

THE EXPERT'S VOICE® IN .NET

# Pro WF

## Windows Workflow in .NET 3.0

*Use Windows Workflow Foundation to develop  
next-generation, workflow-enabled applications*

Bruce Bukovics

Apress®

# Pro WF

## Windows Workflow in .NET 3.0



Bruce Bukovics

## **Pro WF: Windows Workflow in .NET 3.0**

**Copyright © 2007 by Bruce Bukovics**

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-59059-778-1

ISBN-10 (pbk): 1-59059-778-8

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Ewan Buckingham

Technical Reviewer: Sylvain Groulx

Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Dominic Shakeshaft, Jim Sumser, Matt Wade

Project Manager: Beth Christmas

Copy Edit Manager: Nicole Flores

Copy Editor: Jennifer Whipple

Assistant Production Director: Kari Brooks-Copony

Production Editor: Katie Stence

Compositor: Susan Glinert

Proofreader: Nancy Riddiough

Indexer: Julie Grady

Artist: April Milne

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit <http://www.apress.com>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com> in the Source Code/Download section. You will need to answer questions pertaining to this book in order to successfully download the code.

*For Teresa*



# Contents at a Glance

|  |       |
|--|-------|
| About the Author .....   | xix   |
| About the Technical Reviewer .....                                   | xxi   |
| Acknowledgments .....  | xxiii |
| Introduction .....   | xxv   |
| <br>   |       |
| ■ <b>CHAPTER 1</b> A Quick Tour of Windows Workflow Foundation ..... | 1     |
| ■ <b>CHAPTER 2</b> Foundation Overview .....                         | 29    |
| ■ <b>CHAPTER 3</b> Activities .....                                  | 63    |
| ■ <b>CHAPTER 4</b> Hosting the Workflow Runtime .....                | 127   |
| ■ <b>CHAPTER 5</b> Flow Control .....                                | 167   |
| ■ <b>CHAPTER 6</b> Local Services .....                              | 217   |
| ■ <b>CHAPTER 7</b> Event-Driven Activities .....                     | 241   |
| ■ <b>CHAPTER 8</b> Workflow Persistence .....                        | 285   |
| ■ <b>CHAPTER 9</b> State Machine Workflows .....                     | 321   |
| ■ <b>CHAPTER 10</b> Transactions and Compensation .....              | 359   |
| ■ <b>CHAPTER 11</b> Workflow Rules .....                             | 407   |
| ■ <b>CHAPTER 12</b> Exception and Error Handling .....               | 447   |
| ■ <b>CHAPTER 13</b> Dynamic Workflow Updates .....                   | 473   |
| ■ <b>CHAPTER 14</b> Workflow Tracking .....                          | 503   |
| ■ <b>CHAPTER 15</b> Web Services and ASP.NET .....                   | 549   |
| ■ <b>CHAPTER 16</b> Workflow Serialization and Markup .....          | 581   |
| ■ <b>CHAPTER 17</b> Hosting the Workflow Designers .....             | 621   |
| <br>   |       |
| ■ <b>Index</b> .....   | 685   |



# Contents

|                                    |       |
|------------------------------------|-------|
| About the Author .....             | xix   |
| About the Technical Reviewer ..... | xxi   |
| Acknowledgments .....              | xxiii |
| Introduction .....                 | xxv   |

|                  |  |          |
|------------------|--|----------|
| <b>CHAPTER 1</b> | <b>A Quick Tour of Windows Workflow Foundation .....</b> | <b>1</b> |
|                  | Why Workflow? .....                                      | 1        |
|                  | Workflows Are Different. ....                            | 1        |
|                  | Why Windows Workflow Foundation? .....                   | 2        |
|                  | Your Development Environment .....                       | 3        |
|                  | Hello Workflow .....                                     | 3        |
|                  | Creating the Workflow Project. ....                      | 4        |
|                  | Introducing the Workflow Designer .....                  | 5        |
|                  | Using Workflow Activities .....                          | 6        |
|                  | Entering Code .....                                      | 7        |
|                  | Hosting the Workflow Runtime .....                       | 9        |
|                  | Running the Application .....                            | 11       |
|                  | Passing Parameters .....                                 | 11       |
|                  | Declaring the Properties .....                           | 11       |
|                  | Passing Values at Runtime .....                          | 12       |
|                  | Making Decisions .....                                   | 14       |
|                  | Creating a Workflow Library .....                        | 14       |
|                  | Adding Workflow Properties .....                         | 15       |
|                  | Adding IfElse Activities. ....                           | 16       |
|                  | Adding Calculation Logic .....                           | 19       |
|                  | Creating the Calculator Client .....                     | 22       |
|                  | Testing and Debugging the Calculator .....               | 25       |
|                  | Summary .....  | 28       |



