Designed for Microsoft® Windows NT® Windows® 98

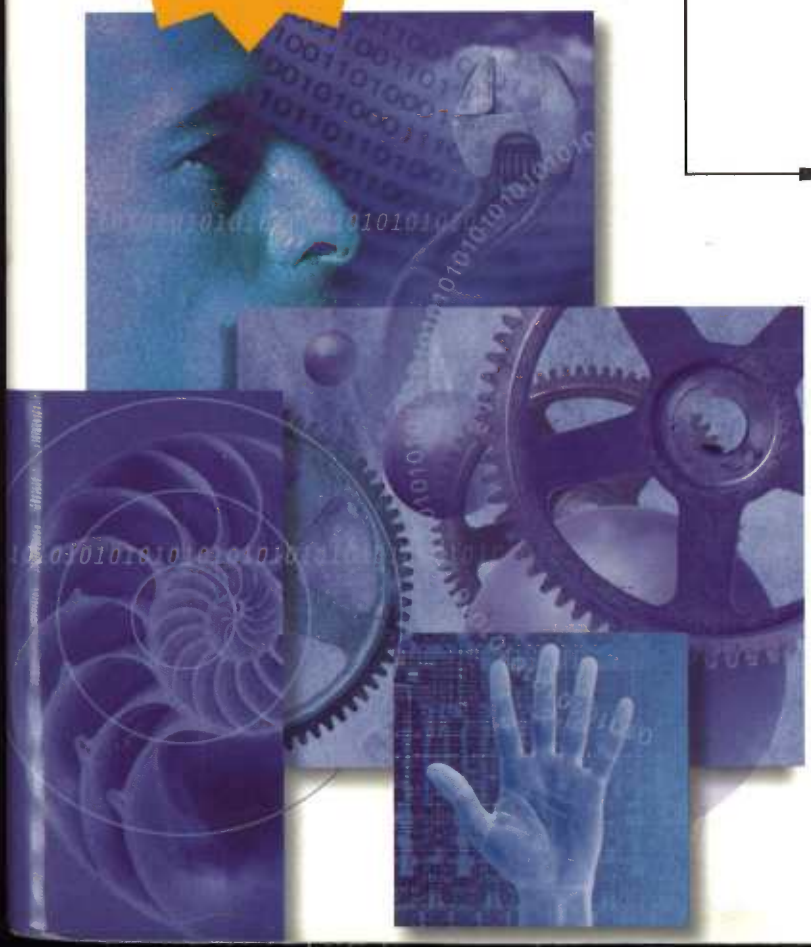
CD-ROM Included

MICROSOFT® PROGRAMMING SERIES

Microsoft®

Programming Microsoft® Visual Basic® 6.0

"Excellent book...a must for any serious VB developer."
Dr. Dobb's Journal



Master object-oriented programming techniques for rapid 32-bit development

Francesco Balena

Foreword by James Fawcette
Publisher of *Visual Basic Programmer's Journal*

Microsoft Corp. Exhibit 1064

DOCKET ALARM

Find authenticated court documents without watermarks at docketalarm.com.

Microsoft® Press

Programming Microsoft® Visual Basic® 6.0

Francesco Balena

Microsoft Corp. Exhibit 1064

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Copyright © 1999 by Francesco Balena

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Library of Congress Cataloging-in-Publication Data
Balena, Francesco, 1960–
Programming Microsoft Visual Basic 6.0 / Francesco Balena.
p. cm.
Includes index.
ISBN 0-7356-0558-0
1. Microsoft Visual BASIC. 2. BASIC (Computer program language)
I. Title.
QA76.73.B3B345 1999
005.26'8--dc21 99-20381
CIP

Printed and bound in the United States of America.

5 6 7 8 9 QWTQWT 4 3 2 1 0

Distributed in Canada by Penguin Books Canada Limited.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at mspress.microsoft.com.

Active Desktop, ActiveX, the BackOffice logo, FrontPage, IntelliSense, Microsoft, Microsoft Press, MS-DOS, Visual Basic, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

The example companies, organizations, products, people, and events depicted herein are fictitious. No association with any real company, organization, product, person, or event is intended or should be inferred.

