equivalent
9	Allow numbers	"0-9"

Character	Meaning	Equivalent
a	Allow lowercase and uppercase	"a-z;A-Z"
A	Allow uppercase	"A-Z"
@	Allow alphanumeric	"a-z;A-Z;0-9"

The following entry filters are equivalent:

&9#

&"0-9"#

&"1;2;3;4;5;6;7;8;9;0"#

The following entry filters are equivalent:

&a#

&"a-z;A-Z"#

The following entry filters are equivalent:

&A#

&"A-Z"#

Placeholders

The number sign (#) is the only placeholder. You use one number sign for each character the user can enter in the field.

For example, the following entry filter allows the user to enter letters in four places:

&a####

The following entry filter allows the user to enter uppercase letters in three places, followed by numbers in two places:

&A###&9##

If you show no placeholders, the filter code allows any number of characters. The following entry filter allows the user to enter only numbers, but it does not limit the length of the entry:

&9

You can set the maximum number of characters allowed in an Alpha field in the Structure editor.

Display Characters

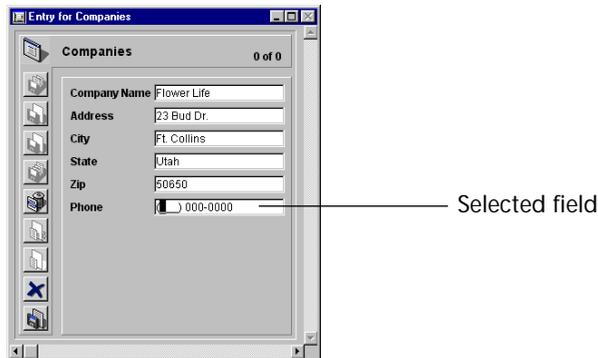
When a field with an entry filter is selected for data entry, 4th Dimension displays an underline () for each placeholder. As the

user types a valid character, each underline is highlighted and replaced with the typed character.

You instruct 4th Dimension which character to substitute for the underline by beginning the entry filter with an exclamation point (!) and the character you want.

You can substitute any character for the underline. For example, if you display “XXXX” and the user types only two of the allowed characters (say they are “AA”), the field will contain “AAXX” when the record is saved.

The following illustration shows a selected field displaying underlines and zeros.



Dead Characters

Any characters, punctuation marks, and spaces can be used as dead characters. Dead characters are displayed during data entry, but they are skipped over by the insertion point and are not entered as part of the data.

The characters you want to use as dead characters are placed before, after, and between placeholders. They are displayed during data entry for clarity.

The phone number entry filter (&9(###) ###-####) uses parentheses, a space, and a dash as dead characters. After you enter a digit immediately preceding a dead character, the insertion point moves directly to the first character following the dead character. The following figure shows how the insertion point skips over the parenthesis and space

after the area code to allow the user to enter the next digit in the phone number.



Custom Entry Filter Formats

You can use a custom format to enter an entry filter. All custom formats are automatically displayed on the Entry Filter drop-down list. To use a custom format as an entry filter, choose its name from the Entry Filter drop-down list or type a vertical bar followed by the format name in the Entry Filter Display area. For example, the entry:

|Part Number

installs the custom format named Part Number as the entry filter for the field.

For information about creating custom formats, see [“Creating Custom Display Formats and Entry Filters”](#) which follows immediately.

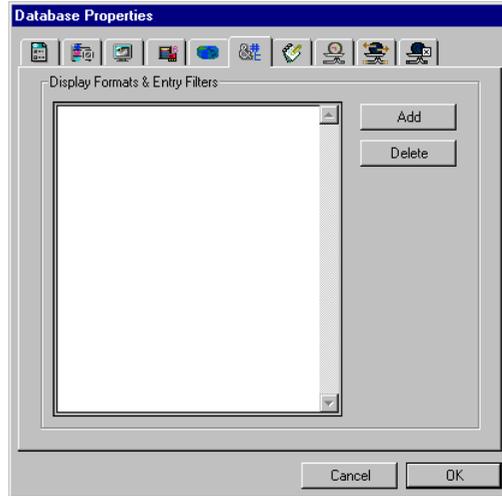
Creating Custom Display Formats and Entry Filters

You can create custom display formats and entry filters that you can refer to by name. You can use a custom format or filter name in place of the code for display formats and entry filters. Custom formats and filters are useful when you use the same display formats or entry filters in several places. If you use fields with the same entry filter in several forms, you can create the entry filter once and specify it by name wherever you need it. In addition, if you decide to change a format or filter, you need only change it in one place and it is updated wherever it is used.

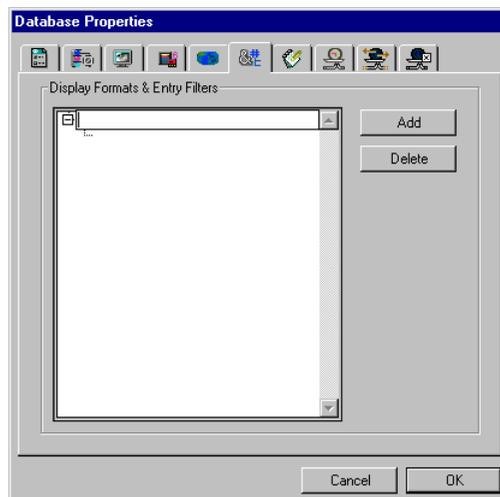
You can also create display formats that correspond to the entry filters and use styles to install them as well.

You create a style in the Display Formats and Entry Filters page of the Database Properties dialog box.

- ▶ To create a custom format or filter:
 - 1 Choose Database Properties from the File menu.
4th Dimension displays the Database Properties dialog box.
For information about the Database Properties dialog box, see the section [“Setting Database Properties” on page 82](#).
 - 2 Click the Display Formats and Entry Filters tab.
The Formats and Filters page appears.



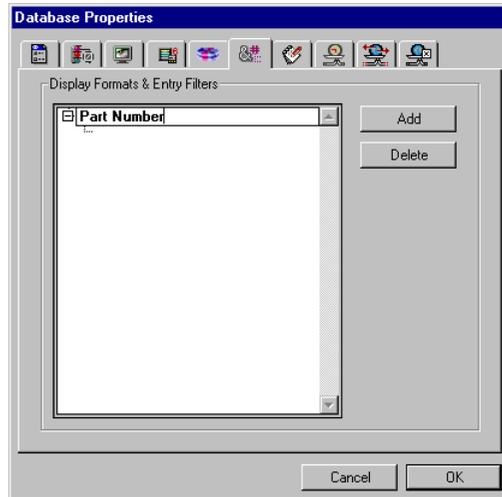
- 3 Click Add.
A blank item is added to the hierarchical list.



4 Type the format or filter name.

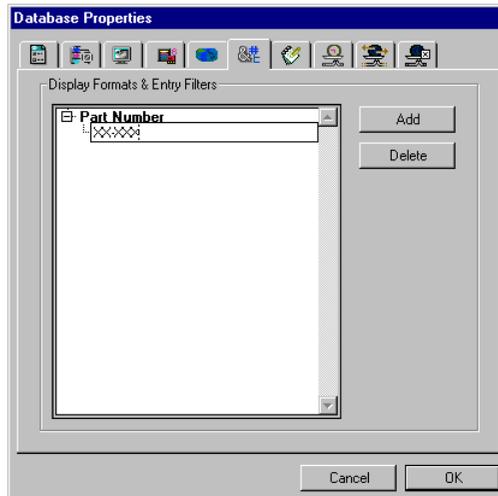
Note To edit the name of a filter or format that already exists, press the Ctrl (Windows) or Command (Mac OS) key and click the name you want to edit.

You can use up to 255 characters.



5 Click the subitem area below the name, hold down the Ctrl key (Command key on Macintosh) and type the display format or entry filter.

You create a display filter or an entry filter just as if you were typing it into the Data Control page of the Object Properties window.



For information about creating display formats and entry filters, see the sections [“Data Entry Controls” on page 331](#) and [“Using Entry Filters” on page 335](#).

For example, if you wanted to create a format for a local telephone number, you would use the following:

```
###-####
```

For example, if you wanted to create a Part Number entry filter for a part number with the format XA-654-1, you would use the following filter:

```
!X&"A-Z"##-!0&"0-9"###-#
```

and the corresponding display format is “##-###-#”.

- 6 If you want to create another filter or format, click the **Add** button. You most often create filters and formats in pairs — one for the entry filter and the other for the display format.
- 7 When you have finished adding filters and formats, click another tab to set other database preferences or the **OK** button. You can edit any filter or format by selecting it and changing the name or the code. You can delete any style by selecting it and clicking the **Delete** button.

Tip Include the word “filter” or “format” in the name to indicate its purpose.

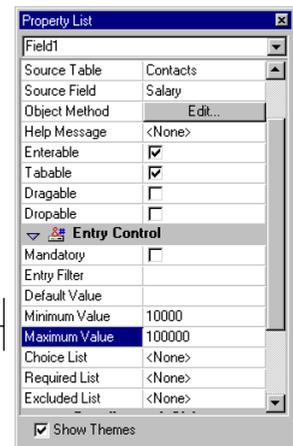
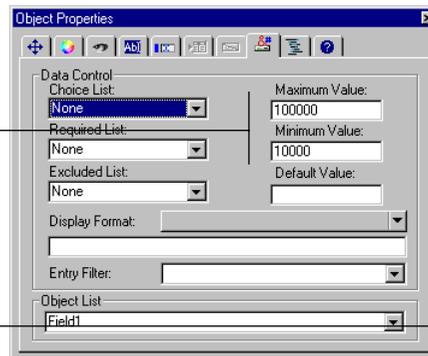
Setting Maximum and Minimum Values

You can restrict a Number, Date, or Time field or enterable object by entering maximum and minimum values in the corresponding entry areas in the Data Control page of the Object Properties window.

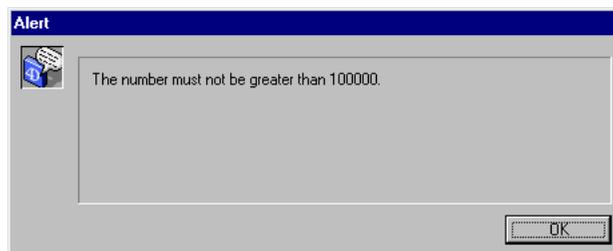
Object Properties window

Property List

Minimum and maximum value entry areas



During data entry, if the user enters a value below the minimum or above the maximum, a warning message is displayed. 4th Dimension returns the user to the field so that a valid entry can be made.



To set a maximum or minimum value, type the value you want to define the limit. Use the data entry format appropriate for the type of field or enterable object for which you are setting a limit. For example, for a Date field or object, use the date entry format to set the maximum or minimum value.

The values you set are inclusive. That is, if the user enters the same value you have set as a maximum or minimum value, the entry is

allowed. Only entries lower than a minimum or higher than a maximum are disallowed. For example, if the value you set as a maximum is 15, the user can enter 15, but not 16.

4D Server Setting a maximum or minimum value changes the maximum or minimum value for all users.

You can also use methods to restrict the values that the user can enter. With a method, you can give more precise and informative feedback to the user or set minimum or maximum values based on other values in the database. For example, a method can check a customer's credit limit before validating a new transaction.

You can also use a required choice list to create unusual ranges of allowable values. For more information, see the sections [“Required Lists” on page 335](#) and [“Creating Lists” on page 588](#).

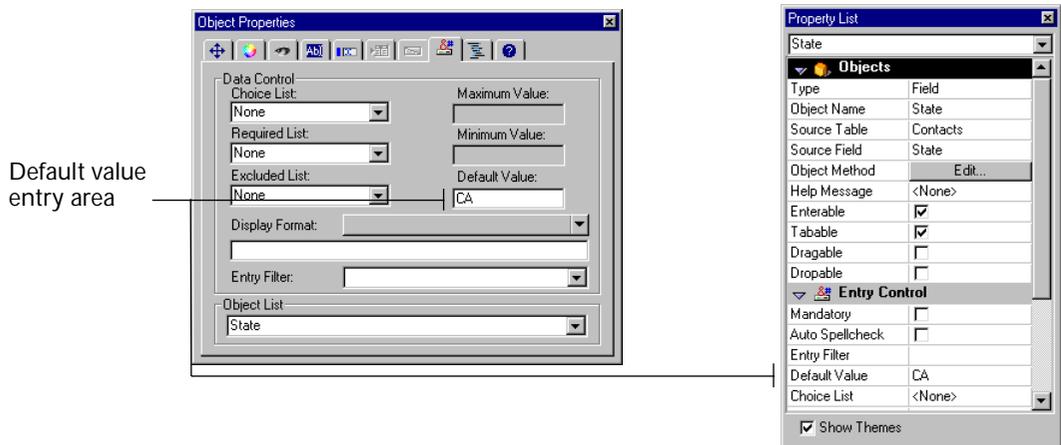
Setting Default Values

You can assign a default value to be entered in a field or enterable object. The default value is entered when a new record is first displayed. You can change the value unless the field or entry area has been defined as non-enterable.

You create a default value by typing the value you want in the Default Value entry area in the Data Control page of the Object Properties window. The default value must be appropriate for the field type.

Object Properties window

Property List



4th Dimension provides stamps for generating default values for date, time, and sequence number. The date and time are taken from the system date and time. 4th Dimension automatically generates any sequence numbers needed. The table below shows the stamp to use to generate default values automatically.

Stamp	Meaning
#D	Current date
#H	Current time
#N	Sequence number

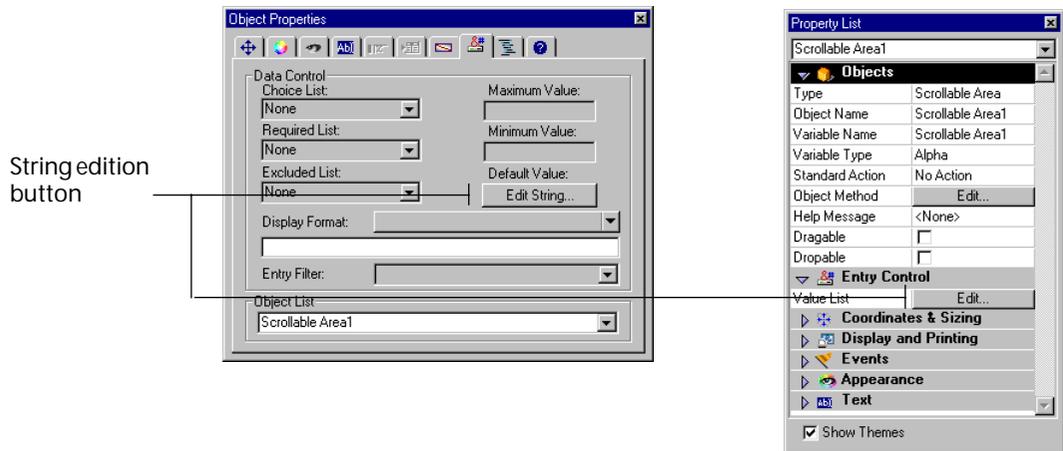
You can use a sequence number to create a unique number for each record. A sequence number is an integer (whole number) that is generated for each new record. The numbers start at one (1) and increase incrementally by one (1). A sequence number is never repeated even if the record it is assigned to is deleted from the table. Each table has its own set of sequence numbers.

Default Lists of Values

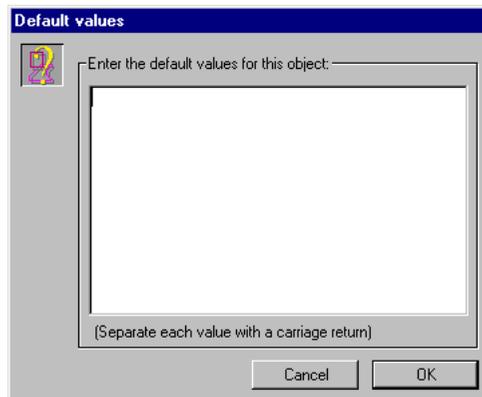
If the object displays a list of values (such as a combo box, scrollable area, pop-up menu, tab control, or drop-down list), you can specify a list of values that will be used as default values. The list will be loaded into the object prior to its being displayed on the form. For objects that accept a list of default values, the Default Value area becomes a button:

Object Properties window

Property List



When you click the string edition button, the Default Values dialog box appears.



Enter the list of default values. Each value should be on a separate line. Click OK to put away the Default Values dialog box and return to the Object Properties window.

When you enter default values into the Default Values dialog box, the values are automatically loaded into an array whose name is the name of the object. Using the language, you can manage the object by referring to that array.

4D Server Setting a default value in the Object Properties window sets the default value for all users.

Using a List to Set Default Values

If the object is a hierarchical list or a tab control, you can use a list that you created using the List editor to set default values.

- ▶ To set default values using a list:
 - 1 Display the object properties in either the data control page of the Object Properties window or the Property List.
This action is described in the section [“Displaying and Setting Form and Object Properties” on page 244](#).
 - 2 Choose the desired list from the Choice List drop-down list.
The user can overwrite an item in the list by Ctrl+clicking (Command-clicking on Macintosh), but the changes are not saved to the actual list.

Setting Default Values Using the Language

You can also set default values using a method. For objects that accept one value, you can assign the default value when the On Load event executes in the object or form method. For objects that accept lists, you can enter the default values using the List editor and then load the contents of the list into an array using the LIST TO ARRAY command. You can load the lists into array when the On Load event occurs or load all lists into arrays when the On Startup database event occurs.

Adding a Scroll Bar to a Text Object

Text fields and enterable objects can contain up to 32,000 characters. 4th Dimension allows you to attach a scroll bar so that the user can

scroll the information. The figure below shows an input form with a text area with a scroll bar.



- ▶ To add a scroll bar to a text object:
 - 1 Display the object properties for the text object in either the Data Control page of the Object Properties window or the Property List. This action is described in the section [“Displaying and Setting Form and Object Properties”](#) on page 244.
 - 2 In the Object Properties window, click the Display tab
OR
In the Property List, expand the Display and Printing theme.
 - 3 Check the Vertical Scroll Bar option.

Note If a text field or enterable object does not have a scroll bar, the user can scroll the information using the arrow keys.

Adding Help to a Field or Object

You can add Balloon Help or a Tip to fields and active objects in your forms to help users work with your database more productively. Depending on your configuration help messages will either appear in tips or help balloons.

Balloon Help can be viewed only when the database is used on a Macintosh running System 7 (or greater), with Balloon Help turned

on. However, you can create Balloon Help even when you are designing a database on Windows. Tips are displayed on all operating systems.

For example, you can create a help message for a Date field which reminds the user to include a separator such as the slash mark (/) between the month, day, and year when entering data. The tip appears in the form whenever the field or object is used.

You add help to a field or object in the Help page of the Object Properties window. The help message appears only in this particular form. For information about adding Balloon Help to a field in all forms in which it appears, see the section [“Choices and Help” on page 147](#).

The addition of existing help messages to objects can be performed in both the Object Properties dialog or the Property List. However the creation or edition of Help messages can only be performed in the Object Properties window.

Dynamic References

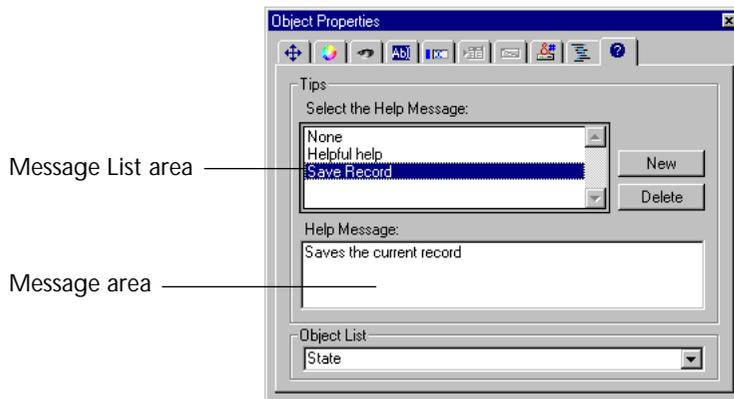
You can insert dynamic contents in a help message. The following dynamic elements can be inserted in a help message:

- a STR# resource reference: the syntax to apply is “:16000,2” where 16000 is the resource number and 2 is its element.
- a table or field label: the syntax to apply is <?[TableNum]FieldNum> or <?[TableName]FieldName>. For more information, refer to [“Inserting Dynamic Table and Field Names” on page 329](#).
- a variable or a field: the syntax to apply is <VariableName> or <[TableName]FieldName>. The current value of the field or variable will be displayed in the help message.

You can, for example, enter the following text in a help message: “Enter <[Family] First_Name>’s age in this area.” When in user environment, 4th Dimension will replace the field reference by the current value for the First_Name field.

- ▶ To add help message to a field or object:
 - 1 Click the Help tab in the Object Properties window.

The Help page appears.



The list of messages contains the names of all current help messages.

- 2 Click New to create a new help message.

OR

Click an existing help message to edit it.

The message list is displayed in alphabetical order. The names of any new help messages are added at the beginning of the message list, before the list is sorted.

- 3 Enter or edit the message in the Help Message area.

The help message appears as a Balloon help on a Macintosh with Balloon Help turned on and as a Tip on any operating system.

4th Dimension stores the text for the help message so that you can reuse the message for other fields and objects.

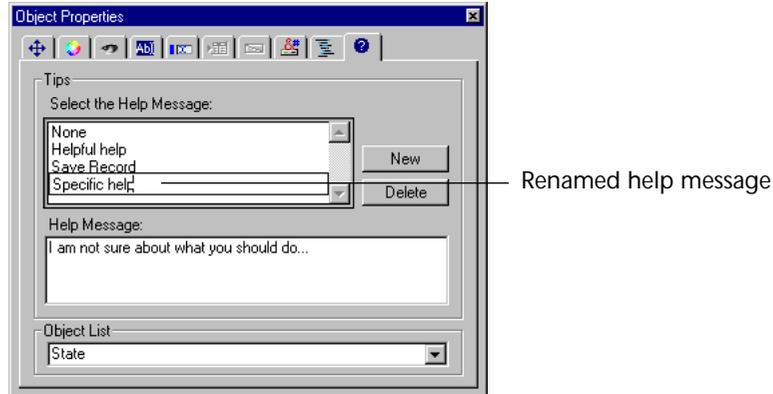
Editing a Help Message To edit an existing help message:

- 1 Click the name of the message you want to modify in the Message List area.
- 2 Edit the help message.
4th Dimension stores your changes to the help message.

Changing the Name of a Help Message To change the name of any help message:

- 1 Display the Object Properties window for the desired field or active object and click the Help tab.
The Help Message page appears.

- 2 Hold down the Ctrl key (on Windows) or Command key (on Macintosh) and click the name of the message in the Message List area.
The name of the message becomes editable.
- 3 Type the new help message name.



If necessary, the list is reordered to maintain the alphabetical order.

Deleting a Help Message To delete a help message so it no longer appears in the list of messages:

- 1 Click the name of the message you want to delete in the list of messages.
The message name is highlighted.
- 2 Click the Delete button.
The help message is removed from the list of messages.

Selecting a Help Message Once you have defined the help messages that you want to use, you can assign them to form objects using the Property List or the Object Properties window.

- ▶ To select the help message you want to use:
 - 1 Select the object to which you want to assign a message.
 - 2 In the Help page of the Object Properties window, click the name of the message you want to select in the list of messages.OR
In the Property List, expand the Objects theme and select a message from the Help Message list.
The message you selected appears as Balloon Help and a Tip for the field or object.

To remove the help message, select **None** in the Message list area of the Object Properties window or from the Help Message list in the Property List.

Display Formats

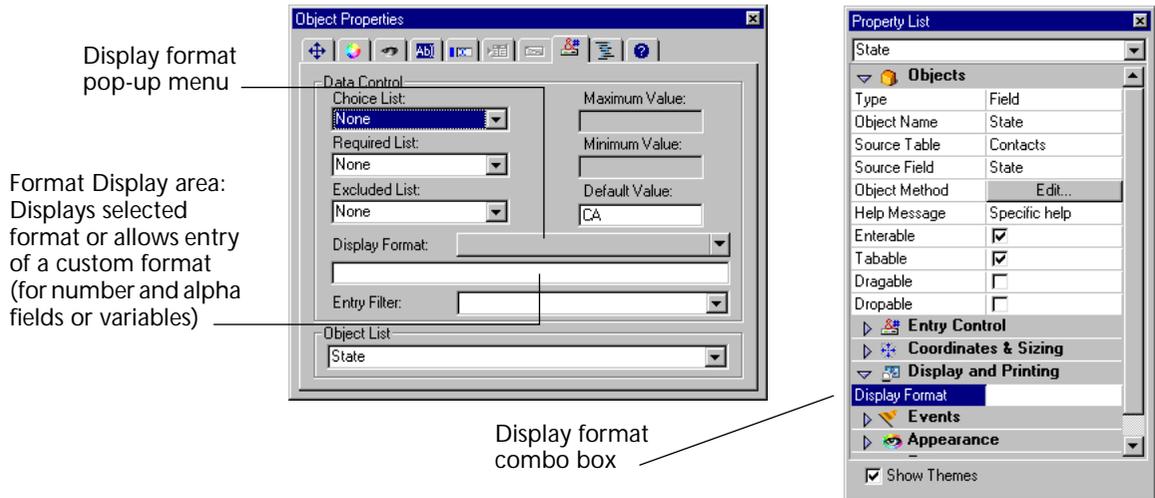
The display formats provided by 4th Dimension give you many choices for screen display and printing. Display formats can be applied to both fields and enterable or non-enterable areas (variables). The format you use to display the contents of a field does not affect the actual value stored by 4th Dimension.

The display format for a field can be different in each form. For example, you may want to show a value without dollar signs in an input form and display it with dollar signs in an output form.

You set display formats in Data Control page of the Object Properties window.

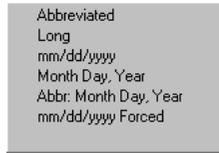
Object Properties window

Property List



Different formats appear in the Display Format list depending on the type of field you select. The built-in formats always appear. Any display formats that were added using the Formats and Filters editor appear in the pop-up menu along with the standard 4th Dimension formats.

Date Field Formats Date formats control the way dates appear when displayed or printed. For data entry, you enter dates in the *MM/DD/YYYY* format, regardless of the display format you have chosen. The figure below shows date formats in the Format pop-up menu.



The table below shows the Date field display formats and gives an example of each format.

Choice	Example
Short	3/25/99
Abbreviated	Wed, Mar 25, 1999
Long	Wednesday, March 25, 1999
mm/dd/yyyy	03/25/1999
Month Date, Year	March 25, 1999
Abbr: Month Date, Year	Mar 25, 1999
mm/dd/yyyy Forced	03/25/1999 ¹

1. The default century is the 20th century, it can be set using the SET DEFAULT CENTURY command

Note Unlike the Alpha and Numeric display formats, the Date display format must be one of the formats shown in the Format pop-up menu.

Time Field Formats Time formats control the way times appear when displayed or printed. For data entry, you enter times in the 24-hour *HH:MM:SS* format or the 12-hour *HH:MM:SS AM/PM* format, regardless of the display format you have chosen. The figure below shows time formats in the Format pop-up menu.



The table below shows the Time field display formats and gives examples.

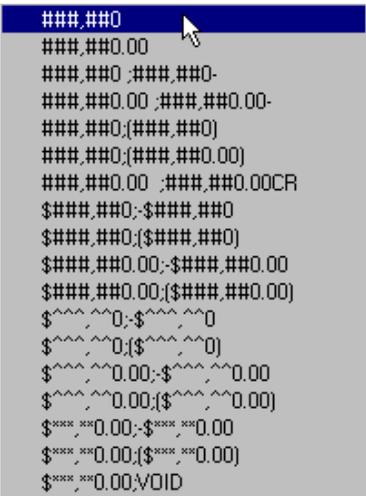
Choice	Example
HH:MM:SS	02:15:34
HH:MM	02:15
Hour Min Sec	2 hours 15 minutes 34 seconds
Hour Min	2 hours 15 minutes
H:MM AM/PM	2:15 AM

Note Unlike Alpha and Numeric display formats, the Time display format must be one of the formats shown in the Format pop-up menu.

Number Field Formats

Number formats control the way numbers appear when displayed or printed. For data entry, you enter only the numbers (including a decimal point or minus sign if necessary), regardless of the display format you have chosen. Number fields include Real, Integer, and Long Integer fields.

The following illustration shows the number formats in the Format pop-up menu.



You can choose the format from the pop-up menu or type it in the Format Display area. You can edit any number format in the Format Display area.

Creating a Custom Number Format

A number field can use any format including a custom format you create with the symbols you see on the Display pop-up menu. Creating a custom number display format is discussed in the following sections.

In each of the number display formats, the number sign (#), zero (0), caret (^), and asterisk (*) are used as placeholders. You create your own number formats by using one placeholder for each digit you expect to display.

For example, if you want to display three numbers, you could use the format ###. If the user enters more digits than the format allows, 4th Dimension displays <<< in the field to indicate that more digits were entered than the number of digits specified in the display format.

If the user enters a negative number, the leftmost character is displayed as a minus sign (unless a negative display format has been specified). If ##0 is the format, minus 26 is displayed as -26 and minus 260 is displayed as <<< because the minus sign occupies a placeholder and there are only three placeholders.

Note No matter what the display format, 4th Dimension accepts and stores the number entered in the field. No information is lost.

Each placeholder character has a different effect on the display of leading or trailing zeros. A *leading zero* is a zero that starts a number before the decimal point; a *trailing zero* is a zero that ends a number after the decimal point.

Suppose you use the format ##0 to display three digits. If the user enters nothing in the field, the field displays 0. If the user enters 26, the field displays 26.

The table below explains the effect of each placeholder on leading and trailing zeros.

Placeholder	Effect for leading or trailing zero
#	Displays nothing
0	Displays 0
^	Displays a space ¹
*	Displays an asterisk

1. The caret (^) generates space character that occupies the same width as a digit in most fonts.

Decimal Points and Other Display Characters

You can use one decimal point in the format. If you want the decimal to display whether or not the user types it in, it must be placed between zeros.

You can use any other characters in the format. When used alone, or placed before or after placeholders, the characters always appear. For example, if you use the following format:

```
$##0
```

a dollar sign always appears because it is placed before the placeholders.

If characters are placed between placeholders, they appear only if digits are displayed on both sides. For example, if you define the format:

```
###,##0
```

a comma appears only if the user enters at least four digits.

Spaces are treated as characters in number display formats.

Formats for Positive, Negative, and Zero

A number display format can have up to three parts allowing you to specify display formats for positive, negative, and zero values. You specify the three parts by separating them with semicolons as shown below:

```
Positive;Negative;Zero
```

You do not have to specify all three parts of the format. If you use just one part, 4th Dimension uses it for all numbers, placing a minus sign in front of negative numbers. If you use two parts, 4th Dimension uses the first part for positive numbers and zero and the second part for negative numbers. If you use three parts, the first is for positive numbers, the second for negative numbers, and the third for zero.

Note The third part (zero) is not interpreted and does not accept replacement characters. If you enter ###;###;# the value zero will be displayed "#". In other words, what you actually enter is what will be displayed for the value zero.

Here is an example of a number display format that shows dollar signs and commas, places negative values in parentheses, and does not display zeros:

```
$###,##0.00;($###,##0.00);
```

Notice that the presence of the second semicolon instructs 4th Dimension to use nothing to display zero.

The following format is similar except that the absence of the second semicolon instructs 4th Dimension to use the positive number format for zero:

`$###,##0.00;($###,##0.00)`

In this case the display for zero would be \$0.00.

Scientific Notation

If you want to display numbers in scientific notation, use the ampersand (&) followed by a number to specify the number of digits you want to display. For example, the format

`&3`

would display 759.62 as

`7.60e+2`

Note The scientific notation format is the only format that will automatically round the displayed number. Note in the example above that the number is rounded up to `7.60e+2` instead of truncating to `7.59e+2`.

Hexadecimal Formats

You can display a number in hexadecimal using the following display formats:

- `&x` This format displays hexadecimal numbers using the “0xFFFF” format.
- `&$` This format displays hexadecimal numbers using the “\$FFFF” format.

Displaying a Number as a Time

You can display a number as a time (with a time format) by using “&/” followed by a digit. Time is determined by calculating the number of seconds since midnight that the value represents. The digit in the format corresponds to the order in which the time format appears in the Format drop-down menu.

For example, the format:

`&/5`

corresponds to the 5th time format in the pop-up menu, specifically the AM/PM time. A number field with this format would display 25000 as:

`6:56 AM`

Format Entered	Positive	Negative	Zero
####	1234	<<<<	
#####	1234	-1234	
#####.##	1234.5	-1234.5	
####0.00	1234.50	-1234.50	0.00
#####0	1234	-1234	0
+#####0;-#####0;0	+1234	-1234	0
#####0DB;#####0CR; 0	1234DB	1234CR	0
#####0 ;(#####0)	1234	(1234)	0
###,##0	1,234	-1,234	0
##,##0.00	1,234.50	-1,234.50	0.00
^^^ ^^	1234	-1234	
^^^ ^^0	1234	-1234	0
^^,^^0	1,234	-1,234	0
^^,^^0.00	1,234.50	-1,234.50	0.00
*****	***1234	**_1234	*****
*****0	***1234	**_1234	*****0
,0	***1,234	*_1,234	*****0
,0.00	*1,234.50	-1,234.50	*****0.00
\$/,**0.00;-\$**,**0.00	\$1,234.50	-\$1,234.50	\$****0.00
\$^^^^0	\$ 1234	\$-1234	\$ 0
\$^^^^0;-\$^^^^0	\$1234	-\$1234	\$ 0
\$^^^^0 ;(\$^^^^0)	\$1234	(\$1234)	\$ 0
\$^,^^0.00 ;(\$^,^^0.00)	\$1,234.50	(\$1,234.50)	\$ 0.00
&2	1.2e+3	-1.2e+3	0.0e+0
&5	1.23450e+3	-1.23450e+3	0.00000

Alpha Field Formats Alpha formats control the way the alphanumeric characters appear when displayed or printed. You can choose a format from the Format list or type a display format into the Format Display area. After choosing a format from the pop-up menu, you can edit it in the Format Display area.

The figure below shows an alpha format being chosen from the format pop-up menu.



The Format pop-up menu contains formats for some of the most common alpha fields that require formats: US telephone numbers (local and long distance), Social Security numbers, and zip codes.

The following table shows the Alpha field formats and the types of fields they are typically used for.

Format	Field type
###-####	Telephone number (local)
(###) ###-####	Telephone number (long distance)
###-###-####	Telephone number (long distance)
###-##-####	Social Security number
00000	Zip Code

You can also enter and edit a format in the Format Display area. You can edit a format after choosing it from the Format pop-up menu, but you can edit a style only in the Styles editor.

The number sign (#) is the placeholder for an Alpha field display format. You can include the appropriate dashes, hyphens, spaces, and any other punctuation marks that you want to display. You use the actual punctuation marks you want and the number sign for each character you want to display.

For example, consider a part number with a format such as:

RB-1762-1

The alpha format would be:

##-####-#

When the user enters “RB17621,” the field displays:

RB-1762-1

The field actually contains “RB17621.”

If the user enters more characters than the format allows, 4th Dimension displays the last characters. For example, if the format is:

(###)

and the user enters “HAPPY,” the field displays:

(PPY)

The field actually contains “HAPPY.” 4th Dimension accepts and stores the entire entry no matter what the display format. No information is lost.

Custom Alpha Formats

You can use a custom format for Alpha fields. Any custom formats that you have created are automatically added to the Format pop-up menu. You can choose a custom format just as you would choose a built-in format.

For information about creating custom formats, see the section [“Creating Custom Display Formats and Entry Filters”](#) on page 344.

Boolean Field Formats

Boolean fields can contain one of two values: TRUE or FALSE. A Boolean field can be displayed as either a pair of radio buttons or as a check box.

If you do not create a display format for a Boolean field, 4th Dimension automatically displays the field as a pair of radio buttons labeled Yes and No.

You use the Format Display area in the Data Control page to create other labels for radio buttons or a check box with any label. If you want to display only the button labels and not the field name, you can delete the field label in the form.

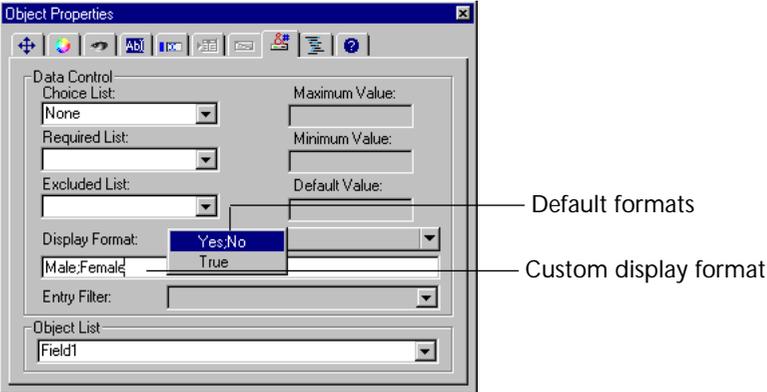
Formatting a Boolean Field as Radio Buttons

You can define a format for a Boolean field using the Object Properties window or the Property List.

- Using the Object Properties window:

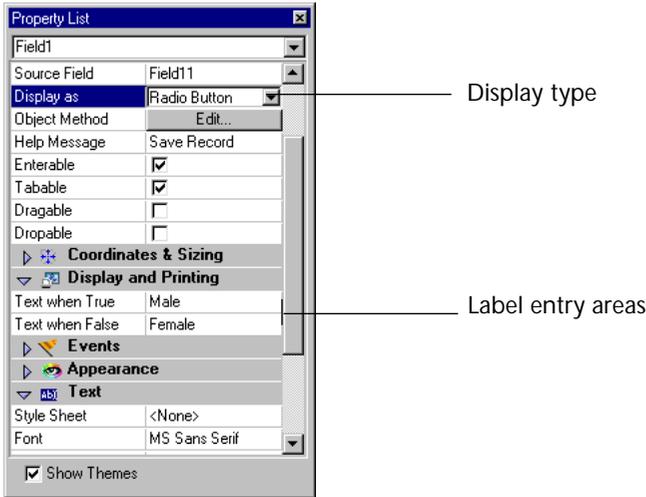
Entering two labels separated by a semicolon (;) in the Format Display area. For example, a field for sex could have one button named Male

and one button named Female. To create these buttons, you would type, “Male;Female.”



■ Using the Property List:

In the Property List, the appearance of Boolean fields is set using the Display as list, located in the Objects theme. Once you have selected Radio button from that list, you can enter the labels in the Text when True and Text when False entry areas.



The buttons are displayed in the Form editor side by side as shown below.



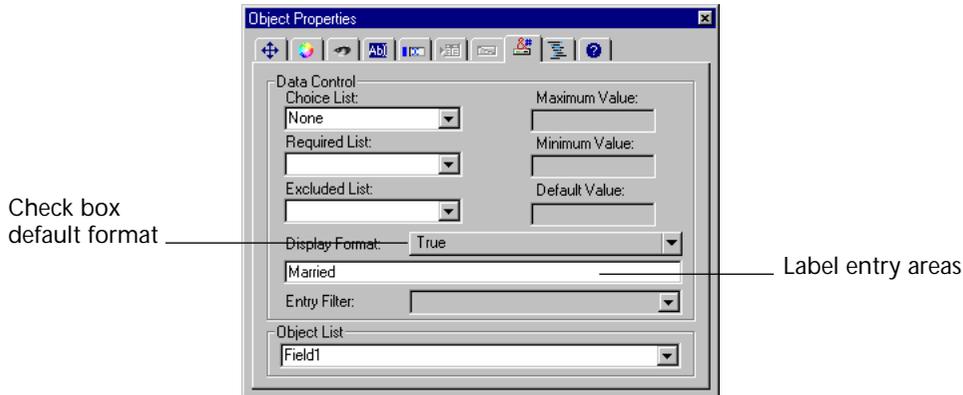
If you use labels with different first letters, you can select the radio button by typing the first letter during data entry. For example, you can press “M” to select Male or “F” to select Female when the field is selected.

The following rules apply when the field is being used for data storage: if the first button is selected, the field is true; if the second button is selected, the field is false. The field is false by default.

Formatting a Boolean Field as a Check Box

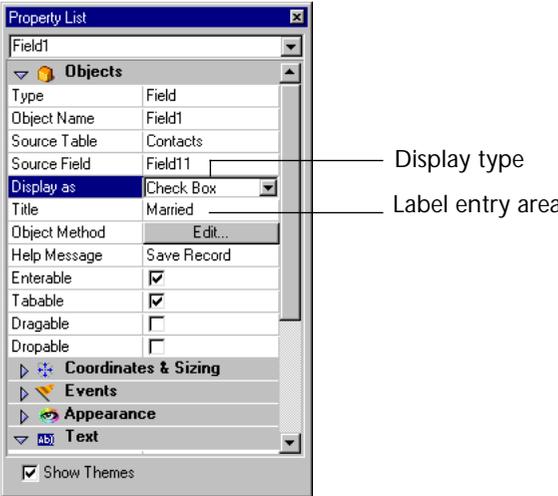
You can set a Boolean field to be displayed as a check box using either the Object Properties window or in the Property List.

- Using the Object Properties window
Enter a single label in the Format Display area. For example, a field for Paid could have one check box labeled Yes. To create this check box you would enter “Yes.” To create a check box with no label, enter a space in the Format Display area. The following illustration shows how a Boolean field is formatted as a check box.



- Using the Property List
Choose Check box from the Display as list located in the Objects theme. Once you have selected that option, an entry area labeled Title

is displayed below the list. This is the entry area in which you enter the label of the check box. The default label is the field's name



The following rules apply when the field is being used for data storage: if the check box is selected, the field is True; if the check box is deselected, the field is False. The field is False by default.

In the User environment, this field is displayed as a check box:



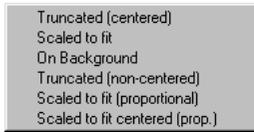
You can format a Boolean field as a check box with no label by entering a space as the display format in the Object Properties window or in the Title area in the Property List. In this case, you add the label for the check box as a separate object. Using that method you can insert dynamic references in the label. For more information about dynamic references, refer to [“Inserting Dynamic Table and Field Names” on page 329](#). In the following example, the Boolean field is on top of the text object.



Picture Field Formats

Picture field formats control how pictures appear when displayed or printed. For data entry, the user always enters pictures by pasting them from the Clipboard, regardless of the display format.

The figure below shows the picture formats available in the Picture Format list of the Property List or in the Display Format list of the Object Properties window.



The truncation and scaling options do not affect the picture itself. The contents of a Picture field is always saved. Only the display on the particular form is affected by the picture display format.

Truncated (Centered and Non-centered)

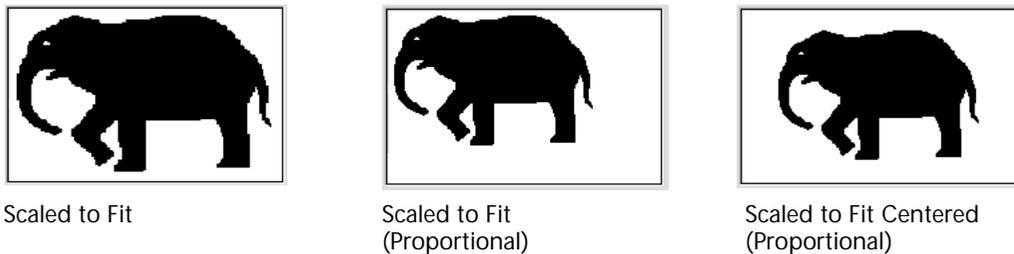
The Truncated (Centered) format causes 4th Dimension to center the picture in the field and crop any portion that does not fit within the field area. 4th Dimension crops equally from each edge and from the top and bottom.

The Truncated (Non-centered) format causes 4th Dimension to place the upper-left corner of the picture in the upper-left corner of the field and crop any portion that does not fit within the field area. 4th Dimension crops from the right and bottom. The figure below compares the Truncated Centered and Non-centered formats.



Scaled to Fit

The Scaled to Fit formats cause 4th Dimension to resize the picture to fit the dimensions of the field area. The figure below compares the three Scaled to Fit formats.



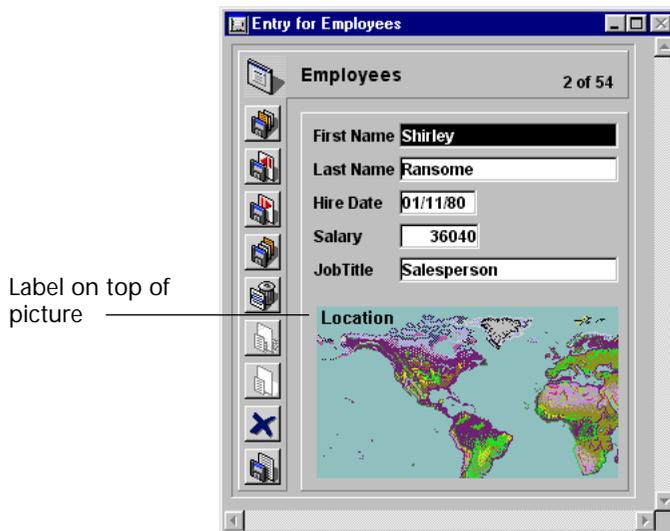
When you use Scaled to Fit (Proportional), the picture is reduced proportionally on all sides to fit the area created for the picture. The Scaled to Fit Centered (Proportional) option does the same, but centers the image in the picture area.

If the picture is smaller than the area defined in the form, the picture will not be modified.

If the picture is bigger than the area defined in the form, the picture is proportionally reduced. Since the picture is proportionally reduced, it will not appear distorted.

On Background

On Background makes the picture transparent. Any objects placed behind the graphic such as fields or variables are visible through the graphic. When a Picture field is in this format, the user can move the picture around the inside of the Picture field by dragging it. 4th Dimension remembers the object's position on the background. The figure below shows a form that includes a picture with the On Background format.

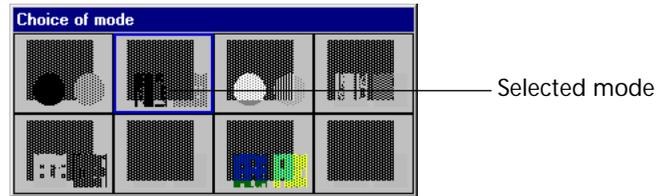


Note If you are printing pictures with the On Background format, they will be printed as bitmaps.

Setting the Choice of Display Mode

When a picture is displayed using the On Background setting, you can select the mode that specifies the interaction between the background and foreground colors. To display the dialog that allows you to select

the desired mode, double-click the picture area in the User environment:



The currently selected color mode is outlined by a blue line. To select a different mode, click it.

If your intent is to use the picture only as a background object for the form (not as data), you can instead paste the picture into the Picture library and then add the picture to the form by dragging. For more information, see [“Picture Library” on page 43](#) and [“Adding Fields to a Blank Page” on page 319](#).

Active Objects on a Form

This section describes the active objects you can use on a form to control database and interface functions. Active objects include buttons, radio buttons, check boxes, tab controls, pop-up menus, drop-down lists, hierarchical pop-up menus and hierarchical lists, combo boxes, scrollable areas, splitters, gauges, and plug-ins. Enterable objects (variables) are treated much the same as fields and are described in the previous section.

This section explains how to create and modify active objects other than fields and enterable objects. The different active object types are discussed in detail in the section, [“Types of Active Objects” on page 381](#).

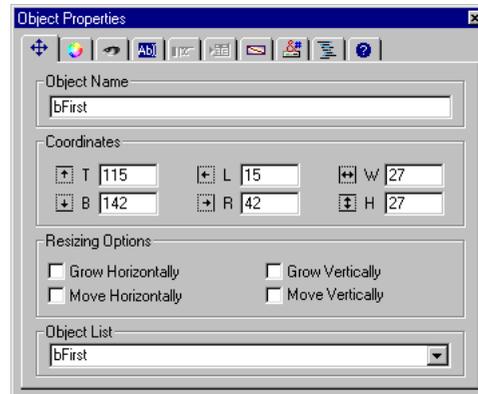
Setting Object Properties

An object’s properties can be modified using the Object Properties window or using the Properties List. These two windows allow you to set numerous options.

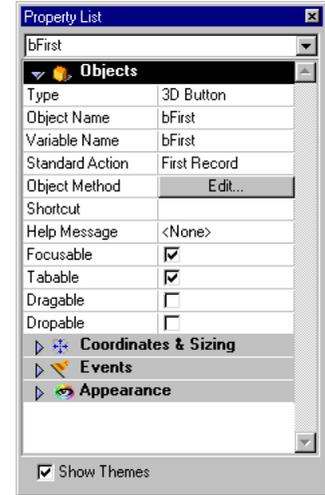
- ▶ To modify the properties of an active object:
 - 1 Select the object that you want to modify and double-click it. Depending on the current display settings, the Object Properties window or the Property List is displayed.

You can use either window to change the object's properties. You can also select another object to display its properties. You can also select several objects to edit their common properties.

Object Properties window



Property List



Note If the object is grouped with other objects, you first need to ungroup it before you can display its properties.

2 Apply your modifications.

Your modifications take effect immediately.

You can name an object, specify its type, define its action, drag and drop properties, resizing or repositioning options, platform interface and appearance, and attach a method for the object.

The Object Properties window and the Property List allow you to set the following properties:

- **Object and Variable** Each active object is associated with a variable. The variable name can be different from the object's name. When using 4th Dimension's language, you can refer to an active form object by its variable name or object name.

You can set the object type in both the Property List and the Object Properties window but the variable type can be set only in the Property List.

You can also assign a standard automatic action to the object. For more complex actions you can write an object method.

- **Attributes** Each object is assigned standard attributes that define how it behaves on the form. This includes, for instance, drag and drop management and the display of the focus. Some objects such as pictures may also have specific attributes.
- **Entry control** For each enterable object you can define entry controls such as entry filters, list of acceptable or unacceptable values, and so on.
- **Coordinates** You can set the size, location, the resizing options, and the positioning options for each object.
- **Colors and Appearance** You can individually set the foreground and background colors, the style, the platform interface, and the appearance of object.
- **Font** For each object that includes some text or a label, you can define the font it uses as well as its alignment. You can also use a style to set font attributes.
- **Subform** These are considered as active form objects. They have a specific set of properties.
- **Events** You can set a list of events for which the object's method will be executed.
- **Help messages** As with fields, you can assign a tip or help balloon to any active object in a form.

Creating an Active Object

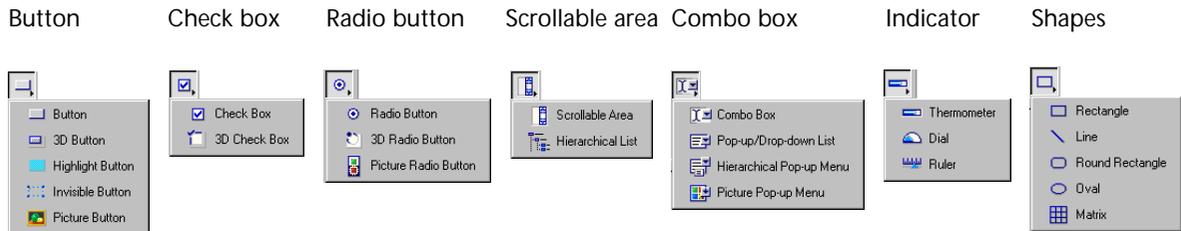
Active objects are created using the Form editor's Tools palette. When using the Tools palette to create an active object you can do it by drawing the object with the tool or by dragging and dropping the tool itself.

The Active objects tools are located in the upper part of the Tools palette.



Using this palette you can create numerous types of active objects: text variables, tab controls, any type of button, check boxes, radio buttons, pop-up menus and drop-down lists, hierarchical menus, combo boxes, scrollable areas, indicators, picture menus, button grids, splitters, and plug-in areas.

Some objects are grouped by types; the subtypes appear in drop-down lists associated with a button in the Tools palette. These drop-down lists are shown below.



For more information about the contents and the use of the Tools palette, refer to [“The Tools Palette” on page 235](#).

► To create an active object in a form:

- 1 Open the form in which you want to create the active object.

For more information on how to open a form, refer to [“Opening a Form in the Form Editor” on page 250](#).

- 2 (Optional) If the desired object type is in a drop-down list, first choose the object from the drop-down list.

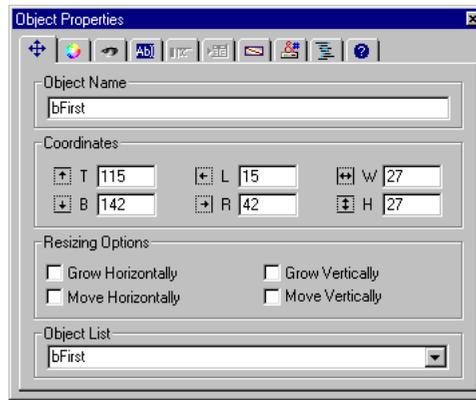
- 3 Select the tool that corresponds to the object you want to create and either drag it onto the form or click it and draw an area on the form.

For more information, refer to [“Creating Objects” on page 276](#).

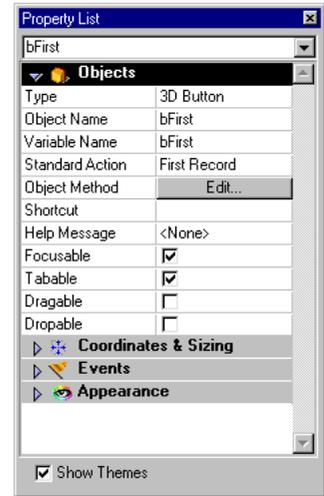
4th Dimension automatically displays the properties of the new object in the Property List or in the Object Properties window (depending on the current settings).

4 Define the object by entering its name or editing the default name.

Object Properties window



Property List



The object's name is used in methods when you refer to that object, so it is necessary to make sure that this name is unique.

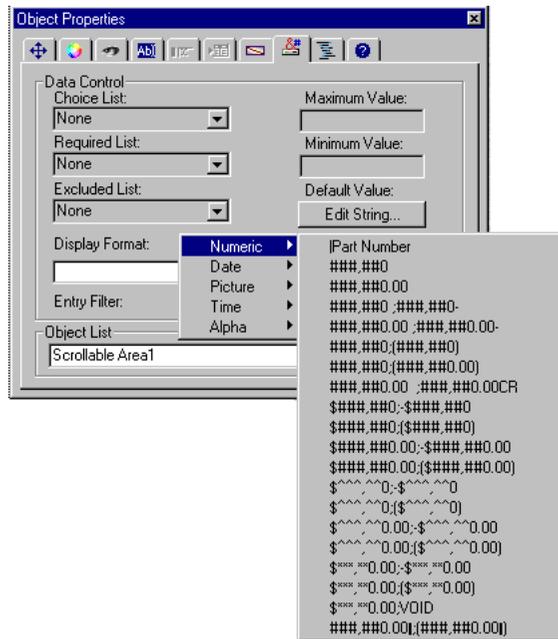
5 Set the object's properties to match your requirements.

For more information about the creation of objects of a particular type, refer to [“Types of Active Objects” on page 381](#).

Display Formats for Objects

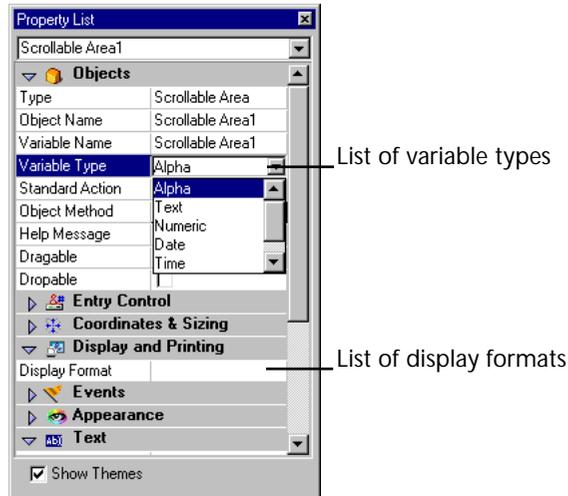
Both the Object Properties window and the Property List provides the same display formats for both enterable and non-enterable objects as it does for fields. The difference is that 4th Dimension does not know what data type is to be displayed or used in the object.

- In the Object Properties window, the Display format pop-up menu is a hierarchical pop-up menu. You choose the data type (Numeric, Alpha, Date, Time, or Picture) and then the display format.



- In the Property List, there is an additional option that lets you select the type of data that will be processed by the variable. By default this

type is Alpha. Once a value is selected the Display format list is updated to reflect the formats available for this type of variable:



The selected format is displayed in the Display Format area. If you choose a numeric or alpha object type, you can edit the format in the usual way. You can type formats for objects directly into the Display Format area. For more information about display formats, see the section [“Display Formats” on page 357](#).

Data Entry Controls for Enterable Objects

The Object Properties window provides data entry controls for enterable objects as well as for fields. These controls allow you to:

- Set an entry filter that defines allowable characters,
- Display a choice list,
- Establish lists of required values or excluded values,
- Set maximum and minimum allowable values,
- Set default values or default lists of values.

These controls work for enterable objects exactly as they work for fields. For complete information, see the appropriate sections earlier in this chapter.

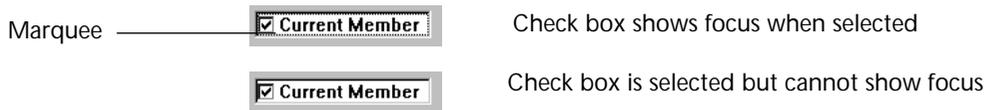
Setting the Tabable and Show Focus Properties

The Tabable and Focusable properties can be set in the Display page of the Object Properties window or in the Objects theme of the Property List. Those two options will affect how the object appears during data entry.

- When the Focusable property is selected for an object, the object is outlined by a gray dotted line when it is selected.
- If an object has the Tabable property, it is included in the data entry order. That is, the user can press the Tab key to select the object.

The Tabable property is only accessible if the Focusable property is selected. That is, any tabable object shows the focus when it is selected. However, some objects can be “focusable” while not being “tabable” (for example, an object can be selected by clicking it and not “tabable”). In this case, the object does not belong to the entry sequence.

For example, the following illustration shows the effect of the Focusable property on check boxes.



Note A non-enterable variable cannot be “tabable”.

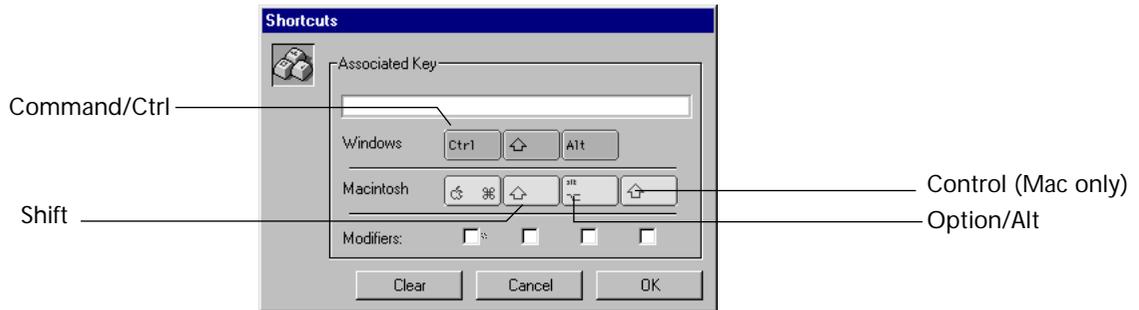
Assigning a Keyboard Equivalent

The Variable page of the Object Properties window allows you to assign a keyboard shortcut for buttons and check boxes. The user can then activate the button or select the check box using the keyboard instead of having to use the mouse.

- ▶ To assign a keyboard shortcut:
 - 1 Click the Keys button in the Variables page of the Object Properties window.
 OR
 In the Property List, click the [...] button in the Shortcuts property located in the Objects theme¹.

1. You may need to click in the Shortcut entry area in the Property List to make the button visible.

The Shortcuts dialog box appears.



2 Type the keyboard shortcut.

For example, if you want to use Ctrl+h, hold down the Ctrl key and press h. The letter h will then appear in the Associated key area and the check box below the Ctrl key will be checked.

You are not required to use modifier keys. You can use any key alone as the shortcut, although this is not recommended in most cases.

If you like, you can manually modify the selection of modifier keys by selecting or deselecting any of the modifier key check boxes.

To start over again, click Clear.

3 When you have finished, click OK.

The Property List displays the keyboard shortcut that was assigned to the object. If you want to change the shortcut later, simply open the Shortcuts dialog box and type the key combination you want to use. 4th Dimension displays the new combination in the Shortcuts dialog box.

Enabling Drag and Drop

Active objects on a form may have drag and drop properties. Two options located in the Display page of the Object Properties window and in the Objects theme of the Property List, are related to drag and drop.

- **Dragable** controls whether the user has the ability to drag the object,
- **Dropable** controls whether the object has the ability to “receive” a dragged object.

If you want to enable drag and/or drop for a particular object, enable the appropriate property. You then must manage the drag and drop action using a method. For more information, see the section on Drag and Drop commands in the *4th Dimension Language Reference*.

Types of Active Objects

4th Dimension provides the following types of active objects (in addition to fields):

- Text variables (enterable and non-enterable),
- Button,
- 3D Button,
- Highlight button,
- Invisible button,
- Picture button,
- Check box,
- 3D Check box,
- Radio button,
- 3D Radio button,
- Picture radio button,
- Scrollable area,
- Hierarchical list,
- Combo box,
- Pop-up menu/Drop-down list,
- Hierarchical menu,
- Picture pop-up menu,
- Tab control,
- Plug-in area,
- Button grid,
- Thermometer,
- Indicator,
- Ruler,
- Splitters.

The sections that follow describe each kind of object in detail.

Enterable and Non-enterable Variables

An enterable variable allows the user to enter a value into a variable and display the value. A non-enterable variable allows you to display the value of a variable. You use methods to manage enterable and non-enterable variables.

Variables are used for temporary storage of data. One common use for a variable is to display calculations that are done using a method such as:

```
vTotal := Quantity * Price
```

You create a variable that displays the result of the calculation, name the object vTotal, and use a method to do the calculation.

An enterable variable accepts data. You can set data entry controls for the object as you would for a field. The entered data is associated with the variable name. You can manage the data with object or form methods using the object's name as a variable.

A non-enterable variable only displays data. The displayed data is associated with the object's name. You control the data with methods, using the object's name as a variable.

To create a non-enterable variable, create a standard variable and deselect the Enterable property. This property can be set in the Variable page of the Object Properties window and in the Options theme of the Property List.

Enterable and non-enterable variables can be of any size. Since they display characters, when the object area is resized, it snaps to a size depending on the object's font size. Variables that contain alphanumeric characters, numbers, dates, times, and pictures can make use of display formats. Text objects can use a scroll bar and can be printed with variable frame. Picture variables can be scaled or truncated.

Buttons

The Form editor lets you add a wide variety of buttons to your forms. When you add buttons to a form, you can associate an automatic action to each button. Automatic buttons let the user accept, cancel, or delete records, move between records, move from page to page in a multi-page form, and open, delete, or add records in a subform.

You normally add buttons when you create the form using the Form Wizard. You can modify these buttons' actions in the Object Properties window or the Property List. For example, you can remove an

automatic action from a button and write an object method that specifies the button's action.

You can also add buttons and assign button actions with the Form editor. For example, if you need more than one subform on the form, you can add the necessary additional subforms and automatic buttons in the Form editor. You simply add each button to the form and associate the automatic action with each button.

You can associate automatic button actions to the following types of buttons:

- **Buttons** These are standard text buttons that are displayed as boxes. Button text is displayed in the selected font, font size, style, and color.



When you create a button, you should type a label into the Button Text area. The label appears inside the button when the form is used in the User environment or custom applications. You can change the text that appears in the button at any time by returning to the Object Properties window and modifying the text in the Button text area.

- **Default Buttons** A default button looks exactly like a button, except that its border is thicker. This indicates to the user that the button is the recommended choice. The following illustration compares a button to a default button.



Notes There can only be one default button per form page.

In the Property List, the Default button object type does not exist. It is a property that you can select for standard buttons.

- **Highlight buttons and 3D buttons** These buttons are designed to be placed on top of graphic objects. They are invisible until clicked. When the user clicks a highlight button, the button is highlighted. Highlight buttons are dimmed in appropriate circumstances in the User environment and in custom applications.

The appearance of highlight and 3D buttons depends on the Appearance settings of the form on which the button appears and the button's Appearance settings. The diagram below illustrates the variations.

Appearance	Highlight buttons	3D buttons
None		
Plain		
Dotted		
Raised		
Sunken		
Double		

- **Invisible buttons** These buttons are invisible and do not highlight when clicked. The resulting action, such as displaying a different page, should indicate that the button has been clicked. An invisible button should be placed on top of text or a graphic that denotes its function; the user clicks on the text or graphic, and the button is activated.

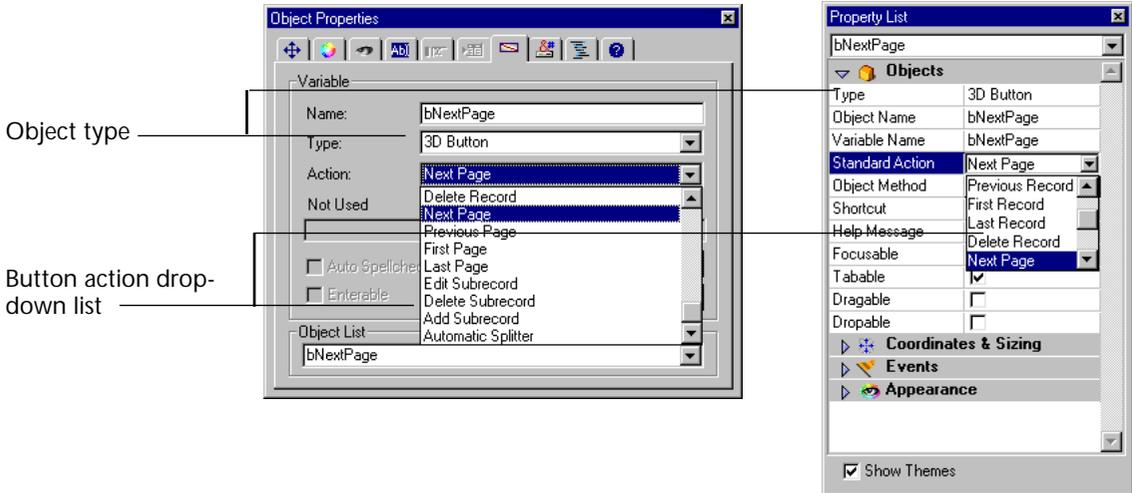
Employee Info Review Info Address Info

Buttons are dimmed when appropriate in the User environment or custom applications. For example, if the first record of a table were displayed, a First Record button would appear dimmed.

You create a button by choosing the desired button type from the Type drop-down list. You then choose the automatic button action you want from the Action drop-down list.

Object Properties window

Property List



Regardless of the type of button, you must select an item from this drop-down list. If you want the button to perform an action not listed in the Action drop-down list, choose “No Action” and write a method that specifies the button’s action. Normally, you would activate the On Clicked event in the Events page and the method would run only when the button is clicked.

Note All variables associated with buttons (including regular buttons, highlight buttons, invisible buttons, radio buttons, radio pictures, or check boxes) are initialized to 0 when the form is first opened in the User environment. When the user clicks a button, it becomes 1. You can associate a method with any button.

Button Actions

This section discusses each automatic action that can be assigned to a button.

- **No Action** Use a No Action button for a button that does not perform an automatic action. Choose No Action when you need to write a method to manage the button. For example, a button that displays a custom Find dialog box in a custom application would have a

No Action automatic action because you must write a method to open the custom dialog box.

- **Accept and Cancel Actions** Clicking an **Accept** button saves a record. It triggers an **On Validate** event. Clicking a **Cancel** button exits the current record without saving any changes.
- **Delete Record Action** Clicking a **Delete Record** button displays an alert asking the user to confirm the deletion. Clicking **Yes** in the alert deletes the current record. If the user is using the input form for a subrecord, **Delete Record** deletes the current subrecord.

After the user clicks a **Delete Record** button, 4th Dimension automatically returns to the output display.

Note A button with this action is automatically disabled when a new record is being added.

- **Record Navigation Actions** The **Next Record**, **Previous Record**, **First Record**, and **Last Record** buttons first accept the current record and then make the specified record current. The specific record made current by these buttons depends on the sort order.

These buttons perform the appropriate actions for subrecords when the user is entering subrecords.

A button of this type is automatically disabled when its action is inappropriate. For example, if the user displays the first record, the **Previous Record** button would be disabled.

- **Page Navigation Actions** The **First Page**, **Last Page**, **Next Page**, and **Previous Page** buttons display the appropriate page in a multi-page form. If there is only one page, these buttons are inactive.

A button of this type is automatically disabled when its action is inappropriate. For example, if the user displays the first page, the **Previous Page** button would be disabled.

- **Subform Actions** The **Subform** buttons affect records in subforms. You can open, delete, or add to records displayed in subforms. Placed on a parent record's form, they affect subrecords or related records in a subform. The following are the Subform button actions:
 - **Open Subform** This button is active when a record in a subform is selected. If the user selects a record and clicks the **Open Subform**

button, the Full Page form for that record opens and the user can modify the record.

- **Delete Subform** This button is active when a record has been selected in a subform. The Delete Subform button does not display an alert, but erases the record immediately.
- **Add to Subform** This button is active when a subform has been selected. When the user clicks an Add To Subform button, 4th Dimension creates a new record in the related table or subtable, scrolls to the record, and places the insertion point in the first enterable field in the subform.