|                  |                                |           |
|------------------|--------------------------------|-----------|
| <b>CHAPTER 2</b> | <b>Foundation Overview</b>     | <b>29</b> |
|                  | Workflow Types                 | 29        |
|                  | Sequential Workflows           | 29        |
|                  | State Machine Workflows        | 30        |
|                  | Choosing a Workflow Type       | 31        |
|                  | Foundation Deliverables        | 32        |
|                  | Class Libraries and Framework  | 33        |
|                  | Runtime Engine                 | 34        |
|                  | Runtime Services               | 34        |
|                  | Design Time Tools              | 35        |
|                  | .NET 2.0 Runtime               | 35        |
|                  | Runtime Environment            | 35        |
|                  | Application Host Process       | 36        |
|                  | Runtime Engine                 | 37        |
|                  | Registered Core Services       | 39        |
|                  | Registered Local Services      | 42        |
|                  | Workflow Instance              | 44        |
|                  | Design Time Environment        | 46        |
|                  | Workflow Authoring Modes       | 46        |
|                  | Project Templates              | 48        |
|                  | Workflow Designer              | 48        |
|                  | Activity Designer              | 54        |
|                  | Rule Condition Editor          | 55        |
|                  | RuleSet Editor                 | 57        |
|                  | Workflow Debugger              | 61        |
|                  | Command-Line Workflow Compiler | 62        |
|                  | Summary                        | 62        |
| <b>CHAPTER 3</b> | <b>Activities</b>              | <b>63</b> |
|                  | Understanding Activities       | 63        |
|                  | A Dual Audience for Activities | 64        |
|                  | Class Hierarchy                | 64        |

|   |     |
|---|-----|
| Exploring Standard Activities .....                   | 65  |
| Custom Workflow Logic .....                           | 66  |
| Flow Control .....                                    | 66  |
| State Management .....                                | 68  |
| Event Handling .....                                  | 69  |
| Local Service Communication .....                     | 70  |
| Rules .....   | 71  |
| Web Services .....                                    | 71  |
| Transactions, Compensation, and Synchronization ..... | 72  |
| Exceptions and Error Handling .....                   | 73  |
| Standard Activities Summary .....                     | 74  |
| Adding Workflow Logic .....                           | 75  |
| Using the CodeActivity .....                          | 76  |
| Creating the Project .....                            | 77  |
| Defining the Workflow Parameters .....                | 77  |
| Validating the Account .....                          | 81  |
| Validating the Product .....                          | 82  |
| Entering the Order .....                              | 83  |
| Running the Workflow .....                            | 87  |
| Evaluating the Approach .....                         | 90  |
| Developing Custom Activities .....                    | 90  |
| Why Custom Activities? .....                          | 91  |
| Designing for Two Audiences .....                     | 91  |
| Creating the Project .....                            | 91  |
| Implementing the Account Validation Activity .....    | 92  |
| Implementing the Product Validation Activity .....    | 95  |
| Implementing the Order Entry Activity .....           | 98  |
| Defining the Workflow Parameters .....                | 101 |
| Defining the Workflow .....                           | 102 |
| Running the Workflow .....                            | 107 |
| Evaluating the Approach .....                         | 111 |
| Enhancing the Design Experience .....                 | 111 |
| Validating the Activity .....                         | 112 |
| Customizing Toolbox Behavior .....                    | 116 |
| Customizing the Designer .....                        | 120 |
| Summary .....   | 125 |

|                  |  |            |
|------------------|--|------------|
| <b>CHAPTER 4</b> | <b>Hosting the Workflow Runtime</b>          | <b>127</b> |
|                  | Overview of Hosting                          | 127        |
|                  | Simple Workflow Hosting                      | 128        |
|                  | Implementing a Test Workflow                 | 128        |
|                  | Implementing a Simple Host Application       | 131        |
|                  | Improved Workflow Hosting                    | 136        |
|                  | Implementing the Workflow Instance Wrapper   | 136        |
|                  | Implementing the Workflow Manager Class      | 139        |
|                  | Hosting with the Workflow Manager            | 147        |
|                  | Configuring the Runtime with App.Config      | 153        |
|                  | Controlling a Workflow Instance              | 158        |
|                  | Synchronous Workflow Execution               | 162        |
|                  | Summary                                      | 165        |
| <br>             |  |            |
| <b>CHAPTER 5</b> | <b>Flow Control</b>                          | <b>167</b> |
|                  | Condition Types                              | 167        |
|                  | Using the IfElseActivity                     | 168        |
|                  | Using an IfElseActivity with Code Conditions | 169        |
|                  | Using an IfElseActivity with Rule Conditions | 176        |
|                  | Using the WhileActivity                      | 180        |
|                  | Implementing the Workflow                    | 180        |
|                  | Testing the Workflow                         | 184        |
|                  | Using the ParallelActivity                   | 186        |
|                  | Implementing the Workflow                    | 186        |
|                  | Testing the Workflow                         | 190        |
|                  | Adding a DelayActivity                       | 192        |
|                  | Using the ReplicatorActivity                 | 194        |
|                  | Implementing the Sequence Workflow           | 195        |
|                  | Testing the Sequence Workflow                | 199        |
|                  | Implementing the Parallel Workflow           | 200        |
|                  | Testing the Parallel Workflow                | 204        |
|                  | Using ReplicatorActivity Events              | 206        |
|                  | Interrupting Execution                       | 206        |
|                  | Using the ConditionedActivityGroup           | 207        |
|                  | Implementing the Workflow                    | 207        |
|                  | Testing the Workflow                         | 211        |
|                  | Using the InvokeWorkflowActivity             | 213        |
|                  | Using the TerminateActivity                  | 214        |
|                  | Using the SuspendActivity                    | 214        |
|                  | Summary                                      | 215        |

