



CONTENTS SUMMARY

Acknowledgments xv

Introduction xvii

CHAPTER ONE

Components **1**

CHAPTER TWO

The Interface **15**

CHAPTER THREE

QueryInterface **37**

CHAPTER FOUR

Reference Counting **63**

CHAPTER FIVE

Dynamic Linking **85**

CHAPTER SIX

HRESULTS, GUIDs, the Registry, and Other Details **103**

CHAPTER SEVEN

The Class Factory **131**

CHAPTER EIGHT

Component Reuse: Containment and Aggregation **159**

CHAPTER NINE

Making It Easier **211**

CHAPTER TEN

Servers in EXEs **247**

CHAPTER ELEVEN

Dispatch Interfaces and Automation **279**

CHAPTER TWELVE

Multiple Threads **309**

CHAPTER THIRTEEN

Putting It All Together **341**

Index 361



TABLE OF CONTENTS

<i>Acknowledgments</i>	xv
<i>Introduction</i>	xvii

CHAPTER ONE

Components 1	
Component Benefits	3
Application Customization	3
Component Libraries	4
Distributed Components	4
Component Requirements	5
Dynamic Linking	6
Encapsulation	6
COM	9
COM Components Are...	9
COM Is Not...	10
The COM Library	11
The Way of COM	11
COM Exceeds the Need	11
Component Conclusions	13

CHAPTER TWO

The Interface 15	
Interfaces Are Everything	16
Reusing Application Architectures	17
Other Benefits of COM Interfaces	17
Implementing a COM Interface	18
Coding Conventions	19
A Complete Example	20
Non-Interface Communication	24
Implementation Details	25
Interface Theory, Part II	27
Interfaces Don't Change	27
Polymorphism	27

Behind the Interface	28
Virtual Function Tables	28
vtbl Pointers and Instance Data	30
Furthermore, Multiple Instances	31
Different Classes, Same vtbls	32
The Building Blocks, Summarized	34

CHAPTER THREE*QueryInterface* **37**

Querying for Interfaces	38
About <i>IUnknown</i>	39
Getting an <i>IUnknown</i> Pointer	40
Getting to Know <i>QueryInterface</i>	40
Using <i>QueryInterface</i>	41
Implementing <i>QueryInterface</i>	41
Everything All Together Now	46
<i>QueryInterface</i> Rules and Regulations	52
You Always Get the Same <i>IUnknown</i>	52
You Can Get an Interface If You Got It Before	53
You Can Get the Interface You Have	53
You Can Always Get Back to Where You Started	54
You Can Get There from Anywhere If You Can Get There from Somewhere	55
<i>QueryInterface</i> Defines the Component	56
You Can't Use the Whole Truth in Any One Moment	56
Handling New Versions of Components	57
When to Make a New Version	59
Naming Interface Versions	59
Implicit Contracts	60
Do You Have Legs?	60

CHAPTER FOURReference Counting **63**

Lifetime Control	64
Reference Counting Overview	65
Reference Counting Interfaces	67
Implementing <i>AddRef</i> and <i>Release</i>	69

When to Reference Count	77
Optimizing Reference Counting	77
Reference Counting Rules	80
The Firefighter's Coat, in Summary	83

CHAPTER FIVE

Dynamic Linking 85	
Creating the Component	86
Exporting a Function from a DLL	86
Loading the DLL	88
Breaking the Monolith	92
Code Listings	93
Bunches of Objects	98
Inflexible Creation, a Summary	100

CHAPTER SIX

HRESULTSs, GUIDs, the Registry, and Other Details 103	
HRESULTSs	104
Looking Up HRESULTSs	106
Using HRESULTSs	108
Defining Your Own Codes	110
GUIDs	111
Why Use a GUID?	112
Declaring and Defining GUIDs	113
Comparing GUIDs	115
Using GUIDs as Component Identifiers	115
Passing GUIDs by Reference	116
The Windows Registry	116
Registry Organization	117
The Registry Editor	118
The Bare Minimum	118
Other Registry Details	119
ProgID	120
Self-Registration	122
Component Categories	123
OleView	125

Some COM Library Functions	125
Initializing the COM Library	126
Memory Management	127
Converting from Strings to GUIDs	127
Summary	128