For more information about using the Subform buttons, refer to [“Adding a Subform to the Form” on page 428](#).

- **Automatic Splitter** This automatic action allows you to create custom splitters on a form.

This action can only be assigned to an invisible button. When an invisible button is assigned this action, it behaves in the same way as a splitter. By pasting a picture in the invisible button, you can create any custom interface for splitters.

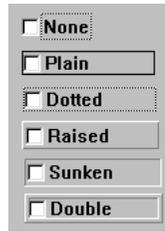
For more information about splitters, refer to [“Splitters” on page 416](#).

Check Boxes and 3D Check Boxes

A check box is used to enter or display binary (true-false) data. A check box is a type of button. A check box is either selected or deselected. The effect of a check box is controlled by a method. Like all buttons, a check box is initialized to 0 when the form is first opened. The method associated with a check box executes when the check box is selected.

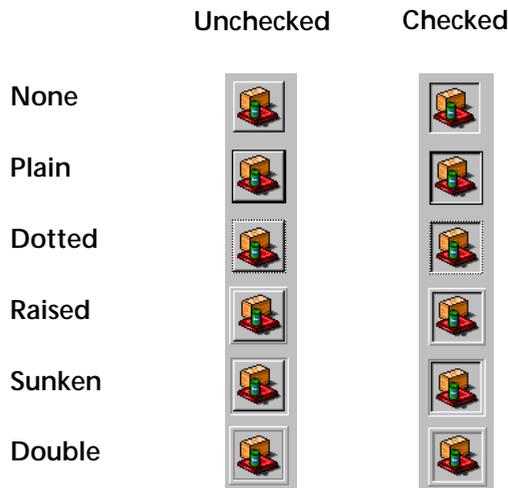
A check box displays text next to a small square. When the user selects the check box, the square is checked (an “X” is placed in it). When a check box is selected, it has the value 1. When deselected, it has the value 0. Any or all check boxes in a form can be selected or deselected. A group of check boxes allows the user to select several items. The fol-

Following illustration shows the effect of each Appearance setting on check boxes.



A 3D check box is similar to a highlight button. You place it on top of a graphic that indicates the function of the check box. When a 3D check box is clicked, its appearance changes according to the Appearance settings in the Display page of the Object Properties window. A 3D check box retains its state (0 or 1) until the user clicks it again.

The following illustration shows the effect of each Appearance setting.



Unlike a Boolean field that is formatted as a check box, the values of the check box variable are not stored automatically. You use a method to manage the variable.

Radio Buttons, 3D Radio Buttons, and Radio Pictures

Radio buttons, 3D radio buttons, and radio pictures are objects that allow the user to select one of a group of buttons or pictures. A radio button shows a small bull's-eye and text. 3D radio buttons and radio pictures display an icon or picture. They are placed on top of a graphic.

Each type of radio button is selected the same way — you click the object to select it. You can also click a selected radio picture to deselect it, but you cannot do this with a radio button.

3D radio buttons and radio pictures are similar to a highlight button in that it is transparent until selected. When selected, it highlights the picture behind it until it is deselected or another radio picture is selected. The user can move the pointer off the radio picture to avoid selecting it.

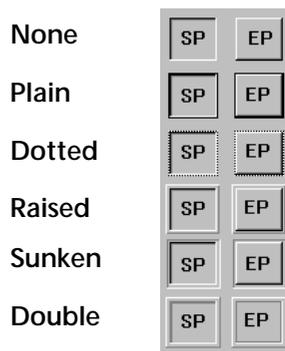
The remainder of this section uses the term “radio button” to mean any type of radio button.

Radio buttons exist in exclusive sets. The name of each object in a group of radio buttons must begin with the same letter (e.g., bRadio1, bRadio2, bRadio3). The effects of radio buttons are controlled with methods. Like all buttons, a radio button is initialized to 0 when the form is first opened. A method associated with a radio button executes when it is selected.

The following is an example of a group of 3D radio buttons used in a video collection database to enter the speed of the recording (SP, LP, or EP).

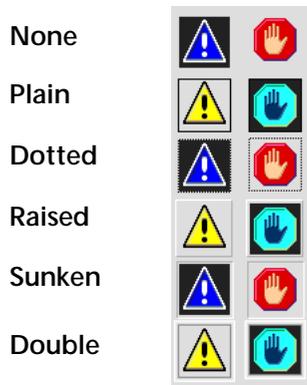


The labels are provided by separate Text objects that are placed on top of each 3D radio button. The “None” Appearance setting was used in the above example. The following illustration shows the effects of each Appearance setting. In each case, the SP tape speed is selected.



Selecting one radio button in a group sets that button to 1 and all the others in the group to 0. Only one radio button can be selected at a time.

The following illustration shows the effects of each Appearance setting on radio pictures. The selected radio picture is black on white.

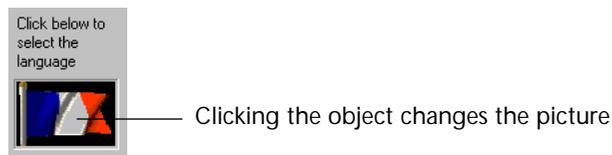


3D radio buttons retain their state (0 or 1) until another radio button in the set is clicked.

Picture Buttons

A picture button lets the user choose among several choices. As the name indicates, each choice is represented by a picture. A picture button can be used in place of a picture menu. With a picture menu, all choices are displayed simultaneously (as the items in the pop-up menu), while the picture button displays the choices consecutively (as the user clicks the button).

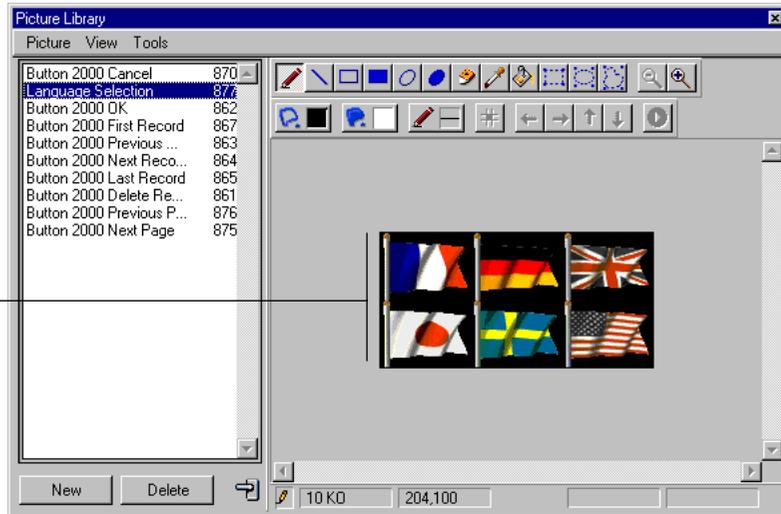
Here is an example of a picture button. Suppose you want to give the users of a custom application the opportunity to choose the interface language for the application. You implement the option as a picture button in your custom Database Properties dialog box, as shown below.



You implement the picture button in the following manner. First, you prepare *one* graphic in which the series of pictures is arranged in a row, column, or row by column grid. You can add the graphic to the Picture

library or as a PICT resource (on Macintosh). The following illustration shows the graphic in the Picture library.

The picture button is designed to display only one picture in the series at a time



You can organize pictures as columns, rows, or row by column grid (as shown above). When organizing pictures as a grid, they are numbered from the left to the right, row by row. For example, the second image of the second row of a grid that consists of four rows and three columns, is numbered 5.

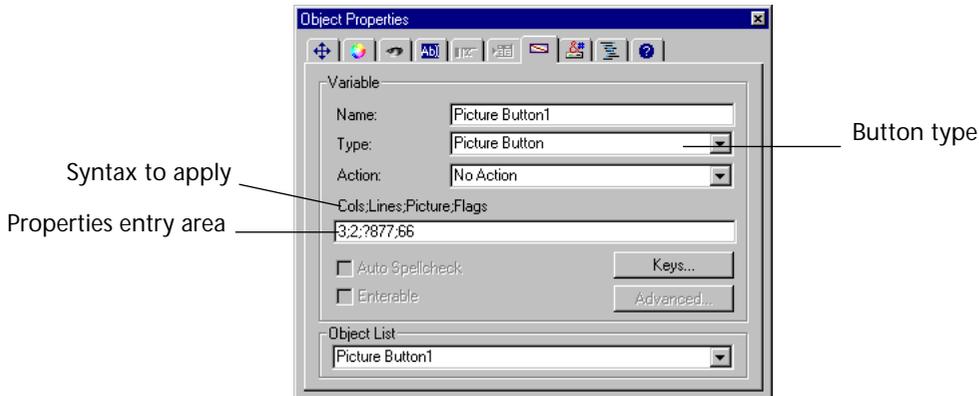
Note The Picture Library includes features that allow you to organize a picture as a table of thumbnails. It also allows you to preview the effects of the current settings.

When a picture was defined as a table of thumbnails, you can create a picture button by dragging the picture name into the form while pressing the Ctrl key (on Windows) or the Command key (on Mac OS). For more information on the Picture Library, refer to [Chapter 11, “Using the Picture Library”](#) on page 601.

The procedure for setting the properties of a picture button in depends on whether you use the Object Properties window or the Property List.

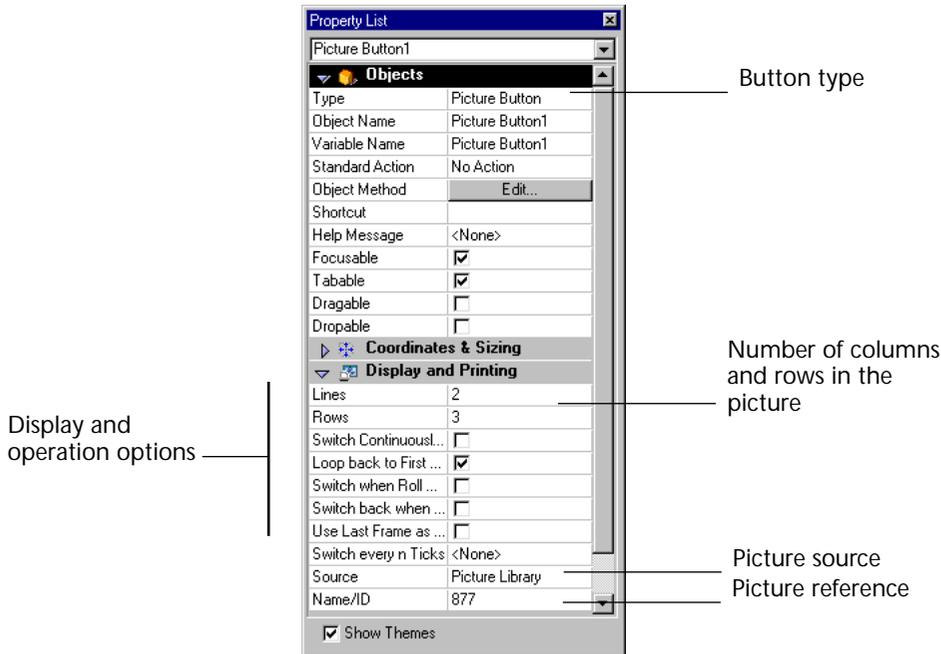
Using the Object Properties window

In the Variable page of the Object Properties window, choose Picture Button as the object type and set the properties as follows.



- *Cols* and *Lines* give the dimensions of the array of pictures. Since this example arranges the pictures in a grid, the specification is 3 columns and 2 rows.
- *Picture* identifies the picture in the Picture library or in the resource file. If the picture is in the Picture library, precede its number with a question mark “?”; if the picture is a resource, precede the resource ID with a colon “:”; if the picture is a variable, enter the variable name.
- *Flags* controls the appearance of the object and the behavior of the object when the user clicks on the object. *Flags* is the sum of any of the following values: 0, 1, 2, 4, 8, 16, 32, 64 and 128. Each of these values represents a display mode or an operation mode. For instance, if you want to enable the modes 1 and 64, enter 65 as flag value. For more information about the display modes, see the section [“Display Modes and Picture Button Operation”](#) on page 394.

Using the Property List In the Property List, select the Picture Button type in the Objects theme.



- In the **Display and Printing** theme, enter the number of columns and rows of the picture.
- In the **Display and Printing** theme, select the source of the picture from the source list. The sources that are available for selection are Variable, Picture Library, or resource. Once the source is set, enter in the **Name/ID** area
 - the name of the variable if the source is a variable, or
 - the resource number or the picture number if the source is a resource of the Picture Library.
- The remaining options allow you to set the display mode as well as the operation of picture buttons. The organization of these options is different from that of the Object Properties window and is described in the next section.

Display Modes and Picture Button Operation

The following table describes the display and operation modes you can set for picture buttons. In the Object Properties window you do this by setting a value for the Flag property and in the Property List you choose the property shown in the second column of the following table.

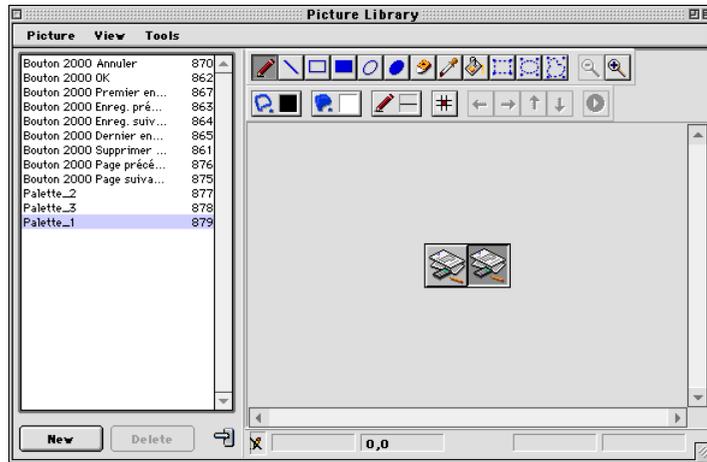
Value for Flag	Option in the Property List	Description
0	None	Displays the next picture in the series when the user clicks; it displays the previous picture in the series when the user holds down the Shift key and clicks. When the user reaches the last picture in the series, the picture does not change when the user clicks again. That is, it does <i>not</i> cycle back to the first picture in the series.
1	Switch Continuously	Is similar to 0 except that the user can hold down the mouse button to display the pictures continuously (i.e., as an animation). When the user reaches the last picture, the object does <i>not</i> cycle back to the first picture.
2	Loop Back to First Frame	Is similar to 0 except that the pictures are displayed in a continuous loop. When the user reaches the last picture and clicks again, the first picture appears, and so forth.
16	Switch when Roll Over	The contents of the picture button are modified when the mouse cursor goes over it. The initial picture is reestablished when the cursor leaves the button's area. This mode is frequently used in multimedia applications or in HTML documents. The picture that is then displayed is the last picture of the thumbnail table, unless the Use Last Frame as Disabled option is selected (128). If that option is selected, the next to last thumbnail that is displayed.
32	Switch Back when Released	This mode operates with two pictures. It displays the first picture all the time except when the user clicks the button. In that case, the second picture is displayed. This mode allows you to create an action button that displays its status (idle and clicked). You can use that mode to create a 3D effect or display any picture that depicts the action.
64	Transparent (Appearance theme)	Controls transparency. If you use 64, the picture button background is transparent. The effect of transparency is shown in the following illustration: <div style="display: flex; align-items: center; justify-content: center;"> <div style="text-align: center;"> <p>Transparency on</p>  </div> <div style="margin: 0 20px;">  </div> <div style="text-align: center;"> <p>Transparency off</p> </div> </div>

Value for Flag	Option in the Property List	Description
128	Use Last Frame as Disabled	This mode allows you to set the last thumbnail as the thumbnail to display when the button is deactivated. When this mode is selected, 4 th Dimension displays the last thumbnail when the button is deactivated. When this mode is used in addition to the modes 0, 1, and 2 the last thumbnail is not taken into account in the sequence of the other modes. It will appear only when the button is deactivated.
Additional parameter located after Flags	Switch every n Ticks	This mode allows you to cycle through the contents of the picture button at the specified speed. In the Object Properties window, this mode is selected by entering a fifth parameter in the button's syntax. Adding this parameter tells 4 th Dimension that the picture button will cycle through the thumbnails every n ticks, where n is the parameter. For example, if you enter "2;3;?16807;0;10" in the Object Properties window, the picture button will display a different picture every ten ticks. In the Property List, you set this parameter by entering a value in the Switch every n Ticks line. When this mode is active, only the Transparent mode can be used (64).

Examples

The following examples illustrate the effects of several of the options described in the above table.

- **Switch Back when Released** (action button) Here is an example on how to set a button in a palette. This example uses the following picture located in the Picture Library.



Once the picture is in the Picture Library, you can set the button that will be used in the palette as follows.

- In the Property List, set the following properties: 1 row, 2 columns and select the Switch Back When Released option.
- In the Object Properties window, enter the following parameters: 2;1;?879;32.

The Palette button will first display the left part of the picture as the default button and will display the right part of the picture when the button is being clicked on.

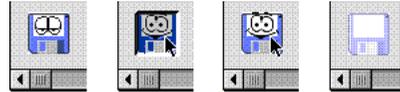


- **Complete picture button** This example explains how to set a button that accepts the following modes: Switch back when Released, Switch when Roll Over and Use Last Frame as Disabled.

In the case of a table of thumbnail that has one row of four columns, each thumbnail correspond to the following statuses: default, clicked, roll over and disabled.

In the Property List you would select the following options: **Switch back when Released**, **Switch when Roll Over** and **Use Last Frame as Disabled**.

In the Object Properties window, you would enter the following syntax: "4;1;?15000;176".



Default Clicked Roll over Disabled

Menu/Drop-down Lists, Pop-up Menus, Drop-down Lists, and Scrollable Areas

Pop-up menus, drop-down lists, and scrollable areas are objects that allow the user to select from a list. You manage the items displayed in the pop-up menu or scrollable area using an array.

An array is a list of values in memory that is referenced by the name of the array.

A pop-up menu/drop-down list displays an array as a list of values when you click on it.

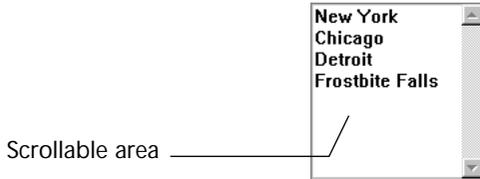
Note On Mac OS, a drop-down list is called a pop-up menu. These two objects are functionally identical.

A scrollable area displays the array in a list box that can be scrolled and used to select an item.

The figure below shows a pop-up menu and a drop-down list:



A scrollable area displays the array in a list box that can be scrolled and used to select an item.



You initialize the object by loading a list of values into an array. You can do this in several ways:

- In the Data Control page of the Object Properties window, click the Edit Strings button to enter the list into the Default Values dialog box. In the Property List, click the Edit button located next to the value List line in the Entry Control theme. For more information, see [“Default Lists of Values” on page 350](#). The default values are loaded into an array automatically. You can refer to the array using the name of the object.
- Before the object is displayed, execute code that assigns values to the array elements. For example,

```
ARRAY TEXT (aCities;6)
aCities{1}:="Philadelphia"
aCities{2}:="Pittsburg"
aCities{3}:="Grand Blanc"
aCities{4}:="Bad Axe"
aCities{5}:="Frostbite Falls"
aCities{6}:="Green Bay"
```

This code could be placed in the object’s method and executed when the On Load form event runs.

- Before the object is displayed, load the values of a list into the array using the LIST TO ARRAY command. For example,

```
LIST TO ARRAY ("Cities";aCities)
```

This code would be run in place of the assignment statements shown above.

If you need to save the user’s choice into a field, you would use an assignment statement that runs after the record is accepted. A complete Case statement in the object method might look like this:

Case of

```

:(Form event=On Load)
  LIST TO ARRAY ("Cities";aCities)
  If (Record number ([People])<0) `new record
    aCities:=3 `display a default value
  Else `existing record, display stored value
    aCities:=Find in array (aCities;City)
  End if
: (Form event=On Clicked) `User modified selection
  City:=aCities {aCities} `field gets new value
:(Form event=On Validate)
  City:=aCities {aCities}
:(Form event=On Unload)
  CLEAR VARIABLE (aCities)

```

End Case

In the Events page of the Object Properties window, you would select each event that you test for in your Case statement.

Arrays always contain a finite number of items. The list of items is dynamic and can be changed by a method. Items in an array can be modified, sorted, and added to.

For information about creating and using an array, refer to the chapter on arrays in the *4th Dimension Language Reference*.

Goto Page Action

You can assign the Goto Page action to a pop-up menu/Drop-down list or a scrollable area. When that action is selected, 4th Dimension will automatically display the page of the form that corresponds to the number of the object that is selected in the drop-down list or scrollable area.

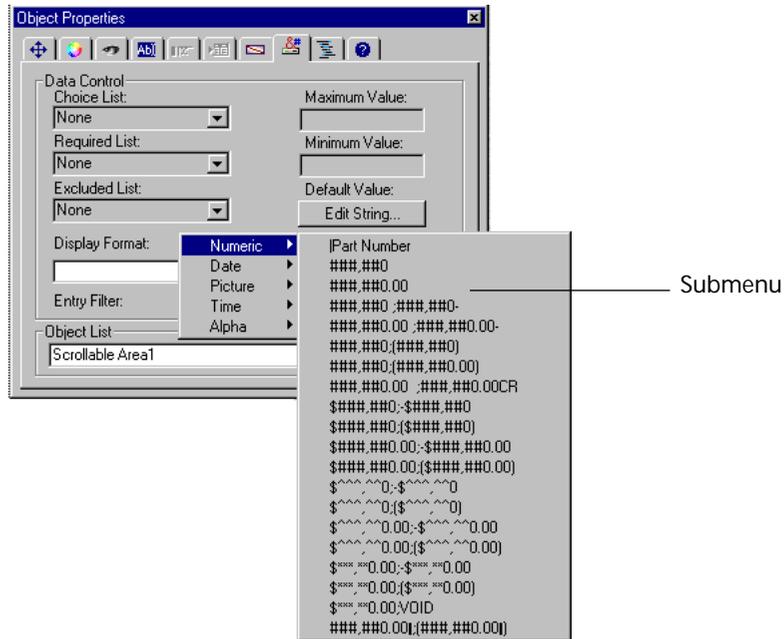
For example, if the user selects the third element of the list, 4th Dimension will display the third page of the current form (if it exists). If you want to manage the effect of the selection of one element, select **No action**.

Combo boxes

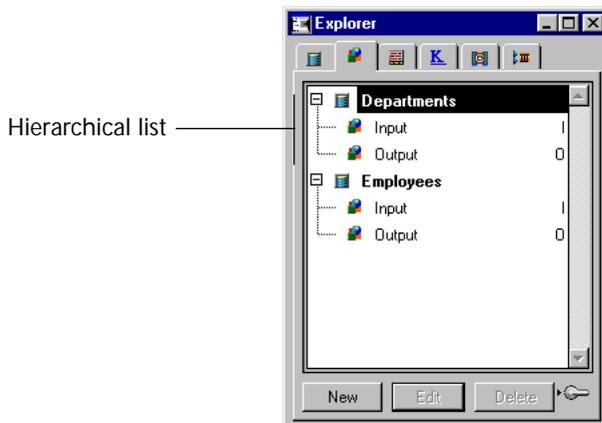
A Combo box is similar to a drop-down list, except that the object accepts text entered from the keyboard. You initialize a combo box in exactly the same way as a drop-down list. If the user enters text into the combo box, it fills the 0th element of the array. In other respects, you treat a Combo box as an enterable area that uses its array as a set of default values. Use the On Data Change event to manage entries into the enterable area, as you would an enterable area object.

Hierarchical Pop-up Menus and Hierarchical Lists

A hierarchical pop-up menu has a submenu associated with each item in the menu. Here is an example of a hierarchical menu:



Similarly, a hierarchical list has a sublist associated with each item in the list. The Explorer is an example of a series of hierarchical lists:



In this example, each item in the list of tables has a sublist of forms. You can expand or collapse the hierarchical list by clicking on the plus or minus sign (on Windows) or arrows (on Macintosh).

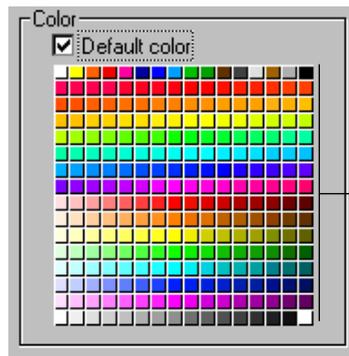
You can control whether an item in a hierarchical list can be modified by the user. If an item in a hierarchical list is modifiable, the user can hold down the Ctrl key (Command key on Macintosh) to edit the text of the item. If you populate a hierarchical list using a list created in the List editor, you control whether an item in a hierarchical list is modifiable using the Enabled/Editable check box in the List editor. For more information, see the section [“Making a Hierarchical List Modifiable or Enabling Tab Control Items”](#) on page 596.

You manage hierarchical pop-up menus and hierarchical lists using the Hierarchical list commands in the language. For more information, see that section of the Language reference.

Button Grids

A button grid is a transparent object that is placed on top of a graphic. The graphic should depict a row by column array. You can use a button grid object to determine where the user clicks on the graphic. Your object method would use the On Clicked event and take appropriate action depending on the location of the click.

In 4th Dimension, a button grid is used as a color palette:

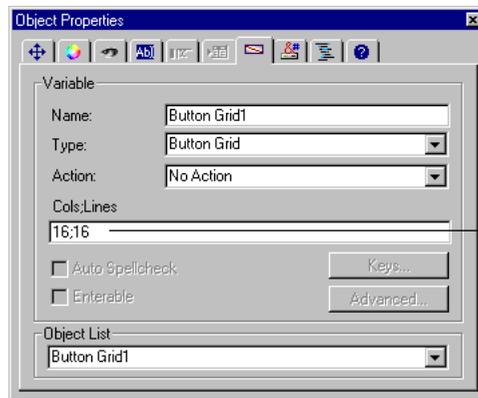


Button grid object on top of background graphic

The buttons on the grid are numbered from top left to bottom right. In this example, the grid is 16 columns across by 16 rows down. The button in the top-left position returns 1 when clicked. If the red button at the far right of the second row is selected, the button grid returns 32.

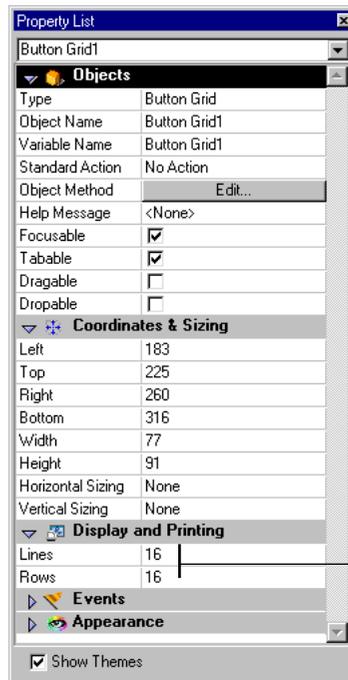
To create the button grid, add a background graphic to the form and place a button grid on top of the graphic.

- In the Variable page of the Object Properties window, specify the number of rows and columns of the grid and the gap between elements of the grid.



Enter the number of columns and rows

- In the Property List, specify the number of lines and columns in the corresponding entry areas of the Display and Printing theme.



Enter the number of columns and rows

Goto Page Action

You can assign the Goto Page action to a button grid. When that action is selected, 4th Dimension will automatically display the page of

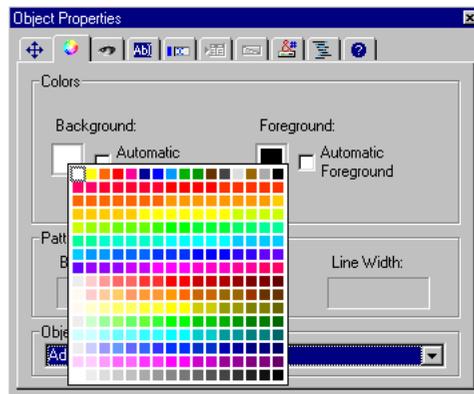
the form that corresponds to the number of the button that is selected in the button grid.

For example, if the user selects the tenth button of the grid, 4th Dimension will display the tenth page of the current form (if it exists). If you want to manage yourself the effect of the selection of one button, select **No action**.

Picture Pop-up Menu

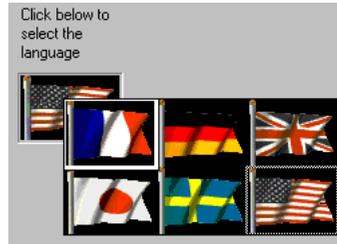
A Picture pop-up menu is a pop-up menu that displays a two-dimensional array of images. A picture pop-up menu can be used to replace a picture menu. The creation of the picture to use with a picture pop-up menu is similar to the creation of a picture for a picture button. The concept is the same as a button grid, except that the graphic is used as a pop-up menu instead of a form object.

The following illustration shows the color palette displayed as a Picture menu.



To create a Picture menu, you need to refer to an image. The following example uses the picture that was defined for picture buttons. In this case, it allows you to select the interface language by selecting it from a

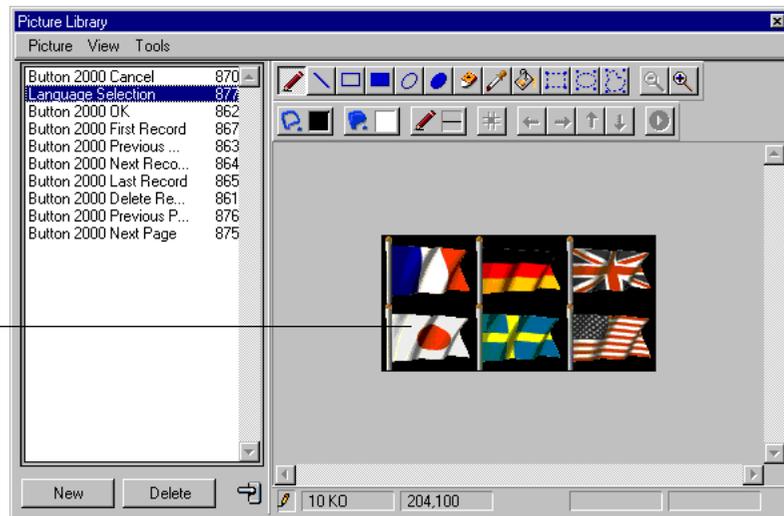
picture pop-up menu. Each language is represented by the corresponding flag.



Picture pop-up menu

As with a picture button, a picture pop-up menu uses a picture that is organized in columns and rows. You can place that picture in the picture library, in a picture variable or in a PICT resource (on Mac OS).

The Picture pop-up menu object displays the whole picture



Note The Picture Library includes features that allow you to organize a picture as a table of thumbnails. It also allows you to preview the effects of the current settings.

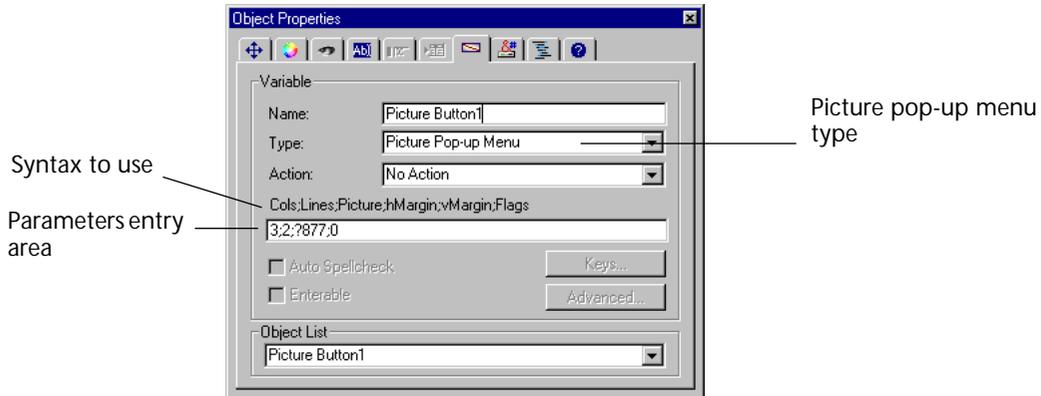
When a picture is defined as a table of thumbnails, you can create a picture pop-up menu by dragging the picture name into the form while pressing the Shift key.

For more information on the Picture Library, refer to [Chapter 11, “Using the Picture Library”](#) on page 601.

The method for setting the properties of a picture pop-up menu in the form editor depends on whether you do it in the Object Properties window or in the Property List.

Using the Object Properties window

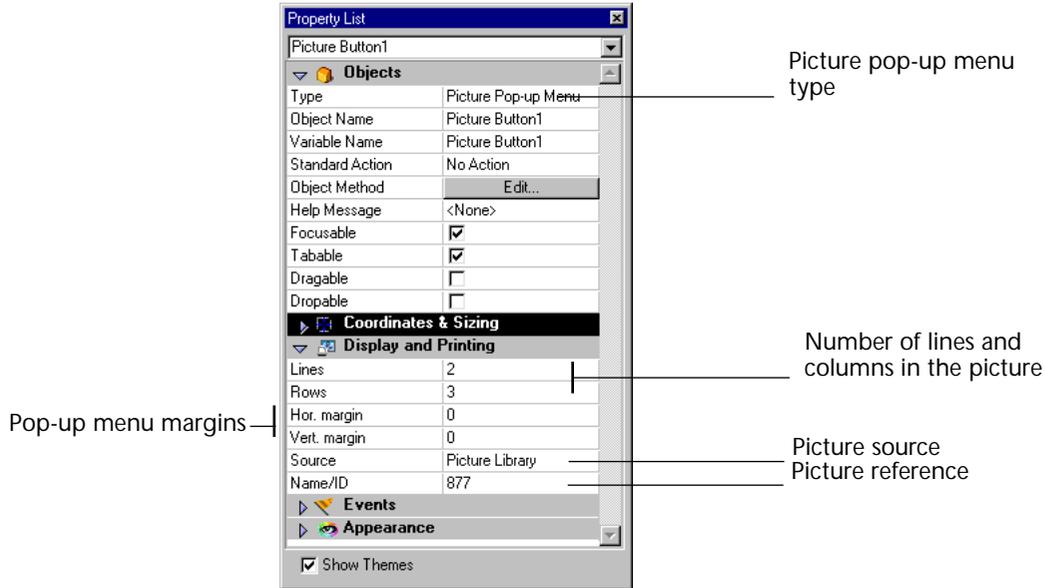
In the Object Properties window, select the Picture Pop-up Menu type in the Variable page. You should then set the parameters as follows:



- The *Cols* and *Lines* parameters define the dimensions of the thumbnail table. Since this example uses three columns and two rows, the values are 3 and 2, respectively.
- The *Picture* parameter defines the picture that is used. It can be located in the Picture library, in a variable, or in a PICT resource:
 - For a variable, enter the name of the picture variable,
 - For a PICT resource in your database, enter the PICT's reference ID as “:1234”. Remember to include a colon, “:”, before the picture's reference ID.
 - For a PICT in your Picture library, enter the reference ID¹ of the picture preceded by a question mark, as shown in the example above.
- The parameters *vMargin* and *hMargin* create a gap between the border of the menu and the picture. Enter values in pixels.
- The parameter *Flags* allows you to specify whether the picture pop-up menu is transparent or not. The two values for *Flags* are:
 - 0 the pop-up menu box is not transparent, or
 - 64 the pop-up menu box is transparent.

1. The Reference ID is displayed in the Picture library.

Using the Property List In the Property List, select the Picture Pop-up Menu type in the Objects theme.



- In the Display and Printing theme, define the number of columns and lines of the thumbnail table.
- In the Display and Printing theme, select the source type from the Source drop-down list. You can choose between three source types: Picture Library, variable, and PICT resource. Once the picture source is set, enter the reference of the picture in the Name/ID entry area. The reference of the picture is its name if its source is a variable or a number if its source is a PICT resource of the Picture Library.
- The two entry areas Hor. Margin and Vert. Margin control the margins between the border of the menu and the picture.
- If you want the pop-up picture menu to be transparent, check that option in the Appearance theme.

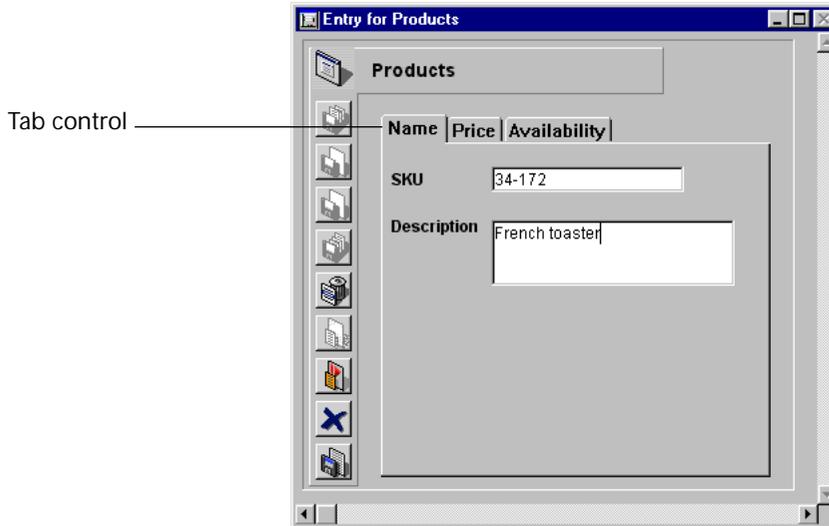
Operation of Pop-up Picture Menu

It is not possible to assign an automatic action to a pop-up picture menu. Pop-up picture menus are managed using methods.

As for button grids, variables associated with pop-up picture menus are set to the value of the selected element in the pop-up picture menu. If no element is selected, the value is 0. Elements are numbered, row by row, from left to right starting with the top row.

Tab Controls

A tab control creates an object that lets the user choose among a set of virtual screens that are enclosed by the tab control object. Each screen is accessed by clicking its tab. The following multi-page form uses a tab control object.

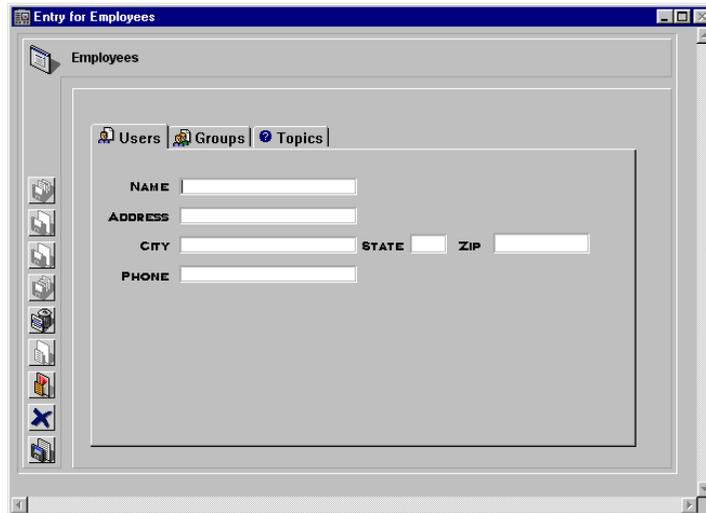


To navigate from screen to screen, the user simply clicks the desired tab.

The screens can represent pages in a multiple-page form or an object that changes when the user clicks a tab. If the tab control is used as a page navigation tool, then the GOTO PAGE command would be used when a user clicks a tab.

Another use of the tab control is to control the data that is displayed in a subform or grouped scrollable arrays. For example, a rolodex could be implemented using a tab control. The tabs would display the letters of the alphabet and the tab control's action would be to load the data corresponding to the letter that the user clicked.

Each tab can display labels or labels and a small icon. If you include icons, the icons appear to the left of each label. Here is an example of a Tab control that uses icons:



When you create a tab control, 4th Dimension manages the spacing and placement of the tabs. You only need to supply the labels in the form of an array or the icons and labels in the form of a hierarchical list.

If the tab control is wide enough to display all the tabs with both the labels and icons, it displays both.

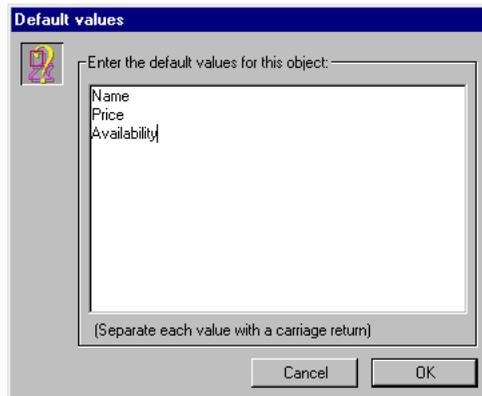
If the tab control is not wide enough to display both the labels and icons, 4th Dimension displays the icons only. If it can't fit all the icons, it places scroll arrows to the right of the last visible tab. The scroll arrows allow the user to scroll the icons to the left or right.

Adding Labels to a Tab Control

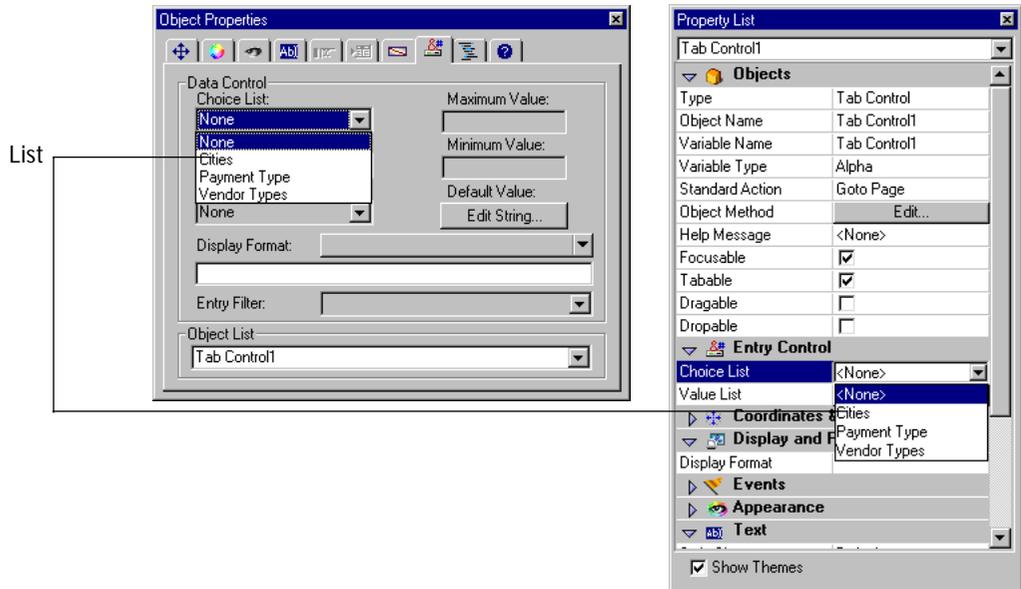
There are several ways to supply the labels for a tab control.

- In the Object Properties window, use the Edit Strings button on the Data Control page. In the Property List, use the Edit button for value

list located in the entry control theme. Here is the Default Values dialog box that creates labels for the Products Tab control.



- Create a list using the List editor and assign the list to the tab control as a choice list, as shown below.



If you like, you can associate a small icon with each list element using the List editor.

- You can create a Text array that contains the names of each page of the form. This code must be executed before the form is presented to the

user. For example, you could place the code in the object method of the tab control and execute it when the On Load event occurs.

```
ARRAY TEXT (asPages;3)
asPage {1}:="Name"
asPage {2}:="Address"
asPage {3}:="Notes"
```

You can also store the names of the pages in a list and use the LIST TO ARRAY command to load the values into the array.

You could also enter the names of the pages as default values using the Object Properties window. For more information, see [“Default Lists of Values” on page 350](#).

- Use the GOTO PAGE command in the Tab control’s method:

```
GOTO PAGE (asPages)
```

The command is executed when the On Clicked event occurs.

- You then clear the array on the On Unload event occurs.

Here is an example object method:

```
Case of
:(Form event=On Load)
LIST TO ARRAY ("Tab Labels";asPages)
:(Form event=On Clicked)
GOTO PAGE (asPages)
:(Form event=On Unload)
Clear variable (asPages)
End Case
```

Goto Page Action

You can assign the Goto Page action to a tab control. When that action is selected, 4th Dimension will automatically display the page of the form that corresponds to the number of the tab control that is clicked.

For example, if the user clicks the third tab control, 4th Dimension will display the third page of the current form (if it exists). If you want to manage the tab control programmatically, select No action.

Graph Areas

A graph area can be used to display a graph in a form. There are many ways to bring values to the graph area for calculation and display, all of which are controlled by the graphing commands in the language. The subject is covered in the *4th Dimension Language Reference*.

Note In the Tools palette there is no graph area tool. When creating a graph area, you first need to create a variable and then assign it the Plug-in area type.

Plug-in Objects

A *plug-in object* is an area on the form that is completely controlled by a 4D plug-in written in C or Pascal.

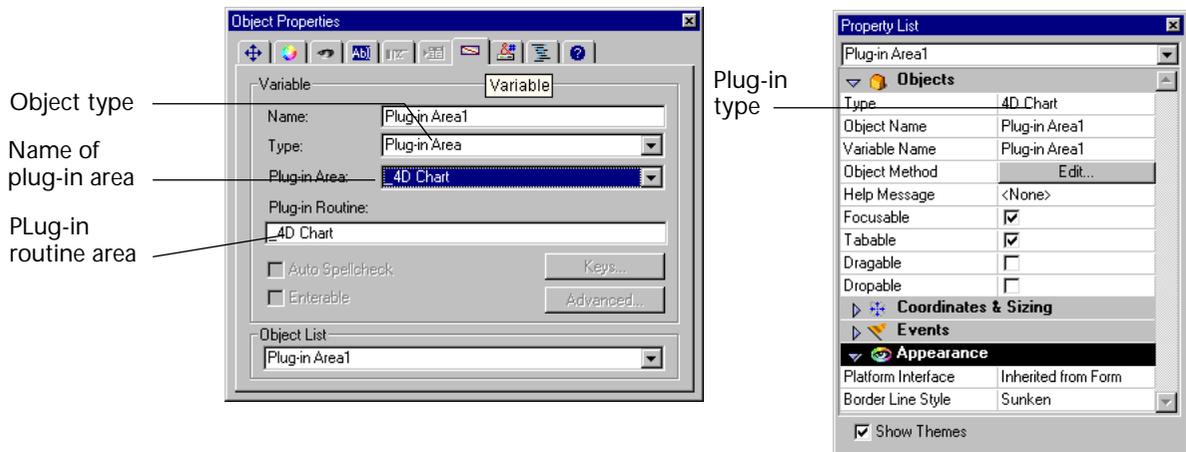
When opening a database, 4th Dimension creates an internal list of the plug-ins installed in your database. Once you have inserted a Plug-in Area in a form, 4th Dimension lists the available plug-ins in the Object Properties window. By default, only 4D Chart is available.

Note Some plug-ins cannot be used in forms or in external windows. When a plug-in cannot be used in a form, it does not appear in the plug-in list of the Object Properties window.

In the Object Properties window, you need to select the name of the plug-in from the Plug-in area list. When you select the plug-in from this drop-down list, 4th Dimension automatically inserts it in the Plug-in Routine enterable area. In the Property List, the plug-in name is directly selected from the Type list:

Object Properties window

Property List



When the object type is Plug-in Area, the Advanced button may be enabled. Advanced options may be provided by the author of the plug-

in. If the plug-in provides additional options, you can click this button to set those options. Because the Advanced options dialog is under the control of the author of the plug-in, information about the Advanced options is the responsibility of the distributor of the plug-in. If the plug-in has no Advanced options, the Advanced button is disabled.

Installing Plug-ins

To install a plug-in in your 4D environment, you first need to quit 4th Dimension. Plug-ins are loaded when you launch 4th Dimension.

On Windows, plug-ins are installed by copying the .4DX and .RSR plug-in files into a WIN4DX folder, located at the same level as the database structure file (.4DB), or located in the ACI folder in the system folder.

On Mac OS, plug-ins are installed by copying the plug-in file into a Mac4DX folder, located at the same level as the database structure file (.4DB), or located in the ACI folder in the Preferences folder (located in the System folder).

Note You may need to enter a serial or expansion number to be able to use certain plug-ins.

For more information about the installation of plug-ins, please refer to to the 4D Product Line Installation Guide.

Using Plug-ins

The ability to incorporate plug-ins into forms gives you unlimited possibilities when creating custom applications. A plug-in can perform a simple task such as displaying a digital clock on a form, or a complex task such as providing full-featured word processing, spreadsheet, or graphics capabilities.

Many of these capabilities are already available for the 4th Dimension environment in the form of plug-ins. The ACI productivity plug-ins set includes:

- 4D Write, a word processing application,
- 4D Draw, an object-oriented drawing application,
- 4D Calc, a spreadsheet application.

For more information, refer to the documentation that comes with the ACI Productivity Plug-ins.

If you are interested in designing your own plug-ins, you can receive extensive information about writing and implementing plug-ins. ACI provides the following information sources:

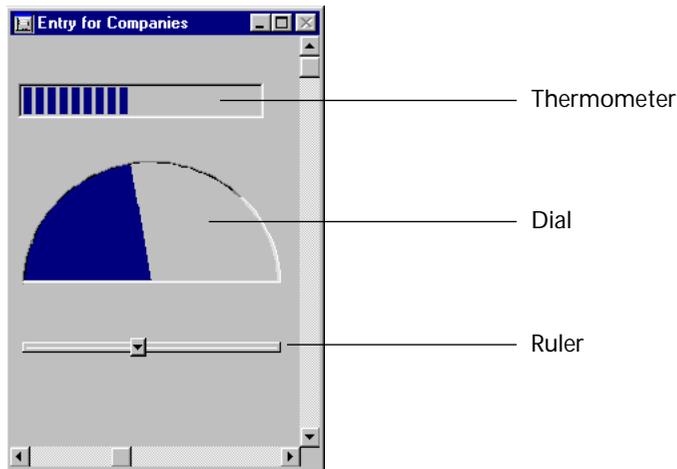
- **Platform Independent 4D Plug-ins Kit** The 4D Plug-ins kit teaches you how to write Plug-ins for 4th Dimension on both Windows and Macintosh.
- **Developer Tools** These developer mailings provide information and examples of plug-ins that can be used and modified for the developer's own needs.

For more information, contact ACI.

Indicators

Thermometers, rulers, and dials are objects that display a value graphically. The three objects work in the same way; they differ only in their appearance. We refer to these three objects as *indicators*.

You can use indicators either to display or set values. For example, if a thermometer is given a value by a method, it displays the value. If the user drags the indicator point, the value changes. The value can be used in another object such as a field or an enterable or non-enterable object.



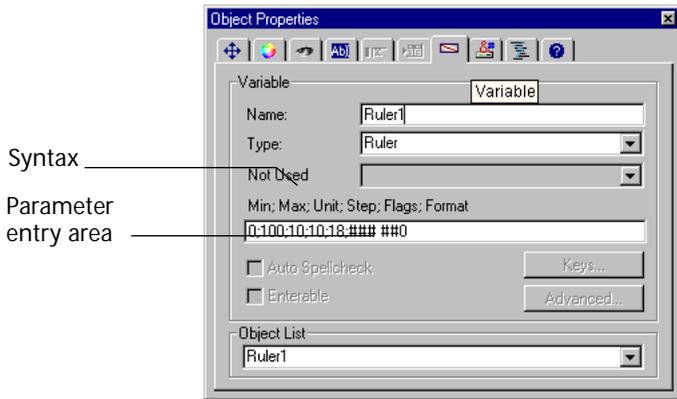
In addition to the standard positioning and appearance settings, you can set some other specific properties for indicators: minimum value, maximum value, units for the tick marks, the minimum steps permitted by the indicator as well as display options.

You can also define the display format of an indicator's label (for more information on display formats, refer to [“Display Formats” on page 357](#)).

In the Object Properties window, those properties are defined using a sequence of parameters. The *Mode* parameter is the sum of several display options (see below). In the Property List, those properties are defined using separate options.

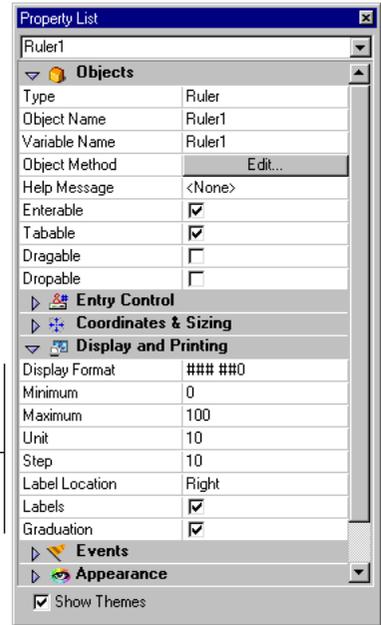
Object Properties window

Property List



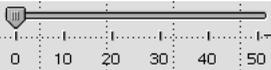
Syntax

Parameter entry area



Parameter areas

Additional display options can be defined for rulers and thermometers (except for the last option, that is only accessible to dials). In the Object Properties window, these options are set using the *Mode*¹ parameter.

Value for Mode	Property List Options	Description
0	No option selected	Does not display the labels.
2	Labels + Label Location= Bottom	Display the units on the right or below the indicator. 

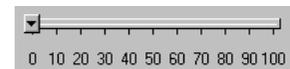
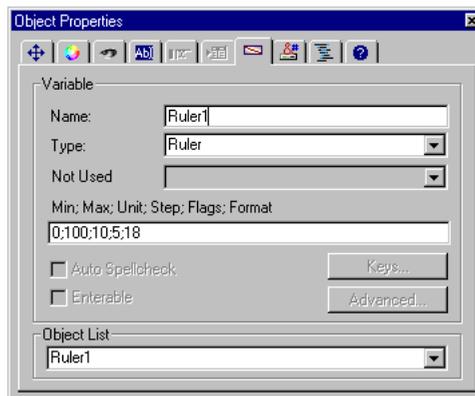
Value for Mode	Property List Options	Description
3	Labels + Label Location= Top	Display the units on the left or above the indicator. 
16	Graduation	Display graduations adjacent to the units, as shown in both illustrations above. Used only if 2 or 3 is also selected.
32	Execute Object Method	On Data Change is executed while the user is adjusting the indicator. If this value is not used, On data change occurs only after the user is finished adjusting the indicator.

Note Option 2 and option 3 cannot be used at the same time.

For example, if you enter:

0;100;10;5;18

the object sets the minimum to 0, the maximum to 100, places tick marks every 10 units on the indicator, and increments the display when the value changes 5 units. The figure below shows these values being set for a thermometer.



The variable associated with the indicator controls the display. You place values into, or use values from the indicator using methods. For example, a method for a field or enterable object could be used to control a thermometer.

1. The *Mode* parameter is a sum of different options. If, for example, the value for Mode is 51 (32+16+3), the indicator will display the indicator's units and the object method be dynamically executed.

The method:

```
vTherm:=[Employees]Salary
```

assigns the value of the Salary field to the vTherm variable. This method would be attached to the Salary field.

Conversely, you could use the indicator to control the value in a field. The user drags the indicator to set the value. The method:

```
[Employees]Salary:=vTherm
```

assigns the value of the thermometer to the salary field. As the user drags the indicator, the value in the Salary field changes.

Splitters

A splitter divides a form into two areas, allowing the user to enlarge and reduce the areas by moving the splitter one way or the other. A splitter can be either horizontal or vertical.

The splitter takes into account each object's resizing properties, which means that you can completely customize your database's interface.

The splitter is generally used in output forms (in the Custom Menu environment) so that columns can be resized:



Here are some of the splitter's general characteristics:

- You can place as many splitters as you want in a form and use a mixture of horizontal and vertical splitters in the same form.
- A splitter can cut (overlap) an object. This object will be resized when the splitter is moved.
- Splitters cannot be used in output forms in the User environment or in subforms.
- If you resize a form using a splitter, the new dimensions of the form are saved only while the form is being displayed. Once a form is closed, the initial dimensions are restored.

- Once it is inserted, the splitter appears as a line. You can modify its border style to obtain a thinner line or change its color. Place the splitter on your form according to your needs.

Interaction with the properties of neighboring objects

In a form, splitters interact with the objects that are around it according to these objects' resizing options:

Resizing options for the object(s)	Object(s) above the horizontal splitter or to the left of the vertical splitter ¹	Object(s) below the horizontal splitter or to the right of the vertical splitter ²
None	Remain as is	Are moved with the splitter; their position relative to the splitter is not modified
Resize	Keeps their original position, but they are resized according to the splitter's new position	
Move	Are moved with the splitter	

1. You cannot drag the splitter past the right (horizontal) or bottom (vertical) side of an object located in this position.
2. The buffer, when moving a horizontal splitter toward the bottom or a vertical splitter to the right is either the window's border or another splitter. This buffer is calculated in such a way that the moved objects remain entirely visible in the form or do not pass under/next to another splitter.

Note An object completely contained in the rectangle that defines the splitter is moved at the same time as the splitter.

Managing splitters programmatically

You can associate an object method with a splitter and it will be executed when the user releases the splitter.

A variable of type `Longint` is associated with each splitter. This variable can be used in your object and/or form methods. Its value indicates the splitter's current position, in pixels, relative to its initial position.

- if the value is negative: the splitter was moved toward the top or toward the left,
- if the value is positive: the splitter was moved toward the bottom or toward the right,
- if the value is 0: the splitter was moved to its original position.

You can also move the splitter programmatically: you just have to set the value of the associated variable. For example, if a vertical splitter is associated with a variable named *split1*, and if you execute the following statement: `split1:=-10`, the splitter will be moved from 10 pixels to the left — as if the user did it manually.

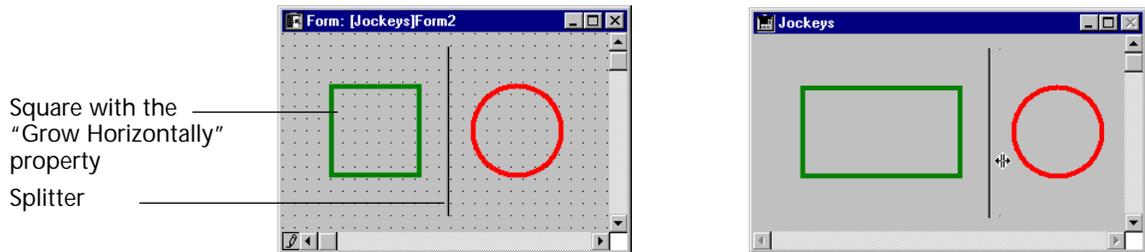
The move is actually performed at the end of the execution of the form or object method containing the statement.

The Automatic Splitter action allows you to create custom splitters in your forms. You can assign this action to an object of type invisible button. When an invisible button is assigned this automatic action, it acts exactly as a splitter. If, for example, you paste a picture on the invisible button, you can create a custom interface for your splitters. For more information about this type of button, refer to [“Buttons” on page 382](#).

Examples

This section gives some practical examples of the use of splitters.

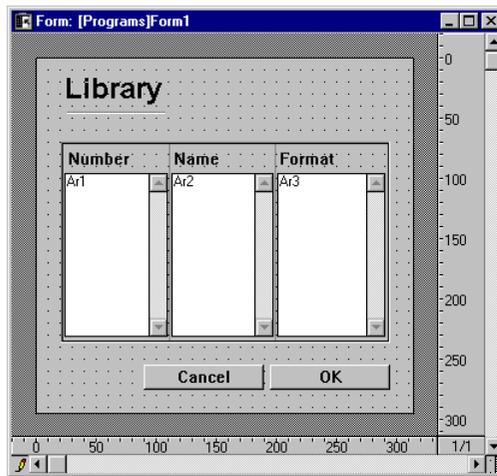
- ▶ **Example 1:** An input form contains a square, a vertical splitter and a circle. The “Grow Horizontally” property is applied to the rectangle. When the splitter is moved to the right or to the left, the rectangle becomes bigger or smaller, the circle is moved and its size is not modified. If the window is resized, the objects do not change position or size.



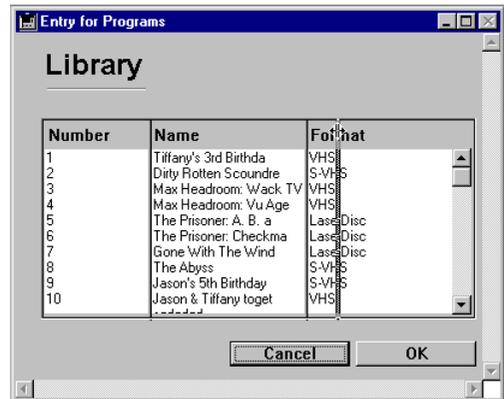
- ▶ **Example 2:** An input form containing three grouped arrays. The “Horizontal sizing” property is applied to all three arrays as well as their titles and a vertical splitter is placed between each column.

In this way, you can modify each column's relative size:

Design mode



User mode



Duplicating on a Matrix

Sometimes you may want to place several similar active objects in a form at the same time, numbering them sequentially so that their names are unique. For example, you may want to create a series of buttons that perform database operations. Duplicating on a matrix has the additional advantage of quickly and easily aligning multiple objects.

You can either duplicate an active object on a matrix manually or use the Duplicate Many dialog box, that allows you to quickly populate the matrix.

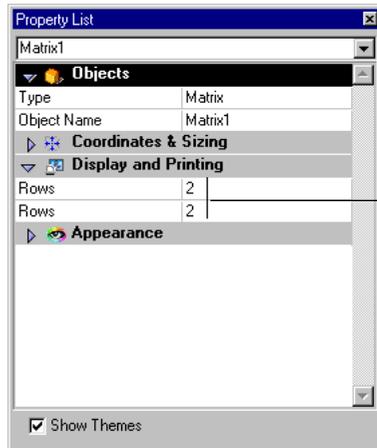
- ▶ To duplicate an object on a matrix:

- 1 Select the Matrix tool  and create a matrix on the form.

You can either drag the Matrix tool onto the form or draw the matrix itself. Make sure that each cell in the matrix is large enough to contain the object you want to duplicate.

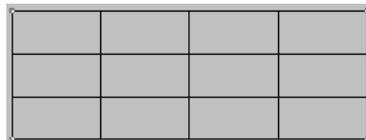
- 2 Display the Property List and select the matrix you just created.

- 3 In the Display and Printing theme, set the number of columns and rows of the Matrix.



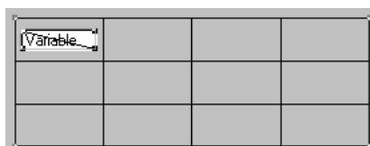
Line and Column settings

- 4 If necessary, set the appearance of the matrix using the options in the Appearance theme.
You can set the platform interface, the border style, the line thickness, the color, and the fill pattern.
- 5 Make sure the cells in the matrix are larger than the object that you want to duplicate.



For more information about resizing objects, refer to [“Resizing Objects” on page 270](#).

- 6 Create a new active object and place it in the upper left cell of the matrix.
Make sure that the object type matches the type and size of the object you want to duplicate.
- 7 Name the object and make sure its name does not end with a number.
- 8 Select both the object and the matrix.



9 Choose Duplicate on Matrix from the Object menu.

4th Dimension copies the active object to each cell in the matrix, giving each active object a unique number.



4th Dimension numbers the active objects from top to bottom in each column. These numbers are added to the object name for each object, thus creating a unique object at every matrix cell.

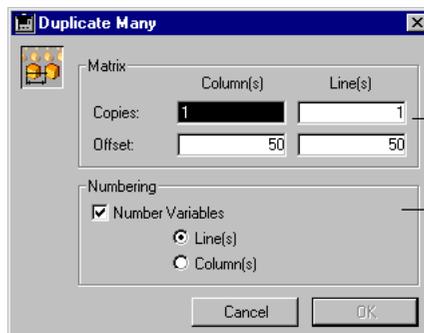


Note To number the series of active objects from left to right in each row, hold down the Alt key (on Windows) or Option key (on Macintosh) when you choose Duplicate on Matrix.

You can now refer to these objects in methods using the names they have been given. You can delete the matrix or leave it in the form.

► To duplicate one or several objects using the Duplicate Many dialog:

- 1 Select the objects you want to duplicate.
- 2 Select Duplicate Many from the Object menu.



Duplication matrix settings

Numbering options for variables (enabled only if a variable is selected)

- 3 In the Matrix area, enter the number of rows and columns as well as the offset between them.

For more information about this point, refer to [“Duplicating Objects” on page 290](#).

4 Select the **Number Variables** option.

This option is enabled only if you selected a variable.

5 Select the order of the numbering.

If you select the **Line(s)** option, 4th Dimension will number active objects from top to bottom (column by column) and left to right.

If you select the **Column(s)** option, 4th Dimension will number active objects from left to right (line by line) and top to bottom.

6 Click the **OK** button.

The objects are copied and numbered according to your settings:



Using Object Methods with Fields and Objects

You can attach a method to any active object in a form. Methods that are attached to individual objects on a form are called *object methods*.

The following are some of the more common uses of object methods:

- Enforce data entry constraints,
- Initialize and manage interface objects such as tab controls, pop-up menus, drop-down lists, combo boxes, hierarchical lists, and pop-up menus.
- Specify the action that takes place when an object is clicked or double-clicked,
- Manage drag and drop operations.

The previous section on types of active objects gives several simple examples of how methods are used to manage various kinds of objects. Here are some simple examples that perform operations on data:

The following method calculates a total based on data in two other fields:

Line Total := Price * Quantity

Here is a method to make all characters in a Name field uppercase:

Name := Uppercase (Name)

Note Built-in functions are displayed in boldface and user-written functions are displayed in italics. For a list of built-in functions, see the *4th Dimension Language Reference*.

The following method concatenates values from a First Name field and a Last Name field and assigns the results to a variable named vName:

```
vName := First Name + " " + Last Name
```

Because each object method is attached to its object, you create object methods from within the Form editor. For information on how to use the Method editor, see the section [“Using the Listing Editor” on page 507](#).

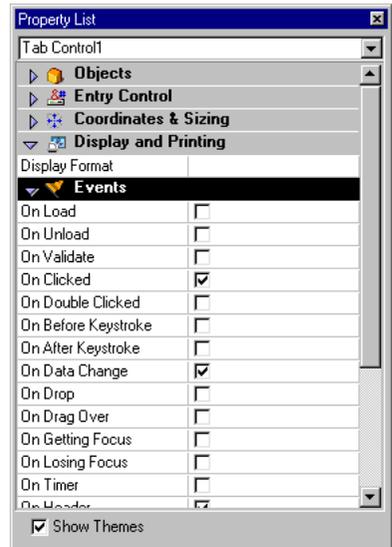
Object Events

Object methods run when certain events occur. For example, the action associated with a tab control makes sense only when a user clicks a tab. In a scrollable area, you may want the method to execute only when the user double-clicks an item. You can specify which events will be executed for a particular object in the Events page of the Object Properties window or in the Property List.

Object Properties window



Property List



The Events list displays all possible events that can lead to the execution of the object method. Here are the events that are relevant for forms that are displayed on-screen:

- **On Load** 4th Dimension is about to display the form on-screen or print the form.
- **On Unload** The form is about to be closed and released.
- **On Validate** After the user clicks OK or a navigation button.
- **On Clicked** The user clicks the object.
- **On Double Clicked** The user double-clicks the object.
- **On Before Keystroke** The user just entered one character in the object that has the focus. The Get Edited Text command returns the object's contents without that character.
- **On After Keystroke** The user just entered one character in the object that has the focus. The Get Edited Text command returns the object's contents with that character.
- **On Data Change** When the user changes the value of an object.
- **On Drop** When the user drops an object on a dropable object.
- **On Drag Over** When a dragged object can be dropped on the object (i.e., the dragged object is in position to be dropped on the object).
- **On Getting Focus** When a form object gets focus (i.e., the user presses Tab to select the object or clicks the object to select it).
- **On Losing Focus** When a form object loses focus (i.e., the user presses Tab to select the next object in the entry order or clicks another object to select it).
- **On Timer** The number of ticks, set using the SET TIMER command was reached.
- **On Header** The form header is about to be printed or displayed.
- **On Printing Break** A form break is about to be printed.
- **On Printing Detail** A form detail area is about to be printed.
- **On Printing Footer** A form footer area is about to be printed.
- **On Display Detail** A record is about to be displayed in an output form.
- **On Plug-in Area** A plug-in area triggers the execution of its object method.
- **On Outside Call** When the form receives a call from CALL PROCESS.

Note 4th Dimension also uses form events that are not available from the Object Properties window or the Property List. For a complete description of events in 4th Dimension, refer to the *4th Dimension Language Reference* manual.

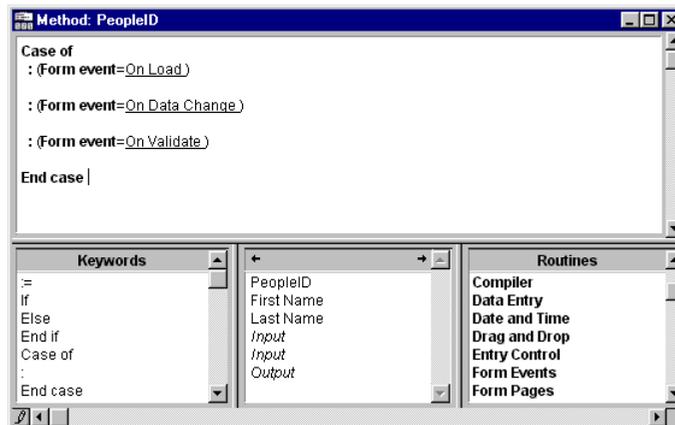
You activate an event by clicking the desired event. A check mark appears to the left of the events you select.

