

.NET Enterprise Solutions

"Interoperability" for the connoisseur

TO

South-Africa and its
passionate IT community



Overall Content

Introduction.....	1
Interop Decomposed.....	11
Technology Chatter.....	21
Interoperability Challenges and Countermeasures	83
Technology Journey Simplified.....	175
Indigo in Easy Steps.....	219
Reference Blueprints & Building Blocks	289
Code Snippets	329
Appendix	343
Glossary	365
Meet the Authors	369
Tables & Figures	373
Index	379

Table of Contents

Introduction.....	1
Audience and Prerequisites.....	2
Foreword.....	3
Acknowledgements.....	4
Introduction.....	5
Conventions in this book.....	6
Structure of this handbook.....	7
DRP Web Site.....	8
Corrections and Comments.....	8
System Requirements.....	8
Quick Reference Guidelines.....	9
Interop Decomposed.....	11
Introduction.....	12
What is interoperability?.....	12
What is Integration?.....	15
Interpreter Integration.....	15
Postal System Integration.....	16
What are interoperability strategies?.....	17
What are interoperability standards?.....	17
What is coupling?.....	17
So what are we saying?.....	18
Conclusion.....	19
Quick Reference Guidelines.....	20
Technology Chatter.....	21
Introduction.....	22
Is Technology a good or bad thing?.....	23
LPC, RPC and QPC.....	24
Technologies supporting Service Oriented Architectures.....	24
Service Oriented Architecture (SOA).....	24
Fundamentals of service tenets.....	24
Elements of the Service Oriented Architecture.....	26
Service Oriented Architecture as an oxymoron.....	27
The Evolution up to and including SOA.....	27
Distributed Application Development (DAD).....	28
So is SOA not actually the same as DAD?.....	29
.NET, COM, COM+ Historical Timeline.....	30
COM.....	32
What is it?.....	32
How does it work?.....	32
Why use it?.....	34
DCOM.....	35
What is it?.....	35
How does it work?.....	35
Why use it?.....	36
COM+.....	37
What is it?.....	37
What features does it introduce?.....	37

Transactions	37
Administration	38
Just-in-time activation (JIT).....	38
Loosely coupled events.....	38
Object construction string.....	39
Object pooling	39
Queued components.....	39
Role-based security	39
Shared Property Manager	39
SOAP services	39
Synchronisation	39
How does it work?	39
Why use it?	40
.NET Web Service.....	41
What is it?	41
Why use it?	41
How does a .NET Web Service work?	42
SOAP as used by Web Services	43
Overview	43
Document Literal versus RPC-Encoded	44
Document/literal Messages.....	44
RPC/encoded Messages	44
RPC/literal Messages.....	45
So which encoding style should we use?	45
Web Service Enhancements.....	46
Web Services interoperability in the real world	46
Interoperability between Web Service vendors and the WS-* Glue.....	47
OASIS.....	47
Common WS-I Standards.....	47
Which should we focus on?.....	48
The roadmap of Web Services to Indigo	49
WSE and Indigo Overview.....	49
Call to action	50
Open Systems Interconnect (OSI) versus Web Services	50
Technologies (some...) used to develop web services	51
.NET Remoting.....	52
What is it?	52
How does it work?	53
Object Activation	53
Singleton Example	54
SingleCall Example	54
Walkthrough: Server Activated – Singleton.....	54
Walkthrough: Server Activated – SingleCall.....	56
Walkthrough: Client Activated	57
Why use .NET Remoting?.....	57
Messaging.....	58
What is messaging?.....	58
When should we use messaging?.....	58

When should we not use messaging?.....	59
MSMQ	59
What is it?	59
What features does it introduce?.....	59
Queue	59
Message	59
Rule.....	59
Trigger.....	59
Why use it?	60
What about interoperability?	60
MQ-SERIES.....	60
Messages.....	60
Queues	60
Understanding Clients and Servers	61
How do we use it?	61
Windows Communication Foundation (Indigo).....	62
Indigo	62
Introduction to the Microsoft Indigo product	62
Definition of Indigo	62
Indigo Architecture	63
What are the objectives of Indigo ... in our opinion?.....	63
Overview	63
What are the Indigo benefits?	66
“ABC” of Indigo.....	66
Introduce development using Indigo.....	67
Generics	67
Simple Unboxing/Boxing & Generics Example.....	68
Lightweight Transaction Manager	70
Indigo Development.....	72
Why should we consider Indigo?	73
What are the areas of caution?.....	73
Thoughts from one of the Chamomile Tea Breaks	74
IBM Service Component Architecture (SCA) for Java	75
SCA Introduction	75
SCA Motivation	75
SCA Architecture	76
Extensibility	76
Summary of SCA Model Characteristics.....	77
WCF and SCA.....	78
Similarities.....	78
Differences.....	79
SCA Conclusion	79
Conclusion.....	80
Quick Reference Guidelines	81
Interoperability Challenges and Countermeasures	83
Introduction	84
Setting the scene ... “A simple business application scope”	84
Overview	84

