

# **Pro Visual C++ 2005 for Developers: Featuring C++/CLI**



Dean C. Wills

**Pro Visual C++ 2005 for Developers: Featuring C++/CLI**

**Copyright © 2006 by Dean C. Wills**

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: 978-1-59059-608-1

ISBN-10: 1-59059-608-0

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: Alvin Chardón

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

Project Manager: Elizabeth Seymour

Copy Edit Manager: Nicole Flores

Copy Editor: Heather Lang

Assistant Production Director: Kari Brooks-Copony

Senior Production Editor: Laura Cheu

Compositor: Dina Quan and Kinetic Publishing Services, LLC

Proofreader: Elizabeth Berry

Indexer: Brenda Miller

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](mailto: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.

# Contents

Foreword.....	xvii
About the Author .....	xix
About the Technical Reviewer .....	xxi
Acknowledgments .....	xxiii
Preface .....	xxv
Introduction .....	xxvii

## PART 1 ■ ■ ■ Fast Track to C++/CLI

■ CHAPTER 1	<b>Hello, World.....</b>	<b>3</b>
	Starting the Visual Studio 2005 Console .....	4
	Retrieving the Source Files .....	4
	Executing HelloCpp.cpp .....	5
	A Quick Tour of the Visual C++ IDE .....	5
	Understanding Projects and Solutions .....	6
	Understanding the Differences .....	7
	Window Layout.....	10
	Building, Executing, and Debugging.....	12
	Summary .....	14
■ CHAPTER 2	<b>There's No Place Like Home.....</b>	<b>15</b>
	Developing the Program.....	15
	Deal Me In.....	15
	The Completed C# Program.....	18
	A Quick Look at the Code.....	19
	Projects and Solutions .....	19
	A Quick Look.....	20
	Building and Executing the Project.....	21
	Binding C++ .....	22

Doing the Shuffle Without the IDE . . . . .	26
Basic Command-Line Compilation . . . . .	26
Using a Module . . . . .	26
Summary . . . . .	27

### ■ CHAPTER 3   **Syntax** . . . . . 29

C# class (Reference Type) . . . . .	31
C# struct (Value Type) . . . . .	31
Caveat . . . . .	31
Lack of Locality . . . . .	32
The C++ Approach . . . . .	33
Types of Member Access Operators . . . . .	34
Keyword Differences . . . . .	34
Arrays and Functions . . . . .	36
The Conversion Operators . . . . .	37
Memory Allocation . . . . .	38
Accessibility and Visibility . . . . .	38
Properties, Events, and Delegates . . . . .	39
Generics . . . . .	39
Built-in Types . . . . .	40
Summary . . . . .	40

### ■ CHAPTER 4   **C# to C++/CLI** . . . . . 41

Add the Keyword namespace After using . . . . .	43
Add the Punctuator ^ to the Reference Type Declarations . . . . .	43
Modify the Built-in Types . . . . .	44
Change the Array Declarations and Type Parameters . . . . .	45
Change the Enumeration Declarations . . . . .	45
Change the Object Allocations and Instantiations . . . . .	46
Change the Pass by Reference Parameters . . . . .	46
Change the Class Member Access Operators for Handles . . . . .	47
Change the Separators for Namespaces and Static Accesses . . . . .	47
Change the Class Declarations . . . . .	48
Add the Function main() . . . . .	48
Summary . . . . .	50

<b>CHAPTER 5</b>	<b>Tools</b>	51
	Lutz Roeder's .NET Reflector	51
	What Is Reflection?	51
	Lutz Roeder's .NET Reflector	52
	Installing and Loading .NET Reflector and the C++/CLI Add-in	54
	Executing .NET Reflector	56
	Instant C++	57
	Tools Shipped with Visual Studio	58
	Microsoft .NET Framework IL Disassembler (ILDasm)	58
	Dependency Walker (Depends)	58
	More Visual Studio Tools	59
	Summary	60
<b>CHAPTER 6</b>	<b>Data Types</b>	61
	C# Types vs. C++ Types	61
	The C++ struct Keyword	62
	Native Classes	62
	Value Types and Reference Types	62
	Dynamic Memory Pools	63
	Garbage Collection	64
	Initialization	64
	Boxing	73
	Constructor Forwarding	77
	C# Partial Classes	78
	Reference Types on the Stack	79
	Basic Types	79
	Basic Type Differences	79
	Missing Keywords	80
	Marshaling Required	80
	Summary	82
<b>CHAPTER 7</b>	<b>Arrays</b>	83
	Native Arrays	83
	Managed Arrays	84