To select or deselect all the events at once, Press the Ctrl key (on Windows) or the Command key (on Mac OS) while clicking any event.

If you need to execute different code segments for several different events, use a Case statement in your method and test for each event you checked in the Events page. To test for an event, you use the Form event function and the Form Event constants in the Constants page of the Explorer.

You can add a constant to your code by opening the Explorer to the Constants page and dragging the desired constant to the desired point in the code. When the Method editor parses the line of code, all constants are underlined.

An example shell for an object method might look like this:



For more information on object methods, see [“Object Methods” on page 480](#) and the section on Form Events in the *4th Dimension Language Reference*.

► To add an object method:

- 1 In the Form editor, select the object to which you want to assign a method.

2 Click the Object Method button in either the Events page or the Field page of the Object Properties window.

OR

Click the Edit button located next to the Object Method line in the Property List.

OR

Choose Object Method from the Objects menu.

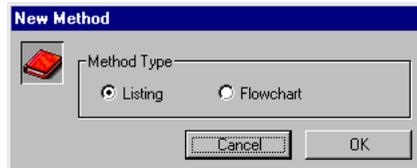
OR

On windows, click the object using the right mouse button and choose Object Method from the contextual menu. On Mac OS, click the object while pressing the Control key and choose Object Method from the contextual menu.

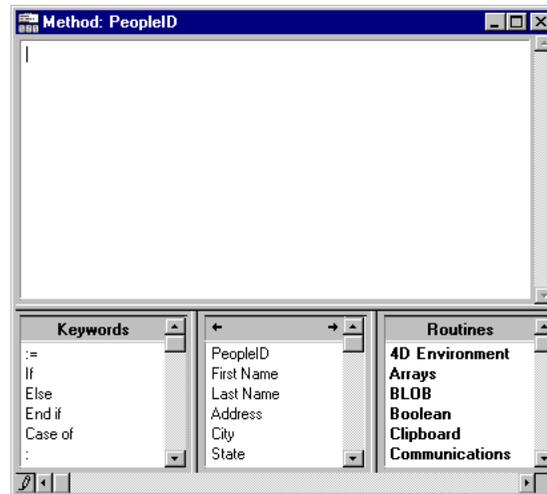
OR

Hold down the Alt key (on Windows) or Option key (on Macintosh) and click the field or object.

If you are creating a new method and if you have not set a default editor type in the Database Properties dialog box, 4th Dimension displays the Method Type dialog box where you can select the preferred Method editor.



When you click OK, 4th Dimension displays a blank Method editor window. The name of the new window includes the word “Method” and the name of the object or field.



Note If an object or field is grouped with another object, it must be ungrouped before you can open its method.

3 Write the method.

You can use several techniques to enter text into the Method editor. You can:

- Type text into the editor,
- Click on keywords, field or table names, or routines in the three scrollable areas below the text area,
- Drag table names, field names, form names, constants, built-in commands, plug-in commands, or project methods into the editor from the Explorer.

For more information on how to use the Method editor, see the section [“Using the Listing Editor” on page 507](#).

4 Close the Method window (optional).

The method is now associated with the field or active object. When an object has a method, a triangle appears in the top left corner of the object.



You can view or modify a method at any time.

- ▶ To open a method for viewing or modification:
 - 1 In the Form editor, select the object whose method you want to open.
 - 2 Click the Edit button located next to the Object Method line in the Property List.
OR
Hold down the Alt key (on Windows) or the Option key (on Macintosh) and click the object to which the method is attached.
OR
Choose Object Method from the Objects menu.
OR
On windows, click the object using the right mouse button and choose Object Method from the contextual menu. On Mac OS, click the object while pressing the Control key and choose Object Method from the contextual menu.
OR
Click the Object Method button in the Field or Events page of the Object Properties window.
The Method editor appears with your method, ready for you to make any changes.

Deleting an Object Method

If you create an object method and then find that you do not need it, you can use the **Clear Object Method** item in the **Object** menu to remove the method. To do so, select the object to which the method is attached and then choose **Clear Object Method** from the **Object** menu.

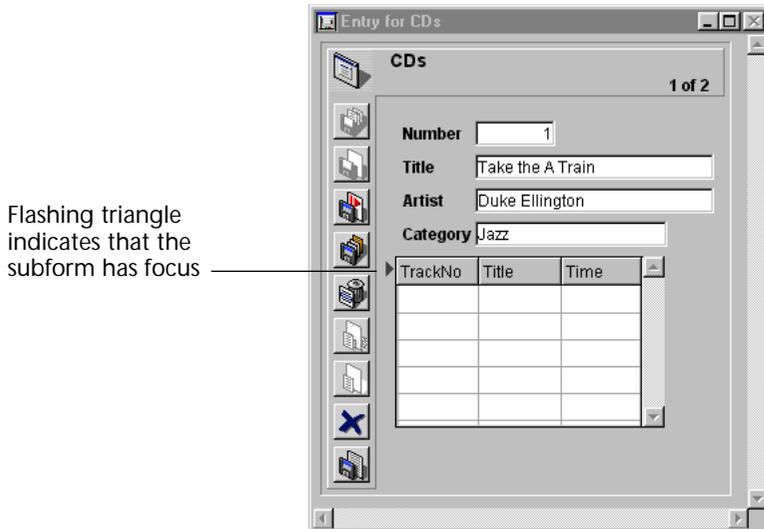
Adding a Subform to the Form

A subform is a List form from another table or subtable in the Master table that is displayed in a Detail form. A subform lets you enter, view, and modify data in other tables. You usually use subforms in databases in which you have established One-to-Many relations. A subform on a form in a related One table lets you view, enter, and modify data in a related Many table. You can have several subforms on the same form and they can belong to different tables or subtables.

For example, a Contacts manager database might use a subform to display all the telephone numbers for a particular contact. Although the telephone numbers appear on the Contacts screen, the information is actually stored in a related table. Using a One-to-Many relation, this database design makes it easy to store an unlimited number of tele-

phone numbers per contact. With automatic relations, you can support data entry directly into the related Many table without programming.

In the figure below, a subform in a Detail form has focus. A small flashing triangle points to the subform title bar and the Add to Subform button is enabled¹.



To add a record to the related Many table, the user clicks the **Add to Subform** button or presses **Ctrl+/'** (Command-Tab on Macintosh).

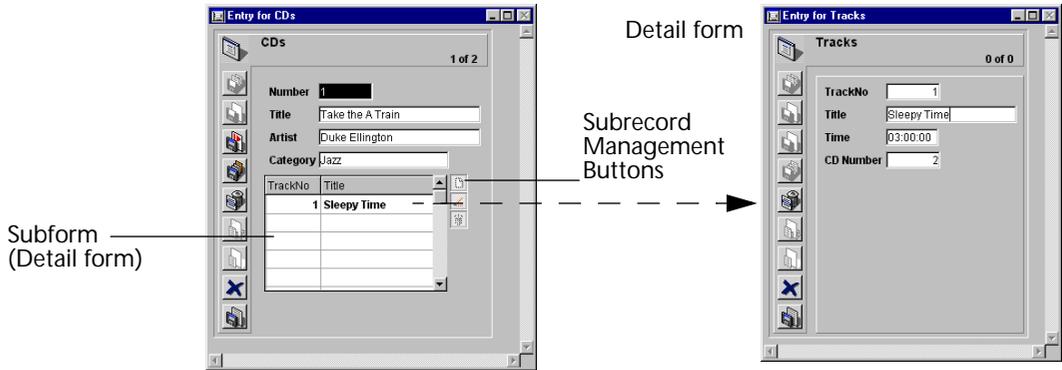
You can create a subform with the Form Wizard when you create a new form or you can add one or more subforms to an existing form using the Form editor. You must have first created the List form you want to use.

Adding a subform using the Form Wizard is described in the section [“Adding a Subform to the Form” on page 217](#).

1. If you have more than one subform on a Detail form, the subform that has focus has a small flashing triangle at its top left corner.

Entering Data in a Subform

Subforms can be used for data entry in two ways: the user can enter data directly in the subform or enter it in an input form. The figure below shows both a subform and an input form associated with it.



The form used as the subform is referred to as the *List form*. The input form is referred to as the *Detail form*.

You can allow the user to enter data through the List form and you can allow the user to double-click a row in the List form to display the Detail form. The following illustration shows the Subform page of the Object Properties window and the Property List:

Object Properties window

Property List

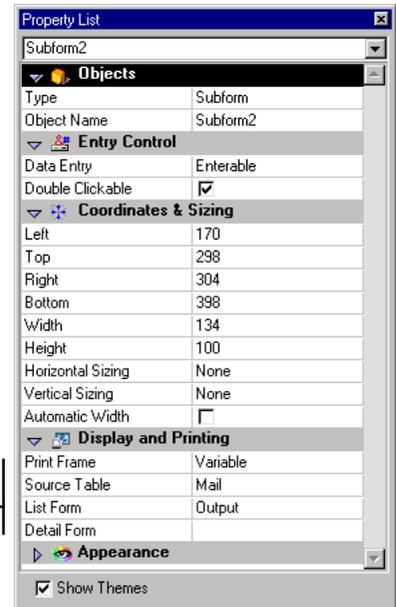
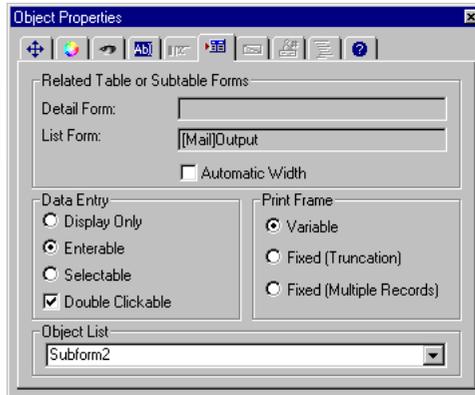


Table and form definition areas

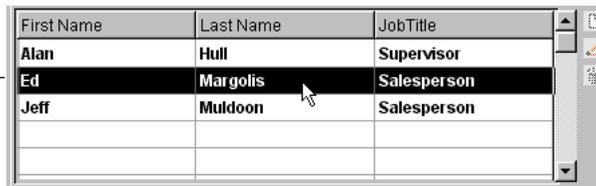
The Related Table or Subtable area shows the List form and Detail form for the subtable area. You specify these forms by dragging them from the Forms page of the Explorer to the subform area on the form. In the Property List, you can specify those forms by dragging them from the Explorer Window.

Data Entry Options for Subforms

The Data Entry area in the Subform page and the Display and Printing theme on the Property List includes the following controls:

- **Enterable** Allows data entry in the List form. If **Enterable** is not checked, the List form is used for display only. When the **Double-clickable** option is selected, the **Enterable** option also allows the user to edit the subrecord in the input form.
- **Display Only** Allows the user to view the records in the List form but not perform data entry. If this option is selected, the user will not be able to display the Detail form, even if the **Double-clickable** option is selected.
- **Selectable** During data entry, clicking on a selectable subform highlights an entire record — similar to highlighting a record in an output form. However, unlike its behavior in an output form, the selected record becomes the current record for its table (or subtable).

With **Selectable** and **Double-Clickable**, a user can double-click to use the Detail form



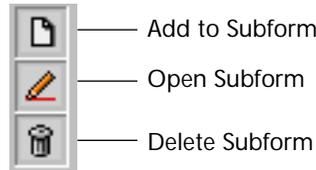
First Name	Last Name	JobTitle
Alan	Hull	Supervisor
Ed	Margolis	Salesperson
Jeff	Muldoon	Salesperson

- **Double Clickable** Allows data entry in the full page form. If **Double Clickable** is not checked, the user cannot use the Detail form.

You normally use an output form as the List form and an input form as the Detail form. If you do not specify a Detail form, 4th Dimension automatically uses the default input form for that table.

You can add custom buttons to control data entry for records in a subform. These button actions are **Open Subform**, **Delete Subform**, and **Add To Subform**. Any type of button—**Button**, **Highlight**, or

Invisible—can be used. For more information about adding these buttons, refer to [“Buttons” on page 382](#).



Displaying Data in a Subform

A subform typically displays related Many records using an automatic One to Many relation.

4th Dimension places a scroll bar on the right of the subform so that you can scroll through the list of records displayed. The scroll bar extends all the way up the right side, even alongside the column headings in the subform.

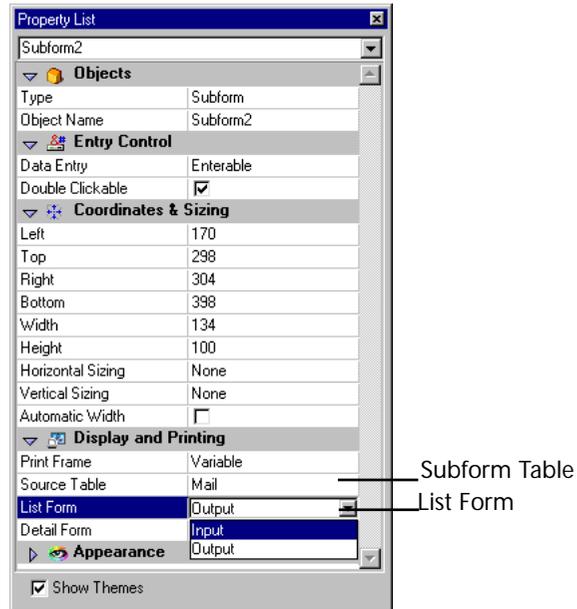
You can adjust the subform as necessary to display the records. The wider you make the area, the more columns can be displayed. The taller you make the area, the more records can be displayed. A subform area automatically displays a scroll bar so that the user can scroll through the records or subrecords.

Creating a Subform

You can create a subform in two ways:

- In the Form Wizard, using the Subform page in Advanced options,
 - In the Form editor, using the form and the Forms page of the Explorer.
For information on the Subform page of the Form Wizard, see the section [“Adding a Subform to the Form” on page 217](#).
- To create a subform in the Form editor:
- 1 Create a subform object using the subform tool  .
You can either drag the tool from the Tools palette or select the tool and draw the subform area in the form.
 - 2 In the Display and Printing theme of the Property List, select the source table for the subform.
OR
Open the Forms page of the Explorer and expand the table that contains the records you want to display.

3 In the Property List, select the List form that you want to use.



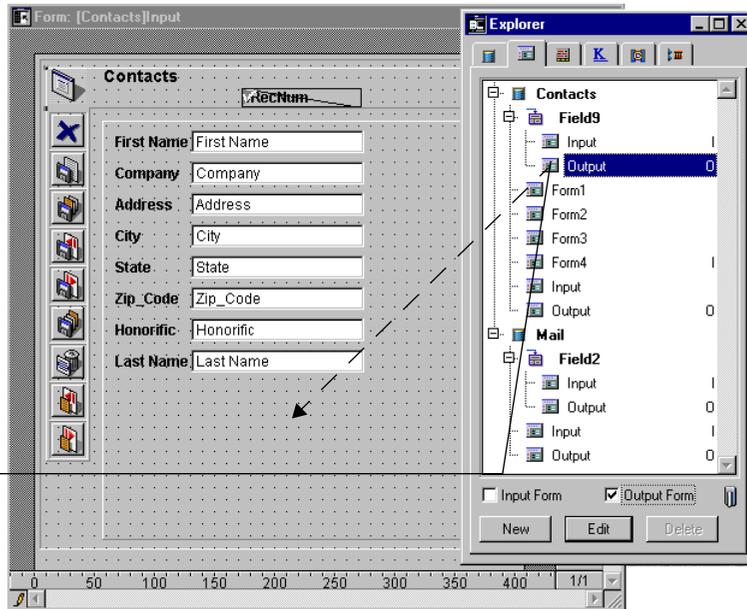
OR

Drag the List form you want to use as the subform from the Forms page of the Explorer to the input form.

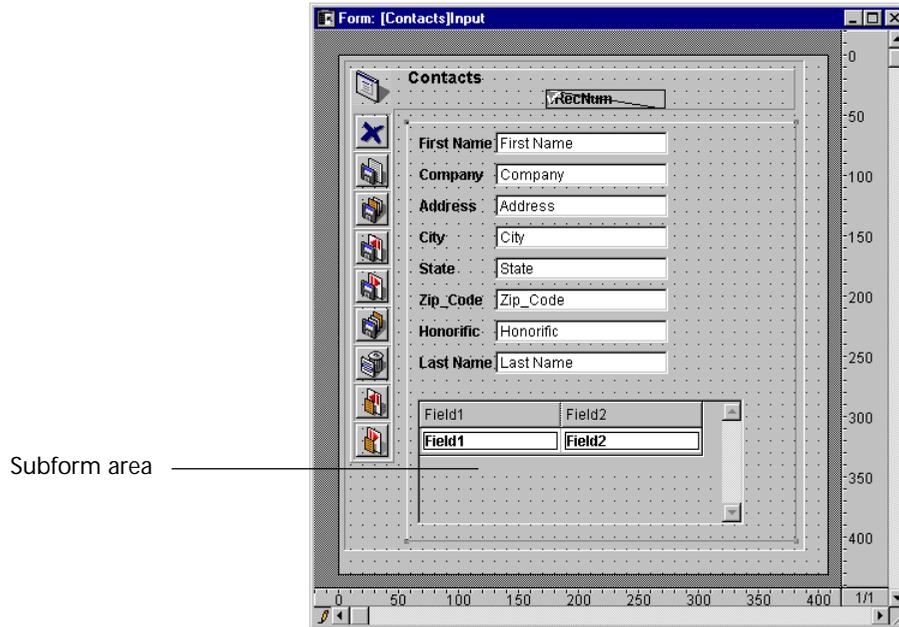
Note You can create a subform directly by dragging the subform from the Explorer. In this case the width of the subform will be defined by

4th Dimension automatically (for more information refer to [“Setting the Width of a Subform”](#) on page 436).

Drag the List form from the Explorer to a blank area on the form



4th Dimension creates a subform area on the form and displays the List form in the subform area.



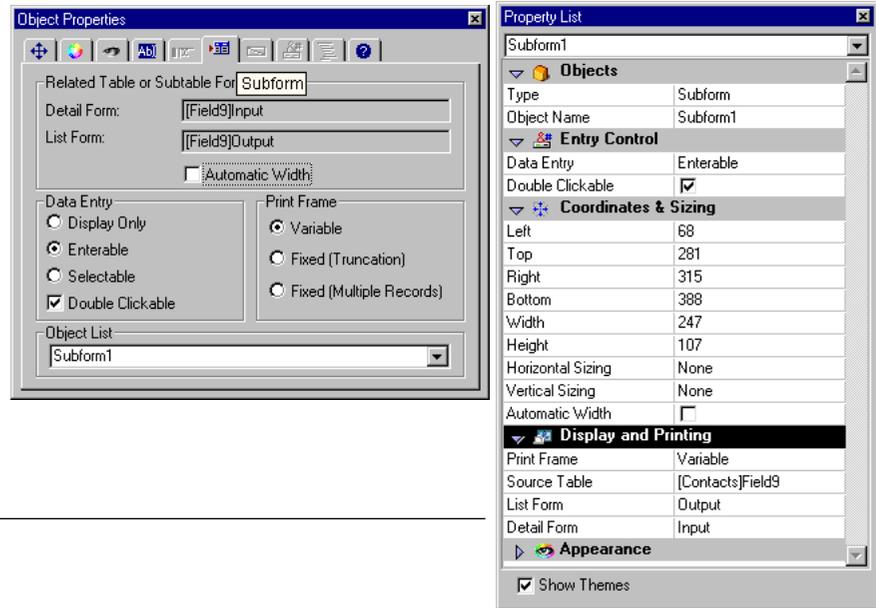
- 4 If you intend to make the subform area double-clickable, select the Detail form you want to use in the Property List.
OR
Hold down the Shift key and drag the name of the Detail form you want to use from the Forms page of the Explorer to the subform area on the form.

The names of both forms appear in the Related Tables area on the Subform page of the Object Properties window.

Object Properties window

Property List

Related Tables area



Dragging the Detail form name does not change the appearance of the subform area on the form itself.

If you drag the wrong List or Detail form to the subform area, you can replace it by dragging the correct form.

Setting the Type of Data Entry for the Subform

You can use the Enterable, Display Only, Selectable and Double-clickable radio buttons to specify the kind of data entry you want to have available for the subform.

For more information on data entry in subforms, refer to [“Entering Data in a Subform” on page 430](#).

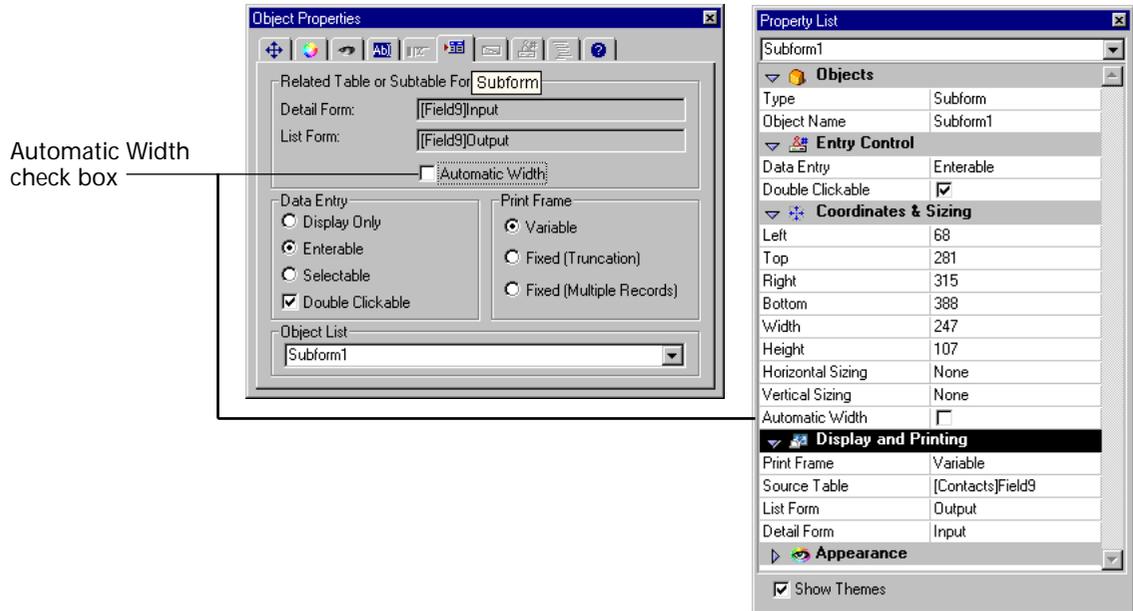
Setting the Width of a Subform

You can resize a subform area as you can any other object or you can let 4th Dimension automatically sizes the subform area so that all the fields in the List form are displayed. To select 4th Dimension's

automatic sizing feature, you can use the Automatic Width property to prevent designers from changing the width of a subform area.

Object Properties window

Property List



Note When you create a subform area by dragging it directly from the Explorer window, 4th Dimension automatically sets the width of the subform.

If you click the Automatic Width check box, you only change the height of the subform area when you drag a resizing handle, leaving the width correct. Changing the height allows you to display more or fewer records.

Changing other Subform Properties

After you have specified the List and Detail forms for the subform and set the data entry and printing properties, you can set other properties of the subform area. The other pages of the Object Properties window work as for any other active object type. For example, you can set the platform interface, appearance, and sizing and resizing options. Use the other pages of the Object Properties window to set any other properties you like. For more information, see [“Setting Object Properties” on page 372](#).

For more information on printing options for forms, refer to [“Printing Subforms, Pictures, and Text Fields” on page 471](#).

Modifying a Subform

You can modify the properties of a subform at any time:

- ▶ To modify a subform:
 - 1 Double-click the subform area in the form.
4th Dimension displays the Subform page of the Object Properties window or the Property List.
 - 2 Make any necessary changes.
You can specify different data entry options or make changes to the way the frame is printed.

6

Output Displays and Reports

Output forms are used for two purposes: listing records on screen and printing reports. This chapter explains the following operations:

- Using the Form Wizard to create output forms for listing records on-screen,
- Using the Form Wizard to create forms for output forms and printed reports,
- Using the Form editor to customize output forms and printed reports.

In many cases, you can create a report more quickly using the Quick Report editor. However, the Form editor gives you greater control over the final appearance of your report. For more information about using the Quick Report editor, refer to the *4th Dimension User Reference*.

Output Forms for Listing Records

A form that displays records as a columnar list contains separate areas:

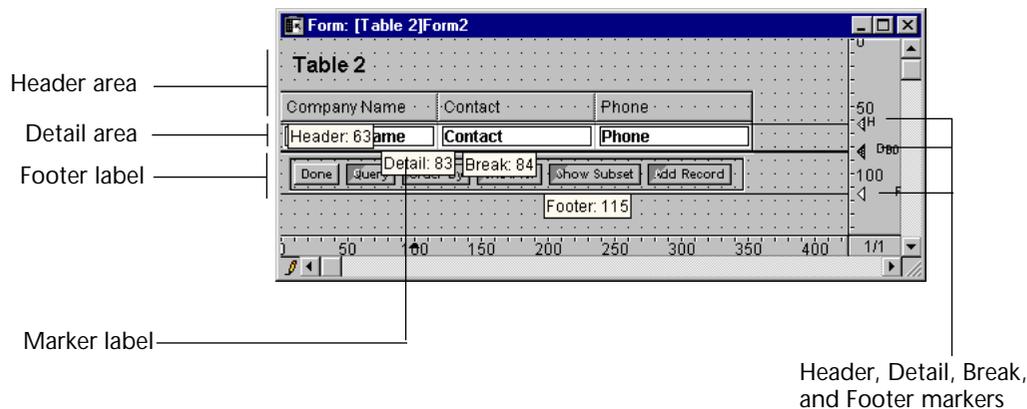
- **Header area** contains the report title, column headers, and form management buttons or objects (in custom applications only),
- **Detail area** contains the body of the report,
- **Footer area** contains buttons or summary calculations based on all the records in the report
- **Break area** contains text or graphics that appear after the list of records and summary calculations based on all the records or sub-groups of records.

When you create a List form using the Form Wizard, it automatically creates these areas for you. It places the form title (the table name) and the field names in the Header area and a one or more buttons in the Footer area¹. The fields you select are placed in a row in the Detail area. A small Break area is created but the Form Wizard puts nothing in it.

When you open the form in the Form editor, you can modify the size of each of these areas, modify the contents of any area, add objects to the Break area, and create additional Break areas for summary calculations.

The areas of the form that function as the Header, Detail, Break, and Footer areas are controlled by *output control lines*. By dragging the output control lines vertically, you can change the size of each area.

The following illustration shows an output form that was created using the Basic screen of the Form Wizard (3D Look).



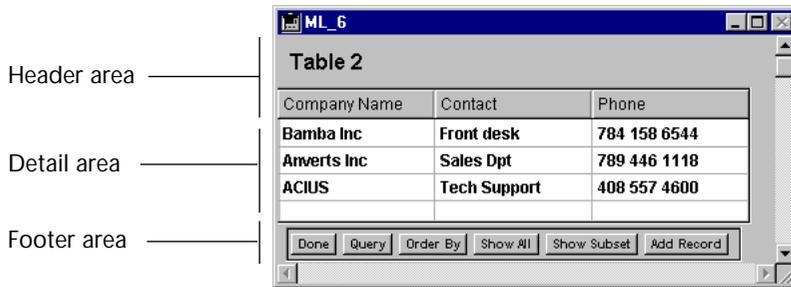
Note You can choose to hide/display markers and their labels. For more information, refer to [“Showing/Hiding Elements in the Form Editor” on page 232](#).

The dotted horizontal lines divide the report into Header, Detail, Break, and Footer areas. The area from the top of the form to the Header line is the Header area. Similarly, the area between the Header and Detail lines is the Detail area, and the Footer area extends from the top of the Break line² (labelled B0) to the Footer line. You adjust the

1. The exact contents of the Header and Footer areas depend on the specific Form Wizard options that you select.

sizes of each area by dragging the Header, Detail, Break, or Footer markers or their labels vertically.

When this form is used in the Custom Menus environment or a custom application, it looks like this:



The Detail area expands dynamically as the window is resized, while the Header and Footer areas remain a fixed size.

These areas work slightly differently in the User and Custom Menu environments. For more information, refer to the next section, [“Output Control Lines” on page 442](#).

In a form that lists records on screen, the Header area can include clickable and non-enterable objects such as buttons, radio buttons, hierarchical lists, and so on. For more information, refer to [“Output Control Lines” on page 442](#).

In a printed report, a Header area often contains the date, the time, and a running title as well as column labels. Records appear in the Detail area. A calculated total may appear in the Break area. The Footer area contains the page number.

2. For more information on Break areas refer to [“Reports with Breaks” on page 460](#).

The following illustration identifies the different areas as they appear in a printed report.

The diagram shows a printed report titled "Consumer Products Employees". The report is divided into four labeled areas:

- Header area:** Contains the title "Consumer Products Employees", the date "Date: 09/16/92", and the time "Time: 10:37 AM".
- Detail area:** Contains a table with 4 columns: First Name, Last Name, Title, and Salary. The table lists 15 employees and their respective salaries.
- Break area:** Contains a summary line: "Total salaries: \$504,597.00".
- Footer area:** Contains the page number "Page: 1" and the report title "Salary Report".

First Name	Last Name	Title	Salary
Smeldorf	Garbando	Clerk	\$19,610.00
Barbara	Doyen	Designer	\$43,210.00
Jeff	Eubanks	Secretary	\$29,310.00
Betsy	Ohren	Supervisor	\$35,590.00
Harry	Conrad	Salesperson	\$35,710.00
Dick	Lookert	Engineer	\$51,772.00
Walter	Brothers	Technician	\$29,320.00
Dave	Arnold	Salesperson	\$35,550.00
Susan	Grambo	Designer	\$36,300.00
Kathryn	Orbach	Salesperson	\$40,030.00
John	Ferras	Secretary	\$23,250.00
Bill	Horton	Engineer	\$41,895.00
Fritz	Ormolu	Fabricator	\$38,050.00
Bill	Conqueror	Engineer	\$45,000.00

A report may have additional Break areas for subtotals and other calculations. A report may also have additional Header areas that appear within the body of the report. The additional Header areas are used to identify subgroups. For an example of a report with several Header and Break areas, see the section [“Creating Additional Control Lines” on page 462](#).

Output Control Lines

You control the Header, Detail, Break, and Footer areas with the output control lines in the Form editor. You move the control lines vertically to allow more or less space for each area. Any object that you place in these areas is displayed or printed at the appropriate location.

The following explains how these areas work when the form is displayed or printed:

- Header area** The Header area is displayed at the top of each screen in the User and Custom Menus environments and is printed at the top of each page of a report. The Header area is defined as the area above the Header control line (H). You make the Header area smaller or larger by dragging the Header control marker vertically. You can use the Header area for column names, for instructions, additional information, or even a graphic such as a company logo or a decorative pattern.

You can also place active objects in the Header area of forms displayed using the DISPLAY SELECTION and MODIFY SELECTION commands.

Note This type of operation does not apply to forms used in the User environment or to subforms.

Only objects that are clickable and non-enterable can be inserted. This includes:

- buttons, 3D buttons, highlight buttons, picture buttons,
- pop-up menus/drop-down lists, picture pop-up menus and hierarchical pop-up menus,
- scrollable areas, hierarchical lists,
- radio buttons, 3D radio buttons, picture radio buttons,
- check box, 3D check box,
- thermometer, rulers, dials.

Note Combo boxes cannot be inserted, since they are enterable objects.

Automatic actions such as Validate, Cancel, or Automatic splitters can be assigned to the inserted buttons. The following events apply to the active objects you insert in the header area: On Load, On Clicked, On Header, On Printing Footer, On Double Clicked, On Drop, On Drag Over, On Unload. For more information on the MODIFY SELECTION and DISPLAY SELECTION commands, refer to the *4th Dimension Language Reference Manual*.

- **Detail area** The Detail area is displayed on the screen once for each record in the User and Custom Menus environments and is printed once for each record in a report. The Detail area is defined as the area between the Header control line and the Detail control line (D). You make this area smaller or larger by dragging the Detail control marker vertically. Whatever you place in the Detail area is displayed or printed once for each record. Most often you place fields or variables in the Detail area so that the information in each record is displayed or printed, but you can place other elements in the Detail area as well.
- **Break areas** Break areas are displayed once at the end of the list of records in the User and Custom Menus environments and are printed once after the records have been printed in a report. In the report above, the Break area is defined as the area between the Detail control line and

the Break control line (labeled B0). There can be other Break areas in your report. You make Break areas smaller or larger by dragging the Break control marker vertically. You can use a Break area to display information that is not part of the records (instructions, current date, current time, etc.) or to display a line or other graphic element that concludes the screen display. In a printed report, you can use a Break area for calculating and printing totals and other summary calculations.

- **Footer area** The Footer area is displayed on screen only in the Custom Menus environment and in custom applications. It is always printed at the bottom of every page of a report. The Footer area is defined as the area between the Break control line (B0) and the Footer control line (F). You make this area smaller or larger by dragging the Footer control marker vertically. You can use the Footer area to print graphics, page numbers, the current date, or any text you want at the bottom of each page of a report. For output forms designed for use on screen, the Footer area typically contains buttons that give the user options such as doing a search or sort, printing records, or putting away the current report.

Whenever any form is used for output, either for screen display or printing, the output control lines take effect and the areas display or print at designated locations. The output control lines also take effect when a form is used as the List form in a subform area.

The output control lines have no effect when a form is used for input.

Methods that are associated with objects in these areas are executed when the areas are printed or displayed as long as the appropriate events have been activated. For example, a object method placed in the Header area is executed when the On Printing Header event takes place.

You can create additional control lines to define additional Break areas and Header areas for a report. These additional areas allow you to print subtotals and other calculations in a report and to display other information effectively. Additional control lines are discussed in the section [“Reports with Breaks” on page 460](#).

Working with the output control lines and the areas they define are described in detail in subsequent sections of this chapter.

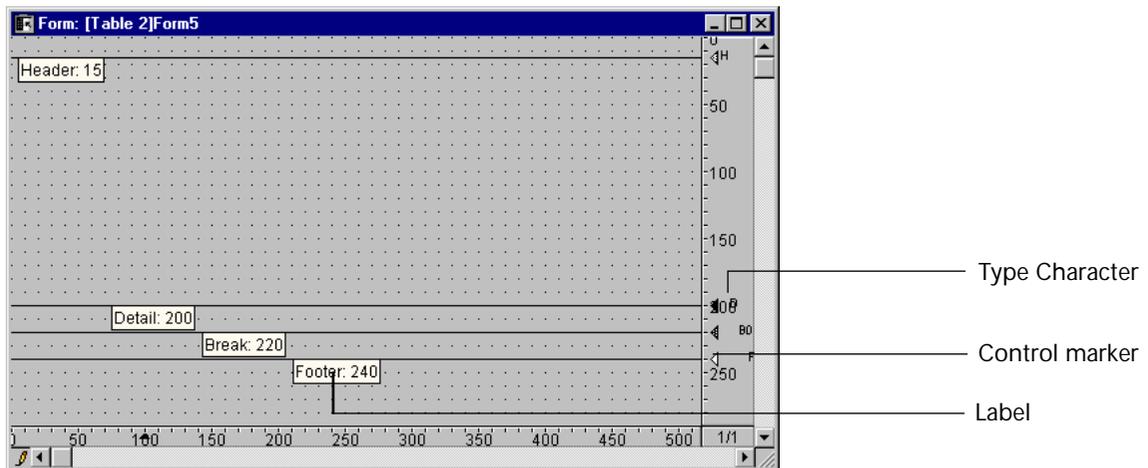
Moving Output Control Lines

You adjust the size of the Header, Detail, Break, and Footer areas by moving the output control markers.

Output control lines are displayed as dotted lines across the form. Each control line has an identifying marker and label that is displayed in the ruler. The control marker is the triangle in the ruler and the label is the letter or letters next to the marker. There is an additional label that you can display, it displays each control line's position and type. When you move a control line, the label displays its new position in real time.

You can also permanently display these marker labels, for more information, refer to [“Showing/Hiding Elements in the Form Editor” on page 232](#). Labels allow you to move control lines even when the rulers are not displayed.

The figure below identifies control markers and labels.



To move a control line, drag the control marker or the marker label vertically.

Holding down the **Shift** key while dragging a control marker moves all control lines below that control marker. For example, to drag all control lines together, hold down **Shift** and drag the Header marker. To move all control lines except the Header control line, **Shift+drag** the Detail marker.

The control lines cannot be dragged out of order. For example, if you attempt to drag a Footer control line higher than a Break control line, the drag operation automatically stops when the Footer marker reaches the Break marker.

You can place markers and control lines on top of one another. Placing one marker on top of another reduces its area to nothing, removing it from the report. For example, if you have nothing to print in a Break area, you can drag the Break marker on top of the Detail marker. Doing so prevents 4th Dimension from creating space for a Break area. The report can thus utilize all the space available on the page.

If you don't want to print any details, drag the Detail marker on top of the Header marker. If you don't need a Header, drag the Header marker to the very top of the form (at point 0).

Creating Output Forms

You use the Form Wizard to create an output form for each table in your database. The process is similar to the process you use for creating input forms. You simply select a different form type. If you use the Wizard's Advanced screen, a different set of options is available.

As with input forms, you can create an output form using either the Basic or Advanced screens of the Form Wizard.

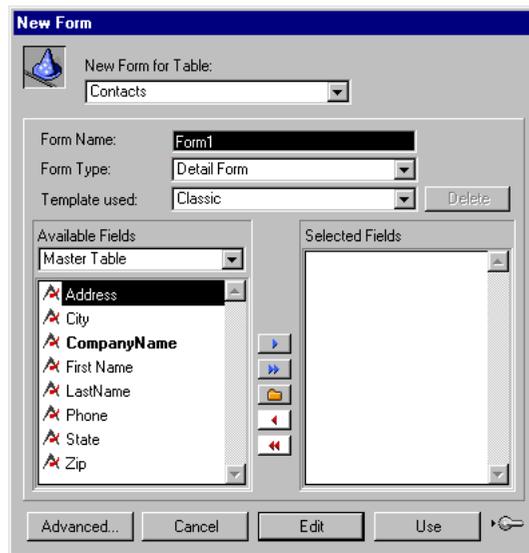
► To create an output form:

1 Choose New Form from the Design menu.

OR

Select the master table for the form in the Forms page of the Explorer and click New.

The Basic screen of the Form Wizard appears.



- 2 If you want to preview the effects of your settings, expand the Preview area by clicking the Preview Area icon.
- 3 If necessary, use the New Form for Table drop-down list to select the master table for the form.
The Fields list changes to reflect your selection.
- 4 Name the form by filling in a name in the Form Name area.
You can refer to the form by name using the language.
- 5 Choose List form from the Form Type drop-down list.
This selection instructs the Wizard to place the output control lines in the correct locations for lists, place the fields in a row in the Detail area, and add buttons to the Footer area (for use in custom applications only).
- 6 Choose a template that is appropriate for output forms.
The template controls several aspects of the appearance of the form, including font attributes, field label placement, the design of decorative rectangles surrounding fields, and platform interface.
4th Dimension ships with several templates. You can also create custom templates with the Form Wizard and add them to this list. For more information about adding custom templates, see the section [“Creating a Form Template” on page 220](#).
- 7 Select the fields you want on your form.

For complete information about selecting fields for the form, refer to the section, [“Selecting Fields for the Form” on page 199](#).

8 If you want to edit the new form in the Form editor, click Edit.

OR

If you want to switch to the User environment to use the form, click Use.

OR

If you want to customize the new form with the Forms Wizard’s advanced options, click Advanced.

Using the Form Wizard’s Advanced Options

When you create a List form, the Advanced screen of the Form Wizard has the following pages:

- **Fields** Similar to the Fields page for Detail forms, except that grouping fields is not relevant for List forms and is not available.
- **Styles** Same as the Styles page for Detail forms.
- **Options** Similar to the Options page for Detail form type, with the following differences.
 - In the Detail and List Form for Printing form type, the Form Size area has a Page Setup button. This button lets you choose the page size for the printed report. When you choose the page size, 4th Dimension adjusts the size of the form and preview area.
 - In the List form type, the Form Size area has a Target Width area. When you use these options, 4th Dimension tries to fit the fields in a particular width. For more information, see the section [“Using the Target Width Option” on page 451](#).

- In the Display Options area, you can add variables to printed reports that display the page number, printing date, and printing time.

List Form

Choose the form size, the label location and the display options:

Form Size

Target Width: 300 points

Truncate if necessary

Label Location

No label

On top of Columns

Display Options

Form Title

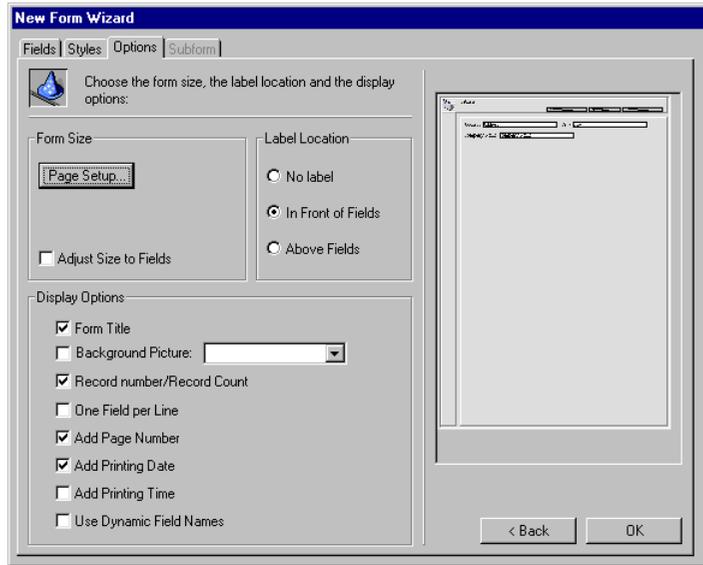
Use Dynamic Field Names

Table 2

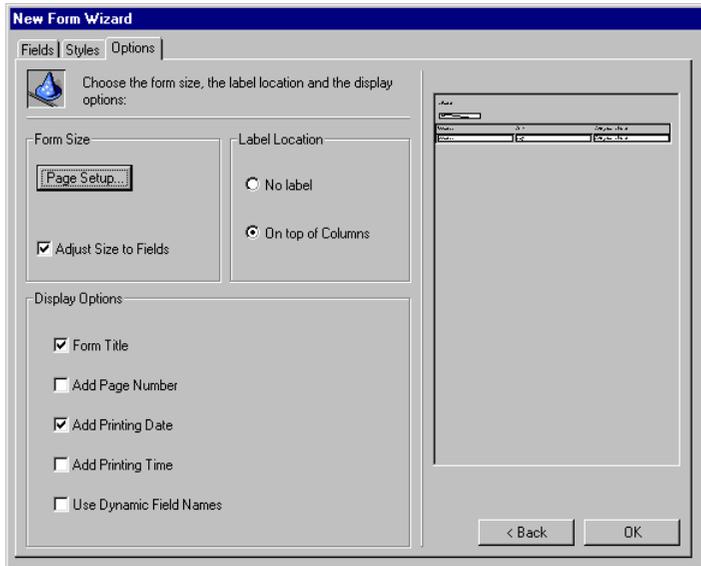
Address	City	Company Name
123 Main St	New York	ABC Corp

< Back OK

Detail Form for Printing

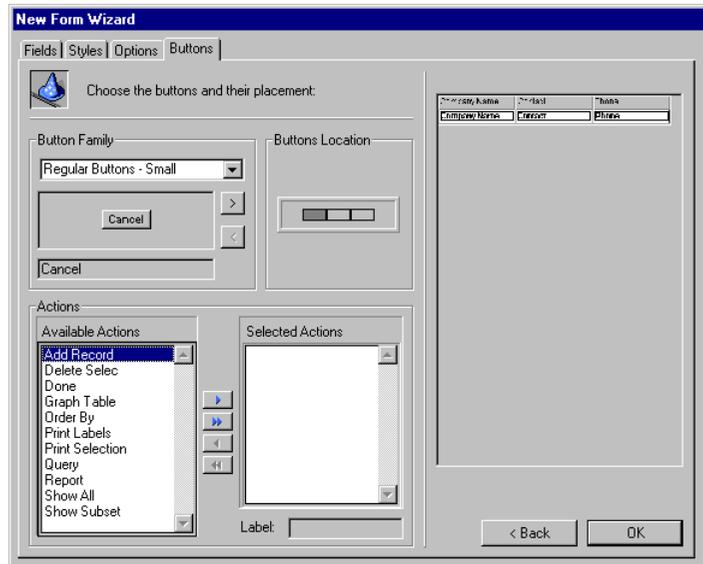


List Form for Printing



- **Buttons** Similar to the Buttons page for Detail forms, except that different buttons are available. The Buttons page is not available for the List and Detail forms for Printing form types.

List Form



The buttons are not automatic in the same sense as buttons for input forms; instead, 4th Dimension automatically creates an object method for each button. Using the Form editor, you can edit the object methods.

For more information on working with the Buttons page, see the section [“Customizing Buttons on the Form” on page 212](#).

Using the Target Width Option

When you create a List form for screen display, you can use an Advanced option to tell the Form Wizard to take a target width into account when it creates the form. If you do not use this option, the width of the output form will be based on the sum of the field lengths of the fields that you add to the form.

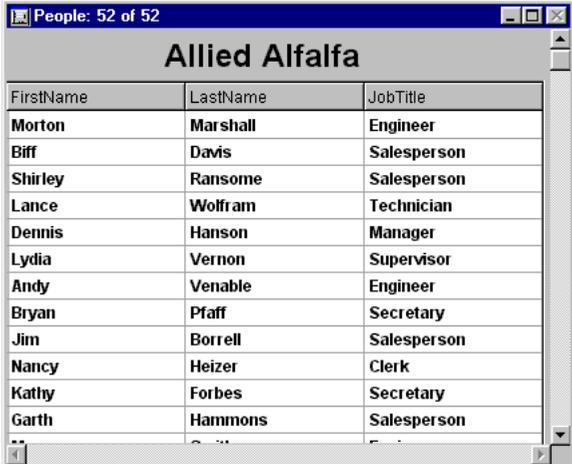
If you click Target Width and enter a target width in pixels, the Form Wizard will try to fit all the fields on the form by reducing the widths of the fields. If you also click Truncate if Necessary, the Form Wizard will also remove one or more fields to make the form width less than or equal to the target width. If you don't check Truncate if Necessary, the form's width may be slightly larger than the target width.

Modifying an Output Form in the Form Editor

A List form that is created using the Form Wizard works well as a listing of records. If needed, you can make the following simple modifications:

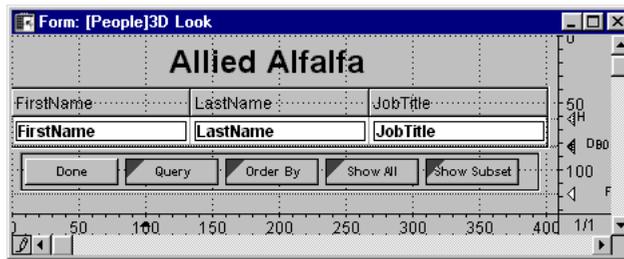
- Replace fields with variables and add methods,
- Use the platform interface, appearance, font attributes, fill, border, or color options to modify individual objects on the form,
- Change the widths of the fields or variables,
- Move the output control lines,
- Add a custom graphic in the Header area,
- Add variables in the Header area (for List forms designed to be displayed on screen in custom applications — refer to [“Output Control Lines” on page 442](#)).

The following is a custom output form being used for displaying records in the User environment. At the top of the screen is a company name as well as column names for the information below.



FirstName	LastName	JobTitle
Morton	Marshall	Engineer
Biff	Davis	Salesperson
Shirley	Ransome	Salesperson
Lance	Wolfram	Technician
Dennis	Hanson	Manager
Lydia	Vernon	Supervisor
Andy	Venable	Engineer
Bryan	Pfaff	Secretary
Jim	Borrell	Salesperson
Nancy	Heizer	Clerk
Kathy	Forbes	Secretary
Garth	Hammons	Salesperson

Here is what the design for this form looks like in the Form editor:



The control lines have been moved to accommodate the company name. This form uses the 9-point font size so that more records fit on a screen.

Displaying Several Lines Per Record

You may want to display fields on more than one line. 4th Dimension allows you to use several lines for each record. Expand the Detail area so that more than one line appears in this area. When you use several lines for a single record, graphic elements such as lines and boxes can be useful to separate fields and records.

Modifying Output Forms for Reports

Printed reports, unlike screen display forms, can make use of the Break area at the end of the report. For material that appears at the bottom of each page, printed reports use the Footer area.

This section provides suggestions and directions specifically for designing forms to be used for printing. It covers lists, reports that print one record per page, expandable areas for printing invoices and variable text fields, form letters, and custom mailing labels.

Printed Columnar Reports

Lists that display columns of information are common in printed reports. You might publish lists of telephone numbers, prices, results, specifications, or parts.

When you create a columnar report using the Form Wizard, you choose List Form for Printing as the Form Type.

Like the lists you design to display records on the screen, a printed list presents columns of information, can include column headings above each column, and can use graphic elements to enhance or clarify the report. The Break area, which is printed once at the end of the report,

is used for calculating totals. For a discussion of using methods for calculating totals, see the section [“An Example Report” on page 468](#).

One Record Per Page Reports

You may need to print one record per page. For example, you may want to use an invoicing database to print a copy of each invoice for your records.

When you need to create such a report, you choose **Detail Form for Printing** as the Form Type in the Form Wizard.

Place the Header (H) control line at the top of the page and arrange the fields and other report elements below it.

If your form displays records in a subform, be sure that the subform is set to print with fixed frame so that the records do not wrap onto additional pages. For information about printing with fixed frame, see the section [“Printing Subforms, Pictures, and Text Fields” on page 471](#).

Drag the Detail (D), Break (B0), and Footer (F) control lines to the bottom of the page to ensure that only one record is printed per page.

Using Subforms

You are probably very familiar with an invoice. A typical invoice shows a name and billing address, a shipping address, a series of items or services purchased, and a total.

An invoicing database includes an output form for printing full-page invoices. For example, an invoice draws information from two tables: an [Orders] table that provides the customer information (bill to and ship to addresses) and a [Line Items] table that provides the line items. The total for the order is calculated and kept in the [Orders] table.

The form for an invoice is created in the [Orders] table and uses a subform area for the line items. The subform area can expand during printing to print all the line items, even if the invoice requires a second page. For more information, see the section [“Printing Subforms, Pictures, and Text Fields” on page 471](#).

The Detail area is expanded to the full-page size. One invoice is printed for each sales order, but as many line items as necessary are printed in the line items area of the invoice.

Report with a Text Field

Many databases allow the user to enter notes or descriptions in a text field. That same text field can be printed as an expandable area on the report.

Text fields can be expanded during printing to accommodate all of the data. For more information about printing text fields, see the section [“Printing Subforms, Pictures, and Text Fields” on page 471](#).

Custom Mailing Labels

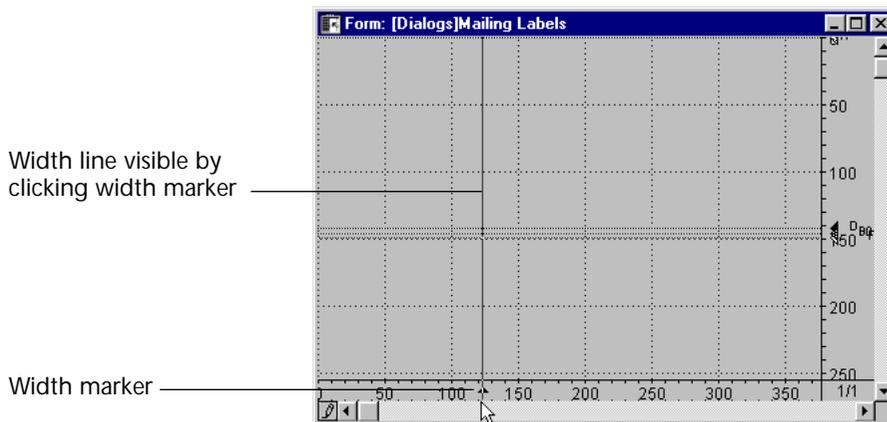
If you want to create special mailing labels, you can design a custom output form for them. The design can use graphic elements, any available fonts, and variables.

Tip It is often quicker and easier to create mailing labels using the Label Wizard in the User environment. For more information about the Label Wizard, refer to the *4th Dimension User Reference*.

► To create a label report form:

- 1 Set the label width by dragging the width marker on the bottom ruler of the Form editor.

The label width determines how many labels 4th Dimension prints across the page. The width marker is shown in the figure below.



- 2 Design the label form to the left of the width marker.
Labels can contain fields, active objects, graphic objects, text, and methods. A subform cannot be printed in a label.
- 3 Go to the User environment.
- 4 Choose Labels from the Report menu.
4th Dimension displays the Label editor.
- 5 Choose the form from the Form to Use drop-down list in the Label editor.

For complete information about using the Label editor, refer to the *4th Dimension User Reference*.

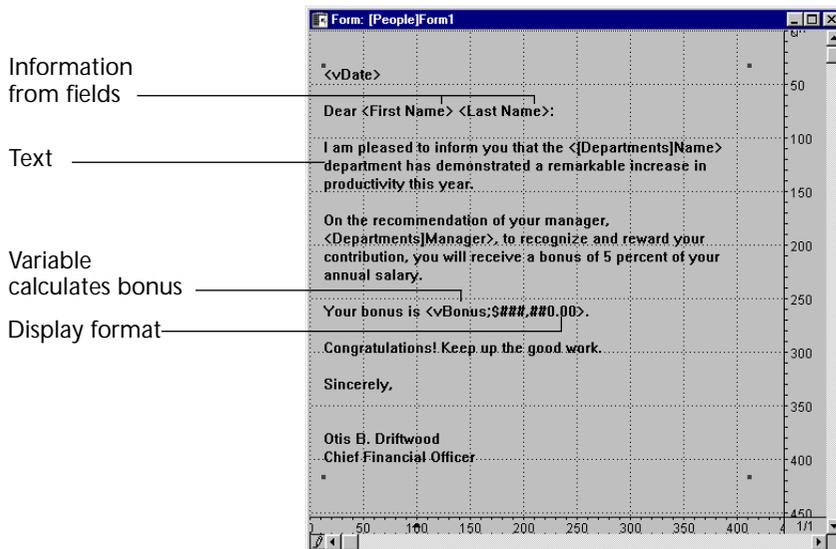
Creating Mail-Merge Documents

You can handle mail-merge tasks using an output form that embeds fields, field or table labels, or variables in a static text area. You can create documents that are the same in every respect except for names, addresses, and any special calculations you want to perform.

Mail-merge documents may be useful when you want to announce a special offer or inform your customers or clients of a business development (such as a move to a new location or a significant personnel change). Another typical use of mail-merge is to inform people that their account is due. You can create a variable and a method to calculate the exact amount.

When you create a form that does mail-merge, use the Detail Form for Printing option in the Form Type drop-down list in the Form Wizard. In the Form editor, create a text area that will contain both the static text and the fields, variables, or table or field labels that will change for each record. You then embed fields or variables in the text area. During printing, values from the fields or variables are inserted in the text.

The figure below shows fields and variables placed in a text area.



- ▶ To create an output form for mail-merge:
 - 1 In the Form Wizard, create a new form using the Detail Form for Printing option in the Form Type drop-down list.
 - 2 Click Edit to open the form in the Form editor.
 - 3 Select the Text Area tool and create a text area in the Detail area.
 - 4 Type the text you want in your form letter, placing field and variable names between less than (<) and greater than (>) symbols where you want information from fields or variables.

You can use a field from any table in the database. Fields from the master table do not have to specify the table name; they can be entered like this: <field>. Fields from other tables must specify the table name; they are entered like this: <[tablename] field>. When the form is printed, the information from the field for each record replaces the <field> element in the text area.

To insert table labels, enter: <?[Table Name]> or <?[N]> where N is the creation order for the table. To insert field labels, enter: <?[Table Name]Field Name> or <?[X]Y> where X is the creation order for the table and Y is the creation order for the field or <?Y> to insert a field of the current table. For more information, refer to [“Inserting Dynamic Table and Field Names” on page 329](#).

A variable must be assigned a value in an object or form method.

You can specify how an embedded field or variable is displayed by inserting a semicolon followed by a display format directly after the field or variable name. For example, the mail-merge document shown above includes a display format for the vBonus variable. The embedded variable <vBonus;\$###,##0.00> calculates the amount of the person’s bonus and displays it in a dollar format. For more information about display formats, see the section [“Display Formats” on page 357](#).

4th Dimension provides a shortcut for inserting fields in a text area. The shortcut allows you to choose the field name from a drop-down menu.

- ▶ To insert a field using the shortcut:
 - 1 Click to place the insertion point where you want the field.
 - 2 Position the pointer in the text area and hold down the Alt key (on Windows) or Option key (on Mac OS) while you press and hold down the mouse button.

4th Dimension displays a pop-up menu of fields from the master table from which you can choose the field you want.

To choose a field from another table in the database, hold down Shift+Alt (on Windows) or Shift-Command (on Macintosh) while you press and hold down the mouse button.

4th Dimension displays a hierarchical menu of tables and fields in the database. You can choose the table and field you want.

4th Dimension places <field> or <[table]field> in the text area at the insertion point.

When the report is printed, the values of the fields and variables embedded in the letter appear.

05/22/92

To Bob Thomas:

I am pleased to inform you that the Engineering department has demonstrated a remarkable increase in productivity this year.

On the recommendation of your manager, Mr. Twilling, to recognize and reward your contribution, you will receive a bonus of 5 percent of your annual salary.

Your bonus is \$2,250.00

Congratulations! Keep up the good work.

Sincerely,

Otis B. Driftwood
Chief Financial Officer

Basic Steps for Creating a Printed Report

You can perform most or all of these steps depending on the nature of the report you want to print. The actual order of steps is not critical. Typically, you design a report, preview it on screen, and then return to the form to make adjustments. The order given here is typical:

- 1 Create a form using the Form Wizard.

For information about creating a form, refer to [Chapter 3](#).

- 2 Move the control lines so that you will have enough space to place the various elements of the report.

For information about moving control lines, see the section “[Moving Output Control Lines](#)” on page 445.

- 3 Move elements into position.

You can place fields, text, non-enterable objects, and graphic objects. Take time to align the objects to one another and to check the justification of data to be displayed in fields and active objects.

4th Dimension displays page size guidelines in the form so that you can place elements for printing.

For information about placing fields and active objects in a form, refer to [Chapter 5](#).

- 4 Adjust the control lines if necessary.

Working with the relationship between control lines and elements to be printed, you can create the right amount of space for printing the Header, Detail, Break, and Footer elements.

- 5 If necessary, create methods to calculate values, concatenate strings of characters, print additional text, and display the date, time, and page numbers.

For information about creating methods, refer to “[Using Object Methods with Fields and Objects](#)” on page 422. For complete information about using object methods, refer to the *4th Dimension Language Reference*.

For information about calculating subtotals and other summaries (averages, counts, etc.), refer to “[Reports with Breaks](#)” on page 460.

Be sure that you activate the appropriate events in the Events section of the object properties for each method you use; otherwise the method will not execute during printing.

- 6 Go to the User environment to test the report.

- 7 Create the selection of records you want to use for the report.

For information about creating a selection of records, refer to the *4th Dimension User Reference*.

- 8 Sort the records according to how you have designed the report.

For information about sorting records, refer to the *4th Dimension User Reference*.

9 Preview the report by printing it to the screen.

The report form uses the Page Setup settings that were in effect when the form was created in the Design environment. If you make any changes to the Page Setup dialog box, be sure to make the same changes in the Design environment.

If the report form needs to be adjusted, return to the Design environment to make any necessary changes.

10 Print the report.

For information about printing in the User environment, refer to the *4th Dimension User Reference*.

Reports with Breaks 4th Dimension can print reports that work with Breaks and Break Headers. A Break is created when you sort the records.

Suppose you have a collection of compact discs that you keep track of in a 4th Dimension database and you want to print a list that arranges the information by artist. When you sort the records by artist, all the records fall into distinct groups. The “Break” occurs after the last record in each group is printed.

Here is how the report looks when previewed to screen:

Album Report by Artist			Page 4
Album Title	Disc #	Record Company	
THE PLANETS	023	Deutsche Grammophon	
THE PLANETS, SUITE DE BALLET, OP. 10	250	Enigma Classics	
Harry Belafonte			
ALL TIME GREATEST HITS VOL.1	025	BMG Music	
PARADISE IN GAZANKULU	028	Capitol Records	
Hector Berlioz			
SYMPHONIE FANTASTIQUE	192	Enigma Records	
Hiroshima			
ANOTHER PLACE	036		
Huey Lewis And The News			
FORE!	031	Chrysalis Records	
INXS			
KICK	203	Atlantic Recording Corporation	
Jean-Luc Ponty			
COSMIC MESSENGER	029	Atlantic Recording Corporation	
Jerry Goodman			
ARIEL	073	Private Music, Inc.	
Joaquin Rodrigo			
CONCIERTO DE ARANJUEZ	233	Enigma Classics	
Joe Sample			
SPELLBOUND	179	Warner Bros. Records Inc.	
Johann Sebastian Bach			
BACHBUSTERS	080	TELARC DIGITAL	
BRANDENBURG CONCERTOS NOS. 1, 2, & 3	125	Enigma Records	
BRANDENBURG CONCERTOS NOS. 4, 5 & 6	163	Enigma Records	
ORGAN FAVOURITES	248	Enigma Classics	
Johann Strauss, Jr.			
STRAUSS FESTIVAL VOLUME 1: FAMOUS WALTZES, PC 096		Enigma Records	
STRAUSS FESTIVAL VOLUME 2	159	Enigma Records	
Johannes Brahms			
HUNGARIAN DANCES NOS. 1-21	243	Enigma Classics	
HUNGARIAN DANCES NOS. 1-21	146	Enigma Records	
SYMPHONY NO.4 ACADEMIC FESTIVAL OVERTURE	127	Enigma Records	
Joni Mitchell			
COURT AND SPARK	108	Asylum Records	
Joseph Haydn			

4th Dimension provides features that you can use to display the information attractively. Here is this form in the Design environment.



In order to generate a report that uses Break levels and Headers, you must first initiate Break processing. For more information about the methods you can use to initiate Break processing, refer to “Initiating Break Processing” on page 464.

A Break Header is printed once before the group of records it refers to and a Break is printed once after the group of records it refers to. In the illustration on the previous page, the Break is called a “level 1 Break” and the Break Header is called a “level 1 Break Header,” because the Break occurs as a result of the first sort field.

You can use up to nine break levels. If you use Subtotal to initiate Break processing, you need to sort on one more field than the number of Breaks you use. In this case, if you use one Break level, you must sort on two fields. If you use three Break levels, you must sort on four fields.

This section explains how to create reports using Breaks. A full-scale example is provided at the end of this section. If you have any trouble understanding anything in this section, try reading the whole section and then coming back to the trouble spot. The concepts are interrelated and are often easier to understand in context than by themselves.

Creating Additional Control Lines

The report examples shown in this section use Break levels and Break Headers. To create areas for these features, you create additional control lines.

The Form editor always starts with the original control lines, labeled H, D, B0, and F. B0 stands for “Break at level 0.” Level zero takes in all the records; it occurs after all the records are printed. Additional Break control lines are designated with numbers. A control line labeled B1 stands for “Break at level 1.” A level one Break occurs after the records grouped by the first sort field are printed.

Label	Explanation	Prints after groups created by:
B1	Break at level 1	First sort field
B2	Break at level 2	Second sort field
B3	Break at level 3	Third sort field

H stands for “Header,” which is printed at the top of each page. Additional Header control lines are associated with Breaks. H1 stands

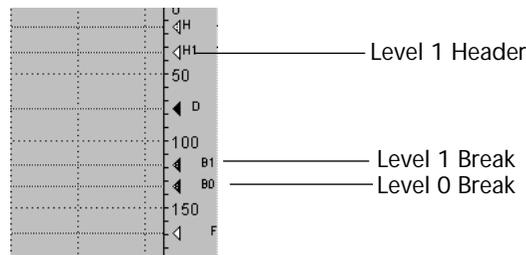
for “Header at level 1.” A level 1 Header is printed just before the records grouped by the first sort field are printed.

Label	Explanation	Prints before groups created by:
H1	Header at level 1	First sort field
H2	Header at level 2	Second sort field
H3	Header at level 3	Third sort field

You create additional control lines by holding down the **Alt** key (on Windows) or **Option** key (on Mac OS) while clicking the appropriate control marker. You use a Break control line to create a Break area for the designated level. You use a Break Header control line to create a Break Header area at the designated level.

If you use the Subtotal function to initiate Break processing, you should create a Break area for every level of Break that will be generated by the sort order, minus one. If you do not need anything printed in one of the Break areas, you can reduce its size to nothing by placing its marker on top of another control line. If you have more sort levels than Break areas, the last Break area will be repeated during printing.

The figure below shows additional control lines.



► To create a new Break or Break Header control lines:

- 1 Hold down the **Alt** key (on Windows) or **Option** key (on Mac OS) and click any Break (**B**) or Break Header (**H**) marker.

4th Dimension creates a new Break or Break Header control line.

The new line is positioned behind the existing control line; to see the new control line, you need to drag the existing line off of it.

- 2 Drag the existing line away from the new line.
- 3 Position both lines where you want them.

- ▶ To delete Break or Break Header control lines that you have created:
 - 1 Hold down the Ctrl key (on Windows) or Command key (on Macintosh).
 - 2 Click on the Break, Break Header, or label of the control line that you want to delete.

4th Dimension deletes the control line and, if necessary, renumbers the remaining lines.

You cannot delete the original control lines H, D, B0, and F.

Initiating Break Processing

To allow 4th Dimension to print Break Header areas, accumulate subtotals, and perform other aspects of Break processing, you must first initiate Break processing in the report form. You initiate Break processing by either:

- Placing the Subtotal function in an object or form method,
- Executing the BREAK LEVEL and ACCUMULATE commands before printing the report.

If the database will be used in interpreted mode only, you can use whichever method you prefer. However, you must use BREAK LEVEL and ACCUMULATE to initiate Break processing if the database will be used in compiled mode. In addition, if you use the Subtotal function, you must also sort the records on one more sort field than the number of Break levels you use. For example, if you use two levels of Breaks in your report, you must sort on three fields.

For more information about initiating Break processing, refer to the *4th Dimension Language Reference*.

Reports with Subtotals

This section describes in detail how the sort order affects reports and how to use additional Break areas for creating subtotals.

4th Dimension can automatically calculate and print totals and subtotals. The figure below shows a report that calculates subtotals for each customer and a total at the end of the report.

Sort level 1:
Customer

Sort level 2:
Product

Subtotal printed at level 1 Break

Total printed at level 0 Break

Z ^{ic} Industries		Sales Report	
Customer	Product	Date	Price
American Data	ATN 700	9/14/88	\$12,450
American Data	STS 1000	3/17/88	\$22,450
American Data	STS Service	4/17/88	\$3,300
American Data	Training Class	6/3/88	\$4,500
Subtotal for American Data			\$42,700
Horizon Services	ATN 850	10/18/88	\$25,364
Horizon Services	STS 1000	11/17/88	\$24,123
Horizon Services	STS 3000	5/7/88	\$74,250
Subtotal for Horizon Services			\$123,737
James Research	ATN 500	6/22/88	\$8,900
Subtotal for James Research			\$8,900
Omni Data Service	ATN 850	1/30/88	\$20,980
Omni Data Service	ATN 850	10/5/88	\$7,900
Omni Data Service	STS 1000	2/14/88	\$24,360
Omni Data Service	STS 3000	6/22/88	\$53,252
Omni Data Service	STS 3000	4/25/88	\$71,025
Omni Data Service	STS 3000	10/1/88	\$55,230
Omni Data Service	STS 3000	9/25/88	\$47,250
Omni Data Service	STS 4000	7/14/88	\$95,420
Omni Data Service	STS 4000	8/3/88	\$89,740
Omni Data Service	STS 4000	5/17/88	\$92,450
Omni Data Service	Training Class	2/5/88	\$4,500
Omni Data Service	Training Class	7/7/88	\$4,500
Subtotal for Omni Data Service			\$566,607
Thomas Info Systems	ATN 700	1/27/88	\$12,780
Thomas Info Systems	STS 2000	6/22/88	\$36,425
Subtotal for Thomas Info Systems			\$49,205
Total			\$791,149

These records have been sorted by customer and product. After the records for each customer have been printed, the subtotal for the customer is calculated and printed. After the records for the entire report have been printed, the total is calculated and printed.

As you can see in the figure above, the subtotals are calculated and printed after the records for each customer. 4th Dimension knows when to perform the calculation and print the subtotal because it has been instructed to do so when the value in the first sort field changes (where it “breaks”).

The customer subtotal is calculated in what is called a level 1 Break because it is based on a change in the first sort level (in this case, the Customer field). The grand total is calculated in a level 0 Break. A

level 0 Break includes all the records and occurs at the end of the report.

The figure below shows another example of calculations during a Break, using the same records, but with a different sort order to create a different Break. This time the records have been sorted by product and customer. The subtotals are calculated when the value in the Product field changes. This is still a level 1 Break, but the Break is on a different field.