Acquisitions Editor: Eric Stroo
Project Editor: Kathleen Atkins
Manuscript Editors: Kathleen Atkins, Sally Stickney
Technical Editor: Marzena Makuta

Microsoft Corp. Exhibit 1064

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

```
    cmdClose.Caption = rs.LoadResString(rsClose)
    cmdRefresh.Caption = rs.LoadResString(rsRefresh)
    ' (Other string assignments omitted...)
    Set imgFlag.Picture = rs.LoadResPicture(rpFlag)
End Sub
```

Because the MyApplication000.Resource class declares enumerated constants for all the strings and other resources in the satellite DLL, you can use IntelliSense to speed up the development phase and produce a more readable and self-documenting code at the same time.

MULTITHREADED ACTIVEX COMPONENTS

Both Visual Basic 5 and 6 can create multithreaded ActiveX components. Components built with the first release of Visual Basic 5, however, could only support multithreading if they had no user interface, which is a serious limitation in some cases. This restriction was lifted in Service Pack 2.

Threading Models

In a nutshell, multithreading is the ability to execute different code portions of an application at the same time. Many popular Windows applications are multithreaded. For example, Microsoft Word uses at least two threads, and the Visual Basic environment uses five threads. Multiple threads are a good choice when you need to execute complex tasks in the background (for example, paginating a document) or when you want to keep the user interface responsive even when your application is doing something else. Multiple threads are especially necessary when you're building *scalable* remote components that have to serve hundreds of clients at the same time.

There are two main types of threading models: *free threading* and *apartment threading*. In the free-threading model, each thread can access the entire process's data area and all threads share the application's global variables. Free threading is powerful and efficient, but it's a nightmare even for most experienced programmers because you must arbitrate among all the shared resources, including variables. For example, even an innocent statement such as

```
If x > 1 Then x = x - 1     ' X should always be greater than 1.
```

can create problems. Imagine this scenario: Thread A reads the value of the *x* variable and finds that it is 2, but before it executes the Then portion of the statement, the CPU switches to Thread B. Thread B happens to be executing the same statement (an unlikely but not impossible circumstance), finds that *x* is 2, and therefore decrements it to 1. When Thread A regains the control of the CPU, it decrements the variable to 0, which is an invalid value that will probably cause other logic errors later in the program's life.

Microsoft Corp. Exhibit 1064

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

The apartment-threading model solves these problems by encapsulating each thread in an *apartment*. Code executed in a given apartment can't access variables belonging to other apartments. Each apartment has its own set of variables, so if two threads are accessing the *x* variable at the same time, they're actually referencing two different memory locations. This mechanism neatly solves the synchronization problem described earlier, and for this reason the apartment-threading model is inherently safer than the free-threading model. In Visual Basic, you can build ActiveX components that support the apartment model only.

Multithreaded ActiveX EXE Components

Visual Basic 5 and 6 let you create out-of-process servers that create an additional thread when a client instantiates a new object. All you need to do to transform a regular ActiveX EXE component into a multithreaded component is select an option in the General tab of the Project Properties dialog box. (See Figure 16-18.) There are three possible settings. The default setting is the Thread Pool option with 1 thread; this corresponds to a single-threaded component.

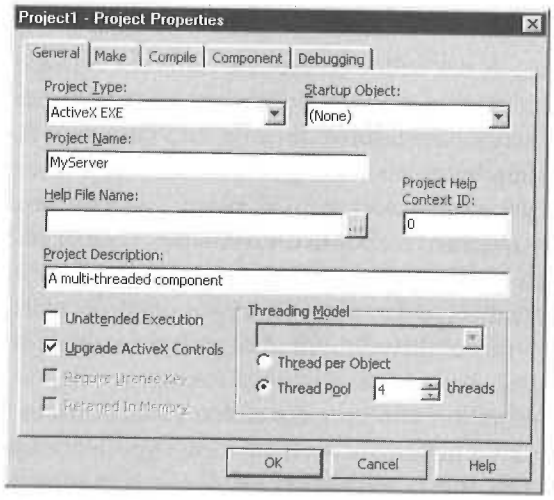


Figure 16-18. *Create a multithreaded component with a few mouse clicks in the Project Properties dialog box.*

If you select the Thread Per Object option, you build a multithreaded component that creates a new thread for every object requested by its clients. Because all objects are executed in their own threads, no client can ever block another client, so these components are highly responsive. The disadvantage of this approach is that too many threads can bring even a powerful system to its knees because Windows has to spend a lot of time just switching from one thread to the other.

Microsoft Corp. Exhibit 1064

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.