|                  |  |     |
|------------------|--|-----|
| <b>CHAPTER 6</b> | <b>Local Services</b>                  | 217 |
|                  | Understanding Local Services           | 217 |
|                  | Implementing a Local Service           | 218 |
|                  | Using a Local Service                  | 219 |
|                  | Implementing and Using a Local Service | 219 |
|                  | Implementing the Account Class         | 219 |
|                  | Declaring the Service Contract         | 220 |
|                  | Implementing the Local Service Class   | 221 |
|                  | Implementing the Workflow              | 223 |
|                  | Testing the Workflow                   | 226 |
|                  | Loading from App.config                | 228 |
|                  | Using a Custom Activity                | 231 |
|                  | Implementing a Custom Activity         | 231 |
|                  | Modifying the Workflow                 | 234 |
|                  | Testing the Workflow                   | 235 |
|                  | Using the CallExternalMethodActivity   | 236 |
|                  | Implementing the Workflow              | 236 |
|                  | Testing the Workflow                   | 238 |
|                  | Summary                                | 239 |
| <b>CHAPTER 7</b> | <b>Event-Driven Activities</b>         | 241 |
|                  | Using Event-Driven Activities          | 241 |
|                  | Using the HandleExternalEventActivity  | 243 |
|                  | Creating the Project                   | 244 |
|                  | Implementing the Event Arguments Class | 244 |
|                  | Defining the Service Interface         | 245 |
|                  | Implementing the Local Service         | 246 |
|                  | Implementing the Workflow              | 248 |
|                  | Implementing the Host Application      | 255 |
|                  | Testing the Workflow                   | 259 |
|                  | Generating Communication Activities    | 260 |
|                  | Generating the Activities              | 260 |
|                  | Modifying the Workflow                 | 261 |
|                  | Manually Controlling Correlation       | 262 |
|                  | Implementing the Event Arguments Class | 264 |
|                  | Defining the Service Interface         | 265 |
|                  | Implementing the Local Service         | 265 |
|                  | Implementing the Workflow              | 266 |
|                  | Testing the Workflow                   | 269 |

|  |            |
|--|------------|
| Using the EventHandlingScopeActivity .....         | 271        |
| Defining the Service Interface .....               | 272        |
| Implementing the Local Service .....               | 272        |
| Implementing the Workflow .....                    | 274        |
| Testing the Workflow .....                         | 280        |
| Summary .....                                      | 283        |
| <br><b>CHAPTER 8 Workflow Persistence .....</b>    | <b>285</b> |
| Understanding Persistence .....                    | 285        |
| Why Persist Workflows? .....                       | 285        |
| Persistence Overview .....                         | 286        |
| Using the SqlWorkflowPersistenceService .....      | 288        |
| Preparing a Database for Persistence .....         | 289        |
| Implementing the Local Service .....               | 289        |
| Implementing the Workflow .....                    | 291        |
| Implementing the Host Application .....            | 293        |
| Testing the Application .....                      | 304        |
| Implementing a Custom Persistence Service .....    | 308        |
| Understanding the Abstract Methods .....           | 308        |
| Implementing the Service .....                     | 310        |
| Testing the Custom Service .....                   | 317        |
| Summary .....                                      | 319        |
| <br><b>CHAPTER 9 State Machine Workflows .....</b> | <b>321</b> |
| Understanding State Machine Workflows .....        | 321        |
| Why a State Machine Workflow? .....                | 322        |
| State Machine Workflow Overview .....              | 323        |
| Implementing a State Machine Workflow .....        | 326        |
| Designing the Car State Machine .....              | 326        |
| Defining the Local Service Interface .....         | 328        |
| Implementing the Local Service .....               | 329        |
| Implementing the Workflow .....                    | 332        |
| Implementing the Host Application .....            | 340        |
| Testing the Application .....                      | 347        |
| Eliminating Duplicate Event Handlers .....         | 349        |
| Refactoring the CarWorkflow .....                  | 349        |
| Testing the Revised Workflow .....                 | 350        |