Sort level 2:
Customer

Sort level 1:
Product

Zac Industries		Sales Report		
Customer	Product	Date	Price	
James Research	ATN 500	6/22/88	\$8,900	
Subtotal for ATN 500			\$8,900	
American Data	ATN 700	9/14/88	\$12,450	
Thomas Info Systems	ATN 700	1/27/88	\$12,780	
Subtotal for ATN 700			\$25,230	
Horizon Services	ATN 850	10/18/88	\$25,364	
Omni Data Service	ATN 850	1/30/88	\$20,980	
Omni Data Service	ATN 850	10/5/88	\$7,900	
Subtotal for ATN 850			\$54,244	
American Data	STS 1000	3/17/88	\$22,450	
Horizon Services	STS 1000	11/17/88	\$24,123	
Omni Data Service	STS 1000	2/14/88	\$24,360	
Subtotal for STS 1000			\$70,933	
Thomas Info Systems	STS 2000	6/22/88	\$36,425	
Subtotal for STS 2000			\$36,425	
Horizon Services	STS 3000	5/7/88	\$74,250	
Omni Data Service	STS 3000	6/22/88	\$53,252	
Omni Data Service	STS 3000	4/25/88	\$71,025	
Omni Data Service	STS 3000	10/1/88	\$55,230	
Omni Data Service	STS 3000	9/25/88	\$47,250	
Subtotal for STS 3000			\$301,007	
Omni Data Service	STS 4000	7/14/88	\$95,420	
Omni Data Service	STS 4000	8/3/88	\$89,740	
Omni Data Service	STS 4000	5/17/88	\$92,450	
Subtotal for STS 4000			\$277,610	
American Data	STS Service	4/17/88	\$3,300	
Subtotal for STS Service			\$3,300	
American Data	Training Class	6/3/88	\$4,500	
Omni Data Service	Training Class	2/5/88	\$4,500	
Omni Data Service	Training Class	7/7/88	\$4,500	
Subtotal for Training Class			\$13,500	
Total			\$791,149	

Subtotal printed
at level 1 Break

Total printed at
level 0 Break

Additional Break Levels

You can provide additional summary calculations by adding another sort level and another Break level.

The following figure shows sales records sorted by customer, product, and salesperson. Summary calculations show two sets of subtotals: one subtotal for each customer, and, within each customer, subtotals for each product for the customer. Finally, this report calculates a total for

the entire company. These are examples of calculations performed at level 2 Breaks, at level 1 Breaks, and at the level 0 Break.

Zic Industries **Sales Report**

Customer	Product	Date	Price
Omni Data Service	ATN 850	1/30/88	\$20,980
Omni Data Service	ATN 850	10/5/88	\$7,900
Subtotal for ATN 850			\$28,880
Omni Data Service	STS 1000	2/14/88	\$24,360
Subtotal for STS 1000			\$24,360
Omni Data Service	STS 3000	4/25/88	\$71,025
Omni Data Service	STS 3000	6/22/88	\$53,252
Omni Data Service	STS 3000	9/25/88	\$47,250
Omni Data Service	STS 3000	10/1/88	\$55,230
Subtotal for STS 3000			\$226,757
Omni Data Service	STS 4000	5/17/88	\$92,450
Omni Data Service	STS 4000	7/14/88	\$95,420
Subtotal for STS 4000			\$187,870
Omni Data Service	Training Class	2/5/88	\$4,500
Omni Data Service	Training Class	7/7/88	\$4,500
Subtotal for Training Class			\$9,000
Subtotal for Omni Data Service			\$476,867
Thomas Info Systems	ATN 700	1/27/88	\$12,780
Thomas Info Systems	ATN 700	6/22/88	\$24,745
Subtotal for ATN 700			\$37,525
Thomas Info Systems	STS 4000	8/3/88	\$89,740
Subtotal for STS 4000			\$89,740
Subtotal for Thomas Info Systems			\$127,265
Total			\$604,132

Annotations:

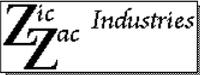
- Subtotal printed at a level 1 Break (points to the subtotal for ATN 700)
- Subtotal printed at a level 2 Break (points to the subtotal for STS 4000)
- Total printed at a level 0 Break (points to the grand total)

The subtotal calculations are performed only for the group of records that precedes the Break. For example, a subtotal is calculated for each product sold to each customer. The subtotal for the customer is calculated for all products sold to that customer.

Summary Reports

You can create a report that prints only summary information. Such a report displays only the subtotals and totals with appropriate

additional text. The following illustration shows a report with only summary information.

 Summary Sales Report	
Subtotal for ATN 850	\$28,880
Subtotal for STS 1000	\$24,360
Subtotal for STS 3000	\$226,757
Subtotal for STS 4000	\$187,870
Subtotal for Training Class	\$9,000
Subtotal for Omni Data Service	\$476,867
Subtotal for ATN 700	\$37,525
Subtotal for STS 4000	\$89,740
Subtotal for Thomas Info Systems	\$127,265
Total	\$604,132
December 1, 1989	

In this report, the records are sorted by customer, product, and date and the calculations are performed during the Breaks created by the sort order. The Detail area for each record is not printed; the records are used only to provide values for the calculations. Notice that 4th Dimension goes through the records from the first to the last during the printing of the report so that it can calculate these figures. (You create this kind of report by placing the Detail control line on top of the Header control line, leaving no space for details to print.)

You can ask 4th Dimension to perform additional calculations on a list including subtotals, averages, maximum and minimum values, page totals to be printed in a footer, and weighted averages. These calculations, and others, use 4th Dimension's summary functions. For information about how to use summary functions, refer to the *4th Dimension Language Reference*.

An Example Report

This section describes an example report and shows how the finished report is related to a form in the Form editor and to the methods that control the printing.

The following illustration shows a finished report.

Detailed Sales Report				02/12/1990
				7:44 PM
American Data				
<i>Product</i>	<i>Purchase Date</i>	<i>Comments</i>	<i>Price</i>	
ATN 700				
	9/14/88	Configured for fast access times	\$12,450	
		Subtotal for ATN 700	\$12,450	
STS 1000				
	3/17/88	Needed additional power of the 1000	\$22,450	
	4/17/88	Plan to purchase additional 1000's	\$3,300	
		Subtotal for STS 1000	\$24,360	
		Subtotal for American Data	\$38,200	
Omni Data Service				
<i>Product</i>	<i>Purchase Date</i>	<i>Comments</i>	<i>Price</i>	
STS 1000				
	2/14/88	Trying out the 1000	\$24,360	
		Subtotal for STS 1000	\$12,450	
STS 3000				
	4/25/88	Needed additional power of 3000	\$71,025	
	6/22/88	Now uses 3000 as standard machine	\$53,252	
	9/25/88	Third 3000	\$47,250	
		Subtotal for STS 3000	\$171,527	
STS 4000				
	5/17/88	Needed 4000 for special application	\$92,450	
	7/14/88	Special configuration	\$95,420	
		Subtotal for STS 4000	\$187,870	
		Subtotal for Omni Data Service	\$383,757	
Thomas Info				
<i>Product</i>	<i>Purchase Date</i>	<i>Comments</i>	<i>Price</i>	
ATN 700				
	1/27/88	First purchase of ATN 700	\$12,780	
	6/22/88	Will use many 700's	\$24,745	
		Subtotal for ATN 700	\$37,525	
STS 4000				
	8/3/88	Needed 4000 for new data center	\$89,740	
		Subtotal for STS 4000	\$89,740	
		Subtotal for Thomas Info	\$127,265	
		Total	\$549,222	
Lightwave Computer Corporation				Page 1

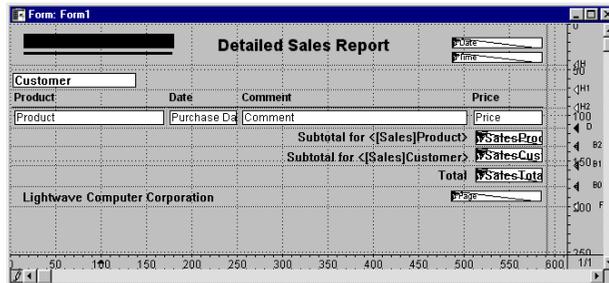
In this example, the page Header contains the date, the time, and the report title. The Break Headers contain the customer name and column headings for the information presented during the first Break. The Detail areas contain data drawn directly from the records. The level 2 Break areas contain subtotals for products for each customer. The level 1 Break areas contain subtotals for each customer and the level 0 Break area contains a total for the report. The Footer contains the page number.

The report is sorted on one more level than Break levels. In this report, the sort fields are Customer, Product, and Date.

4th Dimension requires one more sort level than Break level for the Break processing method used in this report.

The Report Form

The following illustration shows the report form that created the report shown on this page.



Each control line in the form defines the bottom of its area. Whatever is placed in the form is printed at the appropriate place in the report. The Header area contains the elements that will be printed at the top of each page, the Detail area contains the elements that will be printed for each record, and so on. The following table shows what each of these control lines means.

Label	Explanation	Effect
H	Header area	Printed once at the top of each page
H1	L1 Header area	Printed once before each level 1 Break
H2	L2 Header area	Printed once before each level 2 Break
D	Detail area	Printed once for each record in the selection
B2	L2 Break area	Printed once at each level 2 Break (when the value in the second Sort field changes)
B1	L1 Break area	Printed once at each level 1 Break (when the value in the first Sort field changes)
B0	L0 Break area	Printed once at the end of the report
F	Footer area	Printed once at the bottom of each page

You can have additional Break areas and additional Break Header areas.

The Report Object Methods

The non-enterable objects that are placed in the Header, Break, and Footer areas are controlled by object methods. This section describes the methods used to print values in these locations.

Note The following code can be used only in object and form methods. It cannot be used in project methods.

The date is drawn from the system date by placing a non-enterable object named `vDate` in the Header area with this method:

`vDate := Current Date`

The time is drawn from the system clock by placing a non-enterable object named `vTime` in the Header area with this method:

`vTime := Current Time`

The subtotal for sales in the level 2 Break area is calculated and displayed in an object named `vSalesProd` with the following method:

`vSalesProd := Subtotal (Sales)`

The subtotal for sales in the level 1 Break area is calculated and displayed in an object named `vSalesCust` with the following method:

`vSalesCust := Subtotal (Sales)`

The total for sales in the level 0 Break area is calculated and displayed in an object named `vSalesTotal` with the following method:

`vSalesTotal := Subtotal (Sales)`

Note that even though all three objects use the same calculation, they produce different results. Because they are placed in different Break areas, they are executed at different times and perform their calculations for different groups of records. For an explanation of Break levels, see the section [“Reports with Breaks” on page 460](#).

`vPage := "Page " + String (Current form page)`

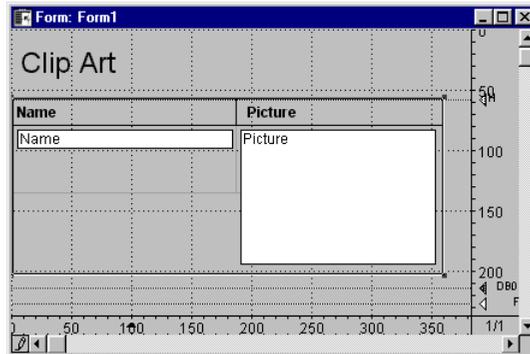
The `Current form page` function returns the page number.

Printing Subforms, Pictures, and Text Fields

You can use subforms, picture fields, and text fields in a report. These objects can be set to print with either a fixed or variable frame. Fixed frame objects print within the confines of the object as it was created on the form. Variable frame objects expand during printing to include the entire contents of the object.

You cannot place more than one variable frame object side-by-side on a form. You can place non-variable frame objects on either side of a variable frame Picture or Text field provided that the variable frame

object is at least one line longer than the other objects and that all objects are aligned to the top, as shown in the following illustration.



Otherwise, during printing, the contents of the other fields will be repeated for every horizontal slice of the variable frame object.

You cannot place objects on either side of a variable frame subform.

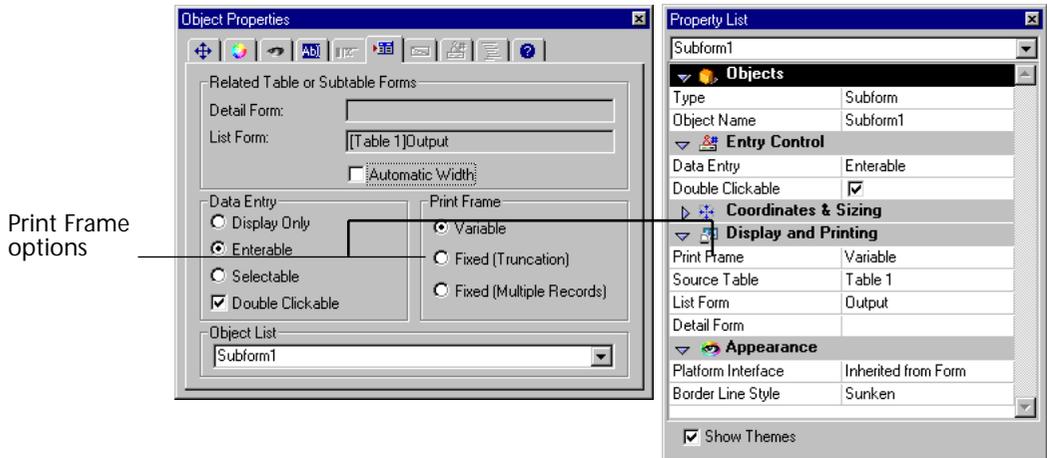
Printing Subform Areas You usually use an output form to print records in a subform.

- ▶ To set the print option for a subform:
 - 1 Double-click on the subform in the Form editor.
Depending on your current display settings, either the Object Properties window or the Property List appears. For more information on this choice, refer to [“Displaying and Setting Form and Object Properties” on page 244](#).
 - 2 In the Object Properties window, click the Subform tab and select a print option as shown below.
OR
In the Property List, expand the Display and Printing theme.

3 Select a printing option:

Object Properties window

Property List



Because there may be more records than can fit in the subform area, 4th Dimension provides the following three check boxes for controlling how records in a subform are printed:

- Variable,
- Fixed (truncation),
- Fixed (multiple records).

If you select the **Variable** radio button, 4th Dimension expands or contracts the subform area to print all the records.

If you select the **Fixed (truncation)** radio button, 4th Dimension prints only the records that fit into the area of the subform. The form is printed only once and those records that are not printed are ignored.

If you select the **Fixed (multiple records)** radio button, the frame remains the same size, but 4th Dimension prints the form several times to include all the records.

Note You cannot place any objects on either side of a variable frame subform. Objects placed on either side of the subform will be repeated for every line of the subform.

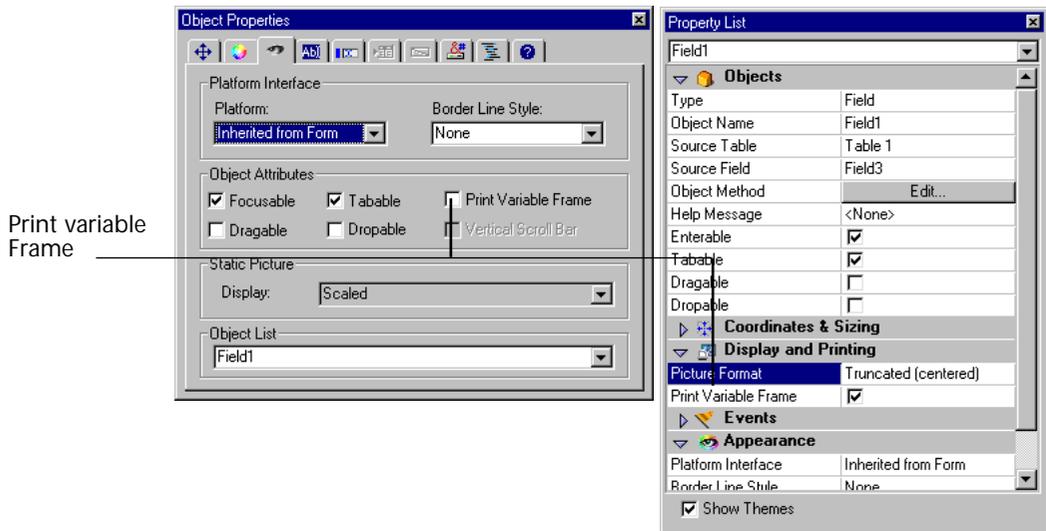
Printing Picture Fields Pictures can be printed with either fixed or variable frames.

- To set a print option for a picture field:

- 1 Double-click the picture field in the form.
Depending on your current display settings, either the Object Properties window or the Property List appears. For more information on this choice, refer to [“Displaying and Setting Form and Object Properties” on page 244.](#)
- 2 In the Object Properties window, click the Display tab.
OR
In the Property List, expand the Display and Printing theme.
- 3 Select a Printing option:

Object Properties window

Property List



- 4 Select the Print Variable Frame check box to print the picture with a variable frame.
When the Print Variable Frame check box is selected, the print frames expand to show the entire picture.
OR
Deselect the check box to print the picture with a fixed frame.

Printing Text Fields

Text fields can be printed with either Variable frame or Fixed frame. Print options for Text fields are selected exactly as they are for Picture fields.

- ▶ To select a print option for a Text field:
 - 1 Double-click the Text field in the form.

Depending on your current display settings, either the Object Properties window or the Property List appears. For more information on this choice, refer to [“Displaying and Setting Form and Object Properties” on page 244](#).
 - 2 Select the Print Variable Frame check box to print the text field with a variable frame.

OR

Deselect the Print Variable Frame check box to print the text field with a fixed frame.

If the Print Variable Frame check box is selected, the Text field expands dynamically to display as many lines as necessary to accommodate the amount of text that was entered.

Printing Labels

You can generate labels with either the Label editor in the User environment or a custom report form. If you use a report form, you have more extensive customization options. You can use variables on the label that use object methods to create the text for the variable and any of the graphics tools available in the Form editor.

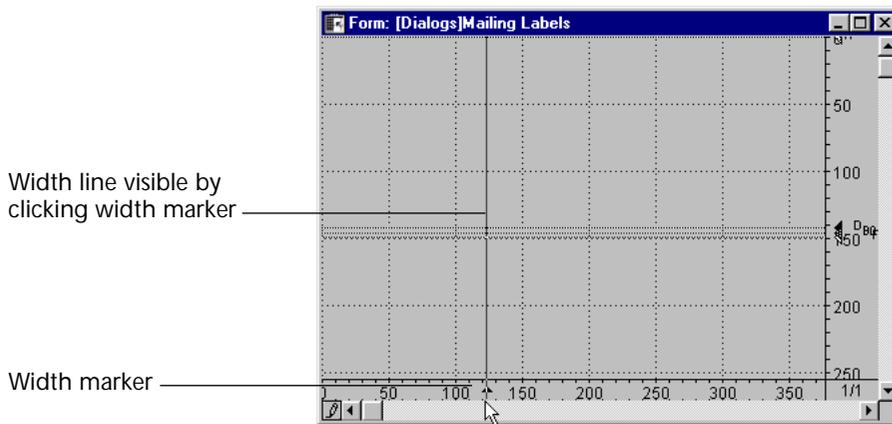
The report format described here can also be used for unusual reports that require side-by-side placement of text.

- ▶ To create a label report form:
 - 1 Set the label width by dragging the width marker on the bottom ruler of the Form editor.

The width marker determines how many labels 4th Dimension prints across the page. The placement of the width marker should correspond to the width of your labels. Be sure to take into account the left and right margins of your labels.

You can determine these margins by calculating the space between each label and dividing this number by two. If you have labels that are 2.25 inches wide with a margin of .125 inches (1/8 inch) on either side of each label, the label width marker should be placed at 2.625 inches

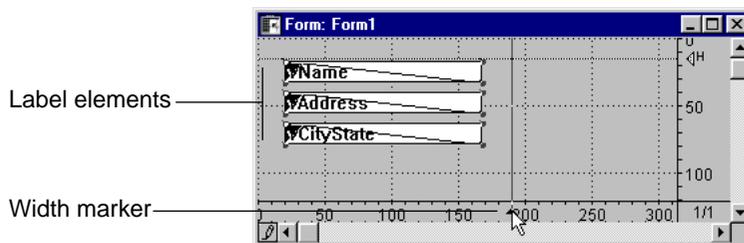
(2 3/8 inches) to ensure that the label text is placed correctly on the labels.



Note You may want to change the ruler units in the form to inches to make it easier to determine the proper placement of the label width marker. To change the ruler units, choose Define Ruler Units from the Form menu and select Inches.

2 Design the label form to the left of the width marker.

Labels can contain fields, active and graphic objects, text, and object methods. A subform cannot be printed in a label. These elements should be centered between the left edge of the form and the width marker so that the space on either side equals the size of the label margins as shown in the label design below.



This design includes variables (active objects) whose values are assigned by object methods. For example, the method for the vName variable concatenates the first name and last name of each person and places a space between the two names.

For more information about object methods, see the section [“Object Methods” on page 480.](#)

- 3 Set the control lines so that the Header control line is above the label and the Detail, Break, and Footer control lines are below the label.

The Header control line should be set at zero inches and the Detail control line should be set to the height of the label. To center the text within the label, center the form elements between the Header control line and the Detail control line.

When you print the labels in the User environment, everything between the Header and the Detail control lines appears on the labels.

- 4 Return to the User environment to print the Labels.
- 5 Choose the label form from the List of Tables window to make it the current output form for the table.

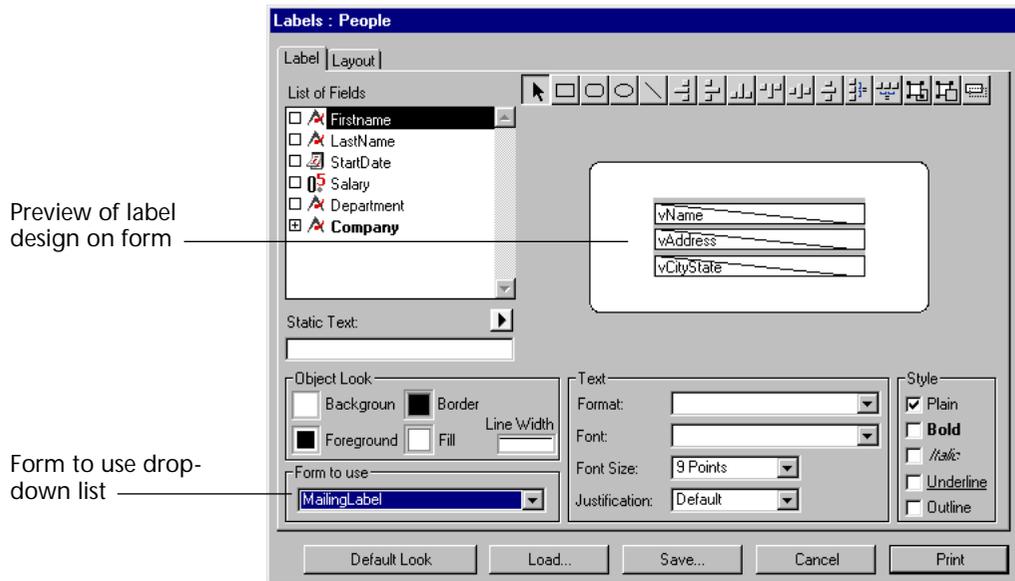
You can display the List of Tables window by pressing Ctrl+Space bar (on Windows) or Command-Space bar (on Macintosh).

Note If you have multiple operating systems installed on your PC and are using Ctrl+Space bar to switch between systems, use Ctrl+Shift+Space bar to display the List of Tables window.

For more information about using the List of Tables window, refer to the *4th Dimension User Reference*.

- 6 Choose Labels from the Report menu.
The Label Wizard appears.
- 7 Choose the label form you designed from the Form To Use drop-down list.

This tells 4th Dimension to use this form to print the labels.



- 8 Make any other changes to the Label editor such as setting the label margins or specifying the font in which the text appears.
For complete information about using the Label editor, refer to the *4th Dimension User Reference*.
- 9 Click the Print button.
4th Dimension prints the current selection of records using your label report design.

7

Creating Methods

You can attach a method to a 4th Dimension object to specify the object's action. A method is a series of instructions that tell the object to do something. For example, you can use methods to:

- Enforce business rules during data entry,
- Calculate values for fields and variables,
- Manage interface elements such as combo boxes, hierarchical lists, and tab controls,
- Manage drag and drop actions,
- Assign actions to custom menu commands,
- Create and manage multiple processes,
- Manage transactions,
- Manage custom reports,
- Regulate multi-user database access.

This chapter provides information about using 4th Dimension's Method editors to create and modify methods. To learn more about 4th Dimension's programming language, refer to the *4th Dimension Language Reference*, which provides detailed information about the programming commands and syntax.

4th Dimension Methods

You can create the following four types of methods:

- **Object methods** Object methods are associated with individual objects on a form, such as fields, buttons, drop-down lists, and tab controls. They can be used for such purposes as assigning initial values, managing and validating data entry, or managing drag-and-drop actions.
- **Form methods** Form methods are attached to individual forms. A Form method can manage everything that happens when a form is used for data entry, screen display, or printing. Alternatively, you can use Object methods to manage individual objects on the form.
- **Table methods** Table methods (triggers) are run when specific events occur at the database engine level. For more information, see [“Triggers” on page 481](#) and [“Triggers” on page 486](#).
- **Database methods** Database methods run automatically when certain worksession-related events occur. For more information about Database methods, see [“Database Methods” on page 483](#).
- **Project methods** Project methods can be called by other methods anywhere in the database. Project methods are also attached to custom menu items to specify the menu item’s action.

Object Methods

An object method is attached to a field or other active object on a form. You create object methods in the Form editor. The object method executes when certain events occur. The method is bound to the object to which it is attached and moves with it when copied.

An object method can perform calculations, get related information from other tables, concatenate data, validate data, display a specific page in a multi-page form, and more. The following are some examples of things you can do in an object method:

- Display the current date on screen or print it in a report,
- Manage data entry from a combo box,
- Display a custom dialog box when the user presses a button on a form,
- Specify the action when a user drags an item in a scrollable list to another object.

An object method can perform data entry control similar to the built-in data entry features (in the Data Control section of the Properties Window). For example, you can write an object method to perform data validation, set the display format, or establish an entry filter.

Form Methods

A form method is a method attached to a form. Each form can have one form method. It is executed whenever the form is used. The method is bound to the form, but does not move with it when a form is copied (form elements are copied, not entire forms).

Note Only certain events occur when a form is used as an output form. For information about events, refer to the *4th Dimension Language Reference*.

A form method manages a form at a higher level than do object methods. Form methods are used to control the interaction between different objects and the form as a whole. One use for a form method is to calculate values based on more than one field. Since the calculation must be performed whenever any of the values involved changes, you place the calculation in a form method.

For example, you might place the following statement in a form method:

```
vTax:= TotalSales * TaxRate
```

As a form method, this statement can be executed every time anything on the form changes. This ensures that the variable vTax is always up-to-date.

Triggers

Triggers are methods that run automatically when certain events occur at the database engine level. Those events are:

- **On Saving New Record** The trigger will be invoked when a record is added to the table. This happens when:
 - Adding a record in data entry (in the User environment or using the ADD RECORD command).
 - Creating and saving a record with CREATE RECORD and SAVE RECORD. The trigger is invoked when you call SAVE RECORD, not when it is created.
 - Importing records (in the User environment or using an import command).

- Calling any other commands that create or save new records (i.e., ARRAY TO SELECTION, SAVE RELATED ONE, etc.)
- Using a plug-in that calls the CREATE RECORD and SAVE RECORD commands.
- **On Saving an Existing Record** The trigger will be invoked when a record of the table is modified. This happens when:
 - Modifying a record in data entry (in the User environment or MODIFY RECORD command).
 - Saving an already existing record with SAVE RECORD.
 - Calling any other commands that save existing records (i.e., ARRAY TO SELECTION, APPLY TO SELECTION, MODIFY SELECTION, etc.).
 - Using a plug-in that calls the SAVE RECORD command.
- **On Deleting a Record** The trigger will be invoked when a record of the table is deleted. This happens when:
 - Deleting a record (in the User environment or calling DELETE RECORD or DELETE SELECTION).
 - Performing any operation that deletes related records through the Deletion control options of a relation.
 - Using a plug-in that calls the DELETE RECORD command.
- **On Loading a Record** The trigger will be invoked when a record of the table is loaded. This includes all situations in which a current record is loaded from the data file¹. You will use this option less often than the three previous ones.

Note This option does not cover the creation of a new record; it only applies to the loading of existing records.

The trigger is a new type of method introduced in version 6. In previous releases of 4th Dimension, table methods (called File procedures) were executed only when a form for a table was used for data entry, display, or printing — they were rarely used. Triggers execute at a much lower level than File procedures. No matter how a

1. This trigger is not invoked when using functions that may use indexes. If the index is used, records are not loaded and, conversely, if the index is not used, records are loaded. This uncertainty about the loading of the records does not allow the trigger to be managed properly. It is therefore never invoked in this case. For more information, refer to the 4th Dimension *Language Reference* manual.

record is created, modified, or deleted — by user actions (such as data entry) or programmatically (such as a call to SAVE RECORD) — the trigger will be invoked.

Database Methods

Database methods run when certain worksession-related events occur. Those events are:

- On Startup (equivalent to the *STARTUP* procedure in previous releases of 4th Dimension)
- On Web connection
- On Server startup
- On Server shutdown
- On Server open connection
- On Server close connection
- On Exit

4th Dimension ships with empty methods that are associated with these events. You can add code to any or all of these methods. For complete information on the uses of database methods, see the section “Database Methods” in the *4th Dimension Language Reference*. For more information on the specific uses of database methods with 4D Server, refer to the *4D Server Reference Manual*.

Note If you are converting a database written with an earlier version of 4th Dimension, it may have a *STARTUP* procedure. If you want that procedure to run automatically at startup, select “Use V3.x.x Startup Method Scheme” in the Design environment page of the Database Properties dialog box and ignore the new database methods architecture¹. Alternatively, you can deselect this option, paste the code from your *STARTUP* procedure into the On Startup database method, and use the new database methods scheme.

Project methods

A project method can be called by any other method or associated with a menu item. A project method can also be executed by choosing Execute Method from the User environment’s Special menu. When a project method is called by another method, the statements in the

1. For more information on this option, see the section “[Design Environment](#)” on page 90.

project method are substituted for the method's name in the method which is calling it.

Project methods are not associated with a table, form, or object. You can create as many project methods as you need.

The following is an example of a project method.

```
` Adds a record to the [Employees] table
INPUT FORM ([Employees];"Input")
OK:=1
While (OK=1)
  ADD RECORD ([Employees])
End while
```

This method is attached to a menu item that is used for entering new records. The method lets the user add new records to the [Employees] table using the form named Input. The user is able to add new records until he or she clicks the Cancel button¹.

A method can also be used as a formula that you apply to the current selection. For information about applying formulas, refer to the *4th Dimension User Reference*.

Events

Database, Table (trigger), Form, and Object methods run only when certain events occur. You specify those events when you set the properties for each object. The Table, Form, and Object Properties windows

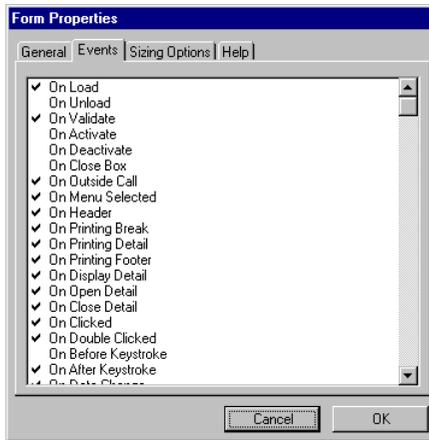
1. Clicking Cancel sets the system OK variable to 0.

each have pages in which you specify the events that will run for the object. The following illustration shows those pages.

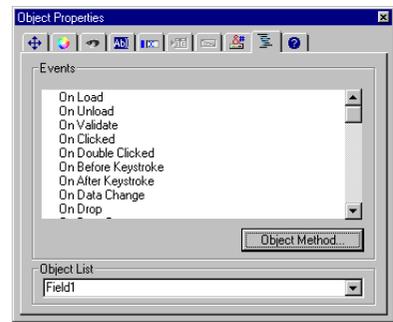
Triggers page of Table properties window



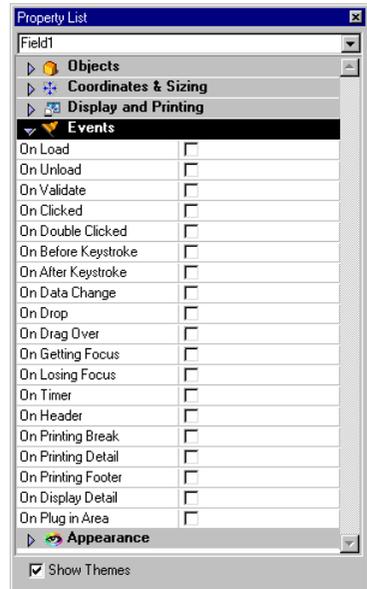
Events page of Form properties window



Events page of Object properties window



Events theme of property List (forms)



Events theme of property List (object)

In addition, 4th Dimension provides blank Database methods which are listed in the Methods page of the Explorer. Each Database method runs only when the corresponding worksession-related event occurs. You can open a blank database method from the Methods page of the Explorer:



Database Methods

You write a Database method by opening the desired blank Database method and entering your code. When you are converting a database written with an earlier release of 4th Dimension, you can copy the code in your *STARTUP* procedure and paste it into the On Startup method.

Triggers

Table events (triggers) are run when particular events regarding saving, deleting, and loading records occur. If you want a trigger to run when a particular event occurs, you must check the appropriate event in the Triggers page of the Table properties window. You then create the trigger by creating a Table method from the Explorer. For more information about creating a Table method, see the section [“Creating a Trigger” on page 499](#).

In the Trigger, you generally need to test for each event that you checked in the Triggers page. To do so, you can use a Case statement and the Database event function. The Constants page of the Explorer lists constants associated with each Table event. The Constants page of the Explorer window lists the constants associated with each trigger in the Database Events theme.

Your Case statement shell would look like this:

Case of

- :(Database event=On Saving New Record Event)
 - ` Perform actions for saving a new record
- :(Database event=On Saving Existing Record Event)
 - ` Perform actions for saving an existing record
- :(Database event =On Deleting Record Event)
 - ` Perform actions for deleting an existing record
- :(Database event=On Loading Record Event)
 - ` Perform actions for loading a record into memory

End case

Note The Design environment page of Database Properties has an option that lets you run Table methods (triggers) according to the rules that were established for File procedures in previous releases of 4th Dimension. If you wish to use Table methods in this manner, select the Use V3.x.x Startup Method Scheme.

Using Triggers

A trigger has two basic functions:

- Performing actions on a record before it is saved, modified, deleted, or just after it is loaded.
- Accepting or rejecting a database operation, such as a save record operation.

For complete information on using triggers, see the *4th Dimension Language Reference*.

Form and Object Events

Form and object methods run when particular form and object-level events occur. If you want a form or object method to run when a particular event occurs, you must activate the appropriate event in the Events page of the Form or Object Properties window.

The following is a list of events for forms used for screen display:

- On Load 4th Dimension is about to display the form on-screen or print the form.
- On Unload The form is about to be closed and released.
- On Validate After the user has accepted the record.
- On Clicked The user clicks the object that has focus.
- On Outside Call When the form receives a call from CALL PROCESS.

- **On Activate** When the form's window becomes the frontmost window.
- **On Deactivate** When the form's window is no longer the frontmost window (i.e., when another window becomes the frontmost window).
- **On Double-clicked** The user double-clicks the object.
- **On Getting Focus** When a form object gets focus (i.e., the user presses Tab to select the object or clicks the object).
- **On Losing Focus** When a form object loses focus (i.e., the user presses Tab to select the next object in the entry order or clicks another object to select it).
- **On Drop** When the object receives a dragged object.
- **On Drag Over** When an object is dragged over the object.
- **On Before Keystroke** When the user presses a key in the object. The Get edited text function returns the contents of the object that has the focus.
- **On After Keystroke** When the user presses a key in the object. The Get edited text function returns the contents of the object that has the focus, including the last keystroke.
- **On Menu Selected** A menu item has been chosen.
- **On Plug-in Area** A plug-in requested that its object method be executed.
- **On Data Change** When the data in the object changes.
- **On Close Box** The window's Close box has been clicked.
- **On Display Detail¹** A record is about to be displayed in an output form.
- **On Open Detail¹** A record is double-clicked in an output form and 4th Dimension is about to display the record in the current input form.
- **On Close Detail¹** 4th Dimension is putting away the input form and is about to redisplay the output form.

1. These events operate only when using the MODIFY SELECTION and DISPLAY SELECTION commands.

- **On Timer** The number of ticks set by the SET TIMER command was reached.
- **On Resize** When the form's window is resized.
- **On Header** When the form's header is either being displayed or printed.

For printed reports only, the following events occur:

- **On Printing Detail** The form's Detail area is about to be printed.
- **On Printing Break** The form's Break areas are about to be printed.
- **On Printing Footer** The form's Footer area is about to be printed.

For complete information on form events, refer to the discussion of the Form event function and the form event constants in the *Language Reference*.

When you write a form or object method, you generally need to test for each event that you activated in the Form or Object Properties window. To do so, you can use a Case statement and the Form event function. The Constants page of the Explorer lists constants associated with each Form event. Your Case statement would look like this:

Case of

```
:(Form event=On Load)
  ` Perform appropriate actions here...
:(Form event=On Data Change)
  ` Perform appropriate actions here...
:(Form event =On Validate)
  ` Perform appropriate actions here...
:(Form event=On Clicked)
  ` Perform appropriate actions here...
```

.
.

.

End case

Introduction to Methods

A 4th Dimension method is a series of instructions that causes 4th Dimension to perform an action or a series of actions. For example, the following project method allows users to add records to a table:

```
INPUT FORM ([Customers];"Input")
```

Repeat**ADD RECORD** ([Customers])

Until (OK=0)

This method would be attached to a menu item in a custom application. When the user chooses that menu item, 4th Dimension runs this method. It makes the [Customers]Input form the current Input form and presents it to the user as a blank form, ready for data entry. The user could continue to enter new record until he or she presses the Cancel button on the form. During data entry, any table, form, or object methods would run when the appropriate events occur.

4th Dimension methods are created with components of a *procedural language*. The following are the elements of the language:

- **Fields** You can use fields from any table. For example, a method can use a value that is stored in a field or it can change that value and store a new value in that field.
- **Object names** You can use the names of objects on a form. For example, you can resize an object, change its color, enable or disable buttons, or modify the font, font size, or style.
- **Variables** You can temporarily store a value in a variable and use it later in the same method or in a different method. You can create a variable in any method and you can use or change its value in any method.
- **Pointers** Pointers let you write generic code that doesn't refer to database objects by name. Instead, a pointer to each object is used. Each time the generic code is used, you can "point" to different database objects. For example, if you substitute a pointer to a table for the table name "[Customers]" in the previous example, you could then reuse the code for any table. For more information on pointers, see the chapter "Arrays and Pointers" in the *4th Dimension Language Reference*.
- **Operators** You can use symbols to instruct 4th Dimension to carry out an operation such as multiplication, addition, and so on.
- **Commands** You can use commands in the language to instruct 4th Dimension to perform an action. For example, the ALERT command displays a message in an alert dialog box. The NEXT PAGE command displays the next page of a multi-page form and the ORDER BY command sorts the records in the current selection.

- **Functions** You can use functions in the language to calculate values. For example, you can calculate an average of several values with the Average function. You can calculate a subtotal for a report with the Subtotal function.
- **Flow of control** You can control when code executes with flow of control structures. The 4th Dimension language includes the following control structures:
 - If...Else...End If
 - Case of...Else...End Case
 - While...End While
 - Repeat...Until
 - For...End For

You use logical tests in these structures to determine whether or how many times code executes. All of these elements are discussed in detail in the *4th Dimension Language Reference*.

Examples

This section describes features common to all types of methods.

Statements

A method is composed of statements, each statement consisting of one line in the method. A statement is an instruction for 4th Dimension to carry out. For example, the following line is a statement:

```
[People]Start Day := Current date
```

This statement places the current date in the Start Day field of the [People] table. Current date is a function that returns the date based on the system date. [People]Start Day is a field.

Notice that the statement specifies the table name, surrounded by square brackets, with the field. When writing project methods, you specify the table name to which a field belongs to avoid possible confusion with other fields with the same name. However, when in a form or object method, you can specify fields from the form's table without specifying the table name. Table names are written within square brackets.

The previous statement is typical of statements that calculate or work with values. It starts with the field in which the value is to be placed and uses the assignment operator to point at the calculation that determines the value. The calculation is performed by whatever follows the

assignment operator. The assignment operator is a colon and equals sign (:=).

You use the assignment operator whenever you need to store a value in a field, an object, or a variable. It takes the following form:

Field/Object/Variable := Calculation

The value container is the field, object, or variable in which you want to place the value. The calculation is the operation that results in the value you want to store. The assignment operator assigns the value that is calculated on its right into the container on its left. You will see several examples of this in the next few paragraphs.

A statement may be simple or complex. Although a statement is always one line, that one line can be as long as needed (up to 32,000 characters).

The following method displays the third page in a multi-page form:

GOTO PAGE (3)

When you use the GOTO PAGE command, you instruct 4th Dimension to display the page indicated in the parentheses.

Notice that the command is in bold capital letters; this is the way that 4th Dimension displays commands in the Method editor. This convention is used in all examples in the 4th Dimension documentation. You do not have to type commands in all capital letters; 4th Dimension automatically changes the display.

Most commands require additional information to carry out the instruction. This additional information is called an argument to the command. An argument contains data that a command needs in order to complete its task. In this case, the GOTO PAGE command needs the page number to go to. An argument is always placed within parentheses following a command.

Executing the Method

The execution of statements in a method always follows a strict order. In a listing method, the order is always line-by-line. When the method is executed, it begins at the first line and works its way down to the last line (Obviously, the order of execution is determined by the flow of control structures in the method). In a flowchart method, the execution order is determined by the path of the flowline (i.e., the line with arrows) and the results of test statements.

This section examines a multi-line method in detail in order to establish some of the terminology, concepts, and common aspects of methods.

The following method computes the total amount due on an invoice:

```
vSales Tax := Total Purchases * Tax Rate
vTotal := Total Purchases + vSales Tax
[Report]Total Due := Round (Total; 2)
```

This method is attached to an object that will be printed on an invoice. As you follow the method line-by-line, you will see that the later lines depend on previous lines. Because of the strict order in which statements are executed, you can depend on a value being available when it is needed.

Here is the first line of the method. It calculates the sales tax for the purchase:

```
vSales Tax := Total Purchases * Tax Rate
```

In this method, “vSales Tax” is a variable. You can create a variable at any time by typing its name. You name the variable on the left of the assignment operator and calculate a value on the right. Subsequently, whenever you need that value, you can simply use the variable name. A variable can be created at any time in a method. It can be assigned a value, as here, which can then be used by a later statement.

The variable “vSales Tax” is a *process variable*. A process variable is a variable which works within the current process. In 4th Dimension, everything occurs within a process, even if the process is one created by 4th Dimension. (You will learn about processes in [Chapter 12](#).)

When a statement creates a process variable such as vSales Tax, a portion of memory is set aside and assigned the name “vSalesTax.” Since it is a process variable, any method in the current process can use the value in vSalesTax. The variable remains in memory until the process is closed. *Local* variables temporarily store values that can be used within the method, but which cannot be used by any other method.

Variables follow the same rules of naming as fields. Typically, you would use a consistent convention so that you know that you are using a variable. In the previous example, the variable is indicated by a lowercase v as the first letter in the variable vSales Tax. For more information about variables and the different types of variables, refer to the *4th Dimension Language Reference*.

For the `vSales Tax` variable the assignment operator (`:=`) assigns it the result of a calculation. In this case we would say, “`vSales Tax gets Total Purchases times Tax Rate.`” The `Total Purchases` field contains the total amount of purchases for the current invoice and the `Tax Rate` field contains the tax rate to be used to calculate the tax. (A `Tax Rate` field could contain different rates based on the address of the customer.)

When an object method uses field names, the instruction is to use the value in that field on the current record. When another record is displayed, used, or printed, the method is executed again, using values in the new record.

The first line of our example method multiplies the values in the `Tax Rate` and `Total Purchases` fields and stores the result in the variable `vSales Tax` for each record that is used (a value entered or an invoice printed).

```
vSales Tax := Total Purchases * Tax Rate
```

Here is the second line of the method. It creates a variable that contains the grand total of purchases plus sales tax:

```
vTotal := Total Purchases + vSales Tax
```

The first component of the statement above is the `vTotal` variable. It is assigned the value generated by the addition of `Total Purchases` (a field) and `vSales Tax` (a variable). It does not matter to 4th Dimension whether a value is stored in a field or in a variable. As long as the field and the variable store data of the same type, the addition operator (`+`) simply adds the two values together.

Notice that this statement uses the `vSalesTax` variable that was created in the previous line. It also creates the `vTotal` variable which will be used in the following line.

Notice also that the value of `Total Purchases` has been used twice: first in the previous line to calculate the `vSales Tax`, and second in this line to calculate the `vTotal`. Nothing happened to the value in the field in either case. In each of these two lines, the assignment operator placed a value in a variable. These statements use the value in the `Total Purchases` field, but do not change that value.

The assignment operator (`:=`) places a value in the field, object, or variable to its left.

Nothing will happen to fields, objects, or variables to the right of the assignment operator.

Here is the third line of the method. It stores a value in a field in a different table:

```
[Report]Total Due := Round (vTotal; 2)
```

The first component is a field. Notice that the table name has been specified because it is different from that of the master table. Table names are always placed in square brackets [like this]. If you have to use parentheses as well, you would use both ([like this]).

The calculation is performed by the Round function. The Round function rounds off a value to a specified number of places.

Most functions require additional information to carry out the calculation. Additional information for a function is the argument to the function. A function may have several arguments. In this case, the Round function needs two arguments: the number to round and another number specifying how many places to round to. Here, the number to round is the value of the vTotal variable, and the number of places to round to is 2. The two arguments are separated by a semicolon. The variable vTotal and the number 2 are *arguments* to the Round function.

Notice that the function's name has only the first letter capitalized. This is the naming convention used for 4th Dimension functions. Just as with commands, you do not need to enter the capital letter. When you press the Enter key or click on another line in the method, 4th Dimension automatically capitalizes the first letter of the function and makes it bold.

Where to Put an Object Method

You can attach an object method to any field or other active object. An object method can refer to values in other fields and objects. The general rule is to attach an object method to the active field or object, the one that receives the data entry or the one that is clicked or otherwise activated. An object method that is to be executed when a button is clicked should be attached to the button object. An object method that capitalizes entries in a field should be attached to the field.

However, if you placed the method

```
Grand Total := Total + Sales Tax
```

in the Grand Total field, the method would not function properly since you would have to type something in the Grand Total field in order for the method to be executed.

You need the statement to be executed whenever the values in the Total or Sales Tax fields change. In order for the calculation to take place, you should place the statement in the form method or use it in object methods for both the Total and Sales Tax fields.

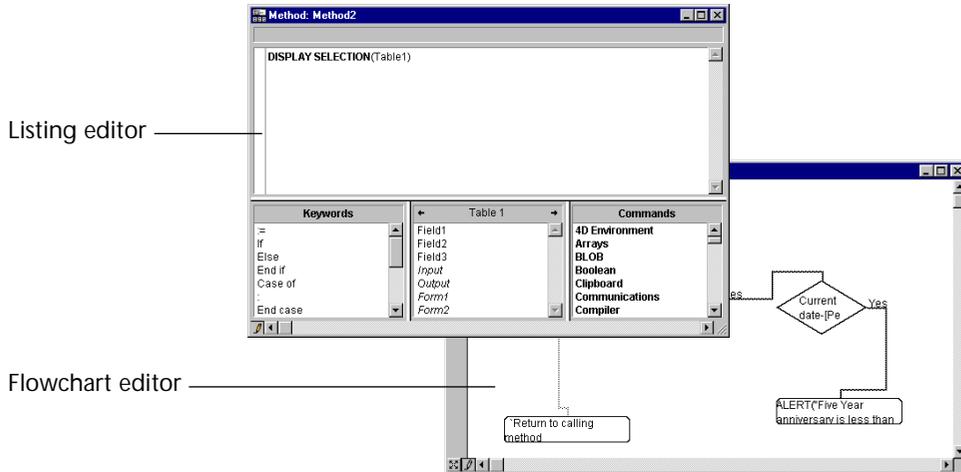
Creating Methods

When you create a method, you use one of the two Method editors. The Method editors provide you with tools to create, test, and edit a method.

The Method Editors 4th Dimension provides two editors in which you can build, modify, and test your methods. They are the:

- **Listing editor** You can use a text editor in which the method appears as a list of statements.
- **Flowchart editor** You can use a graphically-oriented editor in which the method appears as a chart of graphic symbols.

The figures below show how the two Method editors appear on screen.



It is advantageous to use the Listing editor unless you are familiar with the flowchart method of programming and like to create methods as

flowcharts. After you create a method, you can open the method only with the same Method editor.

The Listing editor has three additional advantages:

1. Only methods created using the Listing editor can be compiled.

2. The Listing editor allows you to insert break points, which is an essential part of the debugging process.

3. Only the Listing editor provides you with syntax help when you enter commands.

In the General page of the Database Properties dialog box, you can set a default editor for new methods (for more information, see the section [“General” on page 83](#)). If you do not set a default editor, you must choose which Method editor to use just before you create the method.

Creating an Object Method

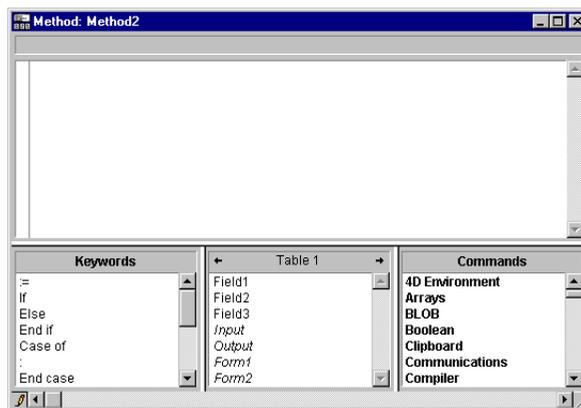
Object methods are created for an object on a form. You start in the Form editor, with a form displayed on the screen.

- ▶ To create an object method:
 - 1 In the Form editor, hold down the Alt key (on Windows) or the Option key (on Macintosh) and click the object.
 - OR
 - Select the object, then choose Object Method from the Object menu.
 - OR
 - On Windows, click the object using the right mouse button and choose Object Method from the contextual menu.
 - OR
 - On Mac OS, press the Control key while clicking the object and choose Object Method from the contextual menu.
 - OR
 - In the Object Properties window or the Property List, click the object method Edit button.

If you haven't specified a default editor type in the Database Properties dialog box, 4th Dimension displays the Method Type dialog box so that you can select the preferred Method editor.



- 2 Select the desired Method editor and click OK. Your preferred Method editor window appears.



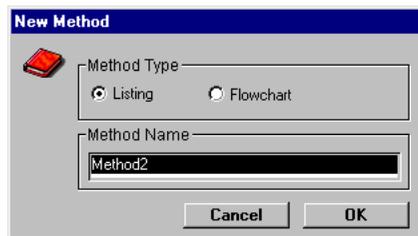
Directions for using the Listing or Flowchart editor are provided in the sections [“Using the Listing Editor” on page 507](#) and [“Using The Flowchart Editor” on page 523](#).

Creating a Project Method

You can create a new project method using a menu command or the Methods page of the Explorer.

- ▶ To create a project method from the Design menu:
 - 1 Choose New Method from the Design menu.

4th Dimension displays the New Method dialog box.



2 Enter a method name.

Method names can be up to 31 characters long. They can include letters, numbers, the space character, and the underline character.

3 Click the OK button.

4th Dimension opens an blank Method editor window where you can begin writing the new method.

► To create a new project method from the Explorer:

1 Click the Methods tab in the Explorer.

The hierarchical list of Database, Project, and Table/Form methods appears.

2 Highlight the Project Methods item.

3 Click New.

The New Method dialog box appears.

4 Click the Listing button and enter the method name.

Directions for using the Listing or Flowchart editor are provided in [“Using the Listing Editor” on page 507](#) and [“Using The Flowchart Editor” on page 523](#).

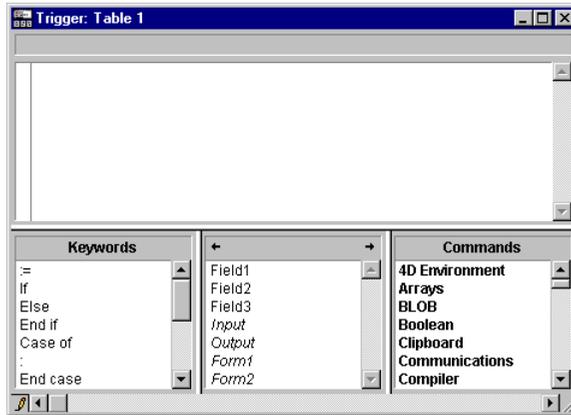
Creating a Trigger

You can create a trigger using a shortcut in the Structure editor or you can create the trigger directly in the Explorer:

► To create a trigger from the Structure editor window:

1 Hold down the Alt key (on Windows) or Option key (on Macintosh) and double-click the table title in the Structure editor window.

If you have specified a default Method editor in Database Properties, the Method editor appears, ready for you to write the trigger.

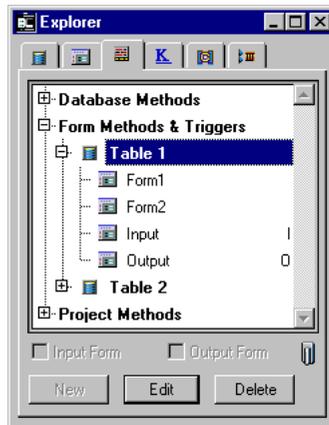


Otherwise, the Method Type dialog box appears.

► To create a trigger from the Explorer:

- 1 If the Explorer is not displayed, choose Explorer from the Tools menu.
- 2 Click the Methods tab.

The hierarchical list of Database, Project, and Form Methods & Triggers appears.



- 3 Highlight the desired table and click Edit.

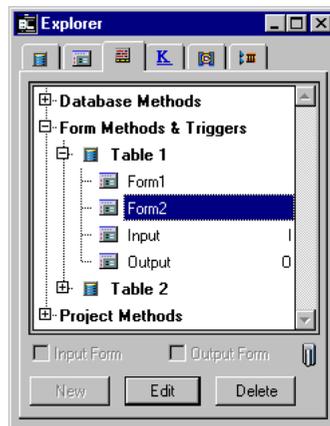
If you have specified a default Method editor in Database Properties, a blank Method editor window appears, ready for you to write the trigger.

Creating a Form Method

You create a Form method from the Methods page of the Explorer.

► To create a Form method:

- 1 Click the Methods tab in the Explorer.
- 2 Expand the table to which the form belongs and highlight the desired form.



3 Click Edit.

If you haven't specified a default Method editor in Database Properties, 4th Dimension displays the Method Type dialog box. Otherwise, 4th Dimension opens a blank Method editor window.

4 If no Method editor is set as the default, click the Listing radio button and click OK.

4th Dimension opens a blank Method editor window where you can begin writing the new method.

► To create a form method from the Form editor:

1 Choose Form Method from the Form menu.

OR

On Windows, click an empty area on the form using the right mouse button and choose Form Method from the contextual menu.

OR

On Mac OS, press the Control key while clicking an empty area on the form and choose Form Method from the contextual menu.

OR

In the Property List, click the Edit button located next to the Form Method line.

If you have specified a default Method editor in Database Properties, a blank Method editor window appears, ready for you to write the form method.

Renaming a Project Method

You can change the name of a project method in the Explorer. Database methods cannot be renamed. Triggers, form methods, and object methods are bound to objects and take their names from the object.

- ▶ To change the name of a project method:
 - 1 Display the Methods page of the Explorer.
 - 2 Expand the list of project methods so that the one you want is displayed.
 - 3 Hold down the Command key (on Macintosh) or Ctrl key (on Windows) and click the project method name.
The method name becomes editable.
 - 4 Type a new name.
 - 5 Press Tab or click anywhere outside the entry area to save your changes.
If a method with the same name already exists, 4th Dimension displays a message saying that the method name has already been used. Otherwise, 4th Dimension changes the name of the method and, if necessary, resorts the list of methods.

Notes Changing a method name can be done in the Method Properties dialog. This dialog is described in the following section.

Changing a method name can invalidate any methods or formulas that use the old method name. Each such item has to be updated in order to work.

4D Server The method name is changed on the server when you finish editing the name. If more than one user is modifying the method name at the same time, the final method name will be the name specified by the last user to finish editing the name.

You may want to specify a method owner so that only certain users can change the method's name.

Setting Access Privileges

You can control access to methods by setting Access and Owner privileges for groups of users. A single group can be assigned for each privilege using the Access and Owner drop-down list in the Method

Properties dialog box. For information about creating a password access system with users and groups, see the section [“An Access Hierarchy Scheme”](#) on page 560.

The Access drop-down list controls which group can execute the method in the User or Custom Menus environment. If a user that is not in this group attempts to execute the method, 4th Dimension displays a message saying that the user’s password does not allow him or her to execute the method.

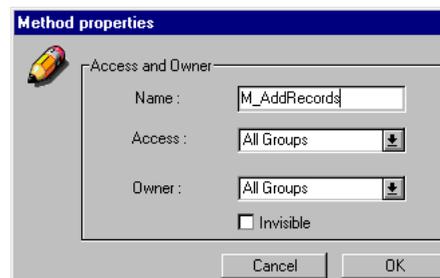
The Owner drop-down list controls which group can edit the method in the Design environment. If a user who is not in this group attempts to edit the method in the Design environment, 4th Dimension displays a message saying that the user does not have the access privilege to edit the method.

Users who are assigned to both groups can use the form in the User, Custom Menus, and Design environments.

- To assign access and owner privileges for a method:

- 1 With the desired method in the frontmost window, choose Method Properties from the Method menu.

The Method Properties dialog box appears.



- 2 Choose the desired Owner and Access groups from the drop-down lists.
- 3 Click OK to save your changes and put away the Method Properties dialog box.

Access or Owner privileges are assigned to the new group.

Making a Method Invisible

There are two menu commands in the User environment that give users the ability to run project methods — the Execute Method command in the Special menu and the Apply Formula command in

the Enter menu. If you don't want any users to run a project method, you can make it Invisible in the Method Properties dialog box. An invisible method does not appear in either the Execute Method dialog box or the Formula editor.

When you make a project method invisible, it is still available to database programmers. It remains listed in the Methods page of the Explorer and in the list of routines in the Method editor.

Opening an Existing Method

After you create a method, you can open it and make changes. 4th Dimension automatically opens the same Method editor that you used to create the method. You can open any existing database, project, table, or form method from the Methods page of the Explorer. You can also open form methods from the Form editor. You can open object methods only through the Form editor.

4D Server Object locking occurs when more than one user attempts to modify the same method at the same time. If a user is modifying a method in the Design environment, the method is locked. Other users cannot modify that same method until the first user frees the method by closing it.

To be able to open a method, you must have access privileges.

Note You cannot delete database methods. To disable such a method, erase all the statements in the method or precede each line with a ` symbol (used to distinguish comments from executable code). For information about deleting an object method, see the section [“Clearing Unwanted Object Methods” on page 507](#).

Opening a Database Method, a Project Method or a Trigger You can open any database method, project method or trigger from the Explorer window. You can also open a trigger from the structure window. Finally, you can also open a form method from the form editor window.

Object methods can be opened only from the forms in which the objects are used.

- ▶ To open a database method, a project method, or a trigger using the Explorer:
 - 1 Select **Edit Method** from the **Design** menu or, if the Explorer is opened, click the **Methods** tab.
You can also use the **Show Table Form Methods and Triggers** menu command from the contextual menu that is displayed when you click a table image using the right mouse button (on Windows) or when you press the **Control** key while clicking a table image (on Mac OS).
 - 2 In the Explorer window, expand the method type that corresponds to the method you want to open (Database methods, Form Methods and Triggers, or Project Methods).
 - 3 Select the method you want to open and click the **Edit** button.
OR
Double-click the method name.
4th Dimension displays the method in the Method editor.
- ▶ To open a trigger from the structure window:
 - 1 Press the **Alt** key (on Windows) or the **Option** key (on Mac OS) and double-click a table name.
4th Dimension displays the trigger in the Method editor.
- ▶ To open a form method from the Form editor:
 - 1 Select **Form Method** from the **Form** menu.
OR
On Windows, click an empty area on the form using the right mouse button and choose **Form Method** from the contextual.
OR
On Mac OS, press the **Control** key while clicking an empty area on the form and choose **Form Method** from the contextual menu.
OR
In the **Property List**, click the **Edit** button located next to the **Form Method** line.
4th Dimension displays the form method in the Method editor.

Opening an Object Method

You open an object method only from the form on which the object is placed.

- ▶ To open an object method:
 - 1 Use the **Explorer** or the **Edit Form** menu item to open the form that contains the object to which the object method is attached.

- 2 Hold down the Option key (on Macintosh) or Alt key (on Windows) and click the object on the form.
OR
In the Object Properties window, double-click the object to open the Object Properties window, click the Events tab, and click the Edit Method button.
OR
In the Property List, double-click the object to open the Property List, expand the Objects theme, and click the Edit button located next to the Object Method line.
OR
On Windows, click the object on the form using the right mouse button and choose Object Method from the contextual menu.
OR
On Mac OS, press the Control key while clicking the object on the form and choose Object Method from the contextual menu.
4th Dimension displays the object's method in the Method editor.

Deleting a Project Method, a Form Method or a Trigger

You can delete a database method, project method or trigger at any time from the Explorer window.

Object methods are deleted in the Form editor (refer to the following paragraph).

Note You cannot delete database methods. To disable such a method, erase all the statements in the method or precede each line with a ` symbol (used to distinguish comments from executable code)

- ▶ To delete a form method, a project method, or a trigger using the Explorer:
 - 1 In the Explorer click the Methods tab.
Make sure the Methods page is displayed, since deletion operations cannot be undone.
 - 2 In the Explorer window, expand the method type that corresponds to the method you want to delete (Database methods, Form Methods and Triggers, or Project Methods).
 - 3 Select the table or form to which the method is assigned.
OR
Highlight the name of the project method you want to delete.
4th Dimension displays the method in the method editor.

- 4 Click the Delete button located at the bottom of the Explorer window. 4th Dimension displays an alert asking you to confirm the deletion.
- 5 Make sure it is the right object you are about to delete and click OK to confirm. 4th Dimension deletes the corresponding method.

Clearing Unwanted Object Methods

You can clear unwanted object methods using the Form editor. In some cases, clearing unneeded object methods can make the database run faster.

- ▶ To delete an unwanted object method:
 - 1 Display the form that contains the method(s) you want to clear.
 - 2 Select the object(s) that have unwanted Object methods attached to them.
 - 3 Choose Clear Object Method from the Object menu. 4th Dimension removes the Object methods from the selected objects.

Note If you clear an Object method by mistake, choose Undo from the Edit menu.

Using the Listing Editor

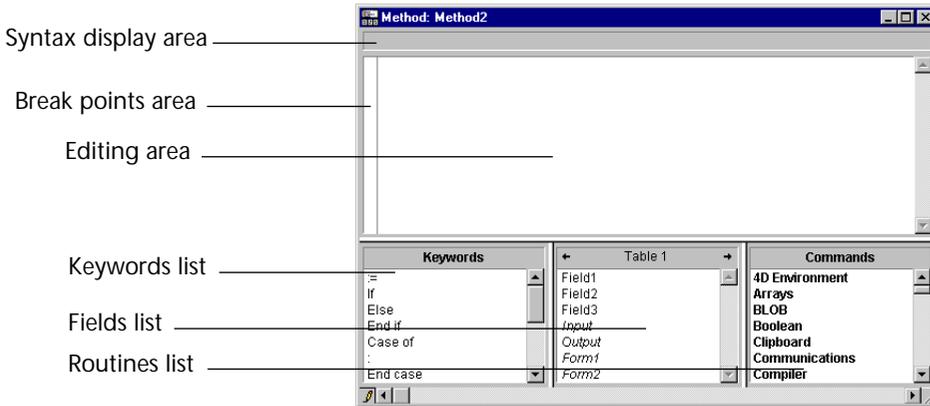
4th Dimension's Listing editor works like a text editor. That is, you can enter and edit text in the editor.

When you create a method with the Listing editor, you write the method as a series of statements. You can also select components of the method from lists provided by the editor.

You can also drag and drop table names, field names, form names, project method names, commands, and constants from the Explorer to the Listing editor.

You can scroll through the method. You can enter up to 32,000 characters in a method.

The figure below shows the Listing editor.



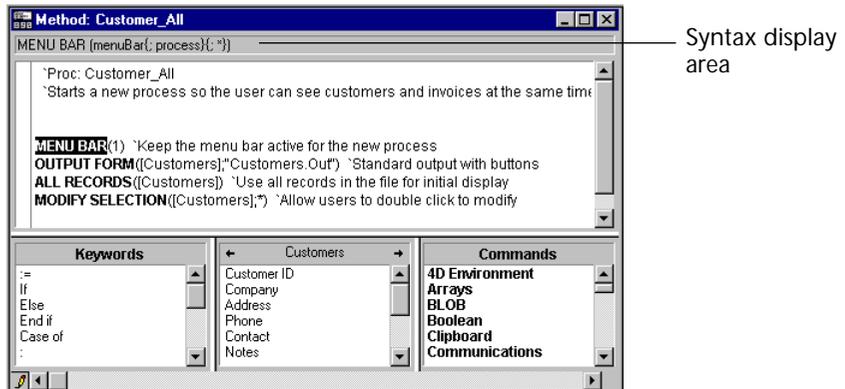
The Listing editor contains six areas: an editing area, a keywords list, a field list, a routines list, a syntax display area, and a break points area.

Editing Area

The editing area contains the text of the method. You enter and modify the method text in this area. The editor automatically indents method text for clear program structure. You can include comments inside the method text for reference.

Syntax Display Area

This area displays the syntax of the 4th Dimension commands—the command name, followed by the parameter names—that you want to insert in your code. For a command syntax to be displayed, you must either select a command in the commands list or highlight it in the Editing area and press the Enter key of the numeric keypad.



The syntax display area is also used to display potential syntax errors detected by 4th Dimension when the method is validated. For more information, refer to [“Writing a Method” on page 511](#).

Keywords List

The Keywords list is a scrollable area that provides the commonly used programming keywords, such as the assignment operator and the flow of control structures. A keyword is a word, symbol, or phrase that has a specific use in a method statement. When you click a keyword, 4th Dimension inserts it at the location of the insertion point in the editing area.

Fields List

The Fields list displays the names of fields and forms in the database. You click field names to enter them at the location of the insertion point in the editing area. The area’s title bar displays the name of the table whose fields and forms appear in the list. If no table name appears in the title bar, the fields displayed are from the master table.

You can display fields and forms from other tables by clicking the cycle arrows in the title bar. You can also press and hold the mouse button down on the title bar and choose a table from the pop-up menu of table names that appears. Clicking a table name inserts it in the method at the location of the insertion point.

When you include a field name in the method by clicking in the Field list, 4th Dimension automatically provides the correct field syntax, adding table names or subtable names as required. If you click the name of a subtable, 4th Dimension opens a window where you can select a subfield name.

Form names are added to the end of the list of fields for each table. Clicking a form name inserts it in the method at the location of the insertion point.

Routines List

The Routines list displays the names of all 4th Dimension commands and functions. The routines are grouped into “themes” according to their function. Each theme is a pop-up menu from which you can choose an individual command or function. To display the commands and functions in an alphabetical list, click the title bar of the Routines list.

In addition to 4th Dimension’s built-in commands and functions, you can access your own methods. The names of your project methods and functions appear at the end of the list of routines.

If you have installed any plug-ins that add their own commands, you can use those plug-in routines in a method. Plug-in routines include the following:

- **4D Internet Commands** These are routines that add features to 4th Dimension's Internet capabilities, especially for e-mail management.
- **4D Chart commands** These are the routines that make up the 4D Chart language.
- **OLE_Tools commands (Windows only)** These are the routines of the OLE Tools plug-in, which comes with 4D.
- **4th Dimension plug-in commands** These are commands that belong to optional plug-ins available for use with 4th Dimension, such as 4D Write, 4D Draw, and 4D Calc.

Note These commands are described in separate documentation.

Plug-in commands appear after the list of user-written methods.

If you prefer writing methods without selecting components from the Keyword, Field, and Routines lists, you can hide these lists by selecting Hide Keywords in the Design Environment page in the Database Properties dialog box. For more information about setting preferences, see the section [“Design Environment” on page 90](#). You can also hide these lists by dragging the window divider downward.

Break Points Area

This area allows you to directly insert break points next to specific instructions. Break points are useful when during the debugging phase of your programming. They stop the execution of your code at specific locations and display the debugger.

To insert a break point, click in the break points area at the location you want it to be. A red dot indicates the presence of a break point.

Note You can display the location of all the break points present in the database in the Runtime Explorer. For more information, refer to [“Break and Catch Pages” on page 78](#).

To temporarily disable a break point or modify its properties, press the Alt key (on Windows) or the Option key (on Mac OS) while clicking on the break point. When you click the break point, the break point property window is displayed.

To delete a break point, click the red dot.

Note Break points remain at the location you defined it, even if you insert or delete a line.

For a complete description of break points, please refer to the 4th Dimension *Language Reference* manual.

Writing a Method

Writing a method is usually a combination of typing text, selecting components, and dragging elements from the Explorer. You can create methods by typing text, selecting components of the method from the Keywords, Fields, and Routines lists, or dragging and dropping components from the Explorer. You can also use the “at” sign (@) wildcard to speed the creation of methods. For more information, see the section [“Using the Wildcard Character” on page 515](#).