Managed Array Details .....	84
A Simple Example .....	84
Arrays as Parameters and Return Values .....	86
Sophisticated Example .....	91
Higher Dimensional Arrays .....	93
Native Arrays .....	97
Summary .....	97

## ■ CHAPTER 8    **Polymorphism and Protection** ..... 99

Polymorphism .....	99
Inheritance .....	100
Interfaces .....	102
Abstract Classes .....	105
Sealed Classes .....	106
Static Classes .....	106
Methods .....	107
Virtual Methods .....	107
Working with Methods .....	109
Virtual Methods Summary .....	115
Accessing Base Class Fields and Methods .....	115
Protection Mechanisms .....	117
Visibility .....	117
Accessibility .....	119
Inheritance .....	119
Declaring ref struct and ref class .....	122
Overload Resolution .....	123
Hide by Name and Hide by Signature .....	124
Summary .....	126

## PART 2 ■ ■ ■ **Details**

### ■ CHAPTER 9    **Pointers and Unsafe Code** ..... 129

The C# View: A Blessing and a Curse .....	129
Pointers: A Definition and a Caveat .....	130
Valid Targets and Syntax .....	131
Common Pointer Operators .....	132
Example of Pointer Usage .....	132
Verifiable Code and Pointer Usage in C# .....	132

Side Effects of Writing Unsafe Code . . . . .	133
Pointer Usage in C++ . . . . .	134
Verifiable Code in C++ . . . . .	134
C++ Handles . . . . .	137
C++/CLI Address Operators . . . . .	138
Complex Examples . . . . .	139
Nightmares . . . . .	140
Summary . . . . .	140
<b>CHAPTER 10 Properties and Events . . . . .</b>	<b>141</b>
A Basic Example Using Properties in C# . . . . .	141
A Basic Example Using Properties in C++/CLI . . . . .	143
A Look Inside the Grammar . . . . .	144
Trivial Properties . . . . .	145
Indexed Properties . . . . .	146
C# Properties . . . . .	149
Advanced Properties of Properties . . . . .	151
Property Caveats . . . . .	158
Miscellaneous Property Details . . . . .	159
Events and Delegates . . . . .	159
Delegates . . . . .	160
Multicast Delegates . . . . .	166
Instance Delegates . . . . .	167
Events . . . . .	168
Summary . . . . .	170
<b>CHAPTER 11 Expressions and Operators . . . . .</b>	<b>171</b>
Operator Overloading . . . . .	173
Complex Numbers, a Basic Example . . . . .	174
A Mathematical Diversion: Numbers Modulo Primes . . . . .	179
Implicit and Explicit Conversions of Built-in Types . . . . .	182
User-Defined Conversions . . . . .	188
CLS-Compliant Operators . . . . .	190
Summary . . . . .	194
<b>CHAPTER 12 The End of the Beginning . . . . .</b>	<b>195</b>
Include Files . . . . .	195
Forward Declarations . . . . .	195
The Problem . . . . .	196

The Solution . . . . .	196
Include File Caveats . . . . .	196
Types of Declarations . . . . .	197
Scope Resolution Operator . . . . .	199
Nullable Types . . . . .	201
Examples in C# and C++/CLI . . . . .	201
The ?? Operator in C# . . . . .	203
Under the Hood . . . . .	204
Checked Expressions . . . . .	205
Anonymous Methods . . . . .	206
Context-Sensitive Keywords . . . . .	206
Method Group Conversion . . . . .	207
Constructor-Initialized Variables . . . . .	208
Expression Statements with No Effect . . . . .	209
Exceptions . . . . .	210
Basic Exception Handling . . . . .	210
Function-Try Blocks . . . . .	213
Summary . . . . .	215

## ■ CHAPTER 13 Fun, Fun, and More Fun . . . . . 217

Dropping Light Bulbs . . . . .	217
Initial Thoughts . . . . .	217
A Rough Approximation . . . . .	218
A Little Algebra . . . . .	218
Discovering a Recursive Algorithm . . . . .	219
Implementation . . . . .	221
First Drop . . . . .	222
Bridge Crossing . . . . .	222
Background . . . . .	222
Algorithm and Implementation . . . . .	223
The Code . . . . .	224
Taking the Algorithm for a Spin . . . . .	226
Dealing with Cannibals . . . . .	226
Assumptions . . . . .	227
The Code . . . . .	227
Summary . . . . .	229