|   |            |
|---|------------|
| Identifying Available Events .....                        | 351        |
| Interrogating the Workflow Queues .....                   | 351        |
| Modifying the CarStateMachine Application .....           | 351        |
| Testing the Application .....                             | 354        |
| Accessing Runtime Information .....                       | 354        |
| Modifying the CarStateMachine Application .....           | 355        |
| Testing the Application .....                             | 356        |
| Summary .....   | 357        |
| <br><b>CHAPTER 10 Transactions and Compensation .....</b> | <b>359</b> |
| Understanding Transactions .....                          | 359        |
| The Way of Transactions .....                             | 360        |
| WF Support for Transactions .....                         | 360        |
| Using the TransactionScopeActivity .....                  | 362        |
| Implementing the AccountAdjustmentActivity .....          | 365        |
| Implementing the AccountTransferWorkflow .....            | 369        |
| Testing the Workflow .....                                | 372        |
| Understanding Compensation .....                          | 377        |
| Using Compensatable Activities .....                      | 379        |
| Implementing the InventoryUpdateActivity .....            | 379        |
| Implementing the OrderDetailActivity .....                | 383        |
| Implementing the OrderEntryWorkflow .....                 | 388        |
| Testing the Workflow .....                                | 395        |
| Participating in a Batch of Work .....                    | 398        |
| Using the IPendingWork Interface .....                    | 399        |
| Implementing the Local Service .....                      | 399        |
| Implementing the BatchedWorkWorkflow .....                | 402        |
| Testing the Workflow .....                                | 403        |
| Summary .....   | 405        |
| <br><b>CHAPTER 11 Workflow Rules .....</b>                | <b>407</b> |
| Understanding Workflow Rules .....                        | 407        |
| Parts of a Rule .....                                     | 408        |
| Why Use Rules? .....                                      | 408        |
| Using Rules in WF .....                                   | 409        |
| Defining Rules .....                                      | 409        |
| Defining RuleSets .....                                   | 412        |
| Identifying Dependencies with Attributes .....            | 414        |

|   |     |
|---|-----|
| Defining Rules with a PolicyActivity .....        | 416 |
| Implementing the SalesItem Class .....            | 416 |
| Declaring the Rules .....                         | 417 |
| Testing the Workflow .....                        | 424 |
| Tracing Rules .....                               | 428 |
| Adjusting Rule Sequence .....                     | 431 |
| Setting the Rule Priority .....                   | 431 |
| Testing the Workflow .....                        | 432 |
| Using Methods Within Rules .....                  | 434 |
| Adding the Access Methods .....                   | 434 |
| Using the Methods in the RuleSet .....            | 435 |
| Identifying Indirect Relationships .....          | 436 |
| Executing a RuleSet in Code .....                 | 437 |
| Implementing the SellItemSerializedWorkflow ..... | 437 |
| Testing the Workflow .....                        | 440 |
| Constructing a RuleSet in Code .....              | 441 |
| Implementing the SellItemInCodeWorkflow .....     | 441 |
| Testing the Workflow .....                        | 444 |
| Summary .....                                     | 445 |

## ■ CHAPTER 12 Exception and Error Handling .....

|   |     |
|---|-----|
| Understanding Workflow Exception Handling ..... | 447 |
| Reviewing Default Behavior .....                | 450 |
| Implementing the ExceptionWorkflow .....        | 450 |
| Testing the Workflow .....                      | 452 |
| Using FaultHandlerActivity .....                | 454 |
| Handling ArithmeticException .....              | 454 |
| Handling DivideByZeroException .....            | 458 |
| Containing the Exception .....                  | 460 |
| Rethrowing an Exception .....                   | 462 |
| Compensation and Exceptions .....               | 463 |
| Implementing the CompensateWorkflow .....       | 464 |
| Test the Workflow .....                         | 466 |
| Using CancellationHandlerActivity .....         | 468 |
| Implementing the CancelHandlerWorkflow .....    | 468 |
| Testing the Workflow .....                      | 471 |
| Summary .....                                   | 471 |