The 4th Dimension method editor provides basic syntax error-checking. Additional error-checking is performed when the method is executed.

Typing Text

4th Dimension uses standard text editing techniques for typing and editing in the Listing editor. As you type, the characters appear at the location of the insertion point. You end each line by pressing the Return key (on Macintosh) or Enter¹ key (on Windows).

Note To enter a numeric value in hexadecimal, type 0x (zero + “x”), followed by the hexadecimal digits.

The Listing editor uses the following conventions for displaying routines, plug-in routines, and user-written methods:

- 4th Dimension keywords and routines are displayed in bold,
- Plug-in routines are displayed in bold-italic,
- Constants are underlined,
- User-written methods are displayed in italic.

1. The Enter key on the numeric keypad behaves differently than the Enter key on the main keyboard. Use the Enter key on the numeric keypad to force 4th Dimension to check the syntax of the line of code without moving the insertion point to the next line.

If you are using a color monitor, you can use the Color menu to colorize various types of text elements.

Note The font and font size are specified in the User Interface page of the Database Properties dialog box. The regular size font is used in the Method editor. For more information, see the section [“User Interface” on page 89](#).

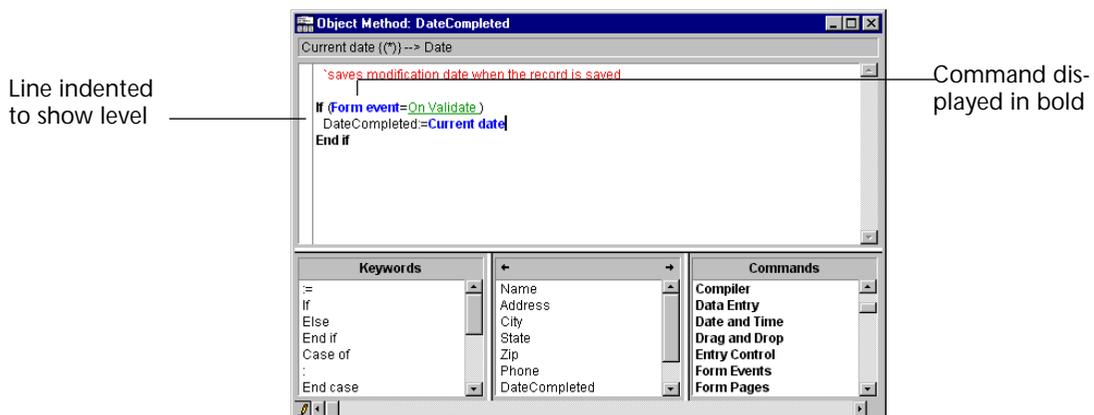
When you press the Return/Enter key, 4th Dimension evaluates the text of the line and formats any routines, constants, keywords, plug-in commands, or methods appropriately. If you are using color to colorize various language elements, the appropriate color will be applied at this time. If you are using flow of control structures, 4th Dimension indents each line to its proper level in relation to the preceding line.

You can move the insertion point by clicking at the location you want. You can select words, whole lines, or several lines by dragging the I-beam over them.

You can use the arrow keys to quickly move from line to line. Using the arrow keys to move across several lines is quicker than clicking because the editor delays evaluating the line for errors.

Note The Listing editor includes numerous navigation shortcuts. These shortcuts are listed in [“Navigational Keyboard Shortcuts” on page 522](#).

The following figure shows a method in the Listing editor. This is an object method attached to the DateModified field in an invoicing database. The value of the field is set to the current date only after the current record is changed (by moving to another record) or accepted.



Brace Matching

The Listing editor has a brace matching option that helps you balance braces, parentheses, quotes, and brackets. There are two levels of brace matching:

- **Small Brace Matching** affects only the opening and closing characters.
- **Big Brace Matching** affects the entire expression enclosed by the opening and closing characters.

When brace matching is active, 4th Dimension tries to find the matching brace, quote, or parenthesis when you type the closing character. When 4th Dimension finds matching characters, either the characters or the entire expression flashes.

For example, when you type

For (\$i;1;Records in selection ([Line Items
and you press the closing bracket “]” to finish the table name expression, 4th Dimension will try to find the opening bracket. When it finds it, it flashes either the opening and closing brackets (Small brace matching) or the table name itself (Big brace matching).

As you continue to type the closing parentheses

For (\$i;1;Records in selection ([Line Items]))

brace matching continues to provide feedback as you complete the arguments for the Records in selection function and the For keyword.

► To use brace matching:

- Choose either **Small Brace Matching** or **Big Brace Matching** from the **Method** menu.

A check mark appears next to the menu command you select. Turn off brace matching by choosing **No Brace Matching** from the **Method** menu.

Adding Method Components from the Scrollable Lists

You can quickly enter keywords, table names, fields, commands, and functions by selecting them from the **Keywords**, **Fields**, and **Routines** lists in the lower portion of the window. Select components from these lists to ensure accuracy and improve speed. In addition to added reliability, when you select a command from the scrollable list, its syntax is displayed in the syntax help area.

When you click a component, 4th Dimension inserts it at the insertion point in the editing area, using the correct syntax for that component. The component can then be modified as normal text.

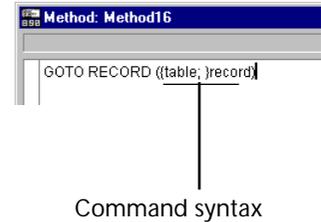
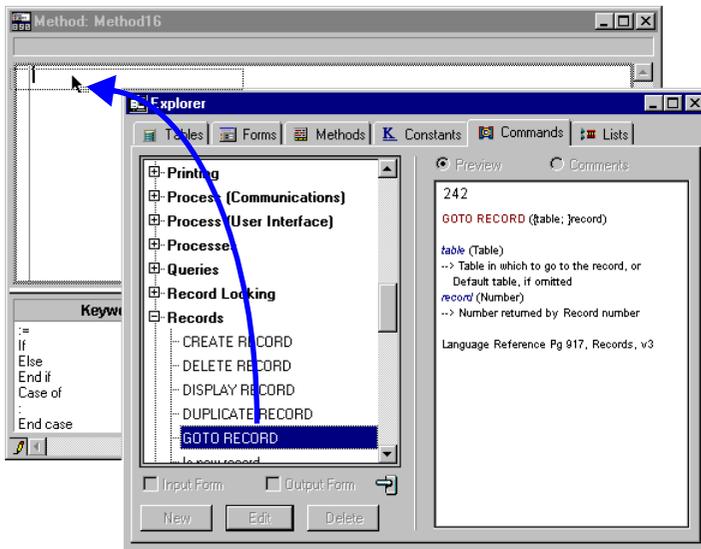
Adding Method Components from the Explorer

You can add components from the Explorer by dragging and dropping components. You can add:

- Table names and field names from the Tables page,
- Table names and form names from the Forms page,
- Project methods, table names, and form names from the Methods page,
- Constants from the Constants page,
- 4th Dimension commands from the Commands page.

When you drag and drop a component, 4th Dimension always uses the correct syntax for the component. For example, if you drag the field name “First Name” from the [People] table, it appears in the Method editor as “[People]First Name”. Similarly, if you drag the Form name “Input” from the People table, it appears in the Method editor as “[People];"Input””.

When you insert a command by dragging it from the Explorer, it appears with its syntax (which consists of all of its parameters) in the Method editor. Of course, you use the syntax that you need to adapt to your usage. This feature reminds you of the parameters that the command expects:



You can choose not to insert the syntax when dragging a command by pressing the **Alt** key (on Windows) or the **Option** key (on Mac OS) when dragging the command.

Using the Wildcard Character

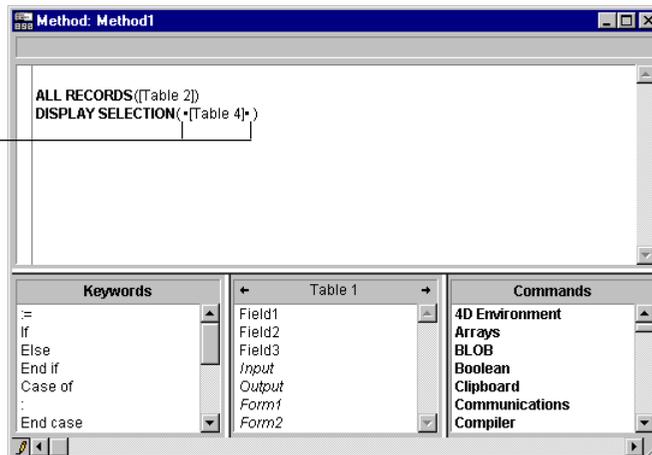
You can enter most commands and table names by typing the first few characters and the wildcard character (**@**). 4th Dimension automatically searches the Keywords, Fields, constants, and Routines list and enters the language component that begins with those characters.

Double-check each entry. You must type enough of the name so that it is distinct from any other name. If more than one name fits the instruction, 4th Dimension picks the middle one in the list of possibilities.

Checking and Correcting Syntax Errors

4th Dimension automatically checks the method syntax to see if it is correct. If you enter text or select a component that is not syntactically correct and then press the **Enter** key (**Return** key on Mac OS) to end the line, 4th Dimension marks the error with bullets, as shown in the following illustration.

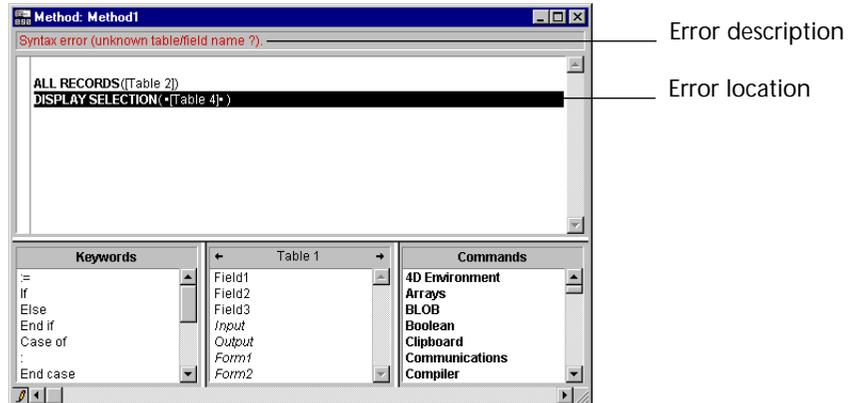
Bullets indicate error (wrong table number, in this case)



You can immediately check the syntax of the current line (without advancing to the next line) by pressing the **Enter** key (on Macintosh) or **Enter** key on the Windows numeric keypad. 4th Dimension evaluates the line, formats it, marks any errors, and places the insertion point at the end of the line. When a line of a method is marked as having improper syntax, fix the entry and press the **Return** or **Enter** key or click any other line in the method. If the line is now correct, 4th Dimension removes the bullets.

The validation of an entire method is done when you close the window. You can also force validation by either selecting **Save Method** *MethodName* from the File menu or pressing the Ctrl+Enter keys (on Windows) or Command+Enter keys (on Mac OS).

When the method is validated, 4th Dimension checks for basic syntax errors and for the structure of the statements (If, End if and so on). When an error is detected, a message is displayed in the syntax display area and 4th Dimension highlights the line that contains the mistake.

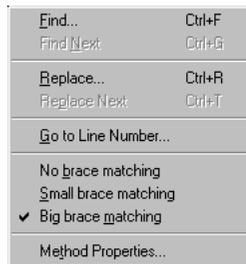


The Listing editor can check only for obvious syntax errors (misspellings and the like). It does not check for errors that occur only during execution. Execution errors are trapped by 4th Dimension when the method is executed. 4th Dimension provides a debugger for handling and correcting these errors. For information about the debugger, refer to the *4th Dimension Language Reference*.

Finding and Replacing Text

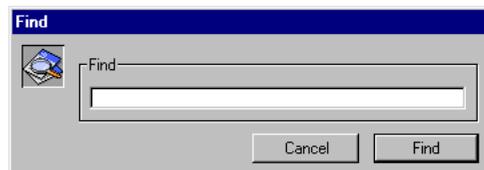
When the Listing method editor is active, 4th Dimension provides the Method menu for performing find-and-replace operations on a method. You can use the Method menu to locate character strings in a method and, if necessary, replace these characters with new ones. You can also use the Method menu to locate a specific line in a method. When you use Find or Replace, 4th Dimension searches only from the current position of the insertion point to the end of the method.

The figure below shows the Method menu.



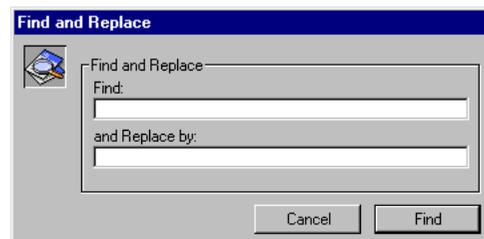
Here is a description of the Method menu commands:

- **Find** Opens a dialog box in which you specify a character string for a search.



When you click the Find button, the editor begins searching from the current text insertion point and selects the first occurrence it finds. If you select text prior to choosing Find, the text is entered as the string to search for. You can use this text or replace it by entering another string.

- **Find Next** Performs the previously defined search again. It begins searching from the point of the last found occurrence of the string.
- **Replace** Opens a dialog box in which you specify a character string for a search-and-replace operation.



The Method editor searches from the current location of the text insertion point to the end of the method. When it finds text that matches the character string, it replaces it with the new character string. If you

select text prior to choosing **Replace**, the text is entered as the string to search for. You can use this text or replace it by entering another string.

- **Replace Next** Performs the previously defined search-and-replace operation again. It begins searching from the point of the last found occurrence of the string.
- **Goto Line Number** Opens a dialog box in which you specify the line number that you want to find. When you click **OK**, the editor finds and highlights that line in the method. This is especially useful when used with 4D Compiler which flags runtime errors by the line number in which they occur.
- **No Brace Matching, Small Brace Matching, and Big Brace Matching** Controls brace matching in the Method editor. For more information, see the section [“Brace Matching” on page 513](#).
- **Method Properties** Opens the Method Properties dialog box, where you can rename the method and set access privileges. For more information about method properties, see the section [“Setting Access Privileges” on page 502](#).

Assigning Colors to Method Elements

When the Listing method editor is active, 4th Dimension provides the **Colors** menu. This menu allows you to set a different color for each of the various elements of a method (fields, tables, interprocess variables, constants, keywords, plug-in commands, and so on). When a color is set for an element, every instance of that element appears in the selected color.

The use of color to separate different types of elements in a method can be very helpful in maintaining clear code and in debugging methods.

- ▶ To set the color of an element:
 - 1 Choose the name of the element from the list of elements in the **Colors** menu.

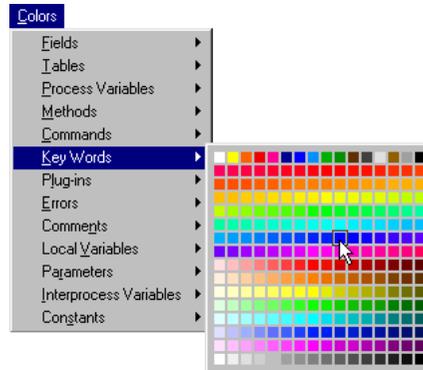
The following illustration shows the Colors menu.



Each element has its own color palette submenu.

2 Select the desired color from a color palette submenu.

The figure below shows a color being selected for Keywords.



Color settings are made for the copy of 4th Dimension that you are running and will appear in methods for all databases opened with that copy. To apply the color setting to the database — regardless of which copy of 4th Dimension is being used, hold down the **Alt** key (**Option** key on Macintosh) while making your color selections.

You must have a color monitor to use the Colors menu. The number of colors that you can choose from depends on the number of colors to which your monitor is set. If your monitor is set to 16 colors, you can use only the first sixteen colors in the cascading menu. If your monitor is set to 256 or 16 million colors, you can use any of the colors in the picture menu.

Opening Methods and Forms

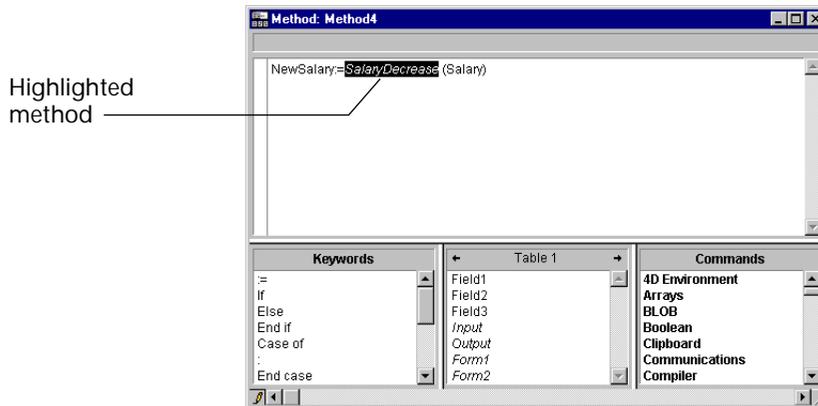
When you are working in the Listing editor, you may find that you need to look at another method or open one of your forms.

4th Dimension provides a quick way to do this with keyboard equivalents.

- ▶ To open a method from the Listing editor:

- 1 Highlight the name of the method that you want to open.

The figure below shows a method name being selected in the Listing editor.



- 2 Press Command-P (on Macintosh) or Ctrl+P (on Windows) to open the method.

The method appears in a Method editor window.

- ▶ To open a form from the Listing editor:

- 1 Highlight the name of the form that you want to open.

Be sure to select the surrounding quotation marks. If you select only the name, the name must be unique to that method or form. If you select the table name along with a form name, 4th Dimension will open the form for the specified table.

- 2 Press Command-L (on Macintosh) or Ctrl+L (on Windows) to open a form.

4th Dimension highlights the form name in the Forms page of the Explorer.

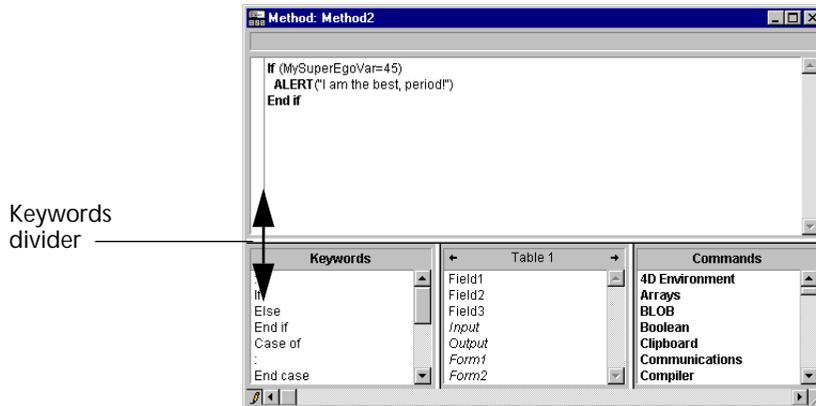
- 3 Press Enter to open the form in the Form editor.

If you have several forms with the same name, 4th Dimension opens the first occurrence of that form.

Managing the Listing Editor Window

You can change the relative sizes of the Editing area and the other scrollable areas in the method window to fit your needs. Place the pointer over the divider until the pointer changes to a divider pointer  and drag the divider that separates the Editing area from the list areas.

The following illustration shows the Keywords divider.



By dragging the divider up, you increase the number of keywords, fields, and routines that you can see. By dragging the divider down, you reduce or remove the Keywords, Fields, and Routines lists completely. You can also hide the Keywords and the other lists by selecting **Hide Keywords** in the Database Properties dialog box.

Navigational Keyboard Shortcuts

Helpful keyboard shortcuts to navigate the code are now available in 4D's Method editor. These shortcuts are also available in all of 4th Dimension's dialog boxes that contain data entry areas.

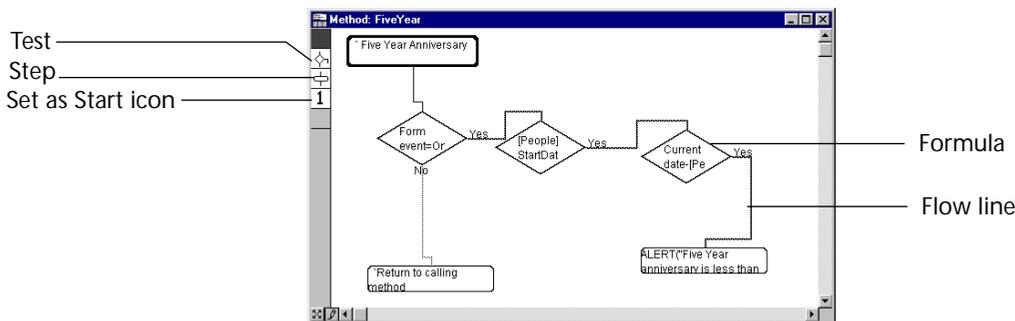
Windows	MacOS	Action
[Shift]+[→]		Create and enlarge the selection, character by character, to the right, or Reduce the selection, character by character, from the left
[Shift]+[←]		Reduce the selection, character by character, from the right or Create and enlarge the selection, character by character, to the left
[Shift]+[↓]		Create and enlarge a selection, line by line, from the top to the bottom
[Shift]+[↑]		Create and enlarge a selection, line by line, from the bottom to the top
[Ctrl]+[Shift] +[→]	[Command]+[Shift] +[→]	Create and enlarge the selection, word by word, from the right
[Ctrl]+[Shift] +[←]	[Command]+[Shift] +[←]	Reduce the selection, word for word, from the right, or create and enlarge the selection, word by word, from the left
[Ctrl]+[→]	[Command]+[→]	Move the insertion point, word by word, from left to right
[Ctrl]+[←]	[Command]+[←]	Move the insertion point, word by word, from right to left
[Home]		Place the insertion point at the beginning of the line
[End]		Place the insertion point at the end of the line
[Ctrl]+[Home]	[Command]+ [Home]	Place the insertion point at the beginning of the method
[Ctrl]+End	[Command]+[End]	Place the insertion point at the end of the method
[Shift]+[Home]		Select all the characters in the line that are to the left of the cursor
[Shift]+[End]		Select all the characters in the line that are to the right of the cursor
[PgUp]		Scroll the contents of the method, page by page, from the bottom to the top (doesn't modify the insertion point)
[PgDn]		Scroll the contents of the method, page by page, from the top to the bottom (doesn't modify the insertion point)

Using The Flowchart Editor

The Flowchart editor lets you create and edit 4th Dimension methods graphically by adding steps and tests to a flowchart. The Flowchart editor lets you create the same kinds of methods that you create with the Listing editor, except that you build your methods visually.

You can think of a flowchart as resembling plumbing in a house. The method, like water, flows through the pipes and its movement is determined by the state of valves in the plumbing. If a valve is open (TRUE), the water (method) will flow through. If the valve is not open (FALSE), the flow will stop at that point, or go elsewhere.

The following illustration shows an example flowchart method.



In a flowchart, there are two types of objects: steps and tests. In addition, there are flow lines that connect steps and tests. Each step and test also has a formula that defines the operation of the step or test. The formula is written in a dialog box. The flowchart palette provides icons that you click to create steps and tests, to set a new starting point, and to duplicate objects.

The following are descriptions of the four major Flowchart method components.

- **Step object** Directs 4th Dimension to perform a specific operation or calculation (a statement). For example, a step can use the NEXT PAGE command to make the next page in the form the current form. Or it can be an assignment statement that assigns a value to a field or variable. Unlike a test, a step is not conditional — the event in the step occurs each time the method executes the step.
- **Test object** Tells 4th Dimension to evaluate a condition (a Boolean condition) to determine the direction that the method will follow. A

test must evaluate to TRUE or FALSE. Each test object has two flow lines leading to other steps or tests — a Yes flow line and a No flow line. For example, a test can be associated with a condition such as “Salary >= 50000.” When 4th Dimension reads the contents of the Salary field, it will follow the Yes flow line if the field contains a value greater than or equal to 50,000.

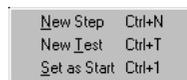
- **Flow lines** Flow lines connect the steps and tests in the flowchart and provide the direction of movement through the flowchart.

The direction of movement is shown by the way a flow line enters and leaves a step or test. The flow line that connects the top of a step or test brings information or instructions to the step or test. The result of a step flows out the bottom of a step. The Yes result of a test flows out the side of a test; the No result of a test flows out the bottom of a test.

- **Formulas** Each step and test must be associated with a one-line formula. The formula defines the precise step to be carried out or test to be performed. The process of creating a formula is described later in this section.

The Flowchart Menu When the Flowchart editor is active, 4th Dimension provides a menu for performing flowchart operations on a method. These menu commands duplicate the functions of the icons in the flowchart palette.

The figure below shows the commands on the Flowchart menu.



The Flowchart menu contains the following menu commands:

- **New Step** Adds a new step object to the flowchart,
- **New Test** Adds a new test object to the flowchart,
- **Set as Start** Specifies a new starting point in the flowchart.

Creating a Flowchart Method

There are three basic steps for creating a method with the Flowchart editor:

- 1 Use the icons in the palette or the Flowchart menu commands to add steps and tests in the Flowchart window.
- 2 Create a formula for each step and test in the window.
- 3 Draw flow lines between the objects.

4th Dimension provides basic syntax error-checking. Other error-checking is performed when the method is executed.

These basic steps are described in detail in the following sections.

Creating Steps and Tests To create steps and tests:

- 1 With the Flowchart editor active, click the Step or Test icon in the palette  .

OR

Choose New Step or New Test from the Flowchart menu.

When you move the pointer into the Flowchart area, it becomes a crossbar.

- 2 Position the crossbar where you want the upper-left corner of the object to appear and click on the location.

4th Dimension creates the test or step. The crossbar becomes the standard pointer again.

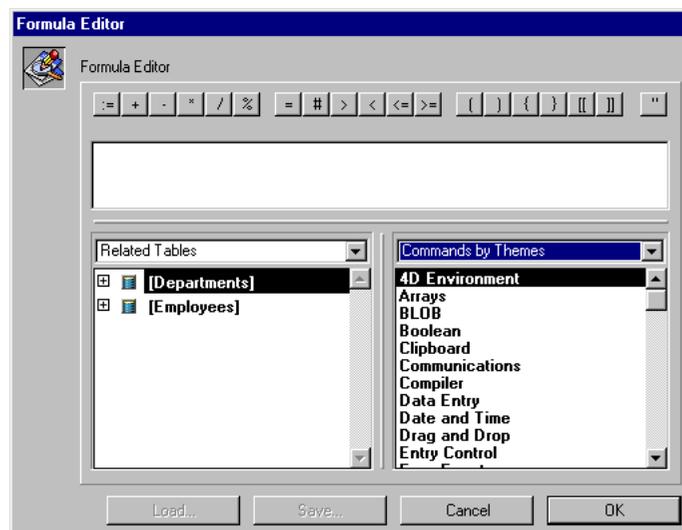
Creating a Formula

You use a formula (a one-line method) to control each step or test.

► To create a formula:

- 1 Double-click the Step or Test object for which you want to create a formula.

4th Dimension displays the Formula editor as shown in the following illustration.



You use the Formula editor in the same way as the Listing editor, except that the Formula editor exists in a dialog box, not an editor window, and it allows only one-line statements. You cannot move or resize the dialog box, nor can you change the relative size of the editing area and the lists at the bottom.

A formula, like a method, can contain up to 32,000 characters.

A step formula must result in an instruction for the computer to do something. The NEXT PAGE statement is an example of a step formula.

A test formula must result in TRUE or FALSE (a Boolean condition). The statement "Salary>50000" is an example of a test formula.

As you create formulas for steps and tests, 4th Dimension automatically checks the formula syntax to see if it is correct. If you enter text or select an object that is not compatible with proper syntax, 4th Dimension marks the error with bullets.

- 2 Write the formula, and then click OK.

4th Dimension displays the Flowchart editor. The formula is displayed in the step or test for which it was created.

Drawing Flow Lines

To define the flow of a method through the chart, you need to draw flow lines between the objects in the Flowchart window.

- To draw a flow line:

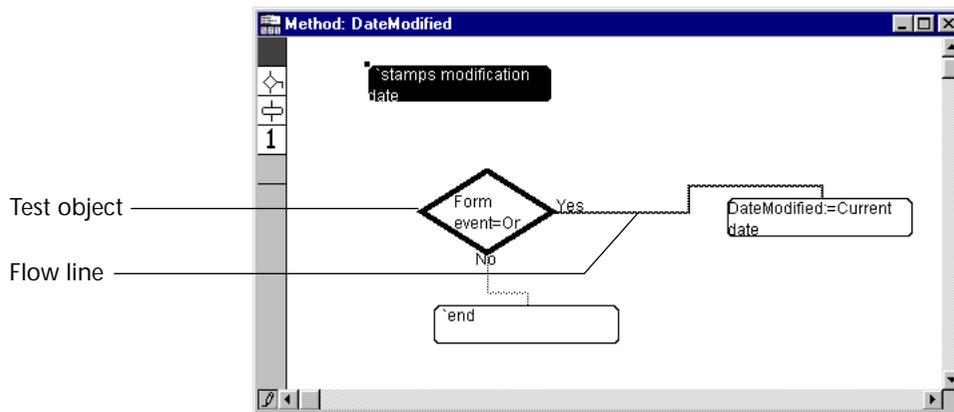
- 1 Position the pointer on the lower boundary of a step object or on one of the corners of a test object.

The pointer becomes a small, upward-pointing arrow.

To draw a Yes line from a test, position the pointer on the left or right corner. To draw a No line from a test, position the pointer on the bottom corner.

- 2 Drag away from the object and towards the object to which you want to connect the flow line.
- 3 Release the mouse button at the border of the object you want to connect.

4th Dimension draws a flow line between the two objects.



4th Dimension lets you draw through other objects to connect objects in the flowchart. If you are drawing flow lines from a test object, 4th Dimension automatically labels the lines as Yes and No.

► To erase a flow line:

- 1 Position the pointer on the flow line you want to delete.
- 2 When the pointer becomes a small, upward-pointing arrow, drag the line to an area between objects (so that the pointer is not touching any object) and release the mouse button.

The selected flow line disappears.

Specifying a First Step

When you create a flowchart method, the first step you create is shown in bold outline indicating that it is the first step. The first step is where the method begins when it executes. You can change the first step at any time. Setting a new first step is a way to isolate parts of the flowchart method in order to debug problems.

When you set a new first step, 4th Dimension ignores all steps and tests before the new starting point.

► To set a new first step:

- 1 Select the step you want to specify as the first step in the method.
- 2 Click the Set as Start icon **1** in the palette.

OR

Choose Set as Start from the Flowchart menu.

4th Dimension displays the step in bold as the new starting point.

Editing a Flowchart Method

You can change the way a flowchart is displayed by moving and resizing components so that the flowchart is clear and easy to read. You can also change the formulas that define the steps and tests.

The following are several ways to select flowchart objects:

- Click the test or step in the Flowchart editor window so that 4th Dimension highlights the selected object.
- Select multiple objects by dragging a selection rectangle around the tests or steps you want to select.
- Select multiple objects by holding down Shift while clicking the tests or steps you want to select.

Moving an Object

To move an object:

- 1 Select the object in the Flowchart editor window.
- 2 Drag the object to its new location in the window.

If there are flow lines connected to the object you are moving, they follow the object.

Resizing an Object

To resize an object:

- 1 Select the object in the Flowchart window.
4th Dimension highlights the step or test and displays a resizing handle in the upper-left corner.
- 2 Move the pointer over the resizing handle on the selected object.
The pointer changes into a multi-directional arrow.
- 3 Drag the handle toward the center of the object to shrink it, or away from the center to enlarge it.
4th Dimension resizes the object.

Deleting an Object

To delete an object:

- 1 Select the object or objects that you want to delete.
- 2 Choose Clear from the Edit menu.

OR

Press the Backspace key.

4th Dimension deletes the selected object or objects. The deletion also removes any flow lines connected to the deleted object.

Editing a Formula

To edit a formula for a step or test:

- 1 Double-click the test or step object whose formula you want to edit. 4th Dimension displays the Formula editor. The existing formula appears in the Formula area.
- 2 Edit the formula, using the editing techniques described in [“Creating a Flowchart Method” on page 524](#).
- 3 Click the OK button to accept the formula and return to the Flowchart window.
Click the Cancel button to discard the formula or editing changes and return to the Flowchart window.

8

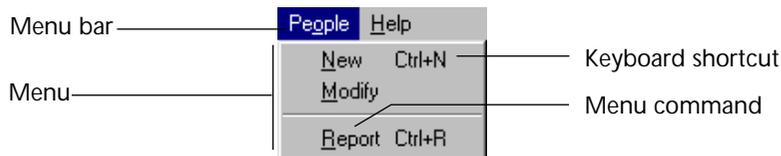
Creating Custom Menus

You can create custom menus for your databases and custom applications. Because pull-down menus are a standard feature of any desktop application, their addition will make your databases easier to use and will make them feel more familiar to users. When you create custom menus, you can also create custom toolbars. With custom menus and toolbars, your databases will perform more like “stand-alone” applications.

When you create a custom application, you must create at least one menu bar with at least one menu. For detailed information about creating custom applications, refer to the *4th Dimension Language Reference*.

Designing Menus

In general, menus provide menu commands that the user chooses to perform database tasks: modifying records, searching for records, printing reports, and so on. The figure below shows an example of a custom menu.



A menu bar is a group of menus that can be displayed on a screen together. Each menu on a menu bar can have many menu commands in it, including separator lines that divide the menu commands into groups and keyboard equivalents. When the user chooses a menu command, it calls a project method that performs an operation.

You can have many separate menu bars for each database. For example, you can use one menu bar that contains menus for standard database operations and another that becomes active only for reporting.

One menu bar may contain a menu with menu commands for entering records. The menu bar appearing with the input form may contain the same menu, but the menu commands are disabled because the user doesn't need them during data entry.

You can also use the Menu Bar editor to create custom toolbars. To do so, you associate an icon with a menu command. The icon appears in the toolbar and the text of the menu command is used as the icon's Tip.

When you use the same menu for more than one menu bar, you can take advantage of the concept of "instances" of a menu to simplify the process of managing menus. For complete information about this method of managing menus, see the section "[Working With Instances of a Menu](#)" on page 543.

When you design menus, keep the following two rules in mind:

- **Use menus for functions that are suited to menus** Menu commands should perform tasks such as adding a record, searching for records, or printing a report.
- **Group menu commands by function** For example, all menu commands that print reports should be in the same menu. For another example, you might have all operations for a certain table in one menu.

You use the Menu Bar editor to create menus. You can perform the following operations in the Menu Bar editor:

- Specify styles for menu commands,
- Specify keyboard equivalents for menu commands,
- Add dividing lines to menus,
- Assign password groups to menu commands,
- Enable or disable menu commands,
- View sample menus while you are creating the menu bar,
- Paste in a custom graphic associated with each menu bar that will be displayed as a splash screen,
- Create a custom toolbar that uses an icon for each menu command,
- Create a connected menu,
- Specify that a new process start when a menu command is chosen.

Each of these tasks is explained in this chapter.

Creating Menus

4th Dimension allows you to create entire menu bars. A menu bar is the collection of menus that appears at the top of your application window. The menu bar displays the menu titles and the menus pull down to display the menu commands. Every menu command should be associated with a project method.

4D Server Object locking occurs when two or more users attempt to modify the same menu bar at the same time. If a user is modifying a menu in a menu bar, the menu is locked. Other users can modify different menus in that menu bar, but they cannot modify the same menu. In addition, if a user is modifying any aspect of a menu bar, other users cannot add any new menus to the menu bar.

Basic Steps for Creating Menus

The following are the basic steps for creating custom menus:

- 1 Create one or more menu bars.
See the next section, [“Creating a Menu Bar” on page 534](#).
- 2 Create the menus that will pull down from the menu bar.
See the section [“Adding Menus” on page 536](#).
- 3 Add menu commands to each menu in the menu bar.
See the section [“Adding Menu Commands” on page 538](#).
- 4 Assign a project method to each menu command.
When the user chooses that menu command, 4th Dimension executes the method. See the section [“Assigning Methods to Menu Commands” on page 541](#).
- 5 Write the project methods that perform the menu commands.
For more information about using the Method editor, see the section [“Using the Listing Editor” on page 507](#). Refer to the *4th Dimension Language Reference* for detailed information about 4th Dimension’s programming language.
- 6 Make any enhancements you want such as special font styles, separator lines, keyboard shortcuts, a toolbar icon, and so on.
See the section [“Enhancing Menus” on page 546](#).
- 7 Assign password access groups to menu commands (optional).
See the section [“Assigning a Group To Database Objects” on page 575](#).

- 8 Specify that a new process be started when q menu command executes (optional).

See the section [“Assigning Methods to Menu Commands”](#) on page 541.

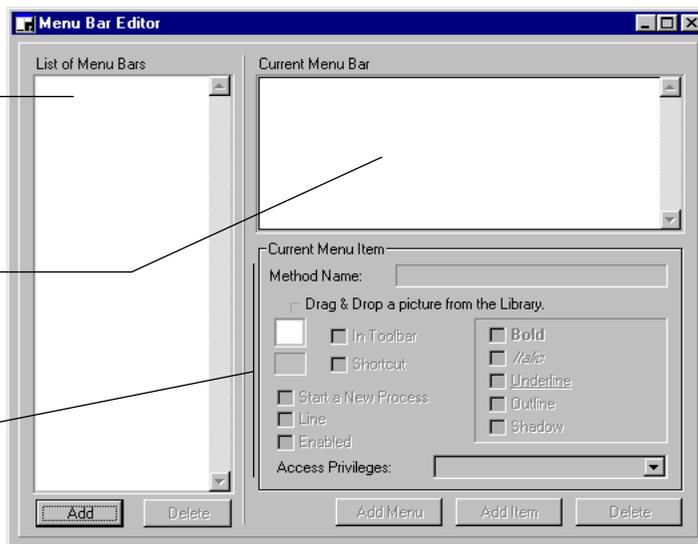
Creating a Menu Bar This section describes the process of creating a custom menu bar.

- ▶ To create a menu bar:
 - 1 Choose Menu Bar Editor from the Tools menu.
 - 4th Dimension displays the Menu Bar editor. If there are any existing menu bars, they are displayed in the panel on the left.

The List of Menu Bars shows the existing menu bars

The Current Menu Bar area shows the menus belonging to the selected menu bar

The Current Menu Item area shows the properties of the selected menu command

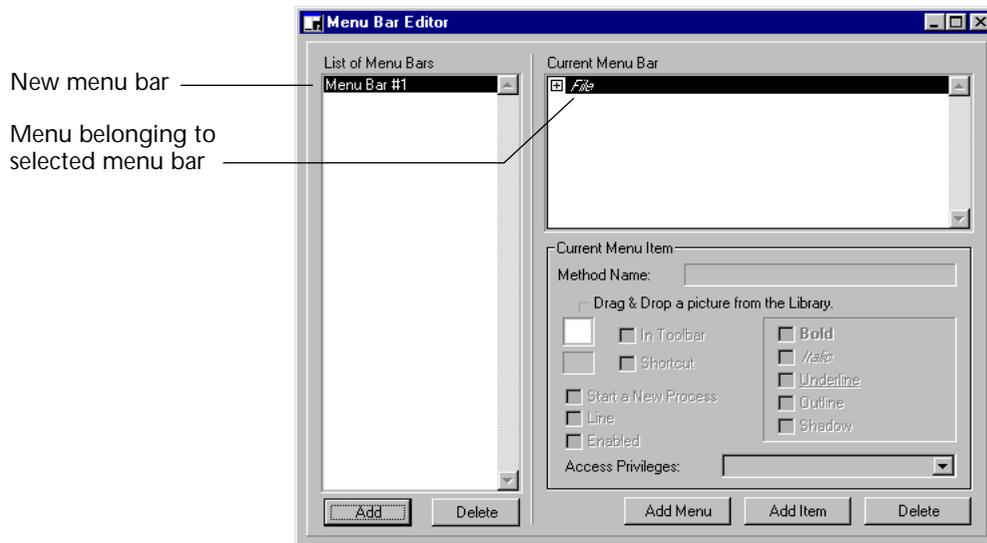


4th Dimension assigns menu bar numbers sequentially — Menu Bar #1 appears first. You cannot change menu bar numbers.

If you delete a menu bar, 4th Dimension automatically rennumbers any remaining menu bars. If you refer to menu bars in a method, you may need to update the method to reflect new menu bar numbers.

- 2 Click the Add button.

The new menu bar appears in the List of Menu Bars, as shown in the following illustration.



At this point, you can begin assigning menus to the menu bar and adding menu commands to the menus.

The Menu Bar editor displays menu information in the following three lists:

- **List of Menu Bars** displays the name of each menu bar.
- **Current Menu Bar list** displays the name of each menu in the menu bar currently selected in the List of Menu Bars. The Current Menu Bar list is a hierarchical list; each menu can be expanded to display the menu commands belonging to the menu.
- **Current Menu Item Properties area** displays the properties of the selected menu command.

Notice that the File menu appears italicized in the Current Menu Bar list. The italics indicate that the text for the menu title is being retrieved from a string resource. If you press the Ctrl key (on Windows) or the Command key (on Mac OS) while clicking the File menu, the string resource number “:79, 1” appears.

The Edit menu is not listed, but is automatically added to the menu bar when it is displayed. You can add menu commands to the File menu, but you cannot edit the Edit menu. When the menu bar is in use, the

File and Edit menus occupy the first two places at the left end of the menu bar.

Note The File menu usually includes Quit as the last menu command.

Adding Menus

You can add menus to either a new or an existing menu bar. You can add a menu to a menu bar in the following two ways:

- Create a new menu,
- Create a connected menu.

When you create a connected menu, you are attaching an exact duplicate of an existing menu to a menu bar. This duplicate is called an *instance* of that menu. For information about creating an instance of a menu, see the section [“Creating Connected Menus” on page 543](#).

Before you can create a connected menu, the first instance of that menu must be created in the Menu editor window.

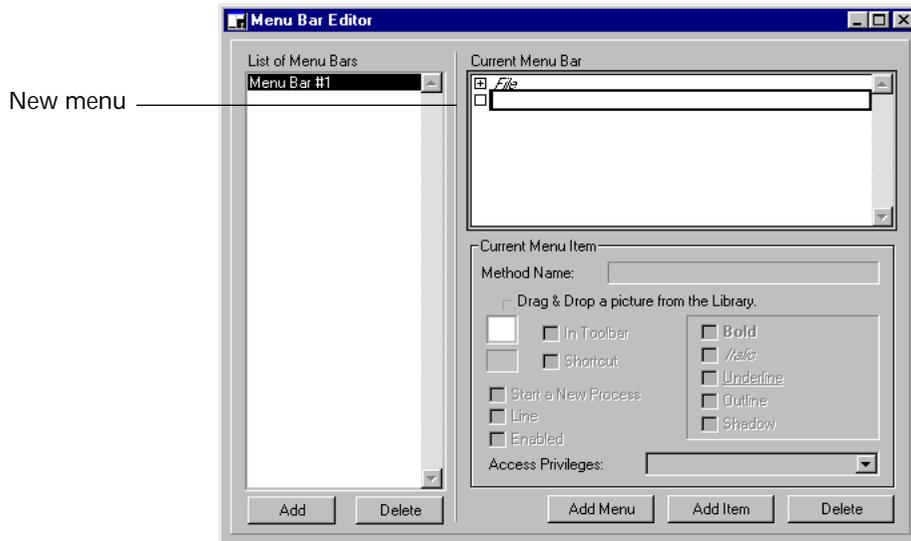
You can create a menu in the following two ways:

- Append a new menu to the end of the current list of menus,
- Insert a new menu anywhere in the current list of menus.

You don't have to create the menus in the order that they will eventually appear. You can rearrange menus after you've created them using drag and drop. For more information, see the section [“Rearranging Menus and Menu Commands” on page 540](#).

- ▶ To add a menu:
 - 1 Choose Add Menu from the Menu menu.
OR
Click the Add Menu button.

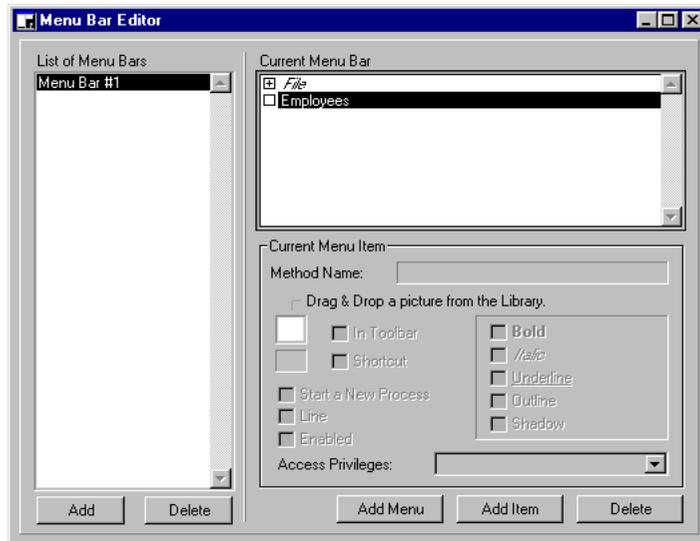
4th Dimension adds a new menu to the Current Menu Bar list so that you can enter the menu title.



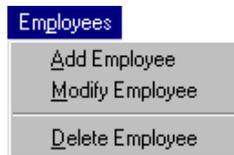
2 Type the name of the new menu in the entry area.¹

1. You can use a STR# resource instead of text. To do so, enter the STR# in the format *STR#, line #* in place of the text of the menu title. When you press tab, the string resource appears in italics. You can also use an interprocess variable containing the STR# resource number in place of *STR#*.

The maximum length for a menu title is 15 characters. Additional characters are ignored.



As you enter the menu name, 4th Dimension displays the name to the right of the current menu bar to show how it will look in the finished application. You can pull down this sample menu in the same way that you pull down active 4th Dimension menus. When you add menu commands, they appear in this sample menu.



- 3 Repeat steps 1 and 2 to add more menus.

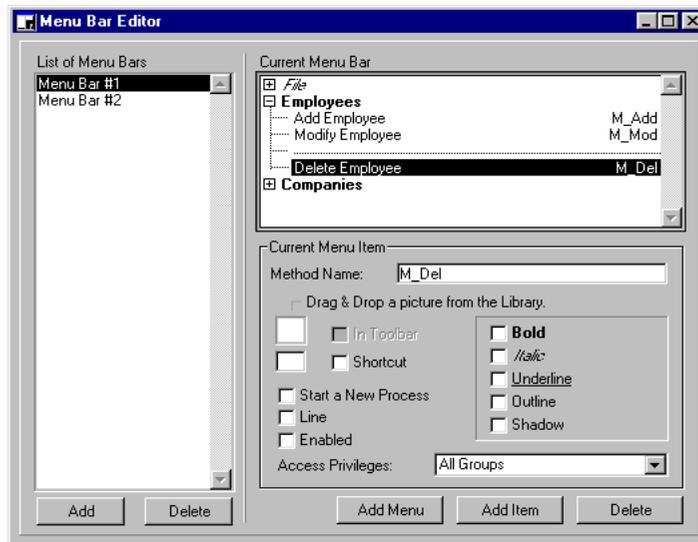
Adding Menu Commands

For each menu in the menu bar, you must create the menu commands that appear when the menus are pulled down.

- To add a menu command:
 - 1 Select the menu that you want to provide with a menu command.
 - 2 Click Add Item.
 OR
 Choose Add Item from the Menu menu.
- 4th Dimension adds a new item to the Current Menu Bar list.

3 Type the name of the new menu command.

The figure below shows Delete Employee being added as a menu command.



Notice that the third menu command appears blank. This menu command is a placeholder for a disabled separator line. For information about creating a separator line, see the section [“Adding Separator Lines”](#) on page 548.

The figure below shows the menu commands and the separator line in the sample menu.



4 Repeat steps 1 through 3 to add more menu commands to the Items list.

Using Control Characters in Menu Labels

You can define the properties of the menu commands by using control characters (metacharacters) directly in the menu command labels. For instance, you can assign the keyboard shortcut Ctrl+G (on Windows) or Command+G (on Mac OS) for a menu command by placing the “/G” characters in the label of the menu item label. Control characters do not appear in the menu command labels. You therefore need to make sure you are not using them when they are not used as control characters. The control characters are the following:

- "("
- "<"
- "!"
- "^"
- "/"

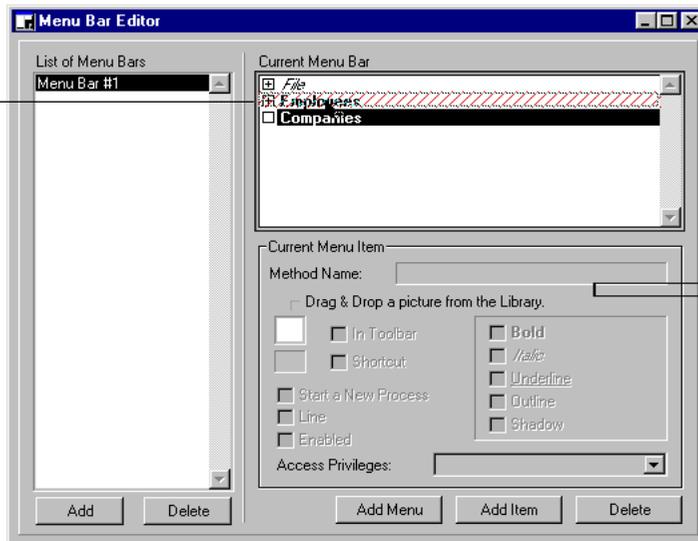
For more information on the use of these characters, refer to the description of the APPEND MENU ITEM in the 4th Dimension *Language Reference* manual.

Rearranging Menus and Menu Commands

After you create the menus for a menu bar and the menu commands for a menu, you can reorder them using drag and drop. To insert a menu command at a different place in the order, simply drag the menu command to the new location. To move a menu, simply drag a menu to another location in the list of menus.

The following illustration shows a menu being dragged.

Companies menu being dragged between File and Employees menus



Assigning Methods to Menu Commands To enable a menu command to perform its function, you must assign a project method to it. These methods perform the functions indicated by the menu commands. For example, the **Monthly Report** menu command can call a project method that prepares a monthly report from a table containing financial data. When a menu command is chosen, 4th Dimension executes the project method assigned to it.

You create the project method in the Method editor. You can create it either before or after you assign it to the menu command. You can assign the method when you create the menu command or at a later time. If you do not assign a method to a menu command, when that menu command is chosen in the Custom Menus environment, 4th Dimension automatically returns to the User environment.

When you have assigned a method to a menu command in the Menu editor, you can open that method by simply selecting the menu command in the Current Menu Bar area and pressing Ctrl+P (on Windows) or Command-P (on Macintosh).

► To assign a method to a menu command:

1 Create the menu command.

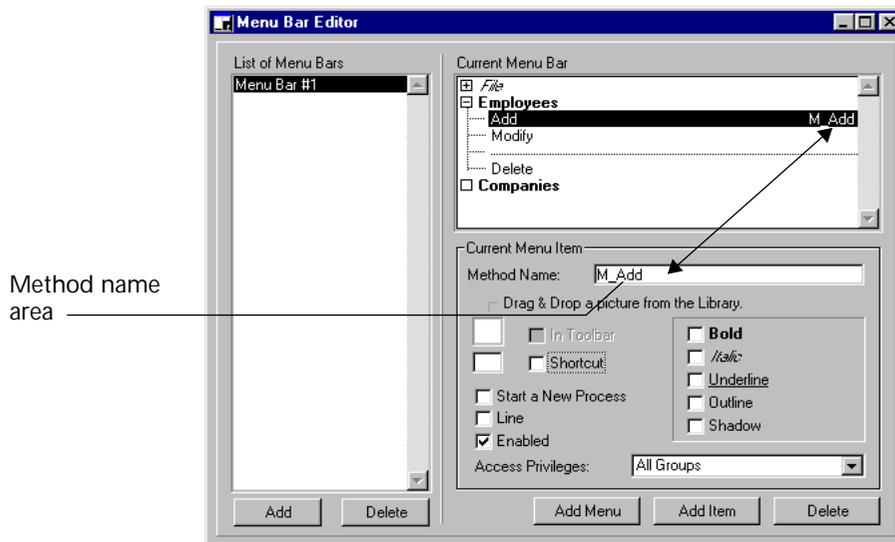
OR

Select it if it already exists.

4th Dimension highlights the selected menu command. The Current Menu Item area changes to show the properties of the selected menu command.

2 Drag a method name from the Methods page of the Explorer to the Method Name area in the Menu Bar editor or type the name of the method in the Method Name area.

If you typed the method name, press Tab or click outside the entry area to save the method name.



You can add a method name before you write the method.

When you finish entering the name, it appears in the Current Menu Bar area.

Note If you change the name of a method that is used in a menu, you must update the method name here in the Menu Bar editor.

3 Click the Start a New Process check box (optional).

If you click the Start a New Process check box, a new process is created when the menu command is chosen.

Normally, a method attached to a menu command executes within the current process unless you explicitly call New process in your code. The Start a New Process check box makes it easier to start a new process.

If you click the Start a New Process check box, 4th Dimension will create a new process when the menu command is chosen. In the Process list, 4th Dimension assigns the new process a default name using the format *M_ProcessNumber*. The names of processes started from a menu are created by combining the prefix "M_" with the process number.

For more information about processes, see [Chapter 12, "Managing Processes"](#) on page 613.

Working With Instances of a Menu

When you write a custom application, you will usually find that you reuse certain menus in several places in your application. The same menu may be attached to several menu bars.

If you create a menu from scratch each time you use it in a menu bar, you must manage each menu separately, on a menu bar by menu bar basis. If you want to change a menu (disable or enable a menu command, place a check mark next to a menu command, and so on), you must change it every place that it occurs.

If you take advantage of the concept of “instances” of a menu by creating connected menus, managing menus becomes much easier. With connected menus, it is possible to update a menu *wherever it occurs* in a single step.

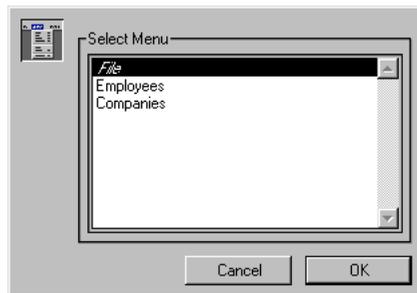
Creating Connected Menus

When you create a menu using the Add Menu command in the Menu menu or the Add Menu button, you create the first instance of that menu.

When you want to reuse the menu in another menu bar, you connect the menu to that menu bar. Connecting the menu attaches another instance of the menu to the menu bar.

- ▶ To connect a menu to a menu bar:
 - 1 Make sure that the menu bar to which you want to connect is the currently selected menu bar in the Menu Bar editor.
 - 2 Select the menu that you want to appear below the connected menu.
 - 3 Choose Connect Menu from the Menu menu.

The Select a Menu dialog box appears.



- 4 Select the name of the menu you want to connect.

5 Click the OK button.

This connects another instance of the menu to the menu bar. The connected menu appears directly above the currently selected menu in the Menu Bar editor.

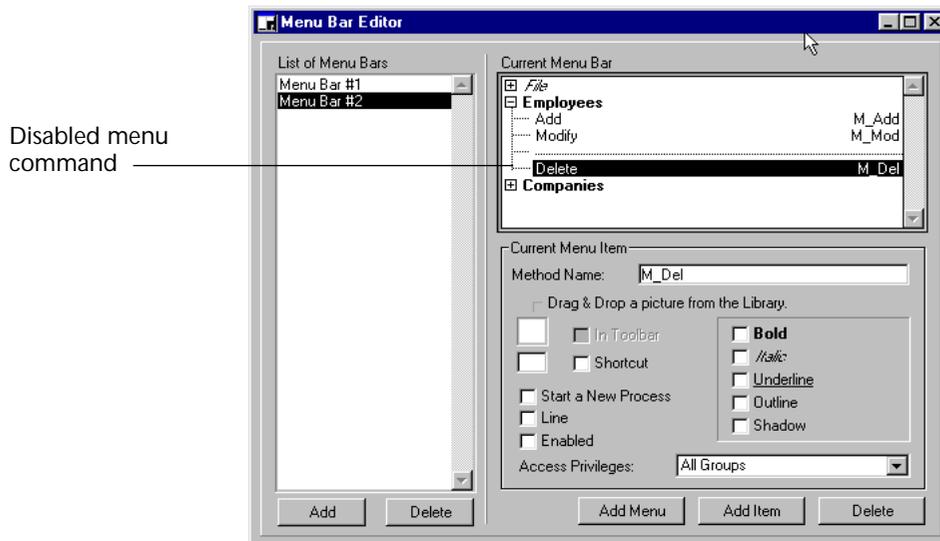
Modifying Connected Menus

The most important thing to remember about connected menus is that all instances of a menu refer to a single menu description. If you modify any instance of a menu, all instances of the menu reflect the change.

Modifications include the following:

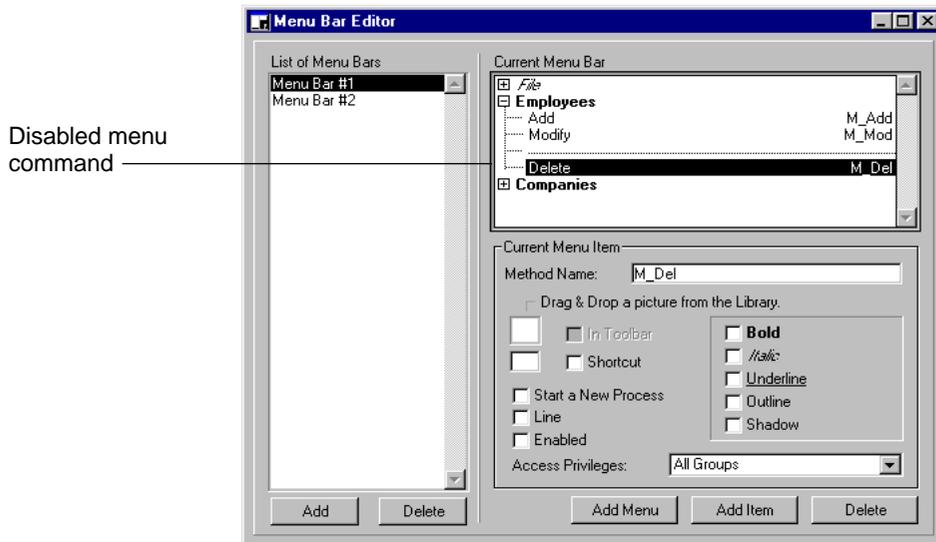
- Changes to the name of the menu, menu commands, or methods,
- Enhancements to the menus, such as adding separator lines, disabling or enabling menu commands, changing fonts or styles, and assigning keyboard equivalents,
- Modifications produced with the 4th Dimension language, such as disabling a menu command by using the DISABLE ITEM command.

The following illustration shows a menu command that has just been disabled in a connected menu.



The menu command is disabled in menu bar #2, but since the menu command belongs to a connected menu, the modification appears in

all other instances of that menu. The following illustration shows the same change reflected in the Employees menu in menu bar #1.



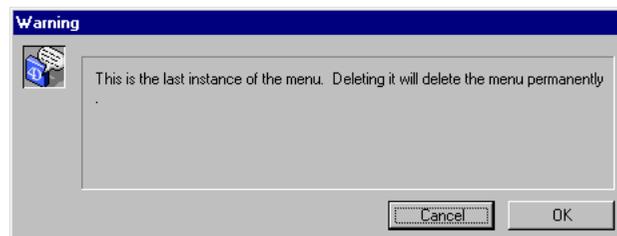
For more information about disabling a menu command, see the section [“Enabling and Disabling Menu Commands”](#) on page 548.

Deleting Connected Menus

Although connected menus refer to a single menu description, there is no main instance of a menu to which other instances refer.

When you delete a menu, you are only deleting an instance of the menu. The menu is not removed from the database — it is only deleted from a single menu bar — unless you delete the last instance of a menu.

If you attempt to delete the last instance of a menu in your database, a confirmation dialog box appears to remind you that it is the last reference to the menu and that deleting it will remove the menu from your database permanently.



For complete information about deleting menus, see the section [“Deleting Menus and Menu Commands”](#) on page 552.

Enhancing Menus

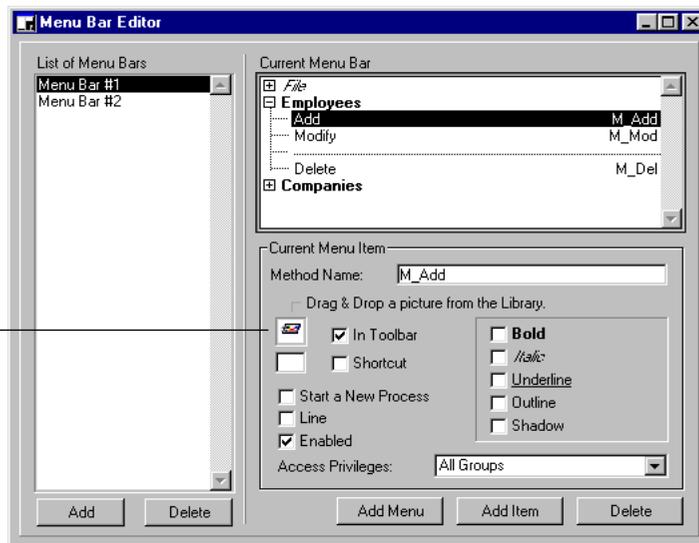
You can change the font style of menu commands, add separator lines between groups of menu commands, assign a keyboard equivalent for a menu command, and enable or disable menu commands. You can also create a custom toolbar by assigning icons to menu commands.

Adding an Icon to a Menu Command

You can associate an icon with a menu command. When you do so, the icon is used as a button in the toolbar that is displayed whenever the menu bar is displayed.

- ▶ To add an icon:
 - 1 Highlight the menu command to which you want to associate an icon. The Current Menu Item area changes to show the properties of the selected menu command.
 - 2 Open the Picture library, select a graphic, and drag the picture from the Picture library to the In Toolbar area.
 - 4th Dimension displays the icon in the area.

Icon from Picture library added to menu item



When the menu bar is used in the Custom Menu environment, 4th Dimension displays the icon in the toolbar and automatically uses the text of the menu command as the Tip for the toolbar button.

Changing Font Styles

4th Dimension lets you customize menus by applying different font styles to the menu commands.

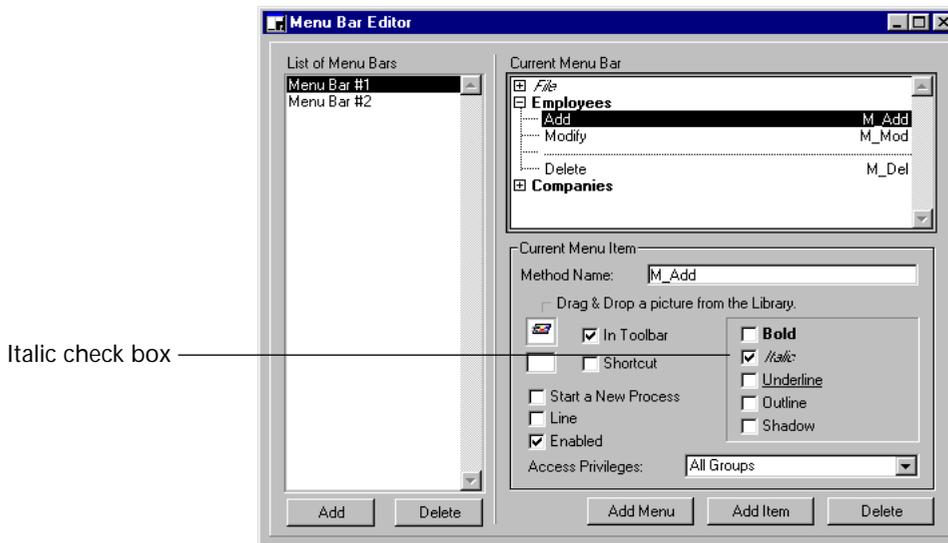
You can customize your menus with these styles:

- Bold,
- Italic,
- Underline,
- Outline (displayed on Macintosh only),
- Shadow (displayed on Macintosh only).

Be cautious when applying font styles to your menus — too many styles will be distracting to the user and give a cluttered look to your application.

To apply a style, select the menu command you want to modify and then choose the style from the check boxes.

The following illustration shows an italic style being applied to a menu command.



The sample menu appears like this:



Enabling and Disabling Menu Commands

You can specify whether a menu command will appear enabled or disabled. An enabled menu command can be chosen by the user; a disabled menu command is dimmed and cannot be chosen. Unless you specify otherwise, 4th Dimension automatically enables each menu command you add to a custom menu¹.

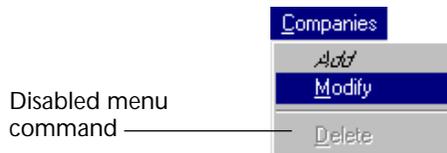
Note You can also enable or disable menu commands using methods.

► To enable or disable a menu command:

- 1 Select the menu command you want to enable or disable.
- 2 To enable the menu command, select the Enabled check box.

To disable the menu command, uncheck the Enabled check box.

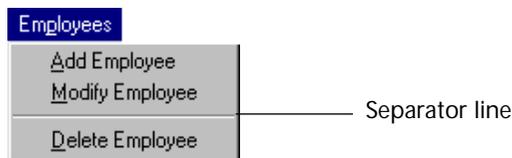
If the Enabled check box is checked, the menu command appears normally in the menu. If the Enabled check box is unchecked, the menu command appears dimmed, signifying that it cannot be chosen.



Adding Separator Lines

Groups of menu commands in a menu can be separated by a separator line. This convention is useful for grouping associated menu commands by function.

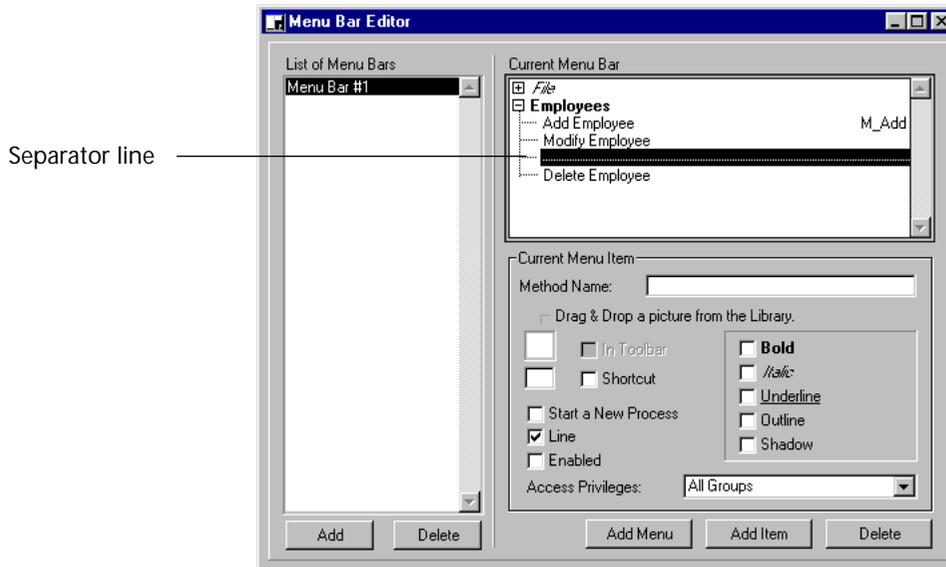
The figure below shows a menu with a separator line.



1. If you add a separator line, the Menu Bar editor automatically disables the line.

You add a separator line by creating a menu command. Instead of entering the menu command's text in the Current Menu Bar area, you simply select the Line property. Instead of text, the line appears in the Current Menu Bar area.

- ▶ To add a separator line:
 - 1 Select the menu you want to change in the Current Menu Bar list. The menu commands for that menu appear in the Items list.
 - 2 Choose Add Item from the Menu menu.
OR
Click Add Item.
4th Dimension creates a new menu command.
 - 3 If necessary, drag the new menu command to the location where you want to place the separator line (optional).
 - 4 Click the Line check box property in the Current Item properties area¹. The line appears in the Current Menu Bar area.



1. On Macintosh, if you use the dash "-" as the first character of a menu item, it will appear as a separator line. This is especially useful when using the APPEND MENU ITEM command.

Assigning Keyboard Equivalents

You can add keyboard equivalents to any menu command. If a menu command has one of these keyboard shortcuts, users will see a symbol for it next to the menu command. For example, “Ctrl+C” appears next to the Copy menu command on the Edit menu. This means you can copy a selection by holding down the Ctrl key and pressing C.

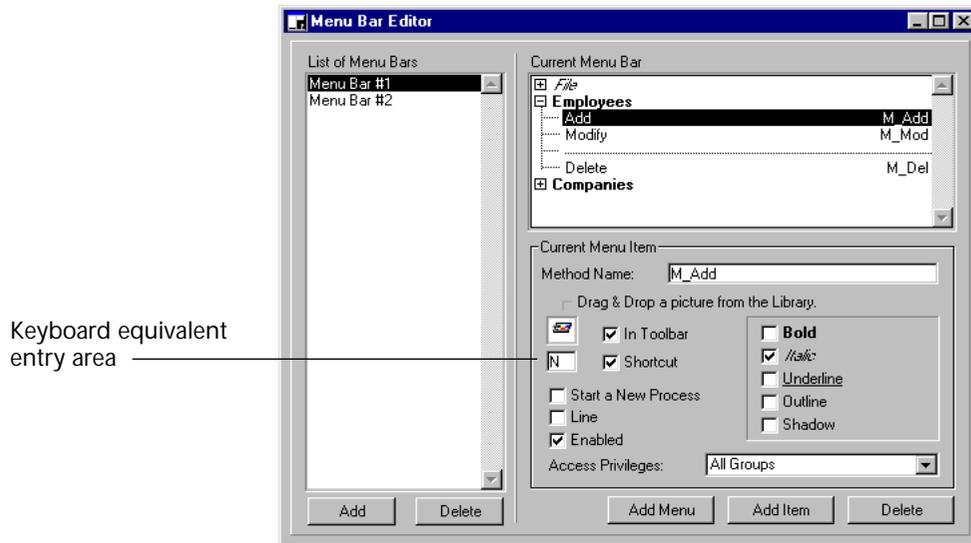
4th Dimension lets you assign keyboard equivalents to the menu commands in your custom menus. You can use any alphanumeric keys in combination with the Ctrl key (Command key on Macintosh) as a keyboard equivalent, except for the keys reserved by standard menu commands that appear in the Edit and File menus and the keys reserved for 4th Dimension menu commands. These reserved key combinations are listed in the following table.