CHAPTER SEVEN

The Class Factory 131	
CoCreateInstance	132
The Declaration of <i>CoCreateInstance</i>	132
Using <i>CoCreateInstance</i>	133
The Class Context	133
Client Code Listing	135
But <i>CoCreateInstance</i> Is Inflexible	137
Class Factories	138
Using <i>CoGetClassObject</i>	138
<i>IClassFactory</i>	139
<i>CoCreateInstance</i> vs. <i>CoGetClassObject</i>	140
Class Factories Encapsulate Creation	141
Implementing the Class Factory	141
Using <i>DllGetClassObject</i>	142
The View from Above	142
Component Code Listing	143
Control Flow	151
Registering the Component	153
Multiple Components in a DLL	154
Reusing Class Factory Implementations	154
Unloading the DLL	156
Using <i>DllCanUnloadNow</i>	156
<i>LockServer</i>	156
Summary	157

CHAPTER EIGHT

Component Reuse: Containment and Aggregation 159	
Containment and Aggregation	160
Containment Overview	161
Aggregation Overview	162
Containment Contrasted with Aggregation	162

Implementing Containment	163
Extending Interfaces	167
Implementing Aggregation	169
<i>QueryInterface</i> Magic	169
Incorrect <i>IUnknown</i>	171
Unknown Interfaces for Aggregation	173
Creating the Inner Component	178
Outer Component Pointers to Inner Component Interfaces	180
A Complete Example	183
Blind Aggregation	200
Aggregation and Containment in the Real World	203
Providing Information About Internal States	204
Simulating Virtual Functions	205
Summary	207

CHAPTER NINE**Making It Easier 211**

Client-Side Simplification	211
Smart Interface Pointers	212
C++ Wrapper Classes	224
Server-Side Simplification	226
The Unknown Base Class	227
The Class Factory Base Class	231
Using <i>CUnknown</i> and <i>CFactory</i>	238
Putting All the Pieces Together Step by Step	243
Summary	245

CHAPTER TEN**Servers in EXEs 247**

Different Processes	247
The Local Procedure Call (LPC)	249
Marshaling	250
Proxy/Stub DLLs	250
Introduction to IDL/MIDL	252
About IDL	253
Example Interfaces in IDL	253
The MIDL Compiler	259

Local Server Implementation	264
Running the Example	264
No Entry Points	265
Start Your Factories	265
Modifying <i>LockServer</i>	269
Going Remote	271
What Does DCOMCNFG.EXE Do?	273
But How Does It Work?	274
Other DCOM Information	274
Summary	277

CHAPTER ELEVEN

Dispatch Interfaces and Automation	279
A New Way of Talking	280
The Old Way of Talking	280
<i>IDispatch</i> , You Dispatch, We All Dispatch	281
Using <i>IDispatch</i>	285
Invoke Arguments	287
Examples	292
The VARIANT Type	294
The BSTR Data Type	297
The SAFEARRAY Data Type	298
Type Libraries	299
Creating a Type Library	299
Using Type Libraries	302
Type Libraries in the Registry	304
<i>IDispatch</i> Implementation	305
Raising Exceptions	307
Marshaling	308
What Do You Want to Do Today?	308

CHAPTER TWELVE

Multiple Threads	311
COM Threading Models	312
Win32 Threads	312
COM Threads	313
The Apartment	313

Apartment Threads	317
Free Threads	317
Marshaling and Synchronization	318
Implementing Apartment Threading	320
Automatic Marshaling	321
Manual Marshaling	322
It's Code Time	323
A Walk Through the Apartment-Thread Example	324
Implementing Free Threading	331
A Walk Through the Free-Thread Example	333
Free-Threading Marshaling Optimization	338
Threading Registry Keys	340
Summary	340
CHAPTER THIRTEEN	
Putting It All Together 343	
The Tangram Program	344
Running Tangram	345
Pieces and Parts	345
The Client EXE	347
The <i>TangramModel</i> Component	348
The <i>TangramGdiVisual</i> and <i>TangramGLVisual</i> Components	348
The <i>TangramGdiWorld</i> and <i>TangramGLWorld</i> Components	349
Demonstrations	350
IDL Files	351
The DLLDATA.C File	352
Circular Reference Counts	352
Don't Call <i>AddRef</i>	353
Use Explicit Termination	354
Use a Separate Component	354
Events and Connection Points	355
<i>IEnumXXX</i>	358
COM Is About Standard Interfaces	359
Phweeeeeee!	359
<i>Index</i>	361