|                   |  |            |
|-------------------|--|------------|
| <b>CHAPTER 13</b> | <b>Dynamic Workflow Updates</b>            | <b>473</b> |
|                   | Understanding Dynamic Updates              | 473        |
|                   | Why Use Dynamic Updates?                   | 474        |
|                   | Applying Dynamic Updates                   | 474        |
|                   | Preventing Dynamic Updates                 | 476        |
|                   | Applying Updates from the Host Application | 476        |
|                   | Implementing the DynamicWorkflow           | 476        |
|                   | Implementing a Custom Activity             | 478        |
|                   | Implementing the Host Application          | 480        |
|                   | Testing the Workflow                       | 484        |
|                   | Restricting Dynamic Updates                | 485        |
|                   | Applying Updates from Within a Workflow    | 486        |
|                   | Implementing the SelfUpdatingWorkflow      | 486        |
|                   | Implementing the Host Application          | 490        |
|                   | Testing the Workflow                       | 491        |
|                   | Updating a Rule Condition                  | 492        |
|                   | Implementing the DynamicConditionWorkflow  | 492        |
|                   | Implementing the Host Application          | 494        |
|                   | Testing the Workflow                       | 498        |
|                   | Replacing a Rule Definition                | 498        |
|                   | Modifying the RuleDefinitions              | 499        |
|                   | Modifying the Host Application             | 500        |
|                   | Testing the Revised Application            | 502        |
|                   | Summary                                    | 502        |
| <b>CHAPTER 14</b> | <b>Workflow Tracking</b>                   | <b>503</b> |
|                   | Understanding Workflow Tracking            | 503        |
|                   | Tracking Services                          | 504        |
|                   | Tracking Event Types                       | 504        |
|                   | Custom Tracking Profiles                   | 505        |
|                   | Using the Tracking Data                    | 506        |
|                   | Benefiting from Workflow Tracking          | 506        |
|                   | Using the SqlTrackingService               | 507        |
|                   | Preparing the Tracking SQL Database        | 507        |
|                   | Developing a Test Workflow                 | 508        |
|                   | Developing the Host Application            | 510        |
|                   | Executing the Host Application             | 512        |
|                   | Retrieving Tracking Data                   | 513        |



|  |     |
|--|-----|
| Creating User Track Points .....                     | 521 |
| Enhancing the TrackingExampleWorkflow .....          | 522 |
| Testing the Revised Workflow .....                   | 523 |
| Tracking Rules Evaluation .....                      | 523 |
| Implementing the TrackingRulesWorkflow .....         | 523 |
| Testing the Workflow .....                           | 525 |
| Extracting Data with a Custom Tracking Profile ..... | 526 |
| Working with Tracking Profiles .....                 | 527 |
| Implementing the TrackingProfileHelper .....         | 528 |
| Creating the Tracking Profile .....                  | 532 |
| Testing the Tracking Profile .....                   | 535 |
| Maintaining the SQL Tracking Database .....          | 536 |
| Partitioning .....                                   | 536 |
| Setting the Partition Interval .....                 | 536 |
| Automatic or Manual Partitioning .....               | 536 |
| Accessing Partitioned Data .....                     | 537 |
| Detaching or Dropping a Partition .....              | 537 |
| Developing a Tracking Service .....                  | 538 |
| Implementing a Tracking Channel .....                | 538 |
| Implementing a Tracking Service .....                | 540 |
| Testing the Custom Tracking Service .....            | 544 |
| Summary .....  | 547 |

## ■ CHAPTER 15 Web Services and ASP.NET .....

|  |     |
|--|-----|
| Publishing a Workflow As a Web Service .....       | 549 |
| Understanding the Web Service Activities .....     | 549 |
| Publishing and Configuration .....                 | 551 |
| Developing a Web Service Workflow .....            | 552 |
| Defining the Web Service Interface .....           | 552 |
| Defining the MathServiceWorkflow .....             | 552 |
| Publishing the Workflow .....                      | 555 |
| Testing the Web Service .....                      | 557 |
| Returning a Web Service Fault .....                | 560 |
| Modifying the MathServiceWorkflow .....            | 560 |
| Testing the Revised Web Service .....              | 562 |
| Developing a Stateful Web Service .....            | 562 |
| Defining the Web Service Interface .....           | 563 |
| Implementing the MathServiceStatefulWorkflow ..... | 563 |
| Publishing the New Workflow .....                  | 567 |
| Testing the Web Service .....                      | 567 |

|   |     |
|---|-----|
| Invoking a Web Service from a Workflow .....    | 567 |
| Implementing the InvokeWebServiceWorkflow ..... | 567 |
| Testing the Workflow .....                      | 571 |
| Using Workflows from ASP.NET .....              | 573 |
| Implementing the DivideNumberWorkflow .....     | 573 |
| Implementing the UseWorkflowWebsite .....       | 574 |
| Testing the Web Site .....                      | 578 |
| Summary .....                                   | 579 |

## ■ CHAPTER 16 Workflow Serialization and Markup ..... 581

|  |     |
|--|-----|
| Understanding Workflow Authoring Modes ..... | 581 |
| Code-Only Authoring Mode .....               | 581 |
| Code-Separation Authoring Mode .....         | 583 |
| No-Code Authoring Mode .....                 | 584 |
| Developing a Code-Only Workflow .....        | 586 |
| Implementing the Workflow .....              | 586 |
| Testing the Workflow .....                   | 588 |
| Reviewing the Generated Code .....           | 590 |
| Developing a Code-Separation Workflow .....  | 592 |
| Implementing the Workflow .....              | 592 |
| Testing the Workflow .....                   | 592 |
| Reviewing the Markup .....                   | 592 |
| Developing a No-Code Workflow .....          | 594 |
| Implementing the Base Workflow Class .....   | 594 |
| Implementing the Custom Activity .....       | 595 |
| Defining the Workflow Markup .....           | 596 |
| Enhancing the WorkflowRuntimeManager .....   | 597 |
| Testing the Workflow .....                   | 599 |
| Using Rules with a No-Code Workflow .....    | 601 |
| Defining the Rule Condition .....            | 601 |
| Modifying the Workflow Markup .....          | 603 |
| Testing the Workflow .....                   | 603 |
| Serializing to Markup .....                  | 604 |
| Compiling a Workflow .....                   | 608 |
| Compiling a Workflow with Rules .....        | 611 |
| Compiling from the Command Line .....        | 614 |
| Deserializing Markup .....                   | 615 |
| Summary .....                                | 619 |