Key ¹	Operation
Ctrl+C	Copy
Ctrl+Q	Quit
Ctrl+V	Paste
Ctrl+X	Cut
Ctrl+Z	Undo
Ctrl+. (period)	Stop action
Ctrl+W	Flushes records to disk in User or Custom Menus environments

1. On Macintosh, use Command instead of Ctrl.

- ▶ To assign a keyboard equivalent:
 - 1 In the Current Menu Bar area, select the menu command to which you want to assign a keyboard equivalent.
 - 2 Click the Shortcut check box.

- 3 Type the alphanumeric character that you want to associate with the menu command in the keyboard equivalent entry area.



Tip If the text of a menu item is followed by the slash and a letter, the Menu Bar editor will automatically use the letter as the keyboard equivalent. For example, if you enter “Add Employee/N” as the text of a menu item, the Menu Bar editor will automatically use Ctrl+N as the keyboard equivalent.

The user can choose the menu command by holding down the Ctrl key (on Windows) or Command key (on Macintosh) and pressing the assigned key.

Note On Windows, keyboard equivalents using the Alt key are automatically handled by the system, not by 4th Dimension.

- 4 Pull down the sample menu to see how the keyboard equivalent appears in the menu (optional).



Note An active object can also have a keyboard equivalent. If Ctrl key assignments conflict, the active object takes precedence. For

information on assigning keyboard equivalents to active objects, see the section [“Assigning a Keyboard Equivalent” on page 379](#).

Deleting Menus and Menu Commands

You can delete an instance of a menu at any time. A deleted menu no longer appears on the menu bar. You might not want a particular menu on a certain menu bar. You might delete a menu after you have placed its menu commands on other menus. Or, you might delete menus that you have placed on different menu bars.

For information about deleting connected menus, see the section [“Deleting Connected Menus” on page 545](#).

► To delete a menu:

- 1 Select the menu you want to delete in the Current Menu Bar list. When you select a menu, the names of menu commands assigned to the menu appear in the Items list.

Double-check to make sure this is the menu you want to delete.

- 2 Click Delete.

OR

Choose Delete from the Menu menu.

4th Dimension removes the menu from the menu bar. The deleted menu will no longer appear in the application’s menu bar.

You can delete a menu command at any time. You might delete a menu command that is out of date. You might delete a menu command after you have placed it on another menu. Or, you might delete a menu command that has been replaced by another menu command.

► To delete a menu command:

- 1 Select the menu command you want to delete.

- 2 Choose Delete from the Menu menu.

OR

Click Delete in the Current Menu Bar area.

4th Dimension removes the menu command from the Current Menu Bar list.

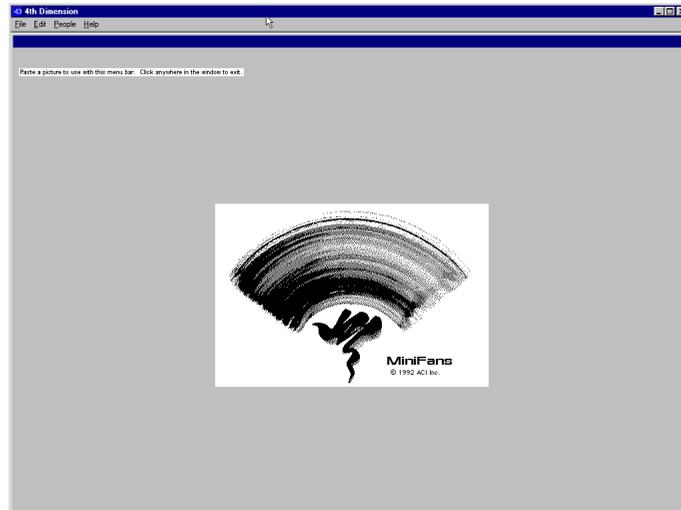
Previewing Menus and Adding a Splash Screen

As you develop your custom application, 4th Dimension lets you view the custom menus and menu bars as they will appear in the application.

You can also embellish each menu bar with a “splash screen,” a custom graphic displayed under the menu bar when it appears. A splash screen can include a company logo or other design elements.

Before you can add a splash screen, you need to create the graphic using a graphics application, by scanning, or by some other means.

- ▶ To preview the menu bar and add a splash screen:
 - 1 Choose **Show Custom Menus** from the **Menu** menu.
4th Dimension displays the menu bar as it will appear in the custom application and prompts you to copy a picture to accompany the menu bar.
 - 2 Open the menus listed on the menu bar to preview the contents of the menus.
 - 3 If you are going to add a splash screen, copy the graphic to the **Clipboard** and paste it into the **Splash Screen** area.
4th Dimension centers the graphic image beneath the menu bar. The following illustration shows an example of a custom splash screen.



- 4 When you are satisfied with your splash screen, click anywhere in the screen to exit the preview.

The splash screen will appear whenever this menu bar is used in the custom application when no form is displayed.

Menus and Custom Applications

Menu bars provide the major interface for custom applications. For each custom application, you must create at least one menu bar with at least one menu. See the *4th Dimension Language Reference* for more information about creating custom applications.

You can create menu bars for the Custom Menu environment regardless of whether you are creating a custom application or simply creating menus for use in the Custom Menu environment. By default, Menu Bar #1 is the menu bar displayed. You can change which menu bar is displayed using a method.

If you define a menu command without assigning it a method, choosing that menu command exits the Custom Menus environment. If you are using the application with 4D Runtime, leaving the Custom Menus environment takes you to the Program Manager (on Windows) or Finder (on Macintosh).

If you are using the full 4th Dimension application, a password access system can be set up to control where each user is placed after leaving the Custom Menus environment. Indeed, you can define an access group for the User Environment in the Database Properties dialog box. Users who do not belong to the access group set for the User Environment will not be able to access it from the Custom menus mode by either selecting a menu command, or by pressing the Alt + F4 keys (on Windows) or Option + F (on Mac OS). When users that do not have the adequate access privileges attempt to switch to the User Environment, 4th Dimension quits.

Note The Designer and the Administrator will always have access to the User Environment, even if they do not belong to the User Environment access group. Users that have access to the Design environment will always have access to the User Environment, even if they do not belong to the User Environment access group.

9

Managing Password Access

If more than one person uses a database, you may want to control access to the database or provide different capabilities and interfaces to different users. If you are designing applications for use in a multi-user environment or the World-Wide Web, it may be essential that you provide security for sensitive data. You can provide this security by assigning passwords to users and creating access groups that have different levels of access to information in the database or database operations.

This chapter provides information about 4th Dimension's Password Access editor. You use the Password Access editor to:

- Specify the users of a database,
- Provide users with passwords,
- Create groups of users with different levels of access to the database,
- Nest groups of users within other groups to create a hierarchy of users,
- Define a default user,
- Specify the group which owns the objects each user creates,
- Assign a startup method for each user,
- Specify a group owner.

After you create access groups, you can manage access to:

- The Design environment,
- The User environment,

- Table properties,
- Record operations,
- Forms,
- Methods,
- Menu commands,
- Plug-ins.

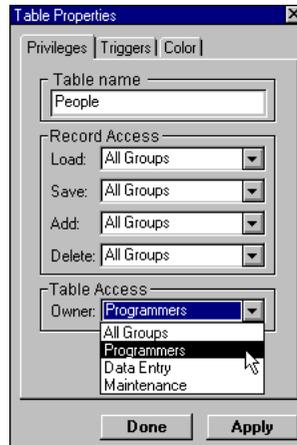
In addition to providing security for your databases, the access system also maintains a user history — the Password Access editor can tell you how many times a user has used the database and the date of the most recent use.

4D Server Object locking occurs when two or more users attempt to modify the password access system at the same time. Only one user can use the Password Access editor at a time.

Access System Overview

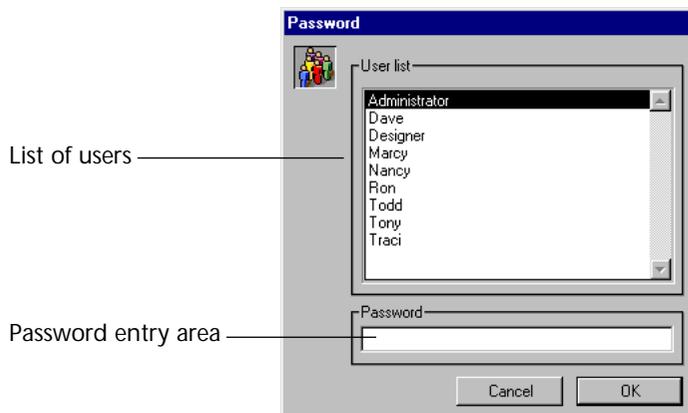
4th Dimension's password access system is based on users and groups. You create users and assign passwords, put users in groups, and assign each group access rights to appropriate parts of the database.

The figure below shows table owner privileges being assigned to a group. Groups can be assigned access privileges to operations on records in the table and to the table definition.



To open the database, a user either selects or types his or her user name and types his or her password¹. Then, depending on which groups the user belongs to and to which parts of the database the groups have been assigned, the user can operate the parts of the database that were specified by the access system.

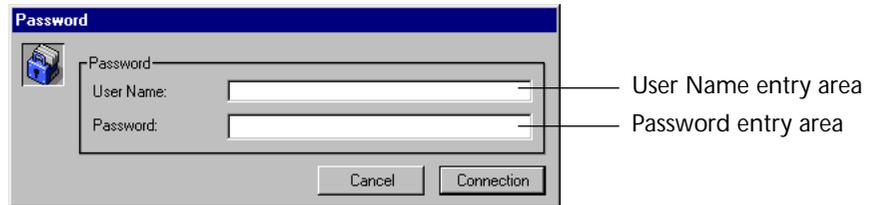
The Data Control page of the Database Properties dialog box allows you to choose which dialog box appears when the user opens the database. By default, the following Enter Password dialog box is displayed.



1. Unless the Default User option is used (for more information, refer to [“Defining a Default User”](#) on page 565).

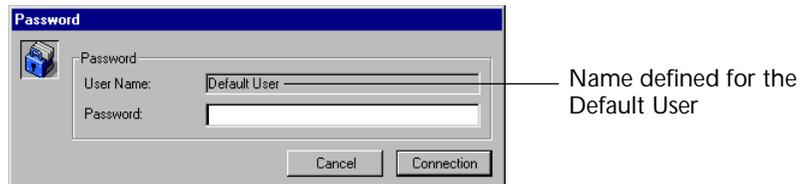
In this dialog box, the user selects his or her name from the list of users and types his or her password in the password entry box.

If you deselect **Display List of Users in Password Dialog Box** in the **Database Properties** dialog box, the **Enter Password** dialog box shown below will be displayed.



In this dialog box, the user must type both his or her name and password.

If you have defined a **Default User** in the **Database Properties** dialog and have assigned it a password, the following dialog box is displayed:

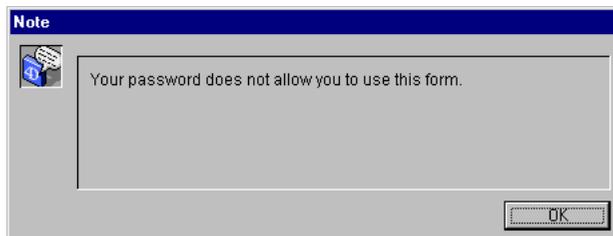


Users only have to enter the password. If no password is assigned to the **Default User**, the dialog is not displayed. Each user has the same privileges and restrictions as defined for the **Default User**.

4D Server After logging in to the database using any of the above dialog boxes, the user may choose to save the location of the server database (the pathname) and (optionally) the password used to log in to the database. The next time that the user double-clicks the 4D Client application icon, the database is automatically run and, if the user has saved his or her password, the user is automatically logged in to the database. For more information, refer to the *4D Server Reference* manual.

The user operates the database in a normal fashion. When the user attempts to use a form, menu command, method, or table that the group is not permitted to use, 4th Dimension displays an error message.

4th Dimension uses two types of dialog boxes to display messages. An example of each type is shown below.



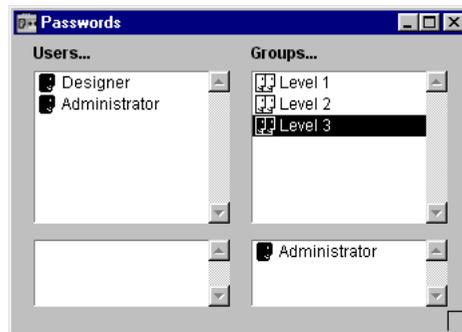
Note If an *ON ERR CALL* method is installed, the error message for methods and tables is not displayed. Refer to the *4th Dimension Language Reference* for more information.

An Access Hierarchy Scheme

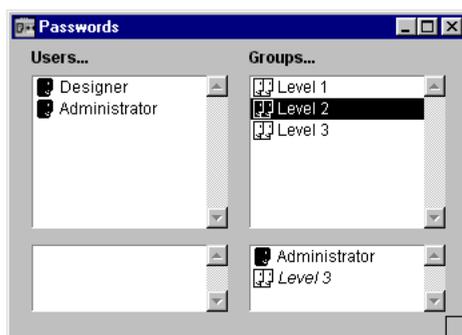
The best way to ensure the security of your database and provide users with different levels of access is to use an access hierarchy scheme. Users can be assigned to appropriate groups and groups can be nested to create a hierarchy of access rights. This section discusses some approaches to such a scheme.

In this example, a user is assigned to one of three groups depending on the user's responsibility. Users assigned to group Level 1 are responsible for data entry. Users assigned to group Level 2 are responsible for maintaining the data, including updating records and deleting outdated records. Users assigned to group Level 3 are responsible for analyzing the data, including performing searches and printing analytical reports.

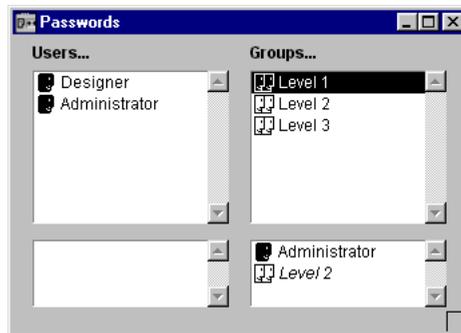
The groups are then nested so that privileges are correctly distributed when the groups are assigned. Level 3 contains only high-level users.



Level 2 contains data maintenance users as well as Level 3 users so that the users in Level 3 have the privileges of Level 2 as well.



Level 1 contains data entry users as well as Level 2 users so that users who belong to Level 2 and Level 3 enjoy the privileges of Level 1 as well.



You can decide which group to assign access privileges to based on responsibility. If you assign group Level 1 to an input form, for example, it means that everyone can use this input form. If you assign group Level 2 to the form, it is restricted to members of Level 2 and Level 3. If you assign group Level 3, only members of Level 3 can use the form.

Such a hierarchical system makes it easy to remember to which group to assign a new user. You only have to assign each user to one group and you use the hierarchy of groups to determine access.

As part of designing your access system, another consideration to keep in mind is the level at which a user should be restricted. You can think of each of the parts of a database to which access can be controlled — methods, forms, tables and table operations — as being part of a natural hierarchy. For example, if only a table is restricted, a user may try to display a form only to be restricted at the table level.

Your access scheme should restrict access at the highest appropriate level, usually at the form level.

The Designer and the Administrator

4th Dimension provides users with certain standard access privileges to the two environments and certain powers within each environment. Once a password access system has been initiated, these standard privileges take effect.

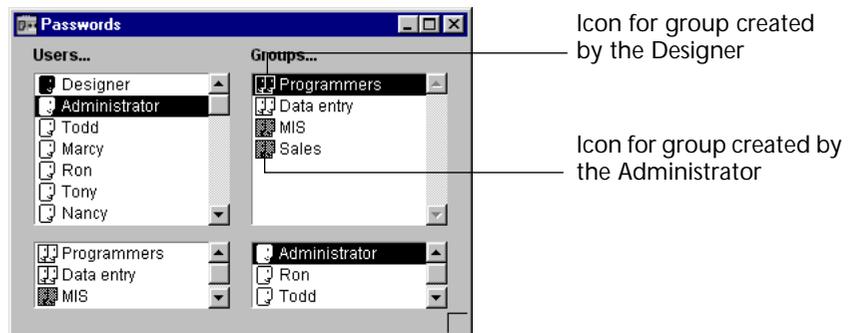
The most powerful user is named Designer. The Designer has control over the design of the database. The Designer can create users and

groups, assign access privileges to groups, and use the both the User and Design environments. No aspect of the database is closed to the Designer, except for users or groups created by other users.

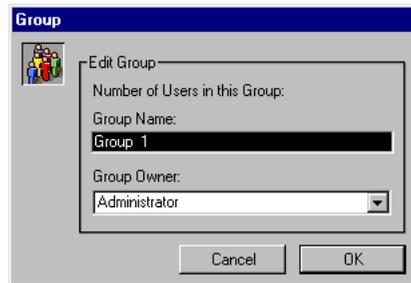
After the Designer, the next most powerful user is the Administrator, who is usually given the task of managing the password access system. When the Password Access editor is first opened, both the Designer and Administrator appear in the list of users. At this point, the Administrator is just a regular user with no special access privileges. To be able to use the password access system, the Administrator must be given the ability to enter the Password Access editor. For information about allowing the Administrator to enter the Password Access editor, see the section [“Administrator and Group Owner Access” on page 581](#). The Administrator is the only user with the ability to save and load groups. For information about saving and loading groups as the Administrator, see the section [“Saving and Loading Groups” on page 574](#).

The Administrator’s access to other parts of the database is limited by group membership — the Administrator must be part of one or more groups to have access privileges in the database. The Administrator is placed in every new group, but you can remove the Administrator’s name from any group.

You can distinguish between users and groups created by the Designer and Administrator by the color of their icons. The icons for users and groups created by the Designer are white while those created by the Administrator are grey. The icons for the Designer and Administrator themselves are both black.



The figure below shows a group being created by the Administrator. The default name of the group is Group 1.



The group owner can change the default name at any time.

The Designer and the Administrator can each create 16,000 groups and 16,000 users.

Group Owners

You can designate an owner for each group. Usually, the owner is the Administrator, but you can designate any group member as the owner.

The group owner can be given the ability to add and remove users from any group he or she owns. The users to be added must already exist. Group owners cannot create users or create or change user passwords. Group owners cannot add or remove other groups.

Like the Administrator, the group owner must be given the ability to enter the Password Access editor, normally reserved to the Designer of the database. For information about allowing a Group owner to enter the Password Access editor, see the section [“Administrator and Group Owner Access”](#) on page 581.

Giving Users Design Environment Access

Normally, the Designer of a database is the only user who has access to the Design environment. However, if you have a number of users who need to be able to modify the database design, you can place those users in a group and give the group access to the Design environment.

You give a group access to the Design environment by selecting the group from the Structure Access drop-down list in the Data Control

and Access page of the Database Properties dialog box. For more information, see the section [“Data Control” on page 85](#).

All other users are ordinary users. When a user opens the database, it opens in the Custom Menus environment or in the User environment (if the access is authorized). The access of a user is limited by group membership.

Giving Access to the User Environment

You can define a group’s access to the User environment. This option allows you to control and protect the access to the User environment when a user is in the Custom Menus environment. A user that is not part of the group that has access to the User environment will not be able to access the User environment from the Custom Menus environment, either by using a menu command or by using the standard shortcut (Option+f on MacOS, Alt+F4 or close box on Windows). If a user tries to access the User mode without having the appropriate privileges, 4th Dimension automatically quits.

The Designer and Administrator always have access to the User environment even if they are not explicitly part of the group that has access to the User environment.

A user that has access to the Design environment always has access to the User environment, even if the user is not explicitly part of the group that has access to the User environment.

► To give a group access to the User environment:

- 1 In the Database Properties window, click on the second tab, “Data Control and Access”.

The corresponding screen appears.

- 2 Choose a group in the User Mode Access drop-down list.

You must have previously defined the access groups in 4th Dimension’s Password editor.

- 3 Accept the dialog box.

For more information, refer to [“Data Control” on page 85](#).

Defining a Default User

You can define a Default User to use your database. When this option is active, the Password dialog box is not presented to the users that open or connect to the database. Moreover, if you have not associated a password with the Default User, the Password dialog box no longer appears, the database opens directly.

Once logged in as the Default User, each user has the access privileges and restrictions defined for the Default User.

This option simplifies access to the database while maintaining a complete data control system.

► To define a Default User:

- 1 In the Design environment, create a user (the name you choose is not important) in the Password editor.

You can associate a password with the user, but it is not mandatory. For more information refer to [“Assigning Users and Groups” on page 566](#).

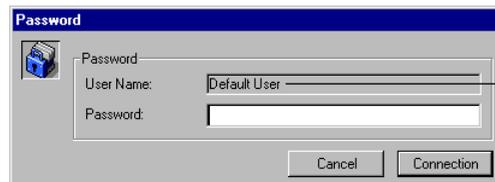
- 2 Using the Design environment editors, choose the access privileges and restrictions for this user.

For more information refer to [“Assigning a Group To Database Objects” on page 575](#).

- 3 In the Database Properties window, click on the Data Control and Access tab.
- 4 Choose your user in the Default User drop-down list (located at the bottom of the window).
- 5 Accept the dialog box.

The access to the database is now no longer customized.

- If you have associated a password with the Default User, a dialog box appears when the database is opened and the Default User’s password must be entered:



— Name defined for the Default User

- If you haven’t associated a password with the Default User, the dialog box doesn’t appear.

Redisplaying the Password Dialog Box

You can force 4th Dimension to display the standard Password dialog box to, for example, connect to the database as the Designer or Administrator.

- ▶ To redisplay the Password dialog box when the Default User mode is active:
 - 1 Open the database while holding down the Shift key.
A Password dialog box appears allowing you to enter a name and password.

Initiating the System

You initiate the 4th Dimension password access system by assigning a password to the Designer.

Until you give the Designer a password, 4th Dimension allows anyone to use any part of the database.

When a password is assigned to the Designer, all the access privileges you have assigned to tables, forms, menus, and methods take effect. In order to open the database, users must enter a password.

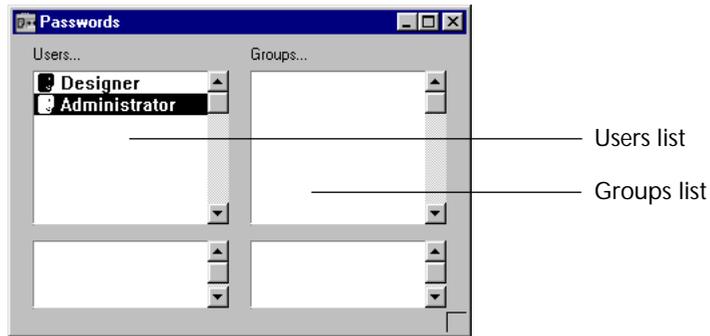
Important ***Do not forget the Designer's password! If you do, you will be unable to open the database in the Design environment.***

Assigning Users and Groups

You use the Password Access editor to create access groups and users and to assign passwords to users.

- To open the Password Access editor, choose Passwords from the Tools menu.

4th Dimension displays the Password Access editor window.



The Password Access editor window displays four scrollable areas. The upper Users list displays user names. The users designated Designer and Administrator appear at the top of the list. The lower Users list displays the names of any groups a selected user belongs to.

The upper Groups list displays the names of access groups. The lower Groups list displays the names of users who belong to a selected group.

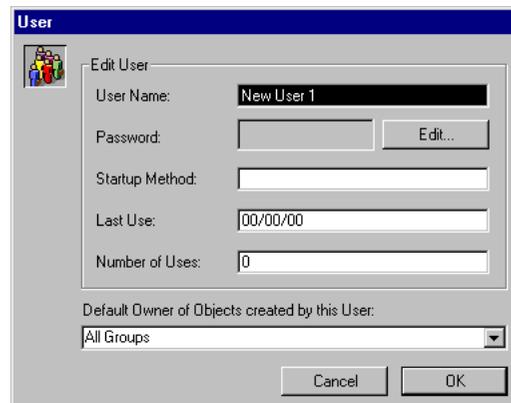
When the Password Access editor is active, the Passwords menu is available. You use the menu commands on this menu to add users and groups.

Adding Users

To add a user and assign user information:

- 1 Choose New User from the Passwords menu.

4th Dimension displays the User dialog box.



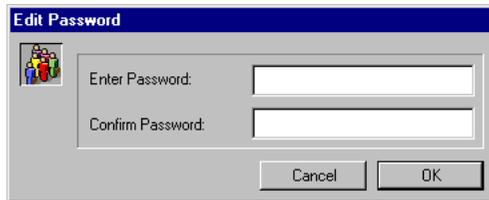
The dialog box provides areas for entering a user name, password, startup method, and the group which owns the objects that the user

creates. In addition, the Designer or Administrator can view information about an individual's use of a database or database application.

2 Type a new user name.

3 Click the Edit button.

The Edit Password dialog box is displayed:



4 Type the password in the Enter Password entry area and enter it again in the Confirm Password area.

You can use up to 15 alphanumeric characters for a password. The Password Access editor is case sensitive — the user must enter the password exactly as it is entered in the Edit Password dialog. For example, if you define a user's password as "HolyCow", the user must enter it with a capital H and capital C, or 4th Dimension will not accept the password.

After you specify a user password, that password is not available for view the next time you open the dialog box. Asterisks display in place of the password characters.

5 Click OK to validate the password.

If the two password entries are different, 4th Dimension plays a Beep and the dialog box is not closed.

6 Type a startup method to be executed when the user opens the database (optional).

7 Choose a group from the Default Owner of Objects Created by This User drop-down list.

This group owns any objects (tables, forms, methods, and so on) that the user creates. For instance, you might specify that the MIS group owns the objects created by each user in the MIS group. If a user from another group attempts to modify a form created by a member of the MIS group, a message appears stating that the user does not have privileges to edit the form.

8 Click the OK button to save the user information.

- ▶ To change user names and passwords:
 - 1 From the Password Access editor, select the user name in the list of users and double-click the user name.
OR
Select the user's name and choose Edit User from the Passwords menu.
4th Dimension displays the Edit User dialog box.
 - 2 Make the modifications you want in the dialog box.
 - 3 Click the OK button to save the changes you made to the user profile.

Users added by the Designer cannot be removed. To remove users added by the Administrator, see the section “Saving and Loading Groups” on page 574.

Creating Access Groups

When you create a group, you can designate a group owner from among the users.

- ▶ To create a group:
 - 1 Choose New Group from the Passwords menu.
4th Dimension displays the Edit Group dialog box.



- 2 Enter a group name in the Group Name box.
The group name can be up to 15 characters in length.
- 3 Click on the Group Owner box and select a user from the drop-down list of users.
The group owner can add users to and remove users from the group.
- 4 Click the OK button to add the group to the access system.
The name of the new group appears in the group list.

- ▶ To change access groups:
 - 1 From the Password Access editor, double-click the group name in the list of groups.
OR
Click on the group name and choose Edit Group from the Passwords menu.
4th Dimension displays the Edit Group dialog box.
 - 2 Enter a new group name or specify a new group owner in the dialog box.
 - 3 Click the OK button to save the changes you made to the group.
The following illustration shows the Password Access editor after a new executive group has been created.

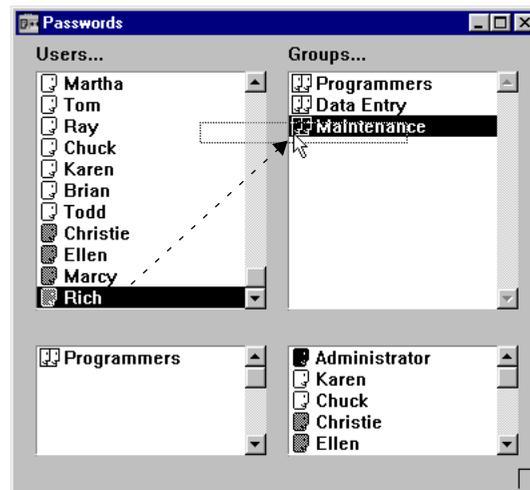


Assigning Users to Groups

You can assign users to any group and you can assign a user to several access groups. You are not required to assign a user to a group.

To assign a user to a group, drag the user name from the Users list over the name of the desired group in the Groups list and release the mouse button.

The figure below shows a user name being added to a group.



The user name appears in the lower Groups list of users assigned to the selected access group. The group name appears in the lower Users list of groups assigned to a selected user. The assigned user now has all the privileges that you provide that group.

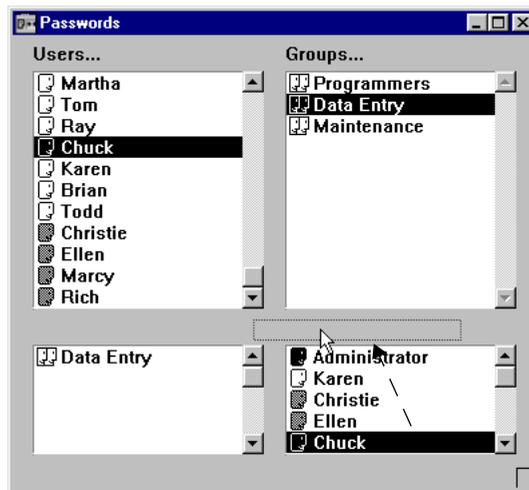
Click any group name to see a list of users who belong to that group.

Removing Users from Groups

To remove a user from an access group:

- 1 Click the group name in the upper Groups list to view the names of users assigned to the group.
- 2 Drag the user name from the lower Groups list to the Users list and release the mouse button.

The figure below shows a user being removed from a group.



The user no longer has the privileges that you provide that group.

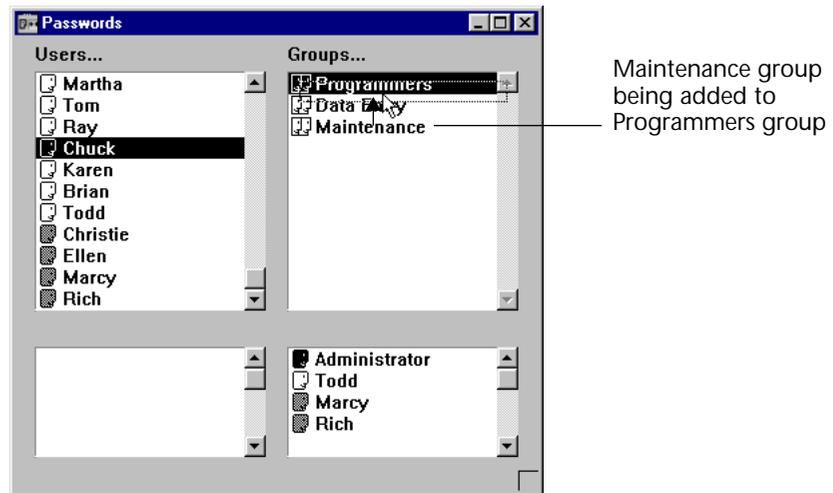
Nesting Groups

To create a user hierarchy, you nest groups, placing one group within another. The users of the nested group obtain the privileges of both groups. For example, if you nested the Executive group inside the Data Entry group, users assigned to the Executive group would automatically get the privileges of Data Entry in addition to the privileges accorded to Executive. However, users inside Data Entry are denied access to the privileges of the Executive group — they have only the privileges assigned to Data Entry.

For further explanation of how a hierarchical access system works, see the section [“An Access Hierarchy Scheme” on page 560](#).

- ▶ To assign a group to another group:
 - Drag the group name from the list of groups over the name of the group whose privileges you want it to assume and release the mouse button.

The figure below shows a group being added to another group.



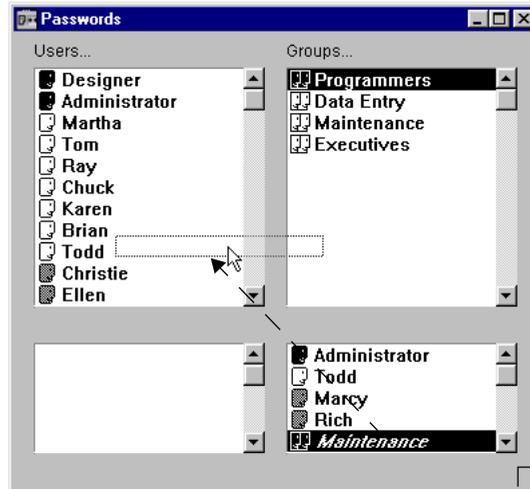
The nested group name appears in italics in the list of users assigned to the selected group. In addition to its assigned privileges, the nested group now has all the privileges of the group it has been placed in.

Removing Nested Groups

To remove a group from another group:

- 1 Select the group name in the upper Groups list to display the names of users and groups assigned to the group.
- 2 Select the group name in the lower Groups list, drag the name to the Users list, and release the mouse button.

The following illustration shows the Maintenance group being removed from the Programmers group.



The group name disappears from the list of users for the selected group.

Saving and Loading Groups

4th Dimension allows the Administrator to save and load any groups that he or she has created or modified. When groups are saved, everything about the current users and groups are saved. Later, when loaded again, the original users and groups are installed.

The ability to save groups means that the Administrator can save the access system of a database and transfer it to a modified version of the same database or to a new database. This is extremely useful for restoring the access system after the Designer has provided a new, updated version of the database. Because the groups can be reloaded, users of the database do not have to learn a new access system. All the user names, passwords, startup method names, groups, group owners, and group memberships are preserved.

Note The Designer cannot save or load groups.

► To save the current groups:

- 1 Enter the database as the Administrator.
- 2 Choose Save Groups from the Passwords menu.

4th Dimension displays a create-file dialog box so that you can name and save the group.

- ▶ To load groups:
 - 1 Enter the database as the Administrator.
 - 2 Choose Load Groups from the Passwords menu.
 - 4th Dimension displays an open-file dialog box so that you can open the Groups file.

Assigning a Group To Database Objects

After you define users and access groups, you can assign groups to the following objects:

- Table properties and operations,
- Forms,
- Methods,
- Menu commands,
- Plug-ins.

You may assign only one group to each object. For this reason, it is important to design the access groups so that more powerful users belong to all the groups below them in the access hierarchy.

For a discussion of how to organize users and access groups, see the section [“An Access Hierarchy Scheme” on page 560](#).

Assigning Access To Record Operations

You can assign different groups to each of the four record operations. You can thus specify which groups can load, save, add, or delete records from a table. These privileges can be very sensitive for some databases.

The following are the record operations that groups can be assigned to:

- **Load** This operation allows users to load and view records. It does not provide the right to modify records, create new records, or delete records.
- **Save** This operation allows users to save modified records. It does not provide the right to load records, add new records, or delete records.
- **Add** This operation allows users to create new records. It does not provide the right to load records, modify existing records, or delete them.
- **Delete** This operation allows users to delete records. It does not provide the right to load records, modify existing records, or add new records.

You must provide Load privileges for any group that has Save or Delete privileges.

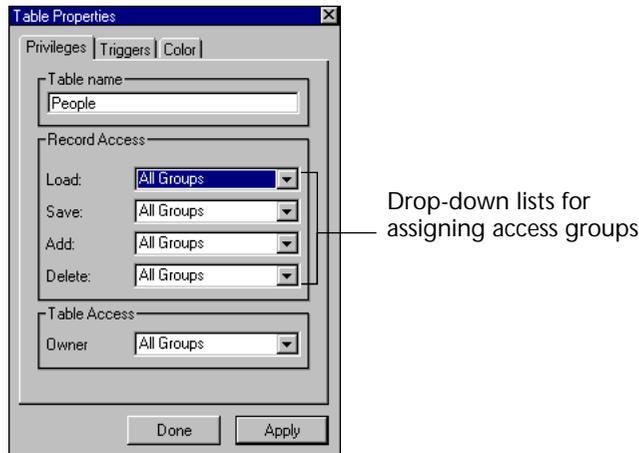
You can allow some users the right to add records without being able to load and view any other records. You can allow others the right to modify records, but not add any new ones.

For each table in a database, you can assign a group to be the owner of the table definition. Users in this group can modify specifications for the table, including the groups given access to record operations.

- ▶ To assign access privileges to record operations for a table and to the table's definition:
 - 1 In the Structure window, select the table image of the table whose access privileges you want to edit¹.
The selected table image is surrounded by a marquee.
 - 2 Double-click the table title.
 - OR
 - Choose Table Properties from the Structure menu.
 - OR
 - Press Ctrl+R (on Windows) or Command-R (on Macintosh).
 - OR
 - On Windows, click the table image using the right mouse button and choose Table Properties from the contextual menu.
 - OR
 - On Mac OS, click the table image while pressing the Ctrl key and choose Table Properties from the contextual menu.

1. You can also double-click the name of the table on the Tables page of the Explorer to view the table properties for the table.

4th Dimension displays the Privileges page of the Table Properties window.



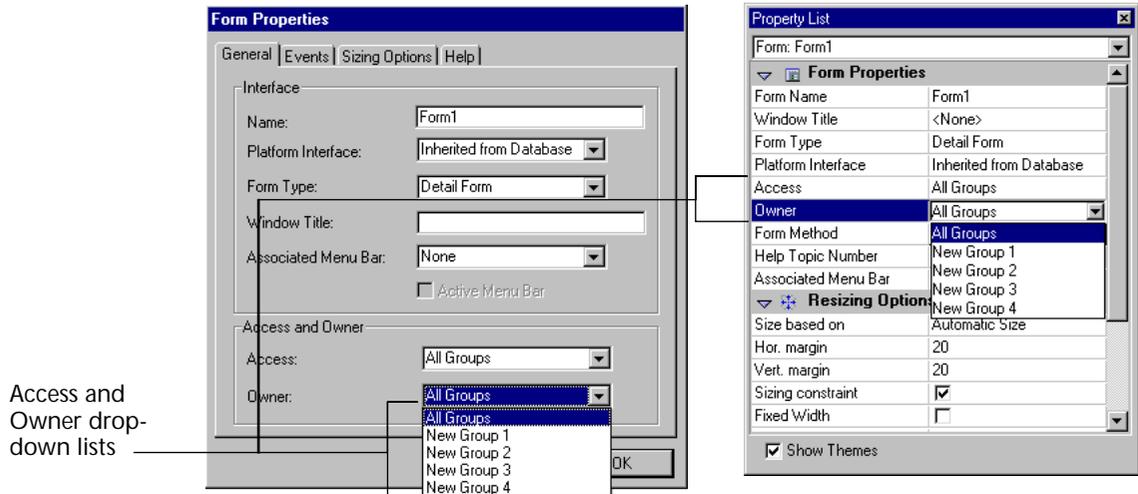
- 3 Choose a group for each database operation from the drop-down lists in the Record Access area.
- 4 Choose a group from the Owner drop-down list in the Table Access area.
Users in this group can modify the definition of the table in the Table Properties window.
- 5 When you have finished, click the Apply button.

Assigning a Group to a Form

When you assign a group access privileges to a form, only users belonging to that group can use that form for data entry. When you assign a group owner privileges to a form, only users belonging to that group can modify that form in the Design environment.

- To assign access and owner privileges:
 - 1 Open the form in the Form editor.
For directions on opening a form, see the section [“Opening a Form in the Form Editor” on page 250](#).
 - 2 Choose Form Properties from the Form menu.
OR
Display the form properties in the Property List.

For more information about this choice, refer to [“Displaying and Setting Form and Object Properties”](#) on page 244.



3 Choose a group from the Access drop-down list.

If you do not choose a group, all groups retain access privileges to the form (the default).

4 Choose a group from the Owner drop-down list.

If you do not choose a group, all groups retain owner privileges to the form (the default).

Assigning a Group to a Project Method

When you assign a group access to a project method, only users belonging to that group can use that method. When you assign a group ownership of a project method, only users belonging to that group can modify that method in the Design environment.

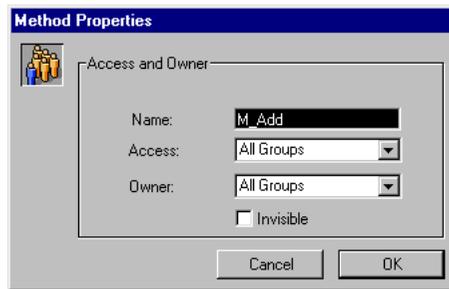
► To assign access and owner privileges:

1 Open the method in the Method editor.

For directions on opening a project method, see the section [“Opening an Existing Method”](#) on page 504.

2 Choose Method Properties from the Method menu.

The Method Properties dialog box appears.



- 3 Choose the group from the Access drop-down list.

If you do not choose a group, all groups retain access privileges to the method (the default).

- 4 Choose a group from the Owner drop-down list.

If you do not choose a group, all groups retain owner privileges to the method (the default).

- 5 Click the OK button.

Assigning a Group to a Menu Command

You can assign an access group to a menu command so that only users in that group can use the menu command in the User or Custom Menus environments.

- To assign an access group to a menu command:

- 1 Choose Menu Bar Editor from the Tools menu.

The Menu Bar editor appears.

- 2 Select a menu bar.

The Current Menu Bars area shows the menus belonging to this menu bar.

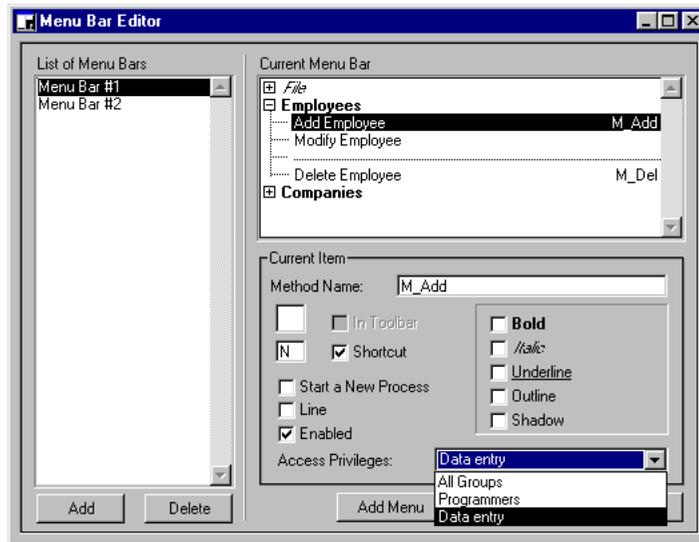
- 3 Expand a menu in the Menus list.

The menu commands and methods for this menu appear.

- 4 Select the menu command for which you want to specify an access group.

- 5 Select the group from the Access Privileges drop-down list.

The figure below shows the Data Entry group being given access privileges to the Add Employee menu command.



Assigning a Group to a Plug-in

You can assign a group privileges to any plug-ins installed in the database. This includes all the 4th Dimension plug-ins and any third-party plug-ins.

- ▶ To assign an access group to a plug-in:
 - 1 Choose Passwords from the Tools menu.
The Password Access editor appears.
 - 2 Choose Plug-Ins Access from the Passwords menu.
The Access to Plug-ins dialog box appears with a list of the plug-ins installed in the database.



- 3 Choose the plug-in for which you want to assign a group.
- 4 Choose the group from the Group Access drop-down list.
- 5 Click the OK button.

System Maintenance

Once a password access system is in place, occasional maintenance of the system is necessary. Users must be added, groups need new members, and passwords need to be changed. The Designer has access to the Design environment and can make any necessary modifications using the Password Access editor.

The Administrator and the Designer can also view the usage history of each user as necessary for maintenance.

Administrator and Group Owner Access

The Administrator does not necessarily have access to the Design environment. However, if the Designer creates a project method that contains the EDIT ACCESS command, the Administrator and group owners can have limited power to control users and groups.

The EDIT ACCESS command can be included in a method that is attached to a custom menu or can be executed by choosing **Execute Method** from the **Special** menu in the User environment. If the method is executed by a user who is not the Administrator or another group owner, it has no effect.

When the method is executed, the result depends on whether the user is the Administrator or a group owner.

If the Administrator executes the method that contains the EDIT ACCESS command, 4th Dimension displays the Password Access editor. The Administrator can use the Password Access editor to create users and groups; edit any users or groups he or she created, including changing user passwords; and add or remove users from any groups he or she created.

The Administrator cannot assign groups to forms, tables or table operations, menu commands, methods, or plug-ins. Only the Designer can assign these access groups.

If a group owner who is not the Administrator executes the method that contains the EDIT ACCESS command, 4th Dimension displays the

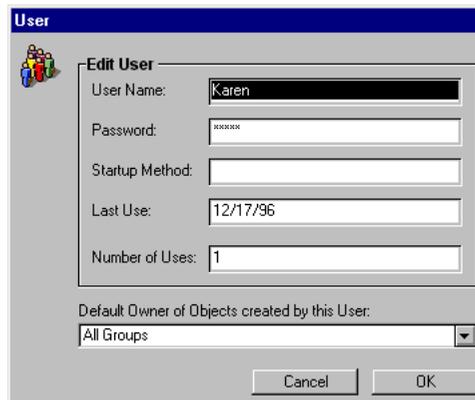
Password Access editor, but displays only the groups that the group owner owns. The group owner can add or remove users from the groups. The group owner cannot create users, edit user information, or add groups. The menu commands for adding and editing users and groups are dimmed.

Viewing Usage

The Edit User dialog box contains the date of the user's last use of the database and the total number of uses. The Administrator or Designer can view this information by opening the Edit User dialog box for any user.

- ▶ To open the Edit User dialog box:
 - 1 Choose Passwords from the Tools menu.
OR
Execute the method that contains the Edit Access command.
4th Dimension displays the Password Access editor.
 - 2 Select the user name you want to see from the Users list and choose Edit User from the Passwords menu.

The Edit User dialog box displays the date of the user's last use of the database as well as the number of times the user has opened the database.



10

Creating Lists

This chapter tells you how to create and use lists. A list is a set of possible values. You can use a list to do the following:

- Provide the user choices from which to select an entry for a field or enterable object,
- Restrict the valid entries to those in the list,
- Exclude the entries in the list from being entered.

4th Dimension lets you associate a small icon with each item in a list or hierarchical list. Where appropriate, the small icon is displayed to the left of the item. For example, you can display the small icons in hierarchical lists.

When a list is used as a choice list for a field or enterable object, the user can simply select from the list instead of typing the entry. For example, you may want to create a choice list for entering job titles in a personnel database.

You can also use lists to provide restrictions on data entry. One list may provide the required values for a field, excluding all others. Another list may provide the excluded values for a field, preventing any value in the list from being entered.

You can also create hierarchical lists. A hierarchical list associates a sub-list with each element of the list.

Your lists can offer up to 8,000 choices in a single database and each choice can contain up to 255 characters.

For information about adding a choice list to a field as a field attribute, see the section [“Choices and Help” on page 147](#). For information about using lists with data entry controls, see the section [“Using Choice Lists” on page 333](#).

Lists are often used in methods. For example, a list is a convenient place to store the elements of an array. An array stores a list of values in memory. You can use lists to store the elements of pop-up menus, hierarchical lists, combo boxes, tab controls, and other multi-valued interface objects. You transfer the contents of the list to the interface object using a method or by assigning the list to the object in the Object Properties window.

You create 4th Dimension lists with the List editor. You use the List editor to do the following:

- Create lists and hierarchical lists,
- Add items to lists,
- Associate small icons with list items,
- Delete lists,
- Delete items from a list,
- Sort items in a list,
- Make a choice list user-modifiable
- Make hierarchical list items editable or a tab control active,
- Specify the spacing between hierarchical list elements,
- Insert hierarchical lists or hierarchical pop-up menus in forms.

4D Server Object locking occurs when two or more users attempt to modify the same list at the same time. If a user is modifying a list in the Design environment, the list is locked. Other users cannot modify the list, the list name, or any of the items in the list, until the first user frees the list by closing it.

Designing Lists for Data Entry

One use of lists is to provide the user with a list of values from which to choose during data entry. The following are some considerations about lists that stem from this purpose:

- You can make a list available for every form or for selected forms.

- You can restrict the possible entries to those in the list or you can allow the user to type additional entries.
- You can allow the user to modify the list or you can prevent the user from modifying the list.

You can attach a list to a field as one of the field properties. Attaching a list to a field causes the list to appear whenever that field is selected during data entry or whenever an output form is used in the Enter in List mode. The user can select an entry from the list. If the list is sorted, the list automatically scrolls as the user types characters at the keyboard. For instance, if the user types “N,” the list scrolls to the first entry starting with “N.” The user can stop typing when the desired choice appears and select it from the list.

If you attach a list to a field using its Field Properties in the Structure editor, the list will also appear when the field is selected in the Query editor. For more information, see the section [“Creating New Fields” on page 131](#).

You can also attach the list to the field as a data entry control in a form. The list will appear only when the field is selected in this input form, not in all forms or the Query editor. The figure below shows a choice list being displayed.



If the number of elements that the list contains is limited, you may not need to use lists. For instance, in the case of a list that has only two values such as Male or Female, you may consider using a Boolean field.

This would allow you to use interface elements such radio buttons or picture radio buttons. Even for cases that include four choices, you can use check boxes.

Hierarchical Lists

4th Dimension allows you to create hierarchical lists. Selecting an element from the parent list displays a sublist.

There is no standard way to use hierarchical lists; it depends on what you use them for. For example, a list of states would take many entries, thus slowing down the selection process. There are several approaches when you want to use hierarchical lists for this type of problems.

Besides their use for entry purposes, you can use hierarchical lists to populate the following objects:

- Hierarchical lists,
- Tab controls,
- Hierarchical pop-up menus,
- Scrollable areas,
- Combo boxes.

You can often divide a list of values in two value categories. In the example mentioned above, you could determine which states are selected the most often. If 80% of the values selected refer to a handful of states, you can place those states in a list and place the remainder in a sublist.

This allows the user to be able to directly select an entry in 80% of the cases, making the entry faster in most cases.

A different approach consists in organizing states in different regions. The first list that is displayed allows the user to select the region and the sublists allow him to select the state. In this case, each selection will require the user to select a region, followed by a state. It still is faster than selecting a state in a list of fifty states.

Required and Excluded Values

Some data entry tasks are not crucial. If you enter a value that does not appear in a list, it may be perfectly acceptable. However, you may have an application in which an entry must be one of the values in a list. Any different value would cause a serious consequence such as delaying bill payment.

4th Dimension allows you to make a list required as part of the data entry controls on a form. This type of data entry control prevents a user from entering any value other than the ones in the list. For example, your company may have a specific group of job titles that are allowable in a personnel database.

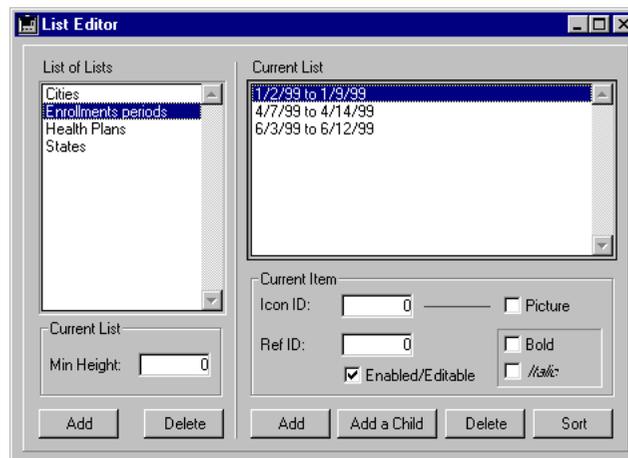
Another data entry control makes it possible to exclude the values in a list. The user then cannot type in a value that should not appear in the field. For example, your company may be prevented from doing business in certain countries. Placing them in an excluded list prevents them from being entered.

Non-Sequential Ranges of Values

One of the most useful data entry controls is the Maximum and Minimum setting for a number, date, or time field. Setting a maximum and minimum value prevents a user from entering a value outside this range.

Suppose you have three acceptable ranges for the field. You can use a list to create such non-sequential ranges. If you then make this list required for a field, values outside the three ranges are not accepted.

The figure below shows a list of ranges.



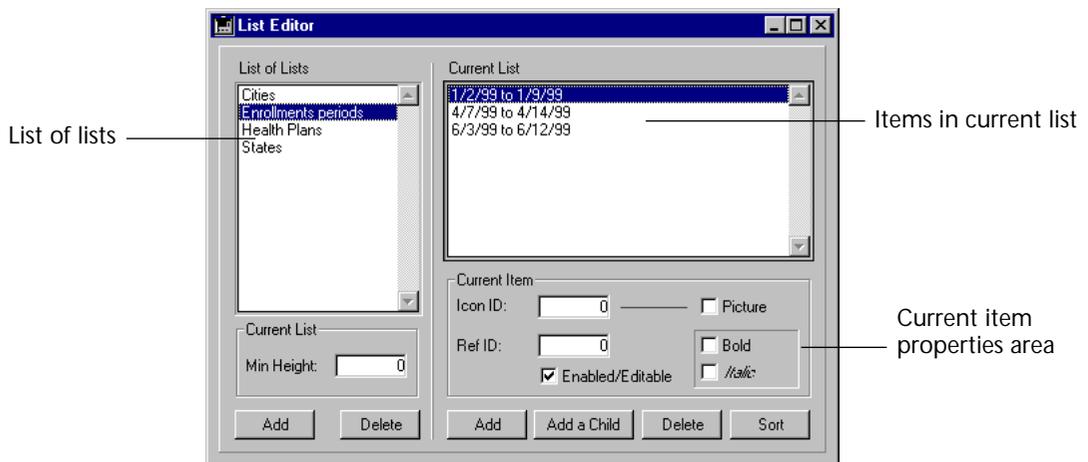
Conversely, you could create a list that specifies the ranges that are not valid. If you then make this list an Excluded list for a field, any entry within the ranges is not accepted.

When defining ranges, you should enter two dots “..” between the minimum and maximum values, otherwise 4th Dimension will not consider them as ranges but as a string value.

Creating Lists

You create lists with the List editor. You can modify any list at any time by returning to the List editor and making changes.

- ▶ To create a list:
 - 1 Choose List Editor from the Tools menu.
- The List editor appears.



The List editor displays the names of existing lists on the left. On the right side of the editor, a list of items in the selected list is displayed. The menu bar provides two menus: Lists and Items.

- 2 Choose New from the Lists menu.
- OR
- Click Add.

4th Dimension creates an empty frame in the Lists of Lists and displays an insertion point in the frame.

3 Type the list name.

You have created a new empty list. Now, you will create the items that will appear in the list as described next.

Adding Items to Lists

When you are adding items to a list, you can append new items to the end of the list or insert them anywhere in the existing list.

► To append items to a list:

1 Select the name of the list to which you want to add items.

If the list already contains items, they appear in the Current List area. If you want to insert a value in an existing list, select the value located after the value you want to insert. The new value will be created after that value.

2 Choose New from the Items menu.

OR

Click Add in the Current List area.

4th Dimension creates an empty frame in the Current List area and displays an insertion point in the frame.

3 Type the item name.

4 To add additional items, repeat the above steps.

5 When you have finished, double-click the Control-menu box to save the current lists and close the editor window (click the Close box on Macintosh).

Once the values entered, you can move the list items by drag and drop.

► To insert an item in a list:

1 Select the item that is to appear above the item you are adding.

4th Dimension inserts the new item above whichever item is currently selected. This allows you to insert a new first item.

2 Click Add in the Current list area.

4th Dimension creates an empty frame in the Items column below the previously selected item.

3 Type the item name.

4 Double-click the Control-menu box to save the current lists and close the editor window (click the Close box on Macintosh).

Creating a Hierarchical List

You can add a list to any list element. The hierarchy is not limited to two levels. You can use a hierarchical list to specify the contents of a hierarchical list interface object.

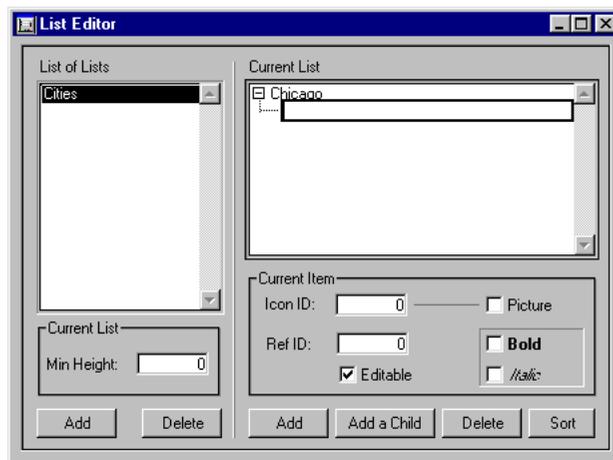
► To create a hierarchical list:

- 1 Select the list item to which the sublist will be attached.
- 2 Click Add a Child.

OR

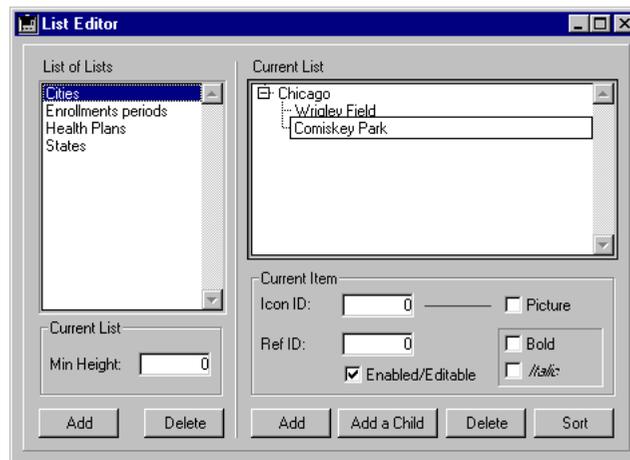
Choose New Child from the Items menu.

4th Dimension expands the selected list element and creates an entry area for the first item on the sublist.



- 3 Enter the item.
- 4 To add another item to the sublist, click Add or choose New from the Items menu.

Repeat these steps as necessary.



If desired, you can attach sublists to sublist items to continue the hierarchy.

► To attach a sublist to a sublist item:

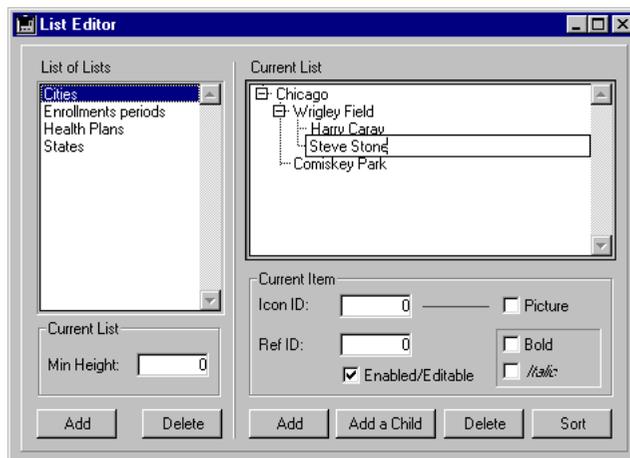
- 1 Select the sublist item.
- 2 Click Add a Child.

OR

Choose New Child from the Items menu.

- 3 Enter the item normally and repeat the process of entering items or attaching sublists to items, as desired.

The following illustration shows a three-level hierarchy.



Deleting Items and Lists

You can delete items at any level of the hierarchy.

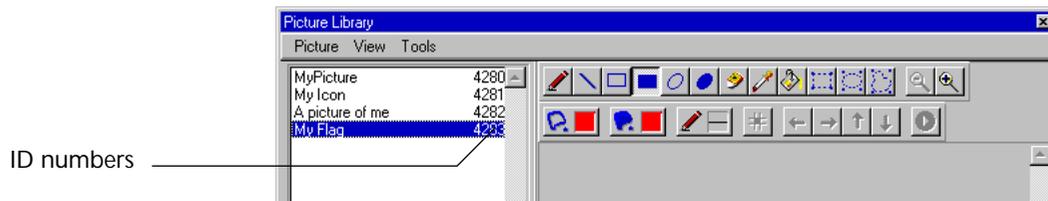
- ▶ To delete an item:
 - 1 With the List editor window open, select the list that contains the item you want to delete.
 - 2 Select the item you want to delete from the Current List area.
If necessary, expand the list.
 - 3 Click Delete in the Current List area.
OR
Choose Delete from the Items menu.
 - 4th Dimension deletes the item from the list and removes the space it occupied in the list.
- ▶ To delete a list:
 - 1 With the List editor window open, select the list you want to remove.
 - 2 Click Delete in the List of Lists area.
OR
Choose Delete from the Lists menu.
 - 4th Dimension deletes the list and removes the space it occupied in the list.

Adding a Small Icon to an Item

You can associate a small icon with an element in a list. The small icons that you use are stored in the Picture library.

Picture library items

When you add a picture to the Picture library, it automatically assigns it an ID number. You can use this number to associate the picture with an item in a list.



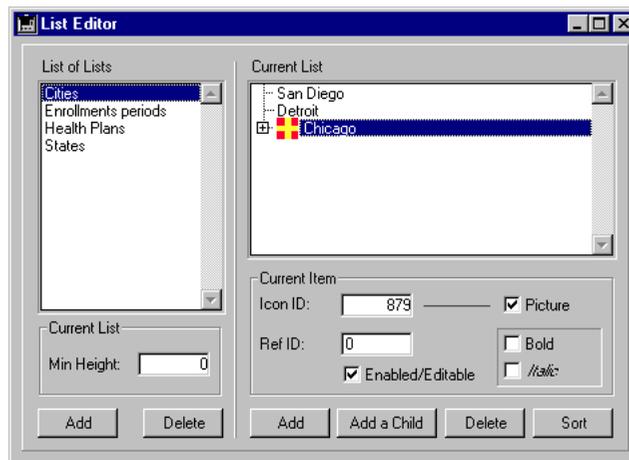
- ▶ To associate a small icon with an item:
 - 1 Select the item in the desired list.
If necessary, first select the list from the list of lists and expand a list element.

The Current Item properties area changes to show the properties of the selected item.

- 2 Check the Picture check box and enter the ID number of the picture in the Icon ID area.
- 3 Press Tab to save the ID number.

The Current List area changes to show the small icon associated with the selected list item.

The following illustration shows the picture with the ID of 9585 (from the Picture library shown above) associated with a list item.



Note Depending on the size of the icons you use, you may need to modify the list's height. For more information, refer to [“Setting the Minimum Height of a List”](#) on page 597.

Adding a Reference ID to an Item

The Current Item Properties area contains an entry area for the item's Reference ID. The Reference ID is designed as a unique identifier for the item. It is of use only when you manage lists using methods.

When you need to use the language to determine which item in a list a user selects (e.g., which item in a hierarchical menu is selected), you can identify the user's choice using the Reference ID of the item. For more information, see the section “Hierarchical Lists” in the *4th Dimension Language Reference*.

Specifying Ranges in a List

4th Dimension allows you to enter ranges of numbers, dates, and times in a list. You can use these ranges as data entry validation ranges by making the list required or excluded in a form.

- ▶ To create ranges in a list:
 - 1 Create the list you want to use for ranges.
 - 2 For each item, enter the minimum value of the range, two periods (.), and the maximum value.
For example:
100..150
defines the range between 100 and 150, inclusive.
 - 3 Continue specifying ranges as separate items until you have defined as many as you need.

Sorting a List

4th Dimension maintains the list of items in the order in which you enter them. You can sort the list alphabetically so that entries are more easily accessible to database users. Since a sorted list automatically scrolls to match characters typed at the keyboard, sorting usually makes data entry easier.

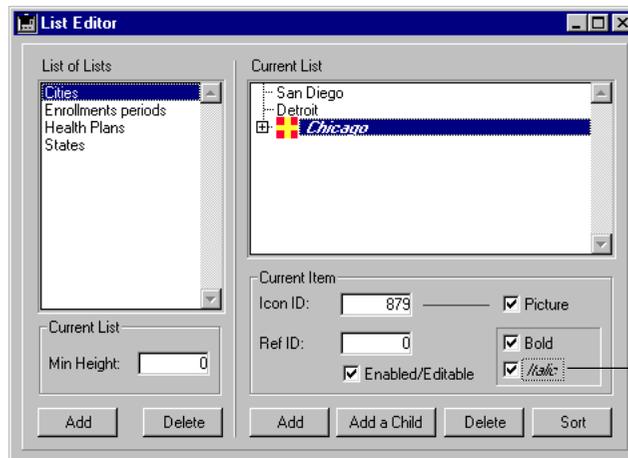
- ▶ To sort a list:
 - 1 In the List editor, select the list that contains the choices you want to sort.
 - 2 Click Sort.
OR
Choose Sort from the Items menu.

4th Dimension sorts the list in ascending order.

To sort the list in descending order, hold down the Shift key when you choose the Sort menu command.

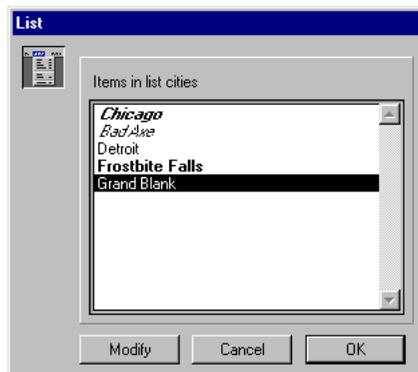
Specifying Font Attributes

When a list is used as a choice list, you can display list items in bold, italic, or bold-italic. The following illustration shows the bold and italic attributes applied to Chicago.



Bold and italic check boxes

To apply font attributes, select the desired item in the Current List area and click the Bold or Italic check boxes. Check both check boxes if you want to use bold-italic. When the list is used as a choice list, the selected style attributes will be used, as shown in the following illustration.



Making a Choice List Modifiable

4th Dimension allows you to specify whether a list of items can be changed by the user when the list is displayed as a choice list. By default, a list is modifiable. 4th Dimension places a check mark in the Lists menu to show that the list is user-modifiable.

If you allow a list to be user modifiable, the user has access to a special List editor in the User environment. The special List editor is for the assigned list only. The user cannot add lists, delete lists, or change any other list. If a list is modifiable, the user can make any change to that list's items.

► To make the list user-modifiable:

1 Select the list that you want to make modifiable.

2 Pull down the Lists menu.

If User Modifiable has a check mark, the list is user modifiable. Release the mouse button without making a menu choice.

If User Modifiable does not have a check mark, go to the next step.

3 Choose User Modifiable from the Lists menu.

4th Dimension adds a check mark to the menu command. The list can now be modified in the User environment.

To prevent the user from modifying a list, select the list and choose User Modifiable to remove the check mark.

Making a Hierarchical List Modifiable or Enabling Tab Control Items

A list can be used to define elements for a hierarchical list (form object) or for tab controls.

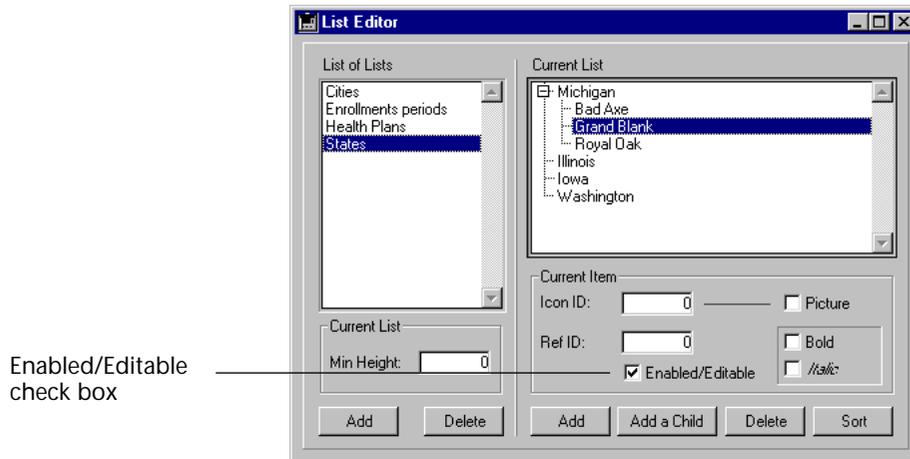
A list can also be used to specify the items in a hierarchical list. When the list is used in this manner, you can control whether each item in the list can be edited by the user. If a list item is editable, the user can hold down the Ctrl key (Command key on Mac OS) and click on the item to get an insertion point. An editable item in a hierarchical list is shown below.



When a list is associated with tab controls, you can enable or disable each tab control that corresponds to an elements if the list. A disabled tab control will be displayed in gray in the form. In the following example, the tab control "New Jersey" is disabled.



If you want to allow users to modify an item in a hierarchical list, click the **Enabled/Editable** check box in the List editor while the desired list item is highlighted, as shown here:



The **Enabled/Editable** option is selected by default. In other words each tab control is enabled or each list value can be edited.

If you do not want users to be able to modify values in the list, deselect the **Enabled/Editable** option for each of these values.

Note When the user edits a list item, the list itself is not changed. The change affects only the text that is displayed until the user accepts or cancels the record. Use a method to manage any user modifications to list items.