## PART 3 ■ ■ ■ Advanced Concepts

■ CHAPTER 14	<b>Generics</b> .....	233
	A Queue of Tasks .....	233
	C# Implementation Under .NET 1.0 .....	233
	Moving to Generics .....	239
	Creating Generic Classes in C++/CLI .....	252
	Summary .....	257
■ CHAPTER 15	<b>Introduction to Templates</b> .....	259
	Comparison with Generics .....	259
	Syntactic Differences .....	260
	Summary .....	260
	Compilation Process .....	261
	Templates Don't Really Exist .....	263
	Constraining Freedom Is a Good Thing .....	264
	The Template Paradigm .....	264
	Specialization .....	265
	Partial Specialization .....	266
	Partial Ordering of Function Templates .....	268
	Nontype Template Parameters .....	270
	Complex Numbers .....	273
	Overview .....	273
	Mathematical Formulas .....	274
	Generic Implementation .....	274
	Templates to the Rescue .....	275
	Core Differences: Templates and Generics .....	279
	Summary .....	279
■ CHAPTER 16	<b>Advanced Generics and Casts</b> .....	281
	Constraints .....	281
	Kinds of Constraints .....	281
	Type Constraints .....	282
	Satisfying Type Constraints .....	284
	The gnew Constraint .....	285
	The value class Constraint .....	285
	The ref class Constraint .....	286
	Summary .....	287

Cast Operators . . . . .	288
Runtime Type Identification . . . . .	288
const_cast<>() . . . . .	289
static_cast<>() . . . . .	290
dynamic_cast<>() . . . . .	290
reinterpret_cast<>() . . . . .	292
safe_cast<>() . . . . .	292
An Adder . . . . .	293
Generic Delegates . . . . .	295
Summary . . . . .	296

## ■ CHAPTER 17 The Preprocessor . . . . . 297

C# Preprocessor Commands . . . . .	297
Code Regions . . . . .	297
Conditional Code Compilation . . . . .	298
C++ Preprocessor Commands . . . . .	298
Full Text Substitution Language . . . . .	299
Debugging Support . . . . .	299
Function Syntax . . . . .	299
Concatenation Operator . . . . .	300
String Operator . . . . .	301
Macros on Macros . . . . .	301
Special Preprocessor Predefined Macros . . . . .	303
#define . . . . .	304
#undef . . . . .	305
Conditional Directives . . . . .	305
#include . . . . .	305
#using . . . . .	306
#error . . . . .	306
#pragma . . . . .	306
Some Useful Pragmas . . . . .	307
Summary . . . . .	314

## ■ CHAPTER 18 Native C++ . . . . . 315

The iostream Library . . . . .	315
Pointers to Members . . . . .	316
Operators new and delete . . . . .	319
/Zc:forScope . . . . .	321

/Zc:wchar_t . . . . .	322
Default Parameters . . . . .	322
C++ Runtime Library Functions . . . . .	323
stdio.h . . . . .	323
stdlib.h . . . . .	324
Other Include Files . . . . .	325
STL . . . . .	326
vector . . . . .	326
deque . . . . .	327
list . . . . .	333
auto_ptr . . . . .	333
Summary . . . . .	333
<b>CHAPTER 19 Multiple Language Support . . . . .</b>	<b>335</b>
.NET Language Integration . . . . .	335
Gathering Metadata . . . . .	336
Tracking Students Example . . . . .	336
Summary . . . . .	343
.NET Interoperability . . . . .	343
Wait for the Beep . . . . .	344
C# Platform Invoke . . . . .	344
C++ Built-in Support . . . . .	346
Using C++ for C# InterOp . . . . .	347
Summary . . . . .	349
<b>CHAPTER 20 Final Thoughts . . . . .</b>	<b>351</b>
Standard Dispose Pattern . . . . .	351
Destructors and Finalizers . . . . .	352
Implementation in C# . . . . .	352
Implementation in C++/CLI . . . . .	354
For Further Study . . . . .	355
Function Objects . . . . .	355
Special Pointers . . . . .	357
Interior Pointers . . . . .	357
Pinning Pointers . . . . .	358
Templates Revisited . . . . .	359
Class Hierarchy Pitfalls . . . . .	361
Type Aliases (typedef) . . . . .	364

Friends..... 366

Volatile Data..... 366

Constant and Mutable Data..... 367

Attributes..... 368

More on IntelliSense and Source Browsing ..... 369

Summary ..... 369

■ INDEX ..... 371