■ **CHAPTER 17**   **Hosting the Workflow Designers** ..... 621

    Understanding the Workflow Designers ..... 621

        Designer Namespaces ..... 622

        Designer Classes ..... 622

        Designer Services ..... 623

    Building a Designer Application ..... 624

        Creating the Designer Project ..... 625

        Implementing WorkflowLoader ..... 626

        Implementing WorkflowMenuService ..... 635

        Implementing WorkflowEventBindingService ..... 639

        Implementing EventPropertyDescriptor ..... 643

        Implementing WorkflowPropertyValueService ..... 648

        Implementing WorkflowToolboxService ..... 650

        Implementing WorkflowDesigner ..... 661

        Implementing MainForm ..... 668

        Implementing AssemblyReferenceForm ..... 674

        Implementing NewWorkflowForm ..... 676

    Using the Designer ..... 679

    Summary ..... 684

■ **INDEX** ..... 685

# About the Author

■ **BRUCE BUKOVICS** has been a working developer for more than 25 years. During this time, he has designed and developed applications in such widely varying areas as banking, corporate finance, credit card processing, payroll processing, and retail systems.

He has firsthand developer experience with a variety of languages, including C, C++, Delphi, Java, VB, and C#. His design and development experience includes everything from mainframe and client/server to widely distributed n-tier and SOA applications. Most recently, he has been immersed in the .NET 3.0 technology stack, leveraging Windows Presentation Foundation (WPF), Windows Communication Foundation (WCF), and, of course, Windows Workflow Foundation (WF) to help build a new generation of applications.

He considers himself a pragmatic programmer and test-driven development evangelist. He doesn't stand on formality and doesn't do things in a particular way just because they have always been done that way. He's willing to look at alternate or unorthodox solutions to a problem if that's what it takes.

He is currently employed at Radiant Systems Inc. in Alpharetta, Georgia, as a lead developer and architect in the central technology group.



# About the Technical Reviewer

■ **SYLVAIN GROULX** is an independent database software consultant based in Montreal, Quebec. He's been an application developer and DBA over the past 20 years. His current interests include the .NET Framework 3.0 Windows Presentation Foundation, Windows Workflow Foundation, and SQL Server 2005 Services (Broker, Integration, Notification, and Reporting).

As the founder of the Microsoft .NET Architecture User Group in Montreal, he has been an active proponent of .NET technologies and community-based learning initiatives. He is a great enthusiast for C#, which is why he has been actively involved with .NET since the first bits were released in 2000.

He enjoyed many great years as a Microsoft MVP before joining Microsoft Consulting Services. His past roles at Microsoft include project lead and developer consultant. He also spent many years architecting and building custom applications for large enterprise customers.

When not sitting in front of a keyboard, Sylvain is busy playing a round of golf whenever possible.



# Acknowledgments

Once again, the folks at Apress have done an outstanding job with this book. It takes a team of very professional people to produce a book like this. You truly made the task of writing this book an enjoyable one. A big thank-you goes out to all of you.

I would like to personally thank Ewan Buckingham who was the editor of this book. Ewan allowed me to write the book that I wanted to write, but he was also there with guidance when I needed it most. Matt Moodie also did some pinch-hitting for Ewan when things got backed up. Jennifer Whipple was the primary copy editor on the book, along with some assistance from Nicole Flores. They both did a great job correcting my prose, refining it without changing my original intent. Katie Stence was the production editor who worked her magic to transform my raw text into the pages that you see before you. And Beth Christmas was the project manager for the book. She was the traffic cop and task master who kept things running smoothly and on schedule.

A good technical book needs a great technical reviewer, and Sylvain Groulx stepped up to the challenge. This was an especially challenging job for Sylvain because I began this book while WF was in a constant state of change, churning with what seemed like monthly CTPs or betas. Yet in spite of this, he was able to keep up with all of the class and namespace changes from one version of WF to the next. Thank you, Sylvain, for making sure that this book is as accurate as possible.

I'd also like to thank all of the folks who contributed to the Microsoft Windows Workflow Foundation forum hosted on MSDN. This active community of developers was an invaluable resource in understanding how some of the undocumented features of WF worked during the early beta and CTP stage.