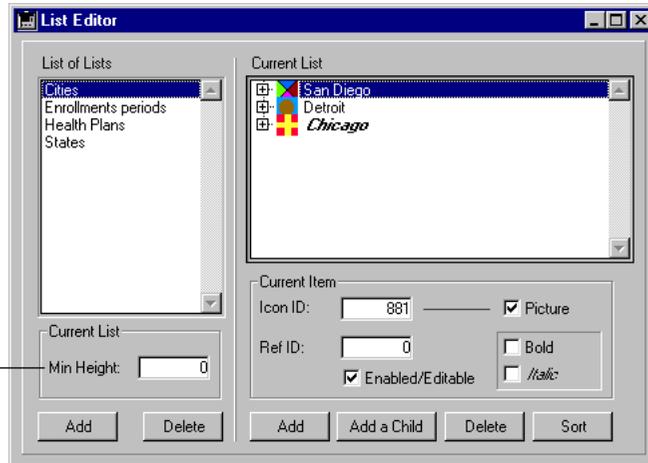
Setting the Minimum Height of a List

When 4th Dimension displays a list, it uses the font size of the hierarchical list object to determine the vertical spacing between adjacent list items. If you use a list to specify the values of a hierarchical list, you can specify a larger vertical spacing. The main reason you would want to do this is to provide additional space for icons that are attached to list items. Or, you can use this feature simply to spread out the list items.

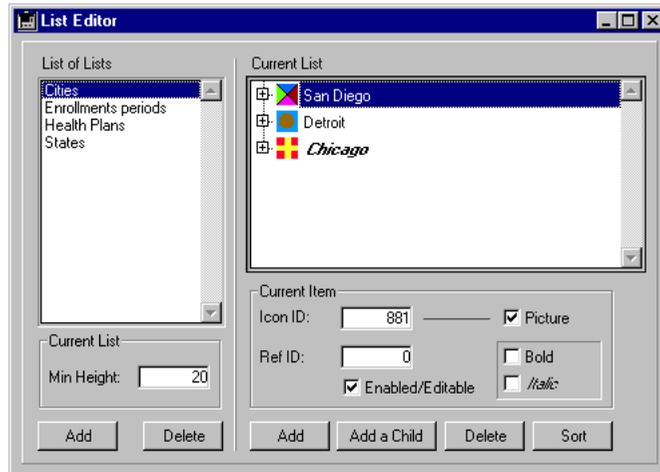
- ▶ To specify a minimum height:
 - Enter a value in points in the Min. Height entry area.
The effects of this value are displayed immediately in the Current List area. The following illustrations show the effect of increasing the minimum height.

Minimum Height: 9

Minimum height entry area



Minimum Height: 20



Dragging a List into a Form

4th Dimension allows you to use shortcuts to create hierarchical lists or drop-down hierarchical menus that are associated with lists. These operations can be performed by dragging lists from the List editor.

- To insert a hierarchical list in a form, drag the list from the List editor to the form.
- To insert a hierarchical menu in a form, drag the list from the list editor to the form while pressing the Shift key.

Note You can use these shortcuts from the Explorer window.

11

Using the Picture Library

Use the Picture library to store graphics that you can use as design elements in forms, as toolbar or list icons, picture menu items, or picture buttons. With the Picture library, you can use a graphic in several places in your database but you need to store it in only one place. When you update an image in the Picture library, all references to the image are updated automatically. This feature can reduce the size of your Structure files and make changes to the database easier to manage.

The picture library also includes a 2D Paint editor that allows you to create or retouch pictures. It is an ideal environment to design buttons and icons.

In addition to the features mentioned above, the Picture library also includes integrated functions that can create or edit tables of thumbnails in order to create picture buttons or picture menus.

Managing and Viewing Pictures

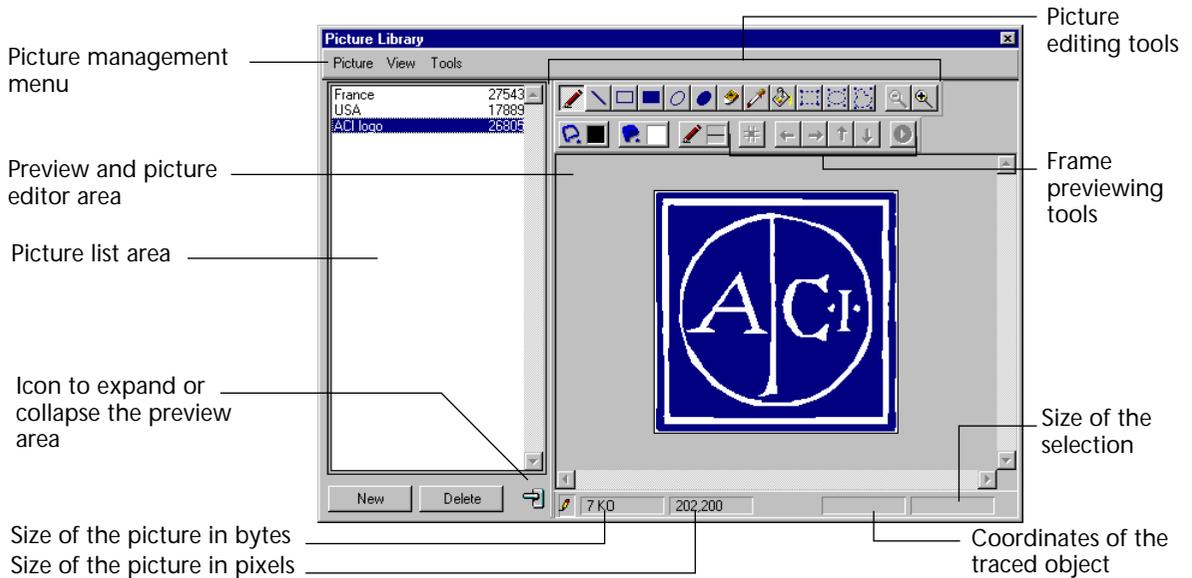
The Picture Library appears as a floating palette, which allows you to work simultaneously with different windows (for example, different forms) while keeping the library in the front.

► To display the Picture library:

- 1 Choose Picture Library from the Tools menu.

The window contains menus that allow you to access picture management functions as well as a toolbar to edit the contents of the pictures.

You can also hide or show the picture's viewing/editing area by clicking on the switch at the bottom of the picture list.

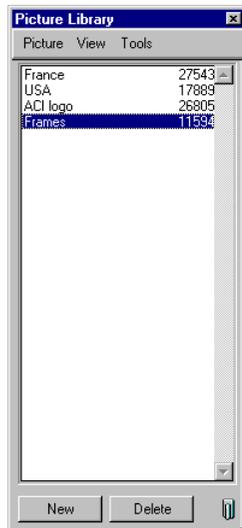


The different tools are described in the following sections.

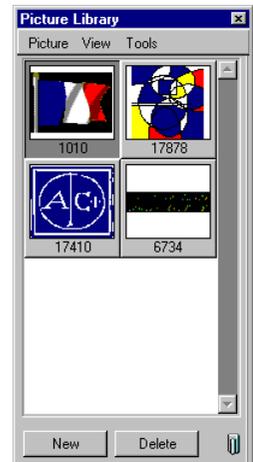
You can define how you want to view the images in the Picture List Area using the View menu:



View as a List
(default mode)



View as a Picture



Note The List display mode must be selected if you want to use the drag and drop features of the Picture Library.

Adding Pictures

You can add pictures to the Picture Library in three ways:

- By importing a picture file.
- By pasting a picture from the Clipboard.
- By creating a new picture in which you draw its contents.

Importing a Picture File

To import a new picture into the Picture library:



- 1 Choose **Import Picture** from the Picture menu of the Picture Library. A standard open file dialog appears. The only picture formats that can be imported are Mac OS formats (*.PIC or *.PCT).
- 2 In the Open File dialog box, select the file and click **Open**. The Picture Properties dialog is displayed. The dimensions that appear in that dialog are the picture's dimensions. For more information on picture properties, refer to [“Setting Picture Properties” on page 604](#).
- 3 If necessary, modify the ID number or other properties and click **OK** to create the picture in the Picture Library.

Note You can modify the ID number of the picture only when it is being created in the Picture library.

Pasting a Picture from the Clipboard

To paste a picture that is stored in the Clipboard:

- 1 Copy a picture to the Clipboard.
- 2 Select the Picture Library list area.
- 3 Press the **Ctrl+v** keys (on Windows) or **Command+v** (on Mac OS).
OR
Choose **Paste** from the Edit menu. The Picture Properties dialog is displayed. The dimensions that appear in that dialog are the picture's dimensions.

For more information on picture properties, refer to [“Setting Picture Properties” on page 604](#).

- 4 If necessary, modify the ID number or other properties and click OK to create the picture in the Picture Library.

Note You can modify the ID number of the picture only when it is being created in the Picture Library.

Creating a New Picture

The Picture library contains an integrated graphics editor. You can create a picture and draw its contents in that editor.

- ▶ To create a new picture:
 - 1 Click the Create button located under the Picture list area.

OR

Choose New Picture from the Picture menu.



The Picture Properties dialog box is displayed. The dimensions that appear in that dialog are the default values for the picture's dimensions.

For more information on picture properties, refer to [“Setting Picture Properties” on page 604](#).

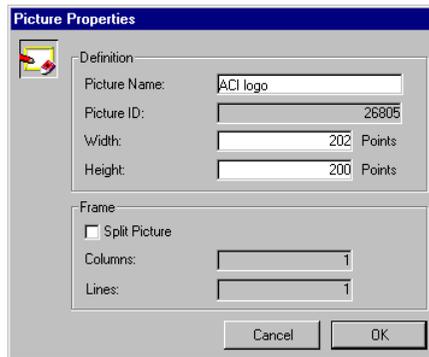
- 2 If necessary, modify the ID number or other properties and click OK to create the picture in the Picture Library.

Note You can modify the ID number of the picture only when it is being created in the Picture Library.

Setting Picture Properties

The Picture Properties dialog box allows you to define or display the the picture's name, ID number, dimensions, as well as its frame attributes.

No matter how you add a picture, the following dialog box appears.



This dialog displays the following properties:

- **Picture Name:** allows you to modify the default name of the picture. You can assign the same name to several pictures, only the ID number has to be different.
- **Picture ID:** allows you to assign an ID number to the picture. This number is the reference number for the picture. It is how you refer to the picture when creating picture buttons, picture pop-up menus, custom toolbars or when you handle pictures programmatically.

WARNING: You can define this number when you create the picture, but you cannot modify it afterwards.

- **Width and Height:** allow you to define the size of the picture. These values are precalculated when you import a picture (from a file or from the Clipboard). When you split the picture (see below), the values correspond to the size of each frame.
- **Frame area:** allows you to create thumbnails from a single image for use in creating an array of buttons or picture menus. For more information, please refer to the [paragraph “Creating Thumbnails”](#), [page 607](#).

To create the picture, accept the dialog box. Once the picture’s properties have been defined, you can modify them at any time by selecting the picture and by choosing **Picture Properties...** from the library’s **Picture** menu.

Creating and Modifying Pictures

The Picture Library has a tool bar (like a Paint program) that allows you to draw and modify pictures.

To create or modify a picture, you just have to click on the view/edit area.

The Picture Library's editor works only in bitmap mode. Of course, you can import a picture — from a file or from the Clipboard — of a different type (for example, a vector graphic) and its characteristics will be retained when you use it in your database. On the other hand, if you modify this picture in the editor, it will be transformed into a bitmap. In this case, its original characteristics will be lost when the modified picture is saved. When you modify an imported picture in the library, an alert dialog box appears to tell you that the picture will be converted into a bitmap (canceling at this point leaves the picture untouched).

The Picture Library's graphic tools are the following:

Icons	Tools	Function	Option keys
	Pen	Draws point by point	Alt (Option): allows you to take the color above which the cursor is located
	Line	Draws a line	Shift: the angles are in multiples of 45°
	Hollow Rectangle + Full Rectangle	Draws a hollow rectangle Draws a full rectangle	Shift: draws squares Ctrl (Command): the rectangle is drawn from its center
	Hollow Oval + Full Oval	Draws a hollow oval Draws a full oval	Shift: draws circles Ctrl (Command): the oval is drawn from its center
	Eraser	Erases by using the current background color	
	Color Picker	Modifies the line color by using a color from the picture	
	Flood Fill	Fills an enclosed area with the current background color	
	Selection tools	Create a selection	
	Zoom	Zoom the picture	

	Outline Color + Fill Color	Line color and background color menus	These options can be modified from the editor's contextual menu (Click with the right mouse button on Windows or Control+click on MacOS)
	Line width	Line width menu	

You can also use the standard editing commands (such as copy, paste, etc.) in the Edit menu, 4th Dimension's tool bar, or the standard keyboard shortcuts.

Note If you use the standard Paste command while the Picture editor area is selected, the Clipboard's contents is inserted into the currently edited picture. If you use the standard Paste command while the Picture list is selected, a new picture is added to the list.

Saving and Cancelling Modifications

Any modification made to a picture is automatically saved in the library as soon as you click outside of the Picture editor area (which means as soon as the editor area "loses" focus).

WARNING: *Once the picture has been saved, you cannot revert any of the modifications that you have made to it.*

While modifying a picture, you can cancel the last modification made to it by choosing Cancel from the Edit menu (or the toolbar) in 4th Dimension, as well as the standard shortcut Ctrl+z (Windows) or Command+z (MacOS).

You can also cancel all the modifications made to a picture by choosing Revert to saved from the library's Picture menu.

Creating Thumbnails

The Picture library contains a set of functions that allow you to create and modify a row by column array of pictures for use in picture buttons or picture menus. The array may be either one- or two-dimensional.

In the Picture library, such an array is called "Frames." Elsewhere in 4th Dimension, an array of pictures may be called "thumbnails" or a "table" of pictures. The idea consists of splitting a picture using rows and columns; each cell is therefore considered a "frame" or "thumbnail." 4th Dimension takes care of displaying the correct frame in the picture button or picture menu according to the specified

parameters (for more information refer to [“Picture Buttons” on page 390](#) and [“Picture Pop-up Menus” on page 403](#)).

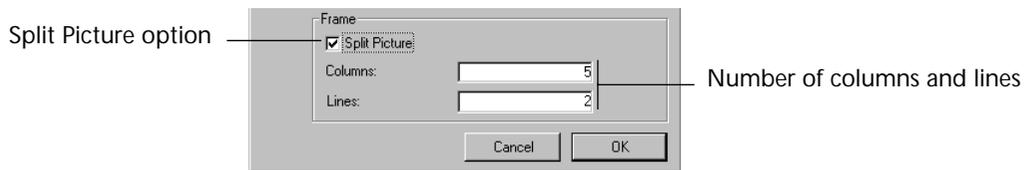
You can define a sequence of frames when creating a picture or even afterwards. You can also insert or delete lines, columns, or frames in an already defined sequence of frames.

Creating a Sequence of Frames

You can create a sequence of frames from a picture already placed in the Picture Library or when you add a picture to the Picture Library.

In both cases, you define how the picture is to be divided into separate frames in the Picture Properties dialog box. If you are creating a picture, the dialog box appears automatically. Otherwise, select the picture and choose **Picture Properties...** from the **Picture** menu.

The **Frame** area allows you to define the number of lines and columns of your frame sequence. To create frames, you must first check the **Split Picture** option:



Size of the Frames

The size of the frames is automatically calculated by 4th Dimension. When you define a sequence of frames, the **“Width”** and **“Height”** areas are modified and the size of each frame is displayed.

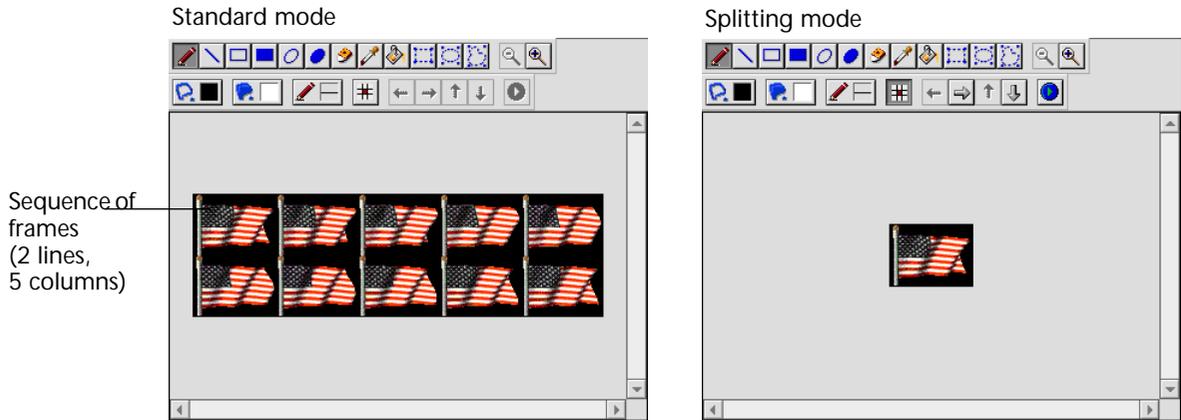
If you want to modify the size of the frame later, you just have to enter new values into the **Width** and **Height** areas without worrying about the global size of the picture. Each resulting frame will be centered automatically (without the picture being distorted) in the new size if it is bigger. If the new size is smaller, each frame will be truncated.

Viewing the Frames

You can preview each frame to check the sequence’s appearance by using specific tools in the **Picture Library’s** tool bar:



When you activate the split mode, 4D splits the picture and displays the first frame (located in the upper left corner). You can then navigate through the frames using the arrow buttons.



You can also display all the frames automatically. This function is particularly useful if you want to create picture buttons that display in a continuous sequence. To do so, click on the Test animation button.

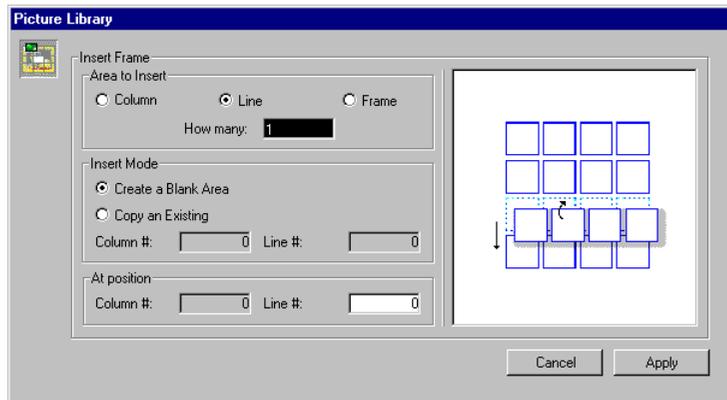
The frames appear in a continuous manner. To stop the test, click the Test animation button.

Inserting and Deleting Frames

The Picture library allows you to insert and delete frames in a previously created sequence of frames. You can insert frames as well as columns and lines.

- ▶ To insert frames:
 - 1 Select the frame sequence to modify.
 - 2 Choose Insert... from the Picture Library's Tools menu.

The following dialog box appears:



- 3 Indicate whether you want to insert a column, line, or frame and indicate how many in the upper portion of the dialog box.
The preview area on the right side of the dialog box shows you an example of the selected operation. Note that inserting an element moves the others (no element is replaced).
- 4 In the Insert Mode area, indicate if you want the inserted element to be blank or if it should contain the contents of an existing element.
In the latter case, you must designate the element to recopy.

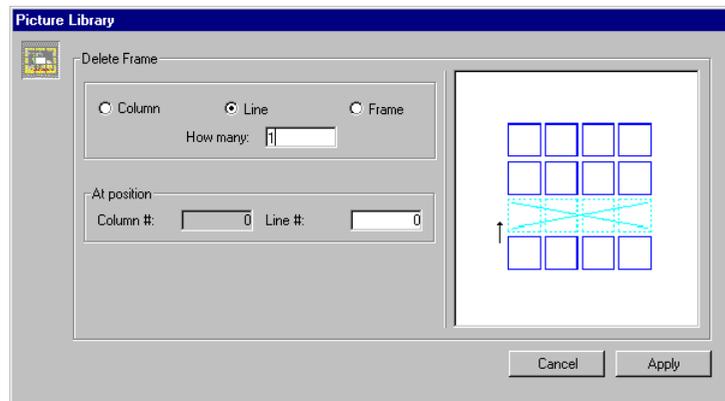
Note The number of the first column and first line is 0.

- 5 Indicate the location in which you want the new element to be inserted and click Apply.
The element is then inserted in the picture.

► To delete frames:

1 Choose Delete... from the Picture Library's Tools menu.

The following dialog box appears:



2 Define the element (column, line, or frame) that you want to delete and indicate how many.

The preview area on the right of the dialog box shows you an example of the selected operation.

3 Indicate the position where you want the element to be deleted and accept the dialog box.

Note The number of the first column and first line is 0.

Shortcuts for Inserting Picture Buttons and Pop-up Menus

When a picture is defined as a sequence of frames, you can use shortcuts to insert it as a picture button or a picture pop-up menu in your forms (for more information refer to [“Picture Buttons” on page 390](#) and [“Picture Pop-up Menus” on page 403](#)). If you want to be able to use these shortcuts, you need to display the pictures as a list.

- To create a picture button, drag the picture from the library and drop it in the form.
- To create a picture pop-up menu, drag the picture from the library and drop it in the form while holding down the **Shift** key.

Note Dragging and dropping a picture that is not defined as a sequence of frames will be inserted as a standard picture.

The new Property List allows you to distinguish between two types of pictures: Picture Library (dynamically updated when the source picture in the library is modified, as in previous versions of 4D.) and Static Pic-

ture (not associated to the library's source picture). Refer to the section [“Placing a Picture from the Picture Library” on page 309](#).

12

Managing Processes

You can increase the functionality of a database by taking advantage of 4th Dimension's multi-tasking capabilities. In a multi-tasking system, database operations can run in separate *processes* — separate 4th Dimension tasks — that operate independently and concurrently.

Multiple processes are executed at the same time, allowing 4th Dimension to carry out several operations simultaneously. For example, one process might print a selection of records while another process allows a user to enter new records. In custom applications, multiple processes are used to manage a multi-window interface. Each window is managed by a different process. The process has its own current selection of records — even if two processes access data in the same table. The number of processes that can be executed at the same time is limited only by available memory.

This chapter explains how to do the following:

- Start a new process,
- View process information including the name, number, status, and total amount of time used by each process,
- Control process execution,
- Control whether a process is visible to users,
- Specify the frontmost process,
- Debug a process.

Processes

A process can be thought of as a 4th Dimension task that is carrying out some action — searching for some records, printing a selection, allowing a user to enter data, and so on. The exact operation that the process performs depends on the method with which it is connected.

As a separate 4th Dimension task, each process has the following elements for data management:

- A current selection for each table,
- A current record for each table
- Process variables,
- Locked records.

In addition, each process can have the following elements:

- Current input and output forms for each table,
- A menu bar,
- One or more windows,
- One active window (the frontmost window).

For a complete list of process elements and information about creating windows and menu bars, refer to the *4th Dimension Language Reference*.

You may notice that all of these elements are found in the 4th Dimension User environment. All processes have the same basic properties as the User environment and allow you to perform the same operations that you can in the User environment.

However, instead of having to perform the operations directly in the User environment, processes allow you to use methods to specify the actions you want 4th Dimension to take.

Anything that can be done with the 4th Dimension language — any user interface that can be created or operation that can be performed — can be done from a process.

Having multiple processes open at once gives you the ability to perform different actions or work with different aspects of a database simultaneously.

Opening multiple processes allows the user to do the following:

- **Work with more than one active window** You can have several active windows open at the same time. For example, you can enter data in one window and receive messages from colleagues in another window.
- **Work with more than one current selection at a time** Because each process has its own current selection, each process can have a different current selection from the same table. For example, an Employees database might contain a list of employees and their occupations. In a single-process database, you could display all engineers at once, or all accountants at once, but you could not make both selections the current selection for the same table. In a multi-process database, you can display the records of all engineers in one process and the records of all accountants in another process.
- **Work with more than one current record at a time** Each process can have a different current record. For example, you might want to compare one employee to another using an input form. In a single-process database, you can display only one employee's record. In a multi-process database, you can display each employee's record in a different process.
- **Start a lengthy operation in a separate process** You can perform a time-consuming operation such as printing a large selection of records in a separate process while you continue to work on your database.
- **Work with more than one input or output form at a time** You can view data in several different forms at once. For example, you could display a selection of records in a standard output form in one process and in a special report form in another process.

Processes Created and Managed by 4th Dimension

4th Dimension automatically creates and manages the following processes which control the operation of 4th Dimension:

- **User/Custom Menus** This process controls the User and Custom Menus environments.
- **Cache Manager** This process controls flushing or caching data to disk.
- **Design** This process controls the Design environment.
- **\$Stat Window** This process manages the Runtime Explorer window. It is created as soon as the Runtime Explorer is opened.

The User/Custom Menus and Cache Manager environments are created automatically when you open a database. The Design process is created automatically when you enter the Design environment.

In addition, 4th Dimension creates and manages the following processes:

- Indexing process,
- On Serial Port Manager process,
- On Event Manager process,
- Web server process.

Unlike user-created processes, the processes created by 4th Dimension are always running and cannot be frozen or aborted. For more information about viewing processes in the Process List editor, see the section [“Using The Process List” on page 621](#).

For more information about the processes created by 4th Dimension, refer to the *4th Dimension Language Reference*.

Time-Sliced Execution

Since in reality more than one process cannot execute at the same time, when you open multiple processes, 4th Dimension slices the total processing time so that execution is divided between all open processes. Execution alternates between processes so rapidly that the processes appear to be executing simultaneously. For instance, processing time is split between the Design process, the User/Custom Menus process, and the Cache Manager so that some milliseconds might be devoted to the User/Custom Menus process, the next to the Design process, the next back to the User/Custom Menus process, and so on.

Starting a New Process

4th Dimension allows you to start your own processes from the User or Custom Menus environments.

Each process that you start can perform a different task or present a different aspect of the data contained in your database.

The functionality of the process can be enhanced by a user interface created using any of the editors in the Design environment or using the 4th Dimension language. For instance, you can display an input form in a process to allow a user to enter records.

► Two start a new process:

1 Create a method.

The specific operation that each process performs depends on the commands and functions in the method. For more information about the 4th Dimension commands, refer to the *4th Dimension Language Reference*.

2 Specify that 4th Dimension start a new process when the method executes.

You can tell 4th Dimension to start a new process in the following three ways:

- Using the New process command in another method,
- Using the Menu Bar editor,
- Using the Execute Method dialog box.

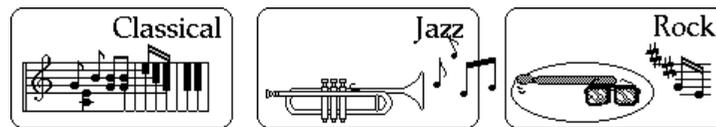
The alternative you select depends on what you are trying to do. Each alternative is described in detail in the following sections.

Starting a New Process Using New Process

There are many circumstances in which you might want to start a new process with the New process command. For instance, you might want to start a new process when a user clicks a button. You can do this by starting the process in the button's object method.

When you start a new process using the New process command, you should place the New process command in the method from which you want to start the new process. When that method executes, 4th Dimension starts a new process for the method specified in the parameters of the command and continues executing the original method.

The figure below shows a set of buttons in a compact disc database.



The object method for the Jazz button uses the New process command to start a new process for the method, *JazzSearch*:

```
myProcess := New process ("JazzSearch";16000;"JazzRecords")
```

The new process, *myProcess*, carries out the actions specified in the method *JazzSearch*. The method *JazzSearch* creates a selection of Jazz

compact discs and displays them in a window. It contains the following statements:

```
QUERY ([Compact Discs];[Compact Discs]Category="Jazz")
If (Records in selection ([Compact Discs])>0)
  RefNo:=Open window (50;50;300;250;8)
  MODIFY SELECTION ([Compact Discs])
  CLOSE WINDOW
End if
```

When a user clicks the Jazz button, 4th Dimension opens a window and displays all the compact discs that contain Jazz music.

The window is running in the new process, myProcess.

If a different button is clicked, another process is started. If the user clicks the Jazz and Rock buttons, 4th Dimension starts two processes and displays the selection for each in its own process window.

The user can double-click records in either window to modify them in the current input form for that process.

For more information about the New process command, refer to the *4th Dimension Language Reference*.

Starting a New Process Using the Menu Editor

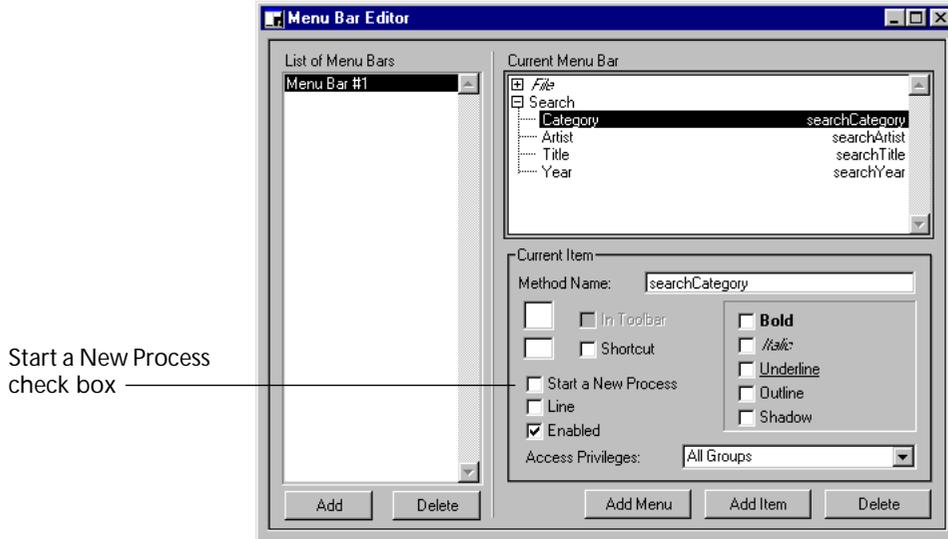
You will often find that you want a new process to start whenever a user chooses a menu command. For instance, you might have a menu command that prints a group of records. Since printing records can be time-consuming, you specify that 4th Dimension start a new process whenever that menu command is selected.

► To start a new process using the Menu Bar editor:

- 1 Choose a menu bar in the Menu Bar editor.
- 2 Select a menu from the Menus list.

The menu commands and methods for that menu are displayed in the Menu editor.

The following illustration shows a menu from the compact disc database.



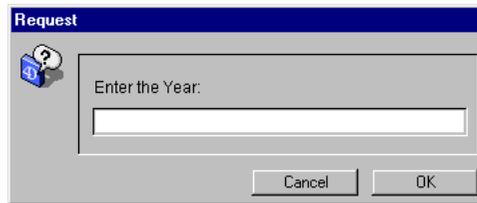
- 3 Select the menu command for which you want to start a new process. When you select a menu command, the Current Item Properties area changes to show the properties of the selected menu item.
- 4 Click the Start a New Process check box.

This specifies that a new process should be started whenever that menu command is selected.

The Search menu allows users to perform various searches. In the above example, the method for the Year menu command allows the user to enter the year. It then searches for all compact discs produced in that year and displays the selection in a window.

```
vYear := Request ("Enter the Year:")
If (OK=1)
  QUERY ([Compact Discs];[Compact Discs]Year=vYear)
  If (Records in selection ([Compact Discs])>0)
    RefNo:=Open window (50;50;300;250)
    MODIFY SELECTION ([Compact Discs])
  End if
End if
```

When the user chooses **Year** from the **Search** menu in the **Custom Menus** environment, a dialog box appears requesting that the user enter a year.

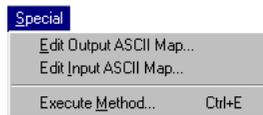


When the user clicks the **OK** button, 4th Dimension displays the selection of compact discs produced the year the user entered. If the user chooses **Category** from the **Search** menu, the user can perform another search based on the type of music the compact disc contains.

Starting a New Process Using Execute Method

You can choose to start a new process when you execute a method using the **Execute Method** dialog box in the **User** environment. One of the advantages to using this method to start a new process is that you can decide on a case-by-case basis whether you want to start a new process for a method.

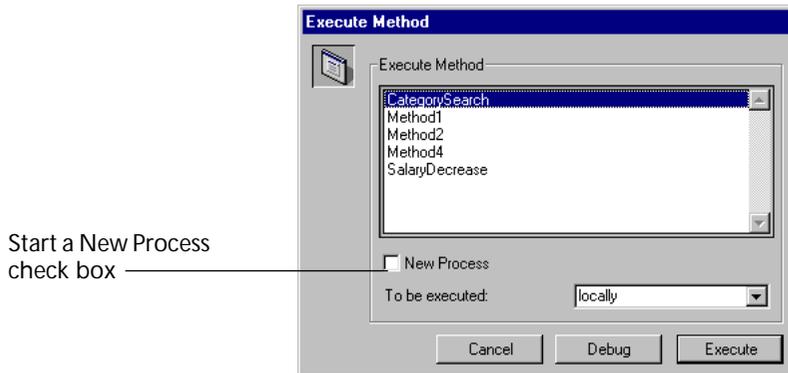
- ▶ To start a new process in the **Execute Method** dialog box:
 - 1 Choose **Execute Method** from the **Special** menu in the **User** environment.



The **Execute Method** dialog box appears.

- 2 Select the method for which you want to start a new process.

The figure below shows the *CategorySearch* method being selected.



- 3 Click the New Process check box.
Clicking the New Process check box tells 4th Dimension to start a new process when the method executes.
- 4 Click the Execute button to execute the method.
If you want to monitor or debug the execution of the method, click Debug.
4th Dimension executes the method within a new process.

Using The Process List

4th Dimension lists processes in the Process page of the Runtime Explorer editor window as soon as they are started. Each process is given a process ID, which is the same as the process number (discussed in the next section). You use this process ID to identify a specific process in commands and functions.

► To view the Process List:

- 1 Display the Runtime Explorer Window.

For more information, refer to [“Runtime Explorer” on page 75](#).

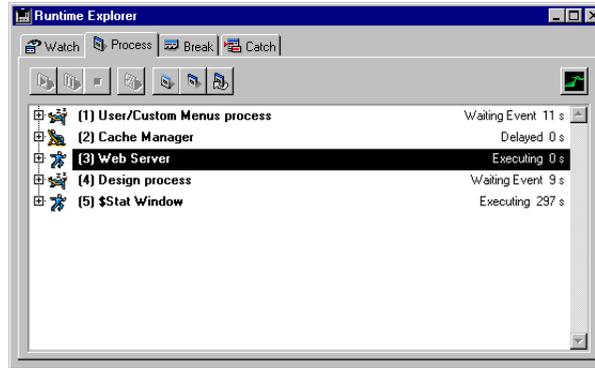
- 2 Click the Process Tab.

When you first run a database, the only processes that are executing are the User/Custom Menus process, the Cache Manager, and, if you have entered the Design environment, the Design process¹. The

1. Depending on the database Properties, the Web Server process may also be launched at startup.

User/Custom Menus and Cache Manager processes are always the first two processes listed in the Process List editor window.

Also, the \$Stat Window process corresponds to the Runtime Explorer window process.



For each process, the Process List editor window gives the following information:

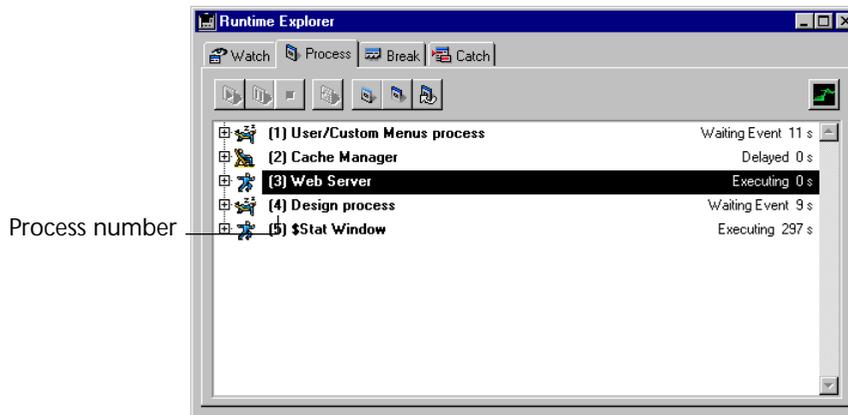
- Process number,
- Process name,
- Current status of the process,
- Total amount of execution time in seconds the process has taken since it was started.
- The graphical representation of the status and CPU time allocated to a process. For each process you can hide or display its graph by clicking on the icon located to the left of the process name.

The process attributes are explained in detail in the following sections.

4D Server 4D Client's Process page of the Runtime Explorer controls processes for a particular client. 4D Server's Process page of the Runtime Explorer controls processes for all clients connected to the server. For information, refer to the *4D Server Reference* manual.

Process Number

The default processes, the User/Custom Menu Process, Cache Manager, and Design Process, are listed first in the window and are processes 1, 2, and 3, respectively¹.



When you start your own process, the process either appears as the next process in sequence or takes the place of a process that has been aborted. For example, suppose processes 4 and 5 are executing. If process 4 is aborted, the next process to be started becomes process 4.

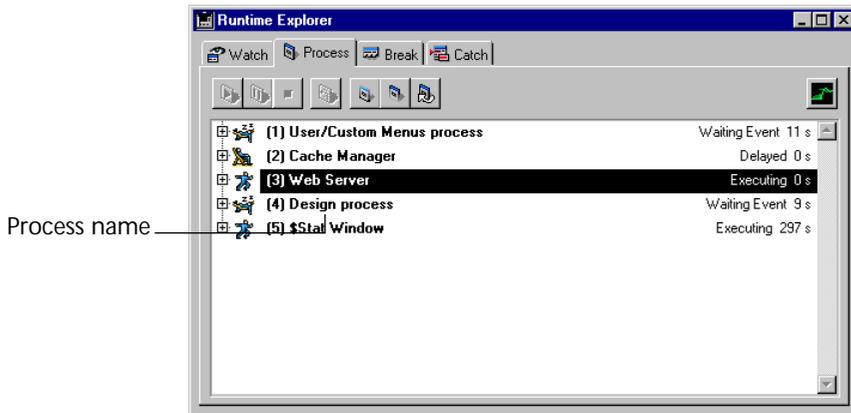
Note Processes are automatically aborted upon completion. You can abort a process before it has completed using the Process List editor. For more information about aborting a process, see the section [“Aborting a Process” on page 627](#).

Process Name

If you start a new process using New process, you can specify its name as a parameter to the New process function. The name specified in the parameter appears as the process name in the Process List editor, as

1. The Design process does not appear until you enter the Design environment.

shown above. For more information about the New process command, refer to the *4th Dimension Language Reference*.



If you do not specify a process name using the New process command, 4th Dimension automatically assigns the process a default name. Default names are based on the method used to start the process, as follows.

- **Processes started from a menu command** If you start a process from a menu command, the process is given the default name “M_*ProcessNumber*.” For instance, if process number 7 is started when a menu command is selected, the process is given the name “M_7”.
- **Processes started from the Execute Method dialog box** If you start a process from the Execute Method dialog box, the method is given the default name “P_*ProcessNumber*.” For instance, if process number 5 is started programmatically, the process is given the name “P_5”.
- **Processes started using New process, but not explicitly named** If you start a process using a method but do not specify the name as a parameter to the New process command, the process name is left blank.

If the name of a process begins with a dollar sign (\$), it is a local process that does not have access to tables or 4D Server.

Process Status

The status of a process is the current state of its execution—what the process is actually doing. In the Runtime Explorer the status of the

process is indicated by the icon located to the left of the process name and by the text located next to the process name.



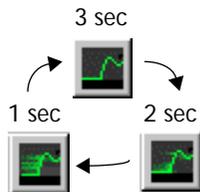
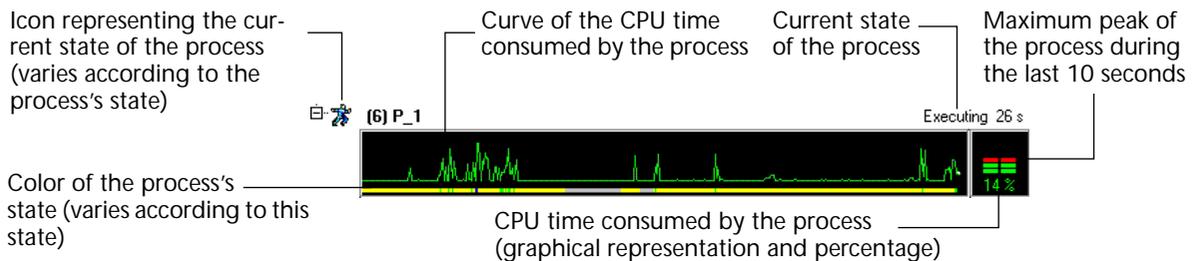
The following is an explanation of each status that can appear in the Process List editor window.

- **Executing** The process is currently executing.
- **Delayed** The process is delayed for a specific amount of time. During the period that the process is delayed, it does not take up any processing time. For information about how to delay a process, refer to the *4th Dimension Language Reference*.
- **Waiting event** The process is waiting for an action from the user such as a button being clicked or a menu command chosen.
- **Waiting for input/output** The process is waiting for some input or output to occur. For example, a process might need to wait while a group of records is being updated to disk.
- **Waiting for internal flag** The process is waiting for the cache manager to finish executing internal database tasks.
- **Paused** The process is paused until you tell it to resume execution. During the period that the process is paused, it does not take up any processing time. For more information, see the section [“Pausing and Resuming a Process” on page 627](#).
- **Aborted** The process has been terminated. When a process is aborted, 4th Dimension frees any locked records, cancels any transactions opened by the process but not yet validated or canceled, and frees the current selection and current record. Processes are automatically aborted upon completion. You can also abort a process before it has completed by using the Process List editor. For more information about aborting a process, see the section [“Aborting a Process” on page 627](#).
- **Hidden With Modal** A process which was displaying a modal dialog box has been hidden so that the user can no longer view the dialog. The process will remain in this state until the dialog is shown.

Process Time

In managing processes, 4th Dimension divides processing time among existing processes so that no single process is executing at every moment. Thus, the process time is the total amount of execution time a process has taken (in seconds) since it started executing. Note that the process time does not reflect the total amount of time that has elapsed since the process started executing since, in reality, execution alternates between all open processes.

The Runtime Explorer displays the processing time for each process. If you expand the process graph, it displays the following information:



You can modify the frequency at which data should be updated — which can be every one, two, or three seconds. To do so, click successively on the icon in the top right portion of the window. The greater the update time is, the more CPU time is consumed by the Runtime Explorer process. The number of processes to graphically represent on screen also influences the CPU time consumed by the process.

Note No CPU time is consumed for a process when its graphical representation is closed.

When you click in the graphical area, a vertical line appears where you clicked and a tip indicates the state of the process at that instant. By holding down the mouse button and moving it from side to side, you can view the changes in the process's state.

Controlling Process Execution

The Runtime Explorer allows you to control the execution of processes by pausing, resuming, or aborting a process. These operations are covered in detail in the sections below.

Note You can also delay a process for a specific period of time. For more information about delaying a process, refer to the *4th Dimension Language Reference*.

Pausing and Resuming a Process

You can temporarily suspend the execution of a process by pausing it. You may want to pause a process to give other processes more execution time or to allow an event upon which the process depends to occur.

For instance, suppose you start a process that prints a selection of records. You then realize that you want to modify the data in one of the records so you first pause the process, finish your modifications, and then resume the process to continue printing the records.

► To pause a process:

- 1 Select the process in the Process page of the Runtime Explorer.
- 2 Click the Pause icon  in the Process Page Toolbar.

The status of the process in the Process List editor window automatically changes to “paused.” The process remains paused indefinitely until you tell it to resume execution.

► To resume execution of a process:

- 1 Select the process in the Process page of the Runtime Explorer.
- 2 Click the Resume icon  in the Process Page Toolbar.

The status of the process returns to the status the process had at the time it was paused. For example, if the process was executing before it was paused, the process begins executing again. If the process was waiting for an event before it was paused, it continues waiting for an event.

Aborting a Process

A process is automatically aborted upon completion. However, you may need to abort a process before it completes for debugging purposes. Processes should not be aborted for any other reason. To stop the process from continuing execution, you abort the process in the Process List editor.

When a process is aborted, 4th Dimension frees any locked records, cancels any transactions opened by the process but not yet validated or canceled, and frees the current selection and current record.

- ▶ To abort a process:
 - 1 Select the process in the Process page of the Runtime Explorer.
 - 2 Click the Abort icon  in the Process Page Toolbar.

The status of the process in the Process List editor automatically changes to “aborted.”

Tracing a Process

You can debug a process by monitoring its execution in the 4th Dimension debugger.

- ▶ To debug a process:
 - 1 Select the process you want to debug in the Process page of the Runtime Explorer.
 - 2 Click the Trace icon  in the Process Page toolbar.

If the process is being executed, the 4th Dimension Debug window appears, allowing you to debug the process by stepping through its execution and evaluating expressions such as the value of fields and variables used in the method.

If the Process was paused, 4th Dimension “stores” the request and displays the Debug window as soon as the execution of the process resumes. For information about using the Debug window, refer to the *4th Dimension Language Reference*.

You cannot debug the Cache Manager, Web Server, or Design processes.

Hiding a Process

You can make a process invisible in the User and Custom Menus environments by hiding it. When a process is hidden, any windows or menus created by the process are invisible to the user while the process is executing.

Hiding a process is useful for operations in which you open a window which you later want to close. Instead of aborting the process to close the window, you can make the window invisible to the user by hiding the process that opened it. Even though the window is hidden, the process continues to execute and complete the operation it began.

- ▶ To hide a process:
 - 1 Select the process you want to hide in the Process page of the Runtime Explorer.
 - 2 Click the Hide icon  in the Process Page toolbar.

The process is now hidden from view in the User and Custom Menu environments.

Note The process continues to execute even though it is hidden.

- ▶ To display again a process:
 - 1 Select the hidden process you want show in the Process page of the Runtime Explorer.
 - 2 Click the Show icon  in the Process Page toolbar.

The process is displayed again in the User and Custom Menu environments.

Bringing a Process to the Front

You can make a window the frontmost window by bringing its process to the front. For instance, if the User/Custom Menu Process is brought to the front, the User or Custom Menu environment is brought to the front of the screen.

You can bring any user-created processes to the front. If you have created a window for a process, the window becomes the frontmost window on the screen. If a menu bar is attached to the window, 4th Dimension brings the menu bar to the front of the screen and makes its menus the current menus. The current menu bar is replaced by the menu bar of the process that is brought to the front.

- ▶ To bring a process to the front:
 - 1 Select the process you want to bring to front in the Process page of the Runtime Explorer.
 - 2 Click the Bring to Front icon  in the Process Page toolbar.

Any windows attached to the process are brought to the front of the screen. In addition, 4th Dimension displays the menu bar for the frontmost process window.

A

Segmenting Data Files

4th Dimension allows you to create data files as large as 128 gigabytes. However, no current microcomputer operating system supports this file size and hard disks of this capacity are not available. For these reasons 4th Dimension and 4D Server allow you to partition your data file into a maximum of 64 segments of 2 gigabytes each. Each segment can be located on a different physical volume. To increase the size of the data file beyond 2 gigabytes, you add data segments. This allows you to place different segments of a data file on different physical volumes.

Segmenting Data Files

You can either segment a data file at the time you create the database or after you begin to use it. You will want to segment a new data file if you expect the data file to become very large. Segmenting a data file allows a virtually unlimited amount of data to be stored.

Note You do not need to create any data segments unless you have more than 2 gigabytes of data or your hard disk cannot accommodate the size of your data file.

When segmenting a data file, you divide the data file into segments and then specify on which volume each segment is to be stored. For example, 4 gigabytes of data could be divided into 2 segments of 2 gigabytes each. Each segment can be limited in size, so you can reserve space on your hard disk for other files and avoid a completely full volume.

4th Dimension transparently fills the data segments in the order in which they were created. When a segment is full, 4th Dimension automatically moves to the next one. If by deleting records you make room in a data segment, the holes created in the segment will be reused.

When all of the data segments are full, you will be prompted with a message stating that there is no more room on the volumes where the segments are located. At this point, you would want to add a data segment.

You can create data segments:

- When you create a new data file,
- By adding data segments to an existing data file.

The following sections describe how to segment both new and existing data files.

Segmenting a New Data File

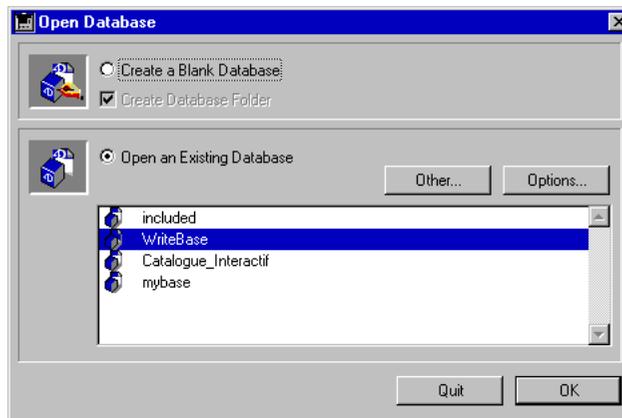
You can segment a data file when you create it. You create a data file when you create a database using 4th Dimension or 4D Server.

Note You can also create a new data file for an existing database. If the data file is missing or has been removed from the database folder, a dialog box appears asking you to either locate your data file or create a new one. You can also force 4th Dimension to allow you to create a new data file by holding down the **Alt** key (on Windows) or **Option** key (on Macintosh) while opening the database.

- To segment a new data file when creating a new database:

- 1 Launch 4th Dimension or 4D Server.

The Open Database dialog box appears.



- 2 Select the Create a Blank Database option and click OK.

A dialog box appears for you to name and select the location of your database.

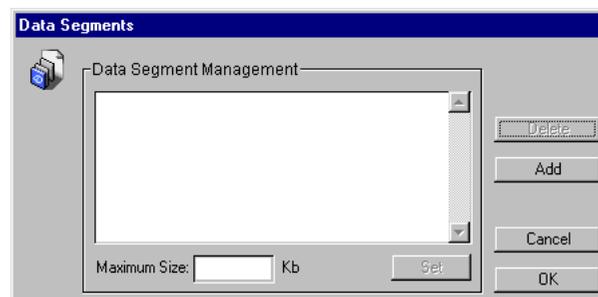
- 3 Type the name of the structure file of your database and choose its location.
- 4 Click the Save button while pressing the Alt key (on Windows) or the Option key (on Mac OS).

The Create a Data File dialog box appears. This dialog box contains a button, Split, that allows you to divide the data file into segments.



- 5 Click the Split button.

The Data Segment Management dialog box appears.



- 6 Click the Add button to create a data segment.

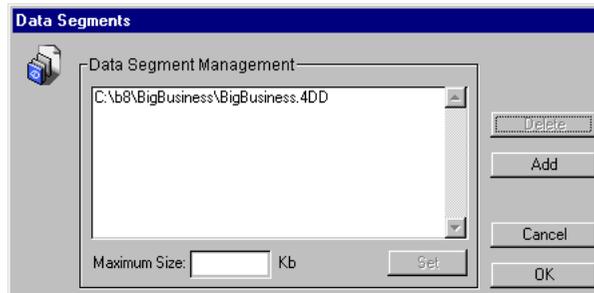
A save-file dialog box appears.



7 Enter the segment name.

8 Click the Save button.

The Data Segment Management dialog box reappears and displays the segment you just created.



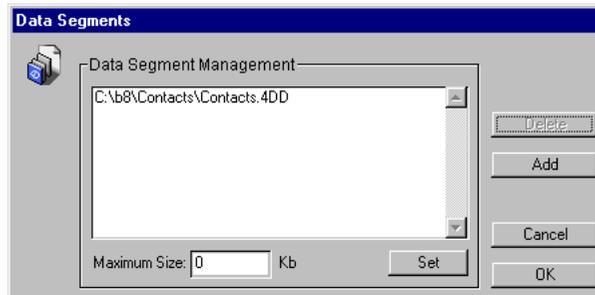
You can continue to create data segments by clicking the Add button and repeating the process.

Segmenting an Existing Data File

To increase the size of an existing data file beyond 2 gigabytes, you can add data segments, each of which can contain up to 2 gigabytes of data.

You create segments for existing data files in the Design environment. If you are using 4D Server, you create data segments on the server machine.

- ▶ To segment an existing data file:
 - 1 Display the Structure window (in 4th Dimension stand-alone only).
 - 2 In 4th Dimension, choose Data Segments from the Structure menu.
OR
In 4D Server, select Segments from the Data menu.
 The Data Segment Management dialog box appears.



- 3 Click the Add button to create a segment.
- 4 Enter the segment name and select its location.

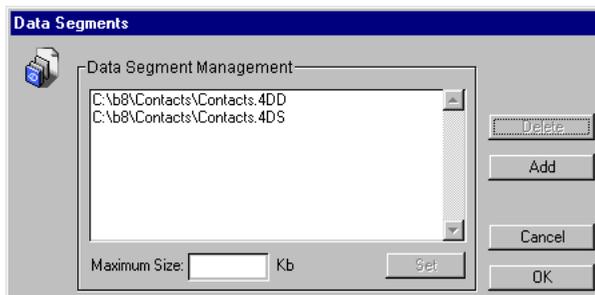


4th Dimension provides a default name for the segment, which is the name of the database followed by the number of the data segment, followed by the prefix “.4DS” (on Windows) or .data (on Mac OS). This naming convention allows you to easily distinguish each data segment. You can change the names of the data segments at any time.

If you are using an operating system that allows only eight characters before the prefix, your filename may be truncated in order to add the segment number. For example, a data file named CONTACTS.4DD will produce the following data segments: CONTACT1.4DS, CONTACT2.4DS, and so on.

5 Click the Save button.

The Data Segment Management dialog box reappears, displaying the new data segment.

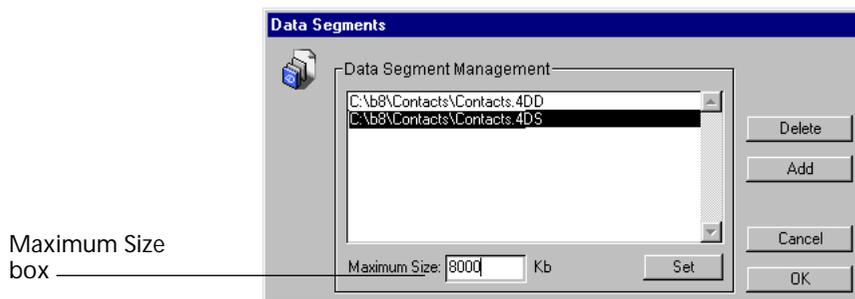


Limiting Data Segment Size

If you do not specify a size limit for data segments, 4th Dimension will fill the data segment until the volume on which it is located is full or until the limit of 2 gigabytes is reached.

Note You can change the limit of a data segment that already contains data. In this case, the limit cannot be less than the size of the data already present in the segment. If you specify a lower size, 4th Dimension will automatically adjust the limit to the current size of the data file when you validate the new limit.

- ▶ To specify a maximum size for a segment:
 - 1 Select the data segment in the Data Segment Management dialog box.
 - 2 Enter a size (in kilobytes) in the Maximum Size box.



- 3 Click the Set button.

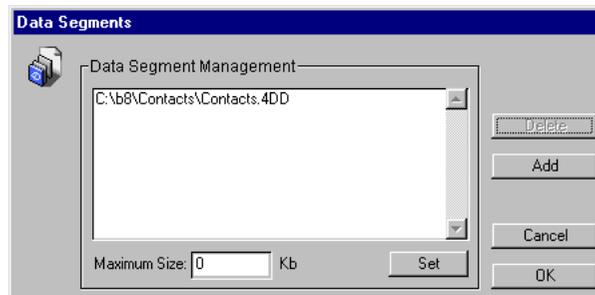
This limits the size of the segment and allows you to reserve space on your drive for other files.

Adding a Data Segment While Indexing

If 4th Dimension reaches the maximum size of the last data segment during an indexing operation, it will present a dialog box allowing you to change the segmentation of the data file.

4D Server If you are using 4D Server, this error message appears on the server machine.

When you click the **Add Segment** button, 4th Dimension displays the **Data Segment Management** dialog box, which allows you to change the maximum size of a data segment or add new data segments.



Note The **Add Data Segment** button is available only if you have access privileges to the Design environment. For more information on access privileges, see [Chapter 9, “Managing Password Access”](#) on page 555.

Deleting Data Segments

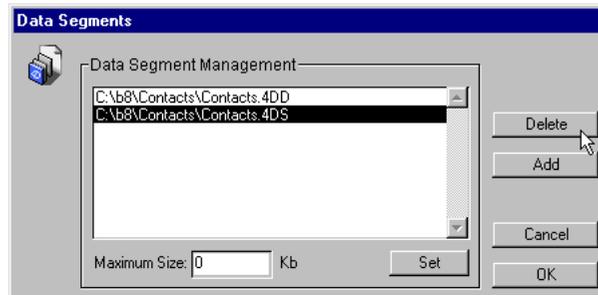
You may want to delete data segments if a large database suddenly decreases in size, or if you initially created more segments than are ultimately necessary. The method that you use to delete a data segment depends on whether it is a new or existing data segment.

Do not delete a data segment in the File Manager or the Finder. For more information, see the section “Missing Data Segments” on page 638.

Deleting a New Data Segment

A new data segment is a data segment that you just created in the Data Segment Management dialog box. You can delete a data segment only while the dialog box is still open. Once you leave the dialog box, the segment becomes part of the data file (in this case, see the following section).

To delete a data segment, click the segment you just created and then click the Delete button.



Deleting an Existing Data Segment

You can delete an existing data segment by using 4D Tools to compact the data file. Once you compact the data file, you can reconfigure the segmentation of your data file. For more information on reconfiguring data segments, see the section [“Reconfiguring Data Segments” on page 639](#).

Missing Data Segments

Do not delete data segments using your operating system! Do not use the File Manager or the Finder. If you delete a data segment using your operating system, 4th Dimension will ask you to locate the missing data segment.

Click the OK button. An Open File dialog box appears.

If you no longer have the data segment, click the Cancel button. Since you could not locate the data segment, 4th Dimension will not allow you to open the database. In this case, you will need to use 4D Tools to repair the database.

Open your database with 4D Tools. The application will ask you to find the missing data segment.

Click the OK button.

In the Open File dialog box, click the Skip button.

4D Tools displays an alert stating that it is going to repair the database by tags.

Click the OK button.

Click the OK button to repair the database by tags.

When you repair by tags, 4th Dimension recreates the data file with the segments present on your disks. This new data file will allow you to reopen your database, but does not guarantee the integrity of your data.

Reconfiguring Data Segments

You may want to reconfigure the segmentation of your data file to accommodate a change in disk size or to change the number of data segments.

For example, suppose you have a 1 gigabyte data segment and you need to switch to two 500 megabyte drives. To do this, you must divide the 1 gigabyte segment into two segments.

The most efficient way to reconfigure data segments is to compact your database using 4D Tools. This method is described in this section.

Before you begin, be sure that you have enough room on your disk for another copy of your database. The process of compacting a database creates a new copy of both the structure and data files. Also, because the compacting process can take some time, be sure to plan accordingly. We recommend that you give 4D Tools the largest possible memory allocation to quicken the compacting process.

► To compact your database and reconfigure your segmentation:

1 Launch 4D Tools and open your database.

2 Choose Compact from the Utilities menu.

OR

Click the Compact button in the Tools Viewer Window.

A save-file dialog box appears.

3 Enter a new name for the structure and select a drive.

4 Click the Save button.

4D Tools creates a copy of your structure file.

A save-file dialog box containing the Split button appears.



5 Click the Split button.

The Data Segment Management dialog box appears.

You can use this dialog box to add new data segments, configuring the data segments as you wish.

B

Assigning a Help File to a 4D Database

4th Dimension allows you to associate a custom on-line help file with each database.

The on-line help system provided with 4th Dimension is compatible with each 4D work environment (stand-alone database or client/server, interpreted or compiled, run using 4D Runtime, or integrated into a 4D engine).

4th Dimension supports several help files formats which correspond to the industry standards (HTML, HLP, and Apple Guide).

Moreover, you can associate a precise section of this help file with each of the database's forms, which allows you to provide contextual on-line help.

Setting a Database On-line Help

File Formats

The format of the files must be one of the following:

- a “.HLP” file, which is the standard Microsoft Help¹ format for Windows. This type of file can be used on MacOS², provided the Microsoft Help for Mac application has been installed.

-
1. For information on how to generate a .HLP file, refer to Microsoft's documentation.
 2. You have to change the document type to HELP and the creator to MSHE. Successfully converting the document depends on the version of the on-line help compiler as well as the version of Microsoft Help being used. Otherwise, this solution doesn't allow you to use contextual on-line help.

- a file “.HTM”, which is the HTML format. With this format you can create an identical on-line help on all platforms. The on-line help is then displayed in a Web browser.
- an Apple Guide file (on MacOS only). This file can use *coachmarks* (allowing you to coach an object in the application window). If Apple Guide has not been installed, 4th Dimension will try to open the file in Microsoft Help format.

Note The Plug-ins can also have a help file, which must be placed in the Win4DX and/or Mac4DX folder for both single-user and client-server applications. The Plug-in's help file must be the name of the Plug-in with the .HLP or .HTM extension.

Assigning the Help File to the Database

Once the Help file generated, you need to associate it to a database so that it is opened when users call the on-line help. To assign a help file to a database, make sure

- The name of the help file is identical to the name of the database's structure file. It must also have the “.HLP” or “.HTM” extension, depending on its format (see above) and platform.
- The help file is placed in the database's folder or in the Win4DX and/or Mac4DX folder.

4D Server If you want the help file to be accessible to all the client workstations, place the file in the Win4DX and/or Mac4DX folder. It will then be transferred to the client workstations automatically.

Creating Contextual On-line Help

Creating contextual on-line help is done by associating a section number with each of your forms. When you call the on-line help from a form, the corresponding help page is displayed. When the user calls the on-line help, the help topic that has the same ID as the form is displayed.

Contextual on-line help is available:

- on Windows, if the on-line help is in HLP format.
- on MacOS, if the on-line help is in Apple Guide format.
- on all platforms, if the on-line help is in HTML format and if the viewing browser is compatible with Internet Config 1.2.

The assignment of an ID number to a form is done in the form properties, for more information, refer to [“Contextual On-line Help” on page 264](#).

Once you have assigned the ID number to the form, you need to assign the same ID number to the help file. This operation varies according to the format you are using:

- for HLP files, refer to the Microsoft on-line help compiler’s documentation (Help Compiler).
- for Apple Guide files, refer to the Apple documentation.
- for HTML files, you must declare each section and assign a number to it.

A section is declared by using a marker of type ``. For example ``.

The URL of the section has the following form `...`. For example `...`

If the section number passed in the form is 0, 4D displays the first page of the help file.

Calling the On-line Help from a Database

In 4D, you can call a database’s custom on-line help in two ways:

- by choosing **DatabaseName Help** in the Help menu (in version 7 of MacOS, this menu appears as a question mark). In this case, the first page of the help file is displayed.
- by pressing the F1 key (MacOS and Windows) or the Help key (MacOS only) when a form is displayed on screen. In this case, if a help field number has been associated with the form, the corresponding page is displayed (contextual help); otherwise the first page of the help file is displayed.