Most importantly, I must thank my wife, Teresa, and son, Brennen, for their support and patience during this project. When I finished my first book, I honestly thought that it would be a long time before I wrote my second. Little did I know that I was about to embark on this little project without any downtime at all between books. Sorry about that. I love both of you very much, and I'm looking forward to having time to spend with you again. I promise: No more books for a while.





# Introduction

I started working with the new Microsoft WinFX technology stack early in the beta and CTP (Community Technology Preview) stage. The foundations in WinFX (Windows Presentation, Windows Communication, and Windows Workflow) have now finally made their way into a shipping Microsoft product: .NET 3.0. I actually started to learn and use all three of these foundations at the same time in my day job. Talk about a massive learning curve.

While I was impressed with the flexibility and capabilities of Windows Presentation Foundation and Windows Communication Foundation, I was somehow inexplicably drawn to Windows Workflow Foundation (WF). WF isn't just a new way to implement a user interface, or a new way to communicate between applications and services. WF represents a completely new way to develop applications. It is declarative, visual, and infinitely flexible. It promotes a model that cleanly separates *what* to do from *when* to do it. This separation allows you to change the workflow model (the *when*) without affecting the *what*. Business logic is implemented as a set of discrete, testable components that are assembled into workflows like building blocks.

Workflow isn't a new concept. But when Microsoft spends years developing a workflow foundation and provides it to us without cost, it is an event worth noting. Other workflow frameworks exist, but WF will soon become the de facto standard workflow framework for Windows applications.

I wrote this book because I'm excited about workflow, and WF in particular. I'm excited about the opportunities that it holds for application developers like us. My hope is that this book will help you to use WF to build an exciting new generation of workflow-enabled applications.

## Who Should Read This Book

This book is for all .NET developers who want to learn how to use Windows Workflow Foundation in their own applications. This book is not a primer on .NET or the C# language. To get the most out of the examples that I present in this book, you need a good working knowledge of .NET 1.1 and preferably .NET 2.0. All of the examples are presented in C#, so you should be proficient with C#.

## An Overview of This Book

The material in this book is a WF tutorial presented in 17 chapters, with each chapter building upon the ones before it. I've tried to organize the material so that you don't have to jump ahead in order to understand how something works. But since the chapters build upon each other, I do assume that you have read each chapter in order and understand the material that has already been presented.

The short sections that follow provide a brief summary of each chapter.

## Chapter 1: A Quick Tour of Windows Workflow Foundation

This chapter provides a brief introduction to WF. In this chapter, you jump right in and develop your first workflow ("Hello Workflow"). You are introduced to some of the fundamental concepts of WF, such as how to pass parameters to a workflow and how to make decisions within a workflow.

## Chapter 2: Foundation Overview

The goal of this chapter is to provide a high-level overview of WF in its entirety. This chapter doesn't teach you how to use each individual WF feature, but it does acquaint you with the design time and runtime features that are available with WF. This chapter is a roadmap for the material that is covered in the remainder of the book.

## Chapter 3: Activities

Activities are the building blocks of WF and are used to construct complete workflows. This chapter provides a summary of the standard activities that are distributed with WF. This chapter also contrasts two ways to introduce business logic into a workflow: the `CodeActivity` and building your own custom activities.

## Chapter 4: Hosting the Workflow Runtime

WF is not a stand-alone application. It is a framework for building your own workflow-enabled applications. This chapter demonstrates how to host the workflow runtime in your own application. Included in this chapter is a set of custom workflow manager classes that assist with hosting of the workflow runtime. These helper classes are used in most of the chapters that follow this one.

## Chapter 5: Flow Control

WF includes a rich set of standard activities that support everything from simple branching decisions and while loops to parallel execution and replication. These flow control activities control the execution sequence within a workflow and are covered in this chapter. Most of these activities support Boolean conditions that can be specified in code or as declarative rule conditions. These two types of conditions are contrasted in this chapter.

## Chapter 6: Local Services

Several core features of WF are implemented as pluggable services. This allows you to choose the implementation of each service that meets your needs. WF also provides for local services that can be implemented by you to serve any purpose. One common use of local services is to facilitate communication between workflow instances and your host application. The focus of this chapter is on implementing and using your own local services.

## Chapter 7: Event-Driven Activities

This chapter covers event-driven activities that allow your workflows to wait for the receipt of an external event. Chapter 6 shows you how to implement local services and invoke methods of those services from workflow instances. This chapter demonstrates how to raise events from those local services and handle the events within a workflow.

## Chapter 8: Workflow Persistence

Workflow persistence allows you to automatically save the state of running workflow instances and then reload them at a later time. The use of persistence is especially important for long-running workflows where a workflow can be unloaded from memory while it is idle and waiting for an external event.

## Chapter 9: State Machine Workflows

WF supports two main types of workflows: sequential and state machine. Up until this point in the book, you have been working with sequential workflows that target system interaction problems. Sequential workflows are best used when the exact sequence of tasks is known at design time. State machine workflows are the focus of this chapter and are designed to easily react to external events. They are especially useful for problems that involve human interaction since the exact sequence of tasks can't be determined at design time.