Index

Symbols

- ! (exclamation point)
 - in entry filter codes 336
- " (quotation marks), in entry filters 341
- # (number sign)
 - as a placeholder 360
 - in display formats 360
 - in entry filter codes 336
 - in entry filters 342
 - placeholder in Alpha fields 365
- \$ (dollar sign)
 - in display formats 361
- & (ampersand character)
 - displaying a number as a time 362
 - for scientific notation 362
 - in entry filter codes 336
 - initiating an entry filter 340–341
- () (parentheses)
 - as dead characters 343
- * (asterisk)
 - as a placeholder 360
- := (assignment operator)
 - in methods 492–495
- ;(semicolon)
 - in number fields 362
 - specifying display formats for embedded fields and variables 457
- < (less than character)
 - in embedded fields and variables 457
 - over maximum number of digits 360
- <?[TableName]> (Dynamic Table name) ... 300, 330
- <?[TableName]FieldName> (Dynamic Field name) . . . 300, 330
- > (greater than character)
 - in embedded fields and variables 457
- @ (wildcard character)
 - in entry filters 339
 - using in methods 515
 - wildcard searches 171–173
- [] (brackets), for table names 491
- ^ (caret)
 - as a placeholder 360
 - generating a non-breaking space 360
- _ (underline character), for placeholders 342

- | (vertical bar character)
 - for display formats and filters 344
 - for entry filters 344
 - for styles 363
- ~ (tilde character)
 - forcing capitalization 341
 - in entry filter codes 336
 - initiating an entry filter 340–341

Numerics

- 3D check boxes 387
- 4D Backup 30
- 4D Chart methods
 - displayed in Method editor 510
- 4D Client 23, 190
 - opening a password-protected database from 559
 - using Process List editor from 622
- 4D Compiler 497, 518
- 4D Internet Commands 510
- 4D Main Memory
 - setting 93
- 4D Open 119
- 4D Open connections
 - allow 87
- 4D Runtime 554
- 4D Server
 - changing form names 227
 - changing method names 502
 - creating data segments with 634
 - field properties 135
 - modifying database objects 111
 - modifying menus 533
 - modifying table definitions 129
 - moving table images 123
 - object locking while modifying forms ... 230, 251
 - object locking while setting database properties 82
 - object locking, with lists 584
 - object locking, with passwords 556
 - object locking. *See also* object locking
 - Process List 622
 - resizing a table image 122
 - saving form pages 323
 - saving server path 559

- setting maximum and minimum values 349
 - 4D Tools
 - compacting a database with 639
 - repairing a database with 638
 - 4D Write
 - used to store text 137
 - 4DVAR MAVAR (Web Server) 102
 - 4th Dimension
 - API 87
 - application icon 24
 - confusing 251
 - flushing data buffers 93
 - 4th Dimension plug-ins 510
- A**
- Abort icon 628
 - Accept button 386
 - Access drop-down list
 - function of 502
 - in Form Properties window 253, 578
 - in Method Properties window 579
 - access group
 - setting for forms 578
 - setting for menu commands 579
 - setting for methods 579
 - access privileges
 - creating an access hierarchy 560–561
 - for 4th Dimension environments 561
 - for Design environment 563
 - for forms 253–255, 577
 - for menu items 579–580
 - for methods 502–503, 578
 - for plug-ins 580–581
 - for record operations 127, 575–577
 - setting for methods 518
 - Access to Plug-Ins dialog box 580
 - ACCUMULATE command 464
 - Action drop-down menu
 - for creating buttons 385
 - Activate Automatic Comments 73
 - active object
 - definition of 326
 - active objects 381–416
 - automatic numbering of in grids 421
 - changing appearance of 297–309
 - copying 292–293
 - creating 374–380
 - creating arrays of 419–421
 - deleting 295
 - display formats for 376–378
 - duplicating 290
 - enterable 382
 - groups of 279
 - keyboard equivalents for 551
 - non-enterable 382
 - on forms 372
 - role of 192–193
 - setting data entry controls 378
 - setting display formats 376
 - types of 326, 381–421
 - active windows
 - working with multiple 615
 - Add button
 - in Data Segment Management dialog box 633
 - Add Field tool/icon 237, 327
 - Add Item menu item
 - Menu menu 538, 549
 - Add Menu menu item 543
 - add record operations
 - access to 575
 - Add to Subform button 162, 387, 431
 - adding a subform to a form
 - from Explorer 61
 - adding fields to form
 - from Explorer 58
 - adding form name to method
 - from Explorer 61
 - adding table names and field names to methods
 - from Explorer 58
 - Administrator
 - access to Password editor 581–582
 - as group owner 563
 - system maintenance role 581
 - using Edit User dialog box 568
 - Advanced button
 - in Object Properties window 411
 - Advanced options
 - in Form Wizard 195
 - Advanced screen
 - in Form Wizard 205–225
 - ALERT command 490
 - Align menu item 241
 - Align to Grid menu item 241, 285, 286
 - aligning objects 280
 - using alignment grid 284–285
 - using alignment tools 281–282
 - alignment grid 284–285

- alignment tools
 - Align Bottom tool 281
 - Align Center Horizontal tool 281
 - Align Center Vertical tool 281
 - Align Left tool 281
 - Align Right tool 281
 - Align Top tool 281
 - Allow Deletion Control
 - in Database Properties dialog box 168
 - Allow deletion control property 86
 - Alpha field formats 364, 364–366
 - entering and editing 365
 - examples 365
 - Alpha fields 136
 - Alt+Click
 - to open object method 426
 - ampersand (&) character
 - displaying a number as a time 362
 - for scientific notation 362
 - in entry filter codes 336
 - initiating an entry filter 340–341
 - appearance
 - setting for objects 298
 - setting for text objects 302
 - setting in Form Wizard 221
 - Appearance drop-down menu
 - in Form Wizard 221
 - Appearance settings
 - for buttons 384
 - for check boxes 388
 - Append Menu menu item
 - Menu menu 536
 - arguments
 - format of 492
 - in entry filters 340–342
 - in methods 492, 495
 - arrow keys
 - moving objects with 269
 - Arrow tool/icon
 - selecting objects 266
 - assignment operator (:=)
 - in methods 492, 494
 - Associating a Comment to an Object 69
 - asterisk (*)
 - as a placeholder 360
 - Auto Assign Related Value in Subform 161–162
 - in Relation Properties window 161
 - Auto Delete Related Many 164
 - Auto One to Many check box 161
 - in Relation Properties window 161
 - Auto Relate One check box 161
 - in Relation Properties window 160
 - Auto Wildcard Support check box
 - in Relation Properties window 160
 - automatic actions 385–387
 - accepting records 386
 - canceling records 386
 - deleting records 386
 - form page navigation 386
 - No Action 385
 - record navigation 386
 - automatic buttons
 - buttons 383
 - Highlight buttons 383
 - Invisible buttons 384
 - Automatic Comments 72
 - automatic foreground and background colors ... 308
 - Automatic Relation check boxes
 - in Relation Properties window 167
 - automatic relations 168, 170, 429
 - Auto Assign Related Value 161
 - establishing 168
 - Many to Many 175
 - One to Many 155
 - One to One 174
 - records selected in related table 181
 - See Also* relations
 - turning off 161
 - automatic repositioning 273
 - automatic resizing 273
 - automatic transactions 86
 - Automatic Width checkbox
 - for subforms 437
- ## B
- background color 310
 - background graphics in picture fields 371
 - background page
 - displaying 316
 - in Detail forms 191
 - on multi-page forms 314
 - placing a picture on 309
 - selecting objects on 265
 - backing up, databases 29–30
 - backup systems
 - for databases 30
 - Balloon Help 147

- Basic screen
 - in Form Wizard 194–195
 - Blob fields 139
 - Boolean field formats 366
 - examples of 366
 - Boolean fields 138
 - as check boxes 368–369
 - as radio buttons 366–368
 - default value 368
 - formatting 366–369
 - Border menu item 241, 307
 - borders
 - setting for form pages 323
 - setting patterns of 306–307
 - brace matching
 - in Method editor 513
 - brackets ([]), for table names 491
 - Break area
 - in output form 439
 - in printed reports 444
 - Break areas 442, 443, 469, 470
 - additional 442, 444
 - additional levels of Breaks 470
 - calculating subtotals 465–466
 - calculating totals in reports 453
 - creating additional 444
 - in printed reports 453
 - in reports 453, 470, 471
 - Break control lines 444, 454
 - creating additional 462–464
 - deleting 464
 - Break Header areas 462, 469
 - Break Header control lines 462
 - adding 462–464
 - deleting 464
 - BREAK LEVEL command 464
 - break levels 462
 - additional 466–467
 - Break List (modification) 78
 - Break Page (Runtime Explorer) 78
 - break processing 463
 - initiating 464
 - breaks
 - examples of calculations in 467
 - for calculations 465–466, 468
 - multiple 462
 - Bring to Front icon 629
 - bullet (•) character 515
 - business rules
 - enforcing 326, 331–352
 - button grid 401–402
 - Button Text area 383
 - buttons 383
 - 3D 383
 - as active objects 382–387
 - creating 385
 - for subform actions 386
 - for subforms 431–432
 - Highlight 383
 - Invisible 384
 - labeling 383
 - radio 388–390
 - standard 383
 - variables associated with 385
 - Buttons page
 - Form Wizard 208, 319
- ## C
- Cache Manager process 615
 - Cache statistics
 - Runtime Explorer 77
 - calculations
 - in reports 454
 - methods for 382, 422–428
 - object methods for 470–471
 - subtotals in reports 465–466
 - CALL PROCESS command 487
 - Can't Delete if Related Many option
 - in Relation Properties window 163
 - Can't Modify attribute 141
 - assigned to primary key field 164
 - in Field Properties window 331
 - in related tables 155
 - Cancel button
 - automatic action 386
 - capitalization
 - tilde (~) character for 341
 - caret (^)
 - as a non-breaking space 360
 - as a placeholder 360
 - Case Sensitive (Find Editor option) 82
 - case sensitivity, in passwords 568
 - Catch Page (Runtime Explorer) 78
 - check boxes 387–388
 - default value of 387
 - formatting Boolean fields as 138
 - values in 387

- Check the Structure File 26
- choice list 331
 - assigning at form level 333
 - assigning in Field Properties window 331
 - assigning to a field 134
 - uses of 583
- Choice List drop-down menu 334
- choice lists 333–335
 - assigning 335
 - See also* lists
- Choices & Help page
 - Field Properties window 147
- Choices attribute 147
- circular relations 181–182
- Clear menu item 295, 528
 - in Edit menu 47
- Clear Object Method menu item 242, 428, 507
- Clipboard
 - copying objects to 47
 - pasting pictures into Picture fields 369
 - storing copied objects 293
 - storing deleted objects 295
 - viewing contents of 47
- Close *EditorName* menu item 51
- Close *FormName* menu item 323
- Close Network Connection setting 96, 97
- Close Structure menu item 46
- closing all Design environment windows 51
- codes
 - entry filter 340–342
- color
 - setting for a field 148
- Color menu item 241
 - Object menu 308
- Color page 148
 - Field Properties window 134
 - Relation Properties window 168
- Color property
 - in Relation Properties window 164
- color table and field names 90
- color table background 90
- colors
 - assigning to method elements 518–519
 - in forms 307
- Colors menu, for method elements 518
- Colors page 306
 - Object Properties window 304, 308, 311
- combo box
 - managing a 399
 - setting default values for 350
- command
 - adding to a method 67
- Command key
 - reserved key combinations 550
- commands
 - DISABLE ITEM 544
 - displayed in Method editor 492
 - New process 542, 617, 624
 - role of 490
- Commands page
 - in Explorer 66–67
- Comments
 - Automatic Comments 72
 - Font Attributes 71
 - Inserting the Date, Time, or User 71
- compacting databases 639
- compatibility
 - of structure and data files 27
- Completely Delete option 129–130
- Connect Menu menu item
 - Menu menu 543
- connected menus 536
 - advantages of 543
 - creating 543
 - deleting 545
 - modifying 544–545
- Connections page
 - Database Properties dialog box 96–97
- constant
 - adding to method 65
- Constants page
 - Explorer 65, 425
- Contextual menus
 - Structure window 49
- Contextual On-line Help 264
- control lines
 - creating additional 462–464
 - in forms 442–446
 - in reports with several breaks 470
 - moving output 445
- Control page
 - Relation Properties window 168
- Coordinates page
 - Object Properties window 269, 274
- Copy menu item 293
 - in Edit menu 47, 550
- copying
 - form objects 292–293

- creating
 - active objects 374–419
 - buttons 385
 - check boxes 387
 - custom formats and filters 344–347
 - custom labels 455–456
 - database structures 119–120
 - databases 22
 - enterable and non-enterable objects 382
 - fields 131–135
 - forms 197–202
 - formulas 525–526
 - labels 455–456
 - lists 588–589
 - menu bars 534–536
 - methods 496–501
 - object methods 497–498
 - radio buttons 388–390
 - table relations 167–168
 - creating relations
 - using Explorer 58
 - crosshair/crossbar pointer
 - for creating objects 277
 - Ctrl+L, opening a form from the Method editor . 520
 - Ctrl+P, opening a method from the Method editor . . 520
 - current date
 - as default value 350
 - Current Item Properties
 - in Menu Bar editor 535, 541, 546
 - Current Item Properties area
 - in List editor 593
 - in Menu editor 619
 - Current List area 589
 - in List editor 589–592
 - Current Menu Bar list 549
 - for deleting a menu 552
 - in Menu Bar editor 535
 - current record 181
 - current selection
 - in unrelated tables 218
 - working with multiple 615
 - current time
 - as default value 350
 - custom applications
 - Custom Menu environment for 34–35
 - definition of 34–35
 - in Custom Menu environment 34–35
 - menu bars for 543, 554
 - requirements for 531
 - custom entry filters 344
 - custom formats
 - for Alpha fields 366
 - for Number fields 363
 - specifying in Object Properties window 363
 - custom formats and filters
 - creating 344–347
 - Custom Formats and Filters editor 357
 - custom menus
 - adding menu items 538–539
 - adding splash screens to 553
 - adding to menu bars 536
 - changing font styles 546–547
 - custom applications and 554
 - example of 531
 - keyboard equivalents for 550–551
 - previewing 553–554
 - Custom Menu environment 34–35
 - applications for 34–35
 - choosing menu items in 541
 - definition of 34
 - entering 34
 - with menus 554
 - Custom Menu item 34, 36
 - custom toolbar 532
 - Cut menu item 47
 - cycle arrows 52
 - in Listing editor 509
- ## D
- dashes
 - as dead characters 343
 - Data Control page
 - Database Properties dialog box . . 85–87, 557, 563
 - Object Properties window 335, 348, 349, 357, 366
 - Data Entry area, for subforms
 - Double Clickable check box 431
 - Enterable check box 431
 - Selectable check box 431
 - data entry controls
 - choice lists 333
 - default values 349
 - entry filters 335
 - excluded lists 335
 - for enterable objects 378
 - Mandatory attribute 333
 - maximum and minimum values 587–588

- setting 331–352
 - using lists as 587
- data entry order 320–322
 - entry order pointer 321
 - grouping objects 322
 - restoring standard order 322
 - setting first object 321
 - viewing and modifying 321
- Data file
 - Creating 26
 - Selecting 26
- data file segmentation
 - reasons for 631
- data files
 - creating new 28
 - locating 26–29
 - maximum size of 631
 - naming 27
 - opening an different 28
 - paths to 29
 - segmenting existing 634–636
- data segment
 - adding while indexing 637
 - deleting a new 638
 - deleting an existing 638
- Data Segment Management dialog box . . . 633–634,
 - 635–636, 636–637
 - for reconfiguring data segments 640
- data segment size
 - limiting 636
- data segments
 - adding 631
 - dealing with missing 638–639
 - deleting 637, 638
 - reconfiguring 639
- Data Segments menu item
 - Structure menu 635
- data validation 326
- database cache memory
 - setting 93
- database methods
 - modifying 62
 - role of 480, 483
 - writing 486
- Database Properties
 - Automatic Comments 73
 - Automatic Form Creation 85
 - Default User 565
 - User Mode Access 564
- database properties
 - Allow Deletion Control 168
 - Automatic Transactions During Data Entry 86
 - choosing a progress indicator 84
 - Close Network Connection setting 96, 97
 - compatibility options 96–97
 - custom display formats and entry filters 95
 - data control 85–87
 - database cache memory 93
 - default Method editor 91
 - editing style sheets 90
 - general 83–98
 - hiding keywords in Method editor 91
 - New memory allocation scheme 93
 - Platform Interface option 105–110
 - Process indicator 84
 - setting 82–95
 - setting 4D main memory 93
 - Show Toolbar property 84
 - Startup environment 83
 - structure 90–92
 - Structure access 87
 - tune up 92–93
 - Use Icons for Field Types 91
 - Use Old File Procedure Scheme 92
 - Use Old Startup Procedure Scheme 92
 - user interface 89–90
- Database Properties dialog box 345
 - default environment 25
 - Formats and Filters page 344, 345
 - setting default Method editor 426
- Database Properties menu item 345
- database structure 113
 - creating 119–120
 - fields 114
 - flat file 115
 - multiple table 115
 - structure files 26
 - subtables in 139–140
 - table relations 117
- Databases
 - Selecting a different data file 26
- databases
 - backing up 29–30
 - compacting 639
 - creating 22
 - creating structure of 119
 - definition of 114
 - fields in 114

- flat file 116
- multi-process 615
- opening 24
- records 114
- relational, defined 117
- relations 31
- structure of 113
- tables in 115–118
- Date fields 137
 - display formats for 358
- dead characters
 - in entry filters 343
- Debug window 628
- debugging processes 628
- decimal points, in number display formats 361
- default data entry order 320
- default editor for methods
 - setting the 497
- default fonts 242
- default forms
 - automatically created by 4th Dimension 190
- default lists of values 350
- default Method editor
 - setting the 91
- default names, with processes 624
- Default Owner of Objects Created by This User drop-down list
 - in Edit User dialog box 568
- default processes 615
- default settings
 - for text objects 301
- default startup environment 83
- Default User (Passwords) 565
- Default Value entry area
 - in Object Properties window 349
- default values
 - automatically generating 350
 - Boolean fields 368
 - setting 349–351
 - setting programmatically 352
 - using current date 350
 - using current time 350
 - using sequence numbers 350
- Default Values dialog box
 - in Object Properties window 351
- default window size
 - automatic size 260
 - Set size 260
 - size based on selected form object 260
- Default Window Size area
 - Form Properties window 261
- Define Grid menu item 284
 - Form menu 284
- Define Ruler Units menu item 276, 476
- Delete button
 - for deleting forms 226
 - keystroke equivalent for 295
- Delete Item menu item, for deleting menu items 552
- Delete menu item
 - for lists and list items 592
- Delete Menu menu item
 - Menu menu 552
- delete operations
 - access to 576
- Delete Page menu item 318
- Delete Record buttons 386
- delete record operations
 - access to 575
- Delete Related Many option
 - in Relation Properties window 163
- DELETE SELECTION command 129, 130
- Delete Subform button 387, 431
- deleting
 - connected menus 545
 - fields 132
 - form pages 318
 - forms 226
 - list items 592
 - lists 592
 - menu items 552
 - menus 552
 - object methods 428, 507
 - objects 295
 - related records 163
- deletion control
 - in several related tables 164
- Deletion control options 168
 - in Relation Properties window 163–164
- Design environment 31–32
 - access to 87, 563
 - editors in 36–44
 - moving to and from 36
 - object locking in 111
 - role of 31
 - toolbar 53–54
- Design environment menus 45–49
- Design Environment page
 - Database Properties dialog box 510

- in Database Properties dialog box 90–92
- Design environment page
 - Database Properties dialog box 483, 487
- Design item
 - Use menu 34, 35
- Design menu 48–49
 - Edit Form menu item 48, 60
 - Edit Method menu item 48
 - list of open windows 49
 - New Form menu item 48, 60, 197, 446
 - New Method menu item 48, 498
- Design process 615
- Designer
 - access to environments 561
 - assigning a password to 566
 - role of 561
 - system maintenance role 581
 - users and groups created by 562
 - using Edit User dialog box 568
- desktop filenames
 - on Macintosh 27
- desktop files 26–29
 - pathnames on Macintosh 29
- Detail area 443, 469
 - explanation of 470
 - in output form 439, 441
 - in reports 470
 - in screen display 454
- Detail control line 454
- detail form
 - defined 187
- Detail Form for Printing form type
 - in Form Wizard 454, 456
- detail forms
 - for subrecords 430
- dials 413–416
 - example of 413
- DISABLE ITEM command 544
- disk capacity
 - change of 639
- Display
 - Comments in the Explorer 69
 - Grid 233
 - Limits 234
 - Marker Labels 234
 - Markers 234
 - Object Properties 233
 - Page 0 233
 - Paper 233
 - Property List 233
 - Rulers 233
 - Tools Palette 233
- display characters
 - for Number field formats 361–364
 - in entry filters 342
- display formats 357–371
 - Alpha field formats 366
 - Boolean field formats 366–369
 - Date field formats 358
 - for objects 376
 - in mail-merge templates 457
 - Number field formats 359–364
 - Picture field formats 369–371
 - Time field formats 358
 - with entry filters 339
- Display Formats and Entry Filters page
 - in Database Properties dialog box 95
- Display List of Users in Password dialog box
 - in Database Properties 88, 558
- Display Only attribute 141, 332
 - in Field Properties window 331
- Display page
 - Object Properties window 299, 388
- display page
 - adding a 315
- display pages
 - selecting objects on 265
- Displaying Runtime Explorer 75
- Distribute tools
 - Tools palette 286
- dollar (\$) sign
 - in display formats 361
- Double Clickable check box
 - for subforms 436
- drag and drop
 - for form objects 380
 - from Explorer 514
 - from Tables page of Explorer 58
 - in Explorer 57
- drop-down lists
 - initializing 350, 398–399
- duplicate fields 133
- Duplicate menu item 241
 - Object menu 290
- Duplicate on Matrix menu item 241, 421
- Dynamic Table and Field Names 329

- E**
- EDIT ACCESS command 581
 - Edit Form menu item 226, 227
 - Design menu 60, 250
 - Edit Group dialog box 569
 - adding groups 569
 - modifying groups 570
 - specifying group owners 570
 - Edit Group menu item
 - Passwords menu 570
 - Edit menu 47
 - Clear menu item 47, 295
 - Copy menu item 47
 - Cut menu item 47
 - Paste menu item 47
 - Select All menu item 47
 - Show Clipboard menu item 47
 - Undo menu item 47
 - Edit Strings button
 - in Object Properties window 398
 - Edit User dialog box 582
 - adding users 567–569
 - changing user name or password 569
 - choosing owner of objects created by users 568
 - modifying users 569
 - specifying STARTUP method 568
 - viewing database usage 582
 - Edit User menu item
 - Passwords menu 569
 - editing methods
 - with Flowchart editor 528–529
 - editor windows 36–51
 - closing 46
 - Flowchart editor 39, 496, 523–529
 - Form editor 37–38, 231
 - List editor 42
 - Listing editor 39, 496, 507
 - Menu Bar editor 40–41, 535
 - Method editor 38–39, 496
 - Password Access editor 41, 567
 - Process List editor 622–626
 - reverting to last saved version of 47
 - saving contents of 47
 - Structure editor 36–37, 120
 - Enabled check box
 - in Menu Bar editor 548
 - Enter in List mode 585
 - Enter key 515
 - Enter Password dialog box 557
 - choosing the default 557
 - Enterable attribute 333
 - setting 332–333
 - Enterable check box
 - for subforms 436
 - in Object Properties window 333
 - enterable objects 382
 - adding scroll bars to 352
 - data entry controls for 378
 - display formats for 376
 - in data entry order 321
 - Enterable Related Fields check box
 - in Form Wizard 333
 - entry filter codes 336, 340–342
 - Entry Filter display area 337, 344
 - Entry Filter drop-down menu 337–339
 - entry filters 335–344
 - arguments 341–342
 - choosing 337–339
 - codes 336
 - custom 344
 - dead characters 343
 - display characters 342
 - initiating 340
 - modifying 339
 - placeholders 342
 - with display formats 339
 - Entry Order menu item 239, 321–322
 - entry order pointer 321
 - environments
 - changing between 32, 36
 - selecting startup 83
 - error messages
 - in password access system 559
 - errors
 - during execution 516
 - syntax 515
 - events 423–425
 - role of in methods 484–486
 - Events page
 - Form Properties window 262, 485
 - Object Properties window 399, 423, 485
 - using the 423
 - Events tab
 - in Object Properties window 267
 - exclamation point (!)
 - in entry filter codes 336
 - Exclude Forms (Find Editor option) 82

- Exclude Methods (Find Editor option) 82
 - Excluded check box 335
 - excluded list 588
 - Execute Method dialog box 617
 - for starting a new process 620, 624
 - Execute Method menu item 620
 - in User environment 483
 - executing
 - methods 492–495
 - object methods 459
 - Exit Design Environment menu item 32
 - File menu 51
 - Explorer 54–68
 - adding a command to a method 67
 - adding a constant to a method 65
 - adding a field with 328
 - adding fields to forms 58
 - adding form name to method 61
 - adding method name to a method 64
 - adding subform to form 61
 - adding table names and field names to methods 58
 - Commands page 66–67
 - Constants page 65, 425
 - creating a new method 62–63
 - creating a relation using 166–167
 - creating new form 60
 - creating relations using 58
 - Delete, New, and Edit buttons 57
 - deleting a form 60
 - designating current input or output form 61
 - Displaying comments 69
 - displaying the 54
 - editing a form 60
 - Forms page 60–61, 226, 250
 - Input Form check box 226
 - Lists page 68
 - Methods page 62–64, 504
 - naming a form using 251
 - Output Form check box 226
 - pages of 54
 - Preview area icon 56, 59
 - previewing a form 61
 - previewing a table image 59
 - renaming a method 64
 - renaming forms 227–228
 - resizing the 56
 - setting current input and output forms 225–226
 - Tables page 58–59, 149
 - using drag and drop 57, 514
 - using to add a constant to a method 425
 - using to name a form 252
 - using to rename objects 55
 - using to write methods 427
 - viewing a table image in Structure editor 59
 - Explorer menu item
 - Tools menu 54
 - Explorer preview area
 - showing and hiding 56
 - Extended ASCII characters
 - Directly sending (Web Server) 102
- ## F
- Fast Mode (new Index mode) 151
 - field
 - adding with the Explorer 328
 - removing from form in Form Wizard 205
 - field attributes 141–147
 - Can't Modify 141
 - Choices 147
 - Display Only 141
 - Enterable 332–333
 - Indexed 142
 - Invisible 143–144
 - Mandatory 141, 332–333
 - Text with Scroll Bar 352
 - Unique 143
 - field display formats
 - See display formats
 - field entry filters. See entry filters
 - field length
 - setting the maximum 133
 - Field names
 - Dynamic 329
 - field names 114, 133
 - changing 149
 - duplicate 133
 - maximum number of characters 133
 - reserved words in 133
 - truncated 133
 - valid 133
 - field properties
 - changing 150
 - modifying 149
 - saving 134
 - Field Properties menu item 149
 - Field Properties window 132, 135, 148, 331

- assigning a choice list 333
- Attributes page 133
- Can't Modify attribute 331
- Choices & Help page 147
- Color page 148
- Display Only attribute 331
- entering help text 147
- for establishing data entry controls 331
- Indexed attribute 331
- Invisible check box 143
- Mandatory attribute 331
- modifying fields 135, 149
- setting data entry controls 331
- setting field attributes 141
- setting field types 136–140
- Unique attribute 331
- field types 135–140
 - Alpha 136
 - Blob 139
 - Boolean 138
 - changing 149–150
 - Date 137
 - Integer 137
 - Long Integer 137
 - Picture 138
 - Real 137
 - role of 131
 - Text 136–137
 - Time 138
- fields 114, 203, 205
 - adding to forms 319
 - as active objects 192
 - assigning a color to 134
 - assigning choice lists 335
 - data entry controls for 331–357
 - deleting 132
 - display formats for 357–371
 - duplicate 133
 - embedding in text areas 456
 - foreign key 154
 - help for 134, 353
 - indexed 142
 - inserting in forms, shortcuts 457
 - invisible 143–144
 - Invisible attribute 132
 - maximum number of 118, 131
 - modifying in forms 328
 - naming 133
 - object locking 135
 - placing on forms 326–328
 - primary key 154
 - related 154–155
 - reordering in a group 205
 - reordering in Form Wizard 202
 - selecting for forms 198, 199–202
 - selecting for output forms 447
 - standardizing entries for 147
 - types of 135–140
 - ungrouping 329
 - using object methods with 422–428
- Fields list
 - in Method editor 509
- Fields page
 - in Form Wizard 206, 209
- File menu 22, 46–47
 - Close *EditorName* menu item 46, 51
 - Close Structure menu item 46
 - Database Properties menu item 46
 - Exit Design Environment menu item 32, 46
 - Open Database menu item 46
 - Page Setup menu item 46
 - Print menu item 46
 - Quit menu item 46
 - Revert to Saved menu item 47
 - Save *EditorName* menu item 47, 111
- Fill menu item 240, 305
- fill patterns
 - setting 305–306
- Find Editor
 - Object types to search 80
 - Scope of the Search 80
 - Searching Options 81
- First Page button 386
- First Record button 386
- fixed-frame printing
 - Picture fields 473
 - Text fields 474
- flags
 - picture button 392
- flat file databases 116
- flow lines 524
 - drawing in Flowchart editor 526–527
- flow of control structures 491
- flowchart components
 - Flow lines 524
 - Formulas 524
 - Step object 523
 - Test object 523

- Flowchart editor 38, 496, 523–529
 - creating formulas 525–526
 - creating steps and tests 525
 - drawing flow lines 526–527
 - example method 523
 - Flowchart palette 523
 - plumbing analogy 523
 - specifying a first step 527
 - using 523–529
- Flowchart editor window 528
- Flowchart menu 527
 - New Step 524
 - New Test 524
 - Set as Start 524
- font
 - used in Method editor 512
- font attributes
 - setting default 299, 301
 - setting default for form 301
 - setting in Form Wizard 206
 - setting with style sheets 222, 303
- Font menu 242
- Font page 303
 - Object Properties window 222, 303
- fonts/font size
 - default 242
 - for objects 242
- Footer area 470
 - explanation of 444
 - in output form 439, 441
 - in printed reports 453
 - in reports 470
- Footer control line 454
- Footer marker 446
- foreground and background color
 - specifying in Form Wizard 206
- foreign key field 158, 160, 166, 167, 169, 171
 - automatic assignment during data entry 162
 - definition of 154
 - in Form Wizard 201
- form
 - creating a new 60, 220
 - deleting a 60
 - designating as input or output 61
 - editing a 60
 - owner of 228
 - previewing in Explorer 61
- Form Editor
 - Choosing a form type 256
 - Display menu command 232
 - Duplicating multiple objects 291
 - Showing or Hiding Elements 232
- Form editor 37–38, 188, 229–231
 - alignment tools 280–282
 - compared to Form Wizard 196
 - creating custom labels 455–456
 - creating labels 455–456
 - customization options 196
 - elements of 230–243
 - Font menu 242
 - Form menu 239–240
 - grouping objects in 279
 - modifying form objects 265
 - Object menu 240–242
 - opening forms in 250–251
 - overview of 188
 - rulers in 231, 275–276
 - Style menu 243
 - Tools palette 235–238
- Form editor menus 238–243
- Form editor window 231
 - adding display pages 316
 - displaying border lines in 323
 - displaying forms in 251
- Form event constants 489
- Form event function 425, 489
- form events 423–425, 487–489
 - activating in Form Properties window 262–264
- form letters
 - creating 456–458
- Form menu 239–240
 - Define Grid menu item 284
 - Define Ruler Units menu item 276
 - Delete Page menu item 318
 - Entry Order menu item 239, 321
 - Form method menu item 239
 - Turn Grid On/Off menu items 239
- form method 262–264
 - creating a 62, 501
- Form method menu item 239
- form methods 481
 - disabling 504
 - for setting data entry controls 332
 - role of 480, 481
 - writing 487–489
- Form Objects
 - Splitters 416
- form objects

- aligning280–286
- assigning a color to307
- changing appearance of297–309
- copying292–293
- creating276–278
- deleting295
- duplicating290
- grouping278–279
- layering293–294
- moving268
- resizing270–273
- selecting265–268
- ungrouping279
- Form Properties
 - Form Type257
- form properties
 - events262–264
 - overview of244–246
 - resizing options260, 262
 - setting251–264
 - setting window margins260
 - sizing options260
 - Window Title257
- Form Properties menu item
 - Form menu239, 251, 577
 - Form scaling239
 - Ruler definition239
- Form Properties window251
 - Access drop-down list253
 - assigning access privileges577
 - assigning owner privileges577
 - Events page262
 - naming a form using251
 - Owner drop-down list253
 - Platform Interface drop-down list256
 - setting the Platform Interface with255, 256
- Form scaling
 - Fixed Ratio297
 - Macintosh to Windows platform296
 - Rescale Pictures297
 - Windows to Macintosh platform296
- Form Scaling dialog box296
- form template198
 - creating220
- Form Template drop-down list
 - in Form Wizard220
- Form Wizard60, 188, 265, 300, 454
 - adding a subform with217, 429
 - Basic screen194–195, 197
 - Buttons page319
 - creating mail-merge templates with457
 - overview of188
 - removing a field205
 - renaming a group204
 - reordering fields202
 - role of37
 - selecting fields in199–202
 - Subforms page199
 - Use Dynamic Field names330
 - used to create multi-page forms315
 - using197–202
- Form Wizard Advanced screens
 - overview of206–208
- Format drop-down menu
 - in Object Properties window357
- Formats and Filters page
 - Database Properties dialog box344, 345
- Formats and Filters tab
 - Database Properties window345
- Forms
 - Form Type256
- forms
 - active objects on326, 372
 - adding fields to319, 326–328
 - adding pages to315
 - alignment grid284–285
 - Break areas in443
 - control lines in442–446
 - copying objects on292–293
 - creating197–202
 - creating new197–202
 - custom labels455–456
 - custom mailing labels455–456
 - default, created by 4th Dimension190
 - definition of37
 - deleting226
 - deleting pages of318
 - Detail area in443
 - Footer area in444
 - group access to577
 - group ownership of577
 - Header area in442
 - input190
 - moving between pages316–317
 - multi-page314–319
 - naming251
 - object locking251
 - object locking while modifying230

- opening 250–251
 - opening from Method editor 519–520
 - output 190–191, 442, 444
 - placing fields on 326–328
 - printing pages of 323
 - printing subforms 471–473
 - role of 187
 - saving 323
 - using subforms 428–432
 - viewing pages in 323
 - Forms page
 - in Explorer 60–61, 226, 250
 - Formula editor
 - for writing Flowchart methods 526
 - formulas
 - creating for steps and tests 525–526
 - Frames Assistant (Picture library) 607
 - functions 491
 - naming convention 495
 - role of 491
- ## G
- General page
 - Database Properties dialog box 83–98, 497
 - GOTO PAGE command 319
 - used with Tab controls 410
 - graph area 410
 - graphic objects
 - changing appearance of 297–309
 - role of 192–193
 - specifying border patterns 306–307
 - specifying fill patterns 305
 - specifying line width 304
 - graphics, on labels 455–456
 - greater than (>) character
 - in embedded fields and variables 457
 - Grid
 - Displaying 233
 - Grid Definition dialog box 284
 - grid, alignment. *See* alignment grid
 - grids
 - creating 419–421
 - group
 - renaming in Form Wizard 204
 - reordering fields 205
 - Group Box tool/icon 237
 - Group menu item 241
 - Object menu 279
 - Group Owner
 - assigning 563, 569
 - grouping fields
 - in Form Wizard 203–205
 - grouping in Form Wizard 203–205
 - grouping objects 278–280
 - groups
 - access to plug-ins 580–581
 - assigning access privileges to 557
 - assigning to database objects 575–581
 - assigning to Design environment 563
 - assigning to forms 577
 - assigning to menu items 579
 - assigning to methods 578
 - assigning to record operations 575–577
 - assigning to table definitions 575–577
 - assigning users to 570–571
 - creating 569–570
 - in access hierarchy 560–561
 - modifying 570
 - nesting 560, 572–574
 - ownership of forms 577
 - removing users from 571
 - saving and loading 574–575
 - Groups list
 - adding users 570–571
 - displaying user names 567
 - removing users 571
- ## H
- Header area
 - adding custom graphics to 452
 - additional levels of headers 470
 - explanation of 442, 470
 - in output form 439, 441
 - in reports 470
 - Header areas
 - additional 442
 - Header control lines 454
 - deleting 464
 - in reports 462
 - help
 - adding to fields 134, 147, 354–357
 - adding to objects 353–357
 - deleting a message 356
 - editing a message 355
 - entering text of 355
 - selecting a message 356

- help message
 - deleting 356
 - editing 355
 - in Field Properties window 147
 - renaming 355
 - selecting 356
- Help page
 - Object Properties window 355
- Help tab
 - in Object Properties window 354
- hexadecimal
 - entering in the Method editor 511
- hexadecimal numbers
 - display formats for 362
- Hide icon 629
- Hide Keywords check box
 - in Database Properties 91
- Hide Keywords option
 - in Database Properties dialog box 510, 521
- hierarchical lists 400, 583
 - creating 590–591
 - expanding and collapsing 55
- hierarchical pop-up menus 400
- Highlight buttons 383
- Home page by default 100

- I**
- icon
 - associating with a list item 583, 592–593
- Include 4D Passwords (Web Server) 101
- Index
 - Fast Mode 151
 - Traditional Mode 151
- Indexed attribute 142
 - adding 150
 - Alpha fields 136
 - assigned to primary key field 164
 - in Field Properties window 331
 - in specifying table relations 154
 - related fields 159
- indexes
 - automatic updating of 142
- Indexing process 616
- indicators
 - assigning values to 416
 - dials 413
 - establishing settings of 413
 - getting values from 416
- rulers 413
- thermometers 413
- using methods 415
- initiators
 - ampersand (&) character 340–341
 - in entry filter codes 340
 - tilde (~) character 340–341
- input form
 - defined 187
- Input Form check box
 - in Explorer 226
- INPUT FORM command 225, 251
- input forms 190
 - lists in 585
 - multi-page 314–317
 - role of 189
 - setting the current 225–226
- Insert an OLE Object menu item 242
- Inserting and Deleting Frames (Picture library) . 609
- instances
 - of menus 543
 - See also connected menus
 - using to manage menus 532
- Integer fields 137
- interface objects 325
- intermediate tables
 - for Many to Many relations 175, 175–181
- Invisible attribute 143–144
- Invisible buttons 384
- Invisible check box
 - in Table Properties 128–129
- invisible fields 143–144
- invisible methods 504
- invisible tables 124, 128
- Items menu 594
 - Delete menu item 592
 - New menu item 589
 - Sort menu item 594

- J**
- justification
 - setting for text objects 303

- K**
- Keep list of users sorted
 - in Password dialog box 88
- key combinations, reserved 550

- keyboard equivalents
 - adding for buttons 379–380
 - adding for check boxes 379–380
 - adding for menu items 550–551
 - reserved combinations 550
 - Keyboard navigation
 - Methods editor 522
 - keyboard shortcut
 - assigning to active objects 379
 - Keys button
 - on Variable page of Object Properties window 379
 - Keywords divider
 - in Method editor 521
 - Keywords list 509
 - hiding in Method editor 91
 - in Method editor 513
- L**
- Label editor
 - using label forms 477
 - label reports
 - creating 475–478
 - creating custom labels with graphics . . . 455–456
 - printing 475–478
 - setting label width 475
 - setting margins 475–476
 - label width marker 231, 475
 - Labels menu item 455, 477
 - language elements
 - assignment operator 494
 - commands 490
 - fields 490
 - flow of control structures 491
 - functions 491
 - object names 490
 - pointers 490
 - statements 491
 - variables 490
 - Large Brace Matching menu item 513
 - Last Page button 386
 - Last Record button 386
 - layering
 - Move to Back menu item 293–294
 - Move to Front menu item 293–294
 - Layering tools/icons 238, 293
 - leading zero 360
 - Leave Related Many Intact 164, 168
 - in Relation Properties window 163
 - less than (<) character
 - in embedded fields and variables 457
 - over maximum number of digits 360
 - Limits
 - Displaying 234
 - Line check box
 - in Menu Bar editor 549
 - line width
 - setting 304
 - Line Width menu item 240, 304
 - List editor 42, 584–596
 - creating Choice lists 147
 - creating lists in 588–592
 - Current List area 589
 - deleting items from 592
 - deleting lists from 592
 - Items menu 588
 - List of Lists 588
 - Lists menu 588
 - role of 42
 - sorting list items 594
 - List Editor menu item
 - Tools menu 588
 - list form
 - created in Form Wizard 440–441
 - defined 187
 - for subrecords 430
 - viewing a record from 187
 - List of Menu Bars
 - in Menu Bar editor 535
 - list of open windows
 - in Design menu 51
 - List of Tables window 477
 - LIST TO ARRAY command 352, 398, 410
 - Listing editor 38, 496, 507–509
 - advantages of 497
 - Editing area 508
 - Fields list 509
 - finding text 516–518
 - Goto Line menu item 518
 - Keywords list 509
 - Method menu 516–518
 - replacing text 516–518
 - Routines list 509
 - syntax error checking in 515–516
 - typographic conventions 511
 - using 508–518
 - Listing editor window 507–509
 - resizing panes in 521

- lists
 - adding items 589
 - attaching to fields 147, 585
 - choice 333–335
 - creating 588–589
 - deleting 592
 - deleting items from 592
 - excluded values 587
 - making modifiable 595–596
 - maximum and minimum values 587
 - non-sequential ranges 587–588
 - object locking 584
 - required 335
 - required values 587
 - role of 583–584
 - sorting items 594
 - specifying ranges in 594
 - used in methods 584
 - Lists menu
 - Delete menu item 592
 - New menu item 588
 - User Modifiable menu item 596
 - Lists page
 - in Explorer 68
 - Load Groups menu item
 - Passwords menu 575
 - load record operations
 - access to 575
 - logical fields. *See* Boolean fields
 - Long Integer fields 137
 - look-ups. *See* wildcard (@) character
- ## M
- Macintosh
 - desktop filenames 27
 - pathnames to desktop files 29
 - reserved key combinations 550
 - scrolling 52
 - mail-merge
 - using text areas for 299
 - mail-merge documents. *See* form letters
 - Managing splitters procedurally 417
 - Mandatory attribute 141, 333
 - in Field Properties window 331, 333
 - setting 141, 332–333
 - Mandatory check box
 - in Object Properties window 333
 - manual relations 155, 170
 - establishing 168
 - multiple link relations 183
 - reasons for using 170
 - Many table 156–158
 - defined 156–158
 - deleting records from 163
 - displaying fields from 218
 - removing relations 169
 - Many to Many relations 174–181
 - creating 175
 - entering data in 176–177
 - intermediate tables 175–181
 - using intermediate tables 175
 - Many to Many structures
 - creating reports 180–181
 - Many to One properties 160–161
 - Many to One relations 156–158
 - Auto Assign Related Value 162
 - creating 165–168
 - entering data with 170–171
 - Many table 157
 - One table 157
 - margins
 - setting in Form Properties 260
 - margins, setting in label reports 475–476
 - Marker Labels
 - Displaying 234
 - Markers
 - Displaying 234
 - marquee
 - for selecting objects 268
 - for setting data entry order 322
 - master table
 - defined 189
 - selecting fields from 199
 - master-detail relationship 190
 - maximum and minimum values 348
 - with lists 587
 - Maximum Size box
 - in Data Segment Management dialog box ... 636
 - measurement scale
 - setting for form rulers 275–276
 - measurement scale, setting for form rulers 276
 - Menu Bar editor 40–41, 532, 534, 535
 - adding menu items 538–539
 - adding menus 536–538
 - adding separator lines 548–549
 - assigning access privileges using 579
 - assigning keyboard equivalents 550–551

- changing menu item font style 547
- creating connected menus 543–544
- creating menus 532
- deleting connected menus 545–546
- Enabled check box 548
- enabling and disabling menu items 548
- enhancing menus 546–548
- Menu list 618
- modifying connected menus 544–545
- Start a New Process check box 542
- starting processes from 542, 618–620
- Menu Bar Editor menu item 534
 - Tools menu 579
- menu bars
 - creating 40, 534–536
 - in custom applications 554
 - object locking 533
 - previewing 553–554
 - See also* menus
- Menu commands
 - See also* menu items
- menu items
 - access privileges to 579–580
 - adding 538–539
 - adding keyboard equivalents to 550–551
 - adding separator lines to 548–549
 - assigning groups to 579–580
 - assigning methods to 541–542
 - deleting 552
 - enabling and disabling 548
 - font style for 547–548
 - rearranging 540
- menus
 - adding 536–538
 - adding keyboard equivalents 550–551
 - adding menu items 538–539
 - connected 536
 - creating 533–539
 - creating connected 543
 - deleting 552
 - deleting connected 545
 - deleting menu items 552
 - designing 531–532
 - example of custom 531
 - for custom applications 554
 - instances of 543
 - length of title 538
 - modifying connected 544–545
 - object locking 533
 - previewing 553
 - rearranging 540
 - reusing 543
- method
 - creating a new 62–63
 - modifying a 63
 - previewing a 63
 - renaming 64
- Method editor 38–39, 496
 - Flowchart editor 496, 523–529
 - Listing editor 496, 507–521
 - showing or hiding keywords in 91
 - specifying font and font size 512
 - specifying the default 91
 - typographic conventions 511
- Method editor window 427
- Method menu 517
 - Find menu item 517
 - Find Next menu item 517
 - Goto Line menu item 518
 - locating text strings 516
 - Replace menu item 517
 - Replace Next menu item 518
- method name
 - adding to a method 64
- Method Properties dialog box 502–503, 518, 579
- Method Properties menu item
 - Method menu 518, 578
- Method Type dialog box 426, 498
- Methods
 - Inserting a command with its syntax 514
- methods
 - arguments in 495
 - assigning access to 578–579
 - assigning colors to elements 518–519
 - assigning to menu items 533, 541–542
 - assignment operator (:=) in 492, 494
 - creating 496
 - creating with Flowchart editor 524
 - definition of 38
 - elements of 489–491
 - example 493–495
 - execution of 492–495
 - Flowchart editor 38, 496, 523–529
 - for indicators 415
 - invisible 504
 - Listing editor 38, 496, 507–521
 - naming 499
 - object locking 504

- opening 504
- opening from Method editor 519–520
- operators in 490
- renaming 502, 518
- renaming with 4D Server 502
- role of 479
- selecting components 513
- statements in 491–492
- syntax errors 515
- types of 480–484
- typing hexadecimal in 511
- typing text in 511
- using process variables 493
- using variables in 493
- writing 511
- Methods Editor
 - Shortcuts 522
- Methods page
 - Explorer 62–64, 486, 498, 499, 500, 502, 504
- MiniFans
 - splash screen for 553
- minimum height
 - of hierarchical list 597–598
- modifying
 - connected menus 544–545
 - field properties 149
 - subforms 438
- Move to Back menu item 241, 293–294
 - setting data entry order 322
- Move to Back tool/icon 293, 322
- Move to Front menu item 241, 293–294
- Move to Front tool/icon 294
- multi-page forms 314–319
 - adding page navigation tools 319
 - background page 309
 - creating with Form Wizard 315
 - printing 192, 314
- multiple processes 190, 613–615
 - current record 615
 - current selection 615
 - multiple active windows 615
 - multiple files 615
 - multiple forms 615
- multiple table structures 116–118
- N**
- negative numbers
 - displaying 360
- formatting 361, 363
- nesting
 - password groups 572–573
- New Child menu item
 - Items menu 590, 591
- New Field menu item
 - Structure menu 121, 132, 135
- New Form menu item 197, 446
 - Design menu 60
- New memory allocation scheme 93
- New Method dialog box
 - for creating project methods 499
- New Method menu item 498
- new process
 - starting from Execute Method dialog box ... 620
- New Process check box
 - in Execute Method dialog box 621
- New process function 542, 617, 624
- New Table menu item
 - Structure menu 124
- New User menu item
 - Passwords menu 567
- Next Page button 386
- NEXT PAGE command 490
- Next Page tool/icon 317
 - Tools palette 316
- Next Record button 386
- No Action button 385
- No Brace Matching menu item 513
- non-enterable objects 382
- number (#) sign
 - as a placeholder 360
 - in entry filter codes 336, 342
- number display, as progress indicator 84
- Number field formats 359–364
 - displaying numbers as times 362
 - displaying symbols and characters 361
 - examples 363
 - placeholders 360
 - positive, negative, and zero formats ... 361–362, 363–364
 - scientific notation 362
 - using custom formats 363
 - using decimal points 361
- Number fields
 - formats 359–364

O

- object events 487–489
- Object List 267
 - choosing another object with 303
- object locking
 - definition of 111
 - of table definitions 129
 - while modifying forms 230, 251
 - while setting database properties 82
 - with database objects 111
 - with field properties 135
 - with menus 533
 - with methods 504
 - with object methods 504
- Object menu 240–242
 - Align menu item 241
 - Align to Grid menu item 241, 286
 - Border menu item 241
 - Clear Object Method menu item 242, 428
 - Color menu item 241
 - Duplicate menu item 241, 290
 - Duplicate on Matrix menu item 241
 - Fill menu item 240
 - Group menu item 241, 279
 - Insert an OLE Object menu item 242
 - Line Width menu item 240
 - Line Width submenu 304
 - Move to Back menu item 241, 293, 294
 - Move to Front menu item 241, 293, 294
 - Object Method menu item 242
 - Show Format menu item 242
 - Show Resource menu item 242
 - Ungroup menu item 241, 280
- Object Method button
 - in Object Properties window 426, 428
- Object Method menu item 242
- object methods 422–481
 - attaching to fields or objects 495
 - clearing 507
 - creating 497–498
 - data entry management 481
 - deleting 428, 507
 - examples with fields 422
 - for controlling data 382
 - in output forms 444
 - in report with breaks 470–471
 - modifying 427
 - object locking 504
 - opening 505
 - placing 495–496
 - role of 480, 480–481
 - uses of 422
 - viewing 267, 427
 - with fields and objects 422–428
 - writing 487–489
- object names
 - referring to in methods 490
- Object Properties
 - Displaying 233
- object properties
 - overview of 193
 - viewing 266
- Object Properties window 222, 303, 306, 326, 372–374
 - Colors page 304, 306, 308
 - Coordinates page 269, 274
 - creating object methods with 480
 - Data Control page 335, 348, 349, 357, 366
 - Display page 299, 388
 - Enterable attribute 333
 - Events page 399, 423
 - Font page 303
 - Help page 355
 - lists, *See* choice lists
 - Mandatory attribute 332–333
 - Mandatory check box 333
 - print frame options 471–473
 - Print Variable Frame option 474, 475
 - setting data entry controls 378
 - setting display formats 376
 - setting entry filters 335–344
 - setting font attributes with 302
 - setting keyboard shortcuts 379–380
 - specifying a custom format 366
 - Subform page 430, 438
 - used to set appearance option 222
 - using custom formats in 363
 - Variable page 379, 402
- objects
 - 3D radio buttons 388–390
 - active 192–193, 381–416
 - aligning 280–285
 - assigning choice lists to 335
 - border patterns 306–307
 - button grid 401–402
 - buttons 382–387, 388–390
 - changing the appearance of 297–309

- check boxes 387–388
- combo box 399
- copying on forms 292–293
- creating active 374–380
- creating in a series 419–421
- creating with Tools palette 278
- deleting 295
- deselecting 267
- dials 413–416
- display formats for 376
- drawing 276–278
- duplicating 290
- enterable 382
- fill patterns for 305–306
- graph area 410
- graphic 192–193
- grouping 278–280
- hierarchical list 400
- hierarchical pop-up menu 400
- layering 293–294
- moving 268
- non-enterable 382
- nudging 269
- picture menu 403–406
- plug-ins 411–413
- pop-up menus 397–399
- radio buttons 388–390
- radio pictures 388–390
- resizing 270–273
- resizing handles 266
- rulers 413–416
- scrollable areas 397–399
- selecting 265–268
- selecting multiple 268
- setting color of 307
- setting line widths of 304
- tab control 407–410
- thermometers 413–416
- ungrouping 279
- using object methods with 422–428
- Objects menu
 - Show Name menu item 242
- OLE_Tools commands 510
- On Activate event 488
- On After Keystroke event 424, 488
- On Before Keystroke event 424, 488
- On Clicked event 401, 424, 487
- On Close Box event 488
- On Close Detail event 488
- On Data Change event 424, 488
- On Deactivate event 488
- On Display Detail event 424, 488
- On Double Clicked event 424
- On Double-clicked event 488
- On Drag Over event 424, 488
- On Drop event 424, 488
- ON ERR CALL method 559
- On Event Manager process, 616
- On Getting Focus event 424, 488
- On Header event 424, 489
- On Load event 410, 424, 487
- On Load form event
 - used to initialize an object 398
- On Losing Focus event 424, 488
- On Menu Selected event 488
- On Open Detail event 488
- On Outside Call event 424, 487
- On Plug-in Area event 424, 488
- On Printing Break event 424, 489
- On Printing Detail event 424
- On Printing Details event 489
- On Printing Footer event 424, 489
- On Resize event 489
- On Serial Port Manager process, 616
- On Timer event 424, 489
- On Unload event 424, 487
 - clearing an array with 410
- On Validate event 424, 487
- One table 156–158, 160
 - defined 156–158
 - deleting records in 163
 - entering data in 170–171
- One to Many properties 161–162
- One to Many relations
 - See also* Many to One relations
- One to One relations 174
- One-to-Many relations
 - using subforms in 428
- On-line Help 264, 642
 - Calling from a 4D Database 643
- Open Data File dialog box 28–29
- Open menu item 22, 24
- Open Subform button 386, 431
- Open window function 260
- opening
 - databases 24
 - object methods 505
- opening forms

- from Method editor 519–520
 - opening methods
 - from Method editor 519–520
 - operators
 - in methods 490
 - Option-clicking
 - to open object method 426
 - Options page
 - in Form Wizard 207
 - ORDER BY command 490
 - output control area
 - removing 446
 - output control lines 231, 293, 440
 - definition of 442–444
 - enabling 444
 - moving 445
 - output control markers 440, 445
 - in Form editor 445
 - output form
 - defined 187
 - Output Form check box
 - in Explorer 226
 - OUTPUT FORM command 225, 251
 - output forms 190–191
 - control lines 442–444
 - creating mail-merge documents 456–458
 - designing labels with 455–456
 - for reports 453–454
 - modifying 452–453
 - setting the current 225–226
 - uses of 439
 - Owner
 - setting for forms 578
 - setting for groups 563
 - setting for methods 578, 579
 - setting for objects created by user 568
 - Owner drop-down list
 - Form Properties window 578
 - function of 502
 - in Form Properties window 253
 - in Method Properties window 579
 - owner privileges
 - for forms 253, 577
 - for groups 563
 - for methods 578
- P**
- Page 0 objects
 - Displaying 233
 - page border lines 323
 - Page buttons 386
 - page navigation actions 386
 - page navigation tools 319
 - Page pop-up menu 318
 - Page Setup dialog box 323, 460
 - page tools/icons 238
 - Paper
 - Displaying 233
 - parent records
 - in subtables 139
 - parent tables 118
 - parentheses ()
 - as dead characters 343
 - Password Access editor 41, 567
 - assigning users and groups 566–569
 - changing access groups with 570
 - changing user name or password 569
 - Groups list 567
 - role of 41, 555
 - Users list 567
 - viewing database usage 582
 - password access system 556–582
 - adding users 567–569
 - Administrator 562
 - assigning groups to forms 577
 - assigning groups to menu items 579–580
 - assigning groups to methods 578–579
 - assigning groups to plug-ins 580–581
 - assigning groups to record operations 575–577
 - assigning groups to table definitions 576–577
 - assigning passwords to users 568
 - assigning users to groups 570–571
 - creating access groups 569–570
 - creating an access hierarchy 560–561
 - Designer 561
 - error messages 559
 - group owners 563
 - initiating 566
 - maintenance of 581
 - nesting groups 572–573
 - object locking in 4D Server 556
 - removing nested groups 573
 - removing users from groups 571
 - role of 556
 - saving and loading groups 574–575
 - user hierarchies 572
 - viewing usage of 582

- Passwords
 - Access to the User mode 564
 - Default User 565
- passwords
 - case sensitivity 568
 - changing 569
 - creating 568
 - entering 557–558
- Passwords menu 567
 - Edit Group menu item 570
 - Edit User menu item 569
 - Load Groups menu item 575
 - New Group menu item 569
 - New User menu item 567
 - Plug-ins Access menu item 580
 - Save Groups menu item 574
- Passwords menu item
 - Tools menu 566, 580
- Paste menu item 47, 293
- paths
 - to data files 29
- Pause icon 627
- picture buttons 390–397
- Picture check box
 - in List editor 593
- Picture field formats 369–371
- Picture fields 138
 - formats for 369–371
 - modules in 138
 - on background 371
 - plug-in areas in 138
 - printing 473–474
 - scaling to fit 370
 - storing data from plug-ins in 139
 - truncating 370
- Picture Library
 - Adding Pictures 603
 - Frames Assistant 607
 - Graphic tools 606
 - Shortcuts 606
- Picture library 43–44, 310, 390, 405, 601–612
 - creating custom toolbar with 546
 - for storing background graphics 372
 - placing a picture on a form 309–311
 - storing small icons in 592
 - using to add pictures to forms 196
- picture menu 403–406
- Pictures
 - Adding 603
 - Editing 606
 - Picture Library (modification) 601
 - Properties 604
 - Viewing 602
- placeholders
 - asterisk (*) 360
 - caret (^) 360
 - in Alpha field formats 365–366
 - in entry filters 340, 342
 - in number field formats 360
 - number (#) sign 360
 - underline () character 342
 - zero 360
- Platform Interface
 - Automatic option 105, 255, 298
 - configuring client stations 109
 - creating style sheets for 224
 - for forms 105
 - for objects 105, 298
 - Inherit from Database option 255
 - Inherited from Form option 298
 - Mac OS option 255
 - Macintosh option 298
 - objects affected by 106–110
 - Platinum option 255, 298
 - setting for forms 255–256
 - setting for text objects 302
 - specifying font attributes 222
 - specifying in Form Wizard 206, 221
 - Windows 95 option 255, 298
 - Windows NT 3.5.1 option 255, 298
- Platform Interface drop-down list
 - database properties 105–110
 - in Form Properties window 256
- plug-in objects 411–413
- plug-in routines
 - listed in Method editor 510
- plug-ins
 - assigning access to 580–581
 - assigning groups access to 580
 - using 412
- plumbing
 - in a house 523
- pointers
 - arrow 266
 - crosshair/crossbar 277
 - entry order 321
 - multi-directional arrow 270
- pop-up menus

- initializing 350, 398–399
 - PostScript™ printer, printing graphics 138
 - power failures 29
 - Preview area icon 61, 63
 - in Explorer 56, 59
 - Preview on screen option
 - for printing reports 460
 - Previous Page button 386
 - Previous Page tool/icon 317
 - Tools palette 316
 - Previous Record button 386
 - primary key field 158, 160, 165, 166, 167
 - definition of 154
 - required attributes for referential integrity ... 164
 - Print Titles check box 91
 - Print Variable Frame check box
 - printing subform areas 472
 - printing text fields 475
 - print window titles 91
 - printing
 - form pages 323
 - labels 475–478
 - Picture fields 473–474
 - subform areas 471–473
 - Text fields 474–475
 - Privileges page
 - Table Properties window 577
 - Process
 - Runtime Explorer 76
 - process
 - elements of a 614
 - Process list 621–629
 - Process List editor 621–629
 - viewing processes 621–626
 - Process List editor window 622
 - Process menu 627
 - bringing a process to the front 629
 - pausing a process 627
 - resuming execution of a process 627
 - tracing a process 628
 - Process Page (Runtime Explorer) 77
 - processes
 - aborting 623, 627–628
 - bringing to front 629
 - controlling execution of 626
 - debugging 628
 - default 615
 - default names of 624
 - definition of 613
 - execution of multiple 616
 - for managing a multi-window interface 613
 - hiding 628
 - multiple 613–615
 - naming 623–624
 - pausing and resuming 627
 - process number 623
 - process status 624–625
 - process time 626
 - role of 613
 - specifying when to execute 617
 - Start a New Process check box, in Menu Bar editor 619
 - starting from a menu item 618–620, 624
 - starting using Execute Method 620–621, 624
 - starting with a method 617–618, 624
 - tracing 628
 - progress indicator
 - choosing a 84
 - project methods
 - assigning to menu items 533, 541
 - creating 62, 498–499
 - renaming 502
 - role of 480, 483–484
 - Prompt if Related One does not exist check box
 - in Relation Properties window 160, 171
 - properties
 - database 82–95
 - of subforms 438
 - Property List
 - Displaying 233
 - Object Types 329
 - Publish Database as Web Server property 98
- ## Q
- Queries (new Index mode) 151
 - Query editor 331, 333
 - quick reports
 - creating with Many to Many relations 180
 - quotation marks ("), in entry filters 341
- ## R
- radio buttons 388–390
 - formatting Boolean fields as 138
 - naming 389
 - radio pictures. *See* radio buttons
 - range of characters

- in filter arguments341–342
- ranges, in lists587–588
- Real fields137
- Record Access area
 - table properties127, 577
- record navigation actions386
- records114
- recovering by tags
 - using the Completely Delete option130
- Reference ID area
 - in List editor593
- referential integrity163, 332
 - Can't Modify attribute164
 - enforcing86
 - Indexed attribute164
 - Unique attribute164
- Related Fields area
 - in Relation Properties window159
- related One table
 - selecting fields from200
- related records, deleting163
- related tables152–154
 - Can't Delete if Related Many option163
 - creating table relations158–168
 - Delete Related Many option163
 - deleting records in163–164
 - entering data in170–171
 - example of153
 - Leave Related Many Intact option163
 - One table and Many table156–158
 - related fields154–155
 - relation types174–181
 - unusual relation types174
 - wildcard look-ups in171–173
- relating tables158–168
- relation line
 - drawing the160
 - setting color of168
- relation properties
 - modifying169
 - setting158–164
- Relation Properties window166
 - Color page168
 - Control page168
 - for enforcing referential integrity332
 - modifying relations using169
 - overview of159
 - Related Fields area159
- relation types
 - Many to Many174–181
 - Many to One156–158
 - One to One174
- relational databases
 - defined117
- relations
 - analyzing181–185
 - between tables31, 153
 - circular181–182
 - creating158–168
 - displaying117
 - foreign key field154
 - multiple links182–183
 - multiple record184–185
 - primary key field154
 - re-establishing169
 - removing169
 - See also* related tables
- renaming forms227–228
- renaming forms or methods55
- renaming methods64
- reports
 - break levels in460–462
 - columnar453–454
 - creating458–460
 - creating custom labels455–456
 - creating from Many to Many structures180–181
 - creating label475–478
 - initiating break processing464
 - object methods470–471
 - one record per page454
 - subforms in454
 - subtotals in465–466
 - summary467
 - Text fields in454
- reports with breaks
 - example of468–469
- repositioning options
 - automatic273
 - in Object Properties window275
- required lists335
- reserved key combinations550
- reserved words
 - in field names133
- Resizable check box
 - in Form Properties window262
- resizing
 - automatic273
 - forms260

- grouped objects 278
 - table images 121
 - resizing handles
 - resizing objects 270
 - selecting objects 266
 - resizing objects
 - with the arrow keys 271
 - with the resizing handles 270
 - resizing options
 - Form Properties window 262
 - Resume icon 627
 - Revert to Saved menu item
 - File menu 323
 - Routines list 509
 - Ruler Definition dialog box 276
 - Rulers
 - Displaying 233
 - rulers 275–276, 413–416
 - assigning values to 416
 - changing units of 231, 275
 - example of 413
 - getting values from 416
 - setting measurement scale 276
 - with output control markers 445
 - Rulers menu item 275
 - Runtime Explorer 75
 - Break page 78
 - Catch page 78
 - Process page 77
 - Watch page 76
- S**
- Save *EditorName* menu item 111
 - Save *FormName* menu item 323
 - Save Groups menu item
 - Passwords menu 574
 - save operations
 - access to 576
 - save record operations
 - access to 575
 - saving
 - forms 323
 - password access groups 574–575
 - Scale command
 - Form menu 296
 - Scaled to Fit picture formats 370
 - scientific notation, in number field formats 362
 - Scrapbook
 - storing copied objects 292
 - scroll bars 52
 - adding to Text fields 137, 352
 - scrollable areas
 - initializing 398–399
 - scrolling 52
 - in forms 323
 - table images 121
 - Searching in the Database 79
 - Searching Options (Find Editor) 81
 - security. *See* password access system
 - segmenting
 - data files 631–640
 - existing data files 634–636
 - new data files 632–634
 - Select a Menu dialog box
 - in Menu Bar editor 543
 - Select All menu item 268, 293
 - in Edit menu 47
 - Selectable check box
 - for subforms 431, 436
 - Selected Process (Runtime Explorer) 76
 - selecting
 - fields for forms 199–202
 - method components 513
 - objects 265–268
 - Selecting a different data file 26
 - selection rectangle. *See* marquee
 - semicolon (;) character
 - in number field display formats 362
 - specifying display formats for embedded fields and variables 457
 - separator lines, adding to menu items 548–549
 - sequence numbers
 - as default values 350
 - generating as default values 350
 - in related tables 155
 - Sequence of Frames (Picture library) 608
 - series
 - creating objects in 419–421
 - Set as Start tool/icon 527
 - SET INTERFACE command 106
 - Shift key
 - to constrain object shape 277
 - Shift+Alt clicking
 - for choosing tables and fields 458
 - Shift-clicking
 - for selecting objects 268
 - Shift-dragging

- moving output control lines 445
- Shortcuts
 - Inserting Picture Buttons 611
 - Inserting Pop-up Menu 611
 - Method Editor 522
 - Picture Library 606
- shortcuts
 - inserting fields in text areas 457
 - keyboard, for buttons and check boxes 379
- Shortcuts dialog box 380
- Show Clipboard menu item 47
- Show Custom Menus menu item
 - Menu men 553
- Show Focus property 379
- Show Format menu item 242
- Show Name menu item 242
- Show Resource menu item 242
- Show Toolbar property
 - in Database Properties dialog box 84
- Showing/Hiding Elements in the Form Editor . . 232
- single-table structures 115
- Size Based On drop-down list
 - in Form Properties window 261
- sizing options
 - for forms 260
- Small Brace Matching menu item 513
- Sort menu item
 - for lists 594
- sorting records
 - for break processing 462
- spaces
 - as dead characters 343
 - in number display formats 361
- Special menu
 - for executing methods 620
- Splash Screen area 553
- splash screens
 - adding 553–554
- Split button
 - in Create a Data File dialog box 633
 - in Data Segment Management dialog box . . . 640
- Splitters
 - Automatic 418
 - Examples 418
 - Interaction with neighboring objects 417
- Start a New Process check box
 - in Menu Bar editor 542, 619
- starting 4th Dimension 22
- Startup environment
 - setting the 83
- startup method
 - assigning to a user 568
- STARTUP procedure 92
- statements
 - in methods 491–492
- Step tool/icon 525
- structure access
 - allow 87
- Structure Access drop-down list
 - in Database Properties dialog box 87, 563
- Structure editor 37, 113
 - creating fields 132–134
 - creating tables 124–130
 - definition of 120
 - renaming tables 127
- structure files 26
- Structure menu
 - creating fields 132
 - creating tables 124
 - displaying table properties 125
- Structure window 37
 - Contextual menus 50, 243
 - creating table relations 165–168
 - for new databases 24, 120
 - reestablishing relations 169
 - removing relations 169
 - showing relations 168–169
 - table images in 24, 120
- Style menu 243
- style sheet
 - applying a 225
 - renaming 224
 - setting font attributes with 303
- Style Sheet Definition dialog box 223–224
- Style Sheet drop-down list
 - in Form Wizard 224
- Style Sheet editor 222–225
 - accessing 222
- style sheets
 - creating 222–224
 - editing 90
 - in Form Wizard 207
- Styles page
 - Appearance drop-down menu 221
 - in Form Wizard 206, 221, 222
- subfields 139
- reasons not to use 140
- subform

- adding to a form 61, 217
 - creating using Form Wizard..... 196
 - defined 190
 - entering data into a 170
 - specifying in Object Properties window 430
 - Subform page
 - in Form Wizard 208
 - Object Properties window 430, 438
 - subform properties 438
 - Subform tool/icon 237
 - subforms 177–180, 428–438
 - Auto Assign Related Value 161
 - Automatic Width check box 437
 - buttons for 386, 431–432
 - creating 432–438
 - creating records in 180
 - displaying information in 177–180, 218, 432
 - entering data in 180, 430
 - modifying 438
 - printing 471–473
 - printing with fixed-frame option 473
 - printing with variable option 473
 - using 428–432
 - using default input forms 431
 - using multiple 429
 - with reports 454
 - Subforms page
 - Form Wizard 199
 - subrecords 118, 139
 - subtable
 - adding subform from 428
 - adding with Form Wizard 208
 - subtables 118–119, 139–140, 199
 - parent records in 118, 139
 - subfields in 139
 - subrecords in 139
 - when not to use 119
 - Subtotal function
 - initiating break processing with 463, 464
 - subtotals
 - computing in a report 470–471
 - summary calculations
 - in reports 439
 - printing 444
 - summary reports 467
 - syntax errors
 - checking and correcting 515
 - checking for 525
- ## T
- Tab control
 - as page navigation tool 319
 - with icons and text..... 408
 - tab control 407–410
 - Tabable property..... 379
 - table
 - definition of 115
 - table definitions
 - assigning access to 576–577
 - table image
 - moving 123
 - previewing in Explorer..... 59
 - resizing 121, 122
 - scrolling 121
 - setting the color of 130
 - viewing in Explorer 59
 - table methods
 - role of 480, 481–482
 - table name
 - setting the color of 130
 - Table names
 - Dynamic 329
 - table names
 - duplicate 127
 - maximum number of characters 127
 - updating in methods 127
 - table properties
 - setting 125–130
 - Table Properties menu item
 - Structure menu..... 121, 125
 - Table Properties window 125, 126–130
 - Completely Delete property..... 129
 - invisible table 128–129
 - Owner 127
 - Privileges page 577
 - Record Access area 127, 577
 - table relations
 - advantages of 153
 - See also* related tables
 - tables 115–118
 - canceling creation of 124
 - creating 124–125
 - displaying different 52
 - master 189
 - maximum number of 118
 - moving images 123
 - naming 127

- parent118
- relating152–168
- renaming.....127
- selected121
- selecting images of121
- subtables118–119
- Tables page
 - Explorer.....58–59, 121, 149, 328
- tags
 - recovering by.....129
- target width
 - specifying in Form Wizard451
- TCP Port
 - specify as Web server port.....98
- template.....198
 - creating a form220
- templates
 - in Form Wizard.....195
- Temporary Folder (Database properties).....94
- Test tool/icon525
- text
 - entering in methods.....511
 - finding and replacing in methods516–518
- Text Area tool/icon237, 301
 - for form letters457
- text areas
 - as graphic objects193
 - changing the appearance of299–304
 - creating on forms299–301
 - embedding fields and variables in ..299, 456–458
 - setting appearance302
 - setting default font attributes.....301
 - setting platform interface299
 - variable frame option471
- text attributes
 - setting for text objects302
- Text fields136–137, 299
 - adding scroll bars to352
 - in reports.....454
 - printing with variable frame.....474–475
- thermometers413–416
 - as progress indicators84, 413
 - example of.....413
 - setting properties of415
- tilde (~) character
 - forcing capitalization341
 - in entry filter codes.....336
 - initiating an entry filter340–341
- Time fields.....138
 - display formats for.....358–359
- time-sliced execution
 - of processes616
- toolbar
 - creating a custom546
 - Design environment53–54
- Tools menu49
 - Explorer item.....54
 - Menu Bar Editor menu item534
- Tools Palette
 - Displaying233
- Tools palette.....235–238, 276
 - Add Field tool/icon237, 327
 - Alignment tools.....238
 - alignment tools/icons281–282
 - Arrow tool/icon.....237, 266
 - Distribute tools286
 - drawing objects276
 - Duplication tool238
 - in Form editor235–238
 - Layering tools/icons238, 293
 - Next Page tool/icon.....316
 - Next Page/Previous Page tools/icons.....317
 - Object method deletion tool.....238
 - Previous Page tool/icon.....316
 - Subform tool/icon237
 - Text Area tool/icon237, 301
- totals
 - calculating in reports.....454
 - object methods for calculating470–471
- Trace icon628
- Traditional Mode (new Index mode)151
- trailing zero360
- triggers
 - creating62, 499, 499–500
 - disabling504
 - for enforcing business rules.....326, 332
 - role of.....480, 481–482
 - setting for a table.....128
 - writing486–487
- Triggers page
 - Table Properties window.....485
- truncated field names133
- truncated picture format.....370
- Tune Up page
 - in Database Properties dialog box.....92–93
- Turn Grid On/Off menu items239
- Type drop-down menu
 - for creating buttons.....385

- in Object Properties window 385
- U**
- underline (_), for placeholders 342
- Undo menu item 295
- unexpected interruptions 29
- Ungroup menu item 241, 280
 - Object menu 280
- Unique attribute 143
 - assigned to primary key field 164
 - changing field properties 150
 - in Field Properties window 331
 - in table relations 154, 155
- unrelated tables
 - adding fields from 201–202
- Update Time (Runtime Explorer) 76
- Updates (new Index mode) 151
- Use Icons for Field Types property 91
- Use menu 35, 48
 - Custom Menus item 34, 36
 - Design item 34, 35
 - for changing environments 32, 34, 36
 - User menu item 32
- Use Old File Procedure Scheme 92, 128, 487
- Use Old Startup Procedure Scheme 92
- Use V3.x.x Startup Method Scheme 483, 487
- User dialog box 567
 - assigning users passwords 568
 - entering a user's password in 568
- User environment 32–34
 - changing input and output forms in 225
 - List editor in 596
 - moving to and from 35
 - role of 33
 - testing report forms 459
- User Interface page
 - Database Properties dialog box .. 89–90, 222, 512
- User menu item
 - Use menu 32
- User Mode
 - Defining a Group Access 564
- User Modifiable menu item
 - Lists Menu 596
- user names
 - changing 569
- User/Custom Menus process 615
- users
 - adding 567–568
 - assigning passwords to 567–569
 - assigning to groups 570
 - removing from groups 571
- Users list 582
 - in Password Access editor 567
- V**
- values
 - maximum and minimum 348
 - setting default 349–351
- variable frame option
 - printing Picture fields 473–474
- variable frames
 - printing Picture fields 473–474
 - printing subforms 472–473
 - printing Text fields 474
- Variable page
 - Object Properties window 379, 402
- Variables
 - Runtime Explorer 77
- variables
 - associated with buttons 385
 - embedded in text areas 456
 - for temporary data storage 382
 - in mail-merge templates 456
 - in methods 493
 - naming of in methods 493
 - process 493
- vertical bar (|) character
 - for display formats and filters 344
 - for entry filters 344
 - for styles 363
- Vertical Scroll Bar check box 353
- volumes
 - storing segments on 631–632
- W**
- Watch Page (Runtime Explorer) 76
- Web browsers 190
- Web Server
 - Generic Web User 101
 - HTML root by default 99
- Web server process 616
- WEDD resource 29
- Whole Object Name (Find Editor option) 82
- wildcard (@) character
 - in entry filters 339

- searches in related tables.171–173
- using in methods515
- wildcard choice field
 - displayed in Wildcard Choice list.172
- Wildcard Choice list
 - in Relation Properties window162–163, 168
 - using in data entry171–173
- window
 - expanding or closing.51
 - making active51
- window margins260
- Window Title
 - setting the257
- windows
 - from several environments36
 - moving51
 - moving between51
- windows, editor. *See* editor windows

Z

- zero
 - as a placeholder.360
 - in number fields361