## Chapter 10: Transactions and Compensation

The goal of this chapter is to demonstrate two ways to control the integrity and consistency of work that is performed by a workflow. Transactions allow you to enlist multiple activities into a single logical unit of work. When transactions are used, all of the work is committed or rolled back together without any partial updates. On the other hand, compensation is the process of undoing work that has already completed. Compensation might be necessary if individual activities in a workflow have completed successfully, but later the workflow determines that the work must be undone.

## Chapter 11: Workflow Rules

WF includes a general-purpose rules engine that you can also use as an alternate way to declare your business logic. Rules are best thought of as simple statements or assertions about data and not as procedural instructions. Individual rules are grouped into rule sets and are evaluated by the rules engine that is included with WF. Each rule allows you to define the actions to execute when the rule evaluates to true, and a separate set of actions when it is false.

## Chapter 12: Exception and Error Handling

Exception handling is important in any application, and WF provides a way to declaratively handle exceptions. The goal of this chapter is to demonstrate various ways to handle exceptions within the workflow model. This chapter also covers cancellation handlers that are used to execute a set of activities when an executing activity is canceled.

## Chapter 13: Dynamic Workflow Updates

Most of the time, you will statically define a workflow and then create instances of it at runtime. WF also provides the ability to dynamically apply updates to an executing workflow, altering the internal structure of the workflow. This chapter demonstrates how to apply dynamic workflow updates from the host application, as well as from within an executing workflow.

## Chapter 14: Workflow Tracking

WF provides an instrumentation framework for tracking the execution of each workflow. The tracking framework supports pluggable tracking services that you can implement to meet your needs. The framework is based on tracking profiles that allow you to customize the amount and type of data tracked for each workflow type. The focus of this chapter is using the standard tracking service and also developing your own custom tracking service.

## Chapter 15: Web Services and ASP.NET

WF allows you to declaratively access web services from within a workflow. You can also expose a workflow as a web service that can be accessed by any web service client. These topics are covered in this chapter along with the use of WF from an ASP.NET Web Forms application.

## Chapter 16: Workflow Serialization and Markup

The goal of this chapter is to demonstrate the use of workflow markup and serialization. Each workflow definition can be declared and expressed in several forms, including markup. Markup declares a workflow in a simple XML form that doesn't require compilation and can be parsed and executed directly by the workflow runtime engine. The advantage of using markup is that it is much easier to modify the workflow definition outside of Visual Studio, since it doesn't require compilation.

## Chapter 17: Hosting the Workflow Designers

After workflow serialization and markup is presented in Chapter 16, this chapter shows you how to build your own workflow designer. WF includes the classes that you need to host the workflow designers in your own application. The bulk of this chapter presents a working designer application that enables you to define and modify markup-only workflows.

## What You Need to Use This Book

To execute the examples presented in this book, you need a development machine with an OS that supports the .NET 3.0 runtime. Currently, that is Windows XP Service Pack 2, Windows 2003 Server Service Pack 1, or Windows Vista. A fully licensed version of Visual Studio 2005 is also necessary in order to use the visual workflow designers and to work with the example C# code. The workflow designers are not supported on the Express editions of Visual Studio or Visual C#.

The runtime for Windows Workflow Foundation is distributed as part of .NET 3.0. In addition to this, you will need to install the designated version of the Windows SDK that supports workflow development. A separate set of extensions to Visual Studio 2005 must also be downloaded and installed to use the workflow designers, project templates, and workflow debugger.

You should be able to locate download links for all of the necessary files from the Microsoft .NET Framework Development Center (<http://msdn2.microsoft.com/en-us/netframework/default.aspx>).

## Obtaining This Book's Source Code

I have found that the best way to learn and retain a new skill is through hands-on examples. For this reason, this book contains a lot of example source code. I've been frustrated on more than one occasion with technical books that don't print all of the source code in the book. The code may be available for download, but then you need to have a computer handy while you are reading the book. That doesn't work well on the beach. So I've made it a point to present all of the code that is necessary to actually build and execute the examples.

When you are ready to execute the example code, you don't have to enter it yourself. You can download all of the code presented in this book from the Apress site at <http://www.apress.com> and go to the Source Code/Download section. I've organized all of the downloadable code into separate folders for each chapter with a separate Visual Studio solution for each chapter. The only exception is one shared project that is referenced by projects in most of the chapters of this book. I'd suggest that you also keep your code separate by chapter as you work through the examples in this book.

## How to Reach Me

If you have questions or comments about this book or Windows Workflow, I'd love to hear from you. Just send your e-mail to [workflow@bukovics.com](mailto:workflow@bukovics.com). To make sure your mail makes it past my spam filters you might want to include the text **ProWF** somewhere in the subject line.