Design Specification	85
Environment	85
“enMatrix” - Overall tenets of designing an enterprise solution	86
Service Oriented Architecture	86
Relevance of plumbing with regards to WCF/Indigo.....	86
Components and Services	86
Inversion of Control.....	86
Service Locator versus Dependency Injection	88
Separating Configuration from Use	89
Have we met the top 10 objectives?	89
Service Oriented Infrastructure	92
“What are the differences or similarities between SOA and SOI (I being Infrastructure)?”	94
Taking a step back ... understand the landscape and context	95
Setting The “Context”	95
Fallacies of Distributed Computing.....	96
Introduction.....	96
The Network is reliable	96
Latency is Zero.....	96
Bandwidth is infinite	97
The Network is Secure	98
Topology does not change.....	98
There is one Administrator	99
Transport cost is zero.....	99
The network is homogeneous.....	100
Summary.....	100
Challenges.....	100
Distributed Objects: Reminiscing.....	100
SOA Timeline	100
Services.....	101
Software Projects – A Summary	102
Design to program against an interface	103
Design Decisions.....	103
Basic Approach	104
Design Tenets.....	104
Abstraction.....	104
Layering	105
Managing Complexity with Architecture	105
Creating Complexity.....	105
Complexity Option 1: Sweep It under a Rug (encapsulate).....	106
Complexity Option 2: Hide it in a Crowd (repository).....	107
Complexity Option 3: Ignore It	108
Complexity Option 4: Slice It (Horizontal Layering)	109
So what does this mean?	110
Configuration.....	111
Back to basics	111
Separating Configuration from Use	112
Main concepts.....	112
Factory Interfaces.....	112

Injection attributes & configuration decoration	112
Injection factories	113
Getting objects!	114
“ioMatrix” – Tenets that are affected by or impact interop	115
What Interoperability is not?.....	115
Is it about being Conventional?	115
It is about Integration?.....	115
So what is it?	116
RPC Interoperability	117
Contexts	118
Interoperability	119
Indigo (WCF).....	121
Form Fulfils Promise	123
Is RPC the right form?.....	125
Interop in Time and Space (Collaboration Structures)	126
Interoperability Solutions.....	132
J2EE to .NET	132
Taking Interop from "Here" to "There"	132
Key approaches to J2EE-to-.NET interoperability	132
Interop Best Practices for Each Approach	133
Java and .NET platforms	134
Introduction	134
Java/.NET Interoperability Options.....	134
Loosely Coupled Integration	134
Tightly Coupled Integration	135
Target Usage Scenarios.....	136
Suns Interoperability with WCF.....	136
Choosing a Transport	137
Transports and Message Encoders.....	137
Decision Points for Choosing a Transport	138
Decision Points for Choosing a Message Encoder.....	139
Unforeseen events?	140
What about Errors?.....	140
Moving Lots of Data	141
The Indigo Approach.....	142
Interoperability By Design.....	142
A few notes about transports in Indigo	142
Cake.....	142
Contract.....	144
Why bother with a contract?	144
Looking at the Service ABC	144
Basics	144
SLE and SLA are sycophantic twins	145
The business model (context) right ... understand it!.....	145
The ABC	146
Context, Contract or Policies?	147
Service Contracts.....	147
WS Standards.....	148

Verbose or Generic interface or mix thereof?.....	148
Patterns.....	151
Patterns pre-date IT	151
Effective pattern adoption	152
Integration patterns.....	152
Design Patterns.....	153
Versioning ... a fallacy?.....	157
What about interoperability?.....	158
What comes first ... code or contract?.....	158
Wrap-Up of contracts	159
Transactions.....	161
Centralised Environments	161
Distributed Environments.....	163
WS-Transaction.....	165
Our conclusion on transactions.....	165
A hypothetical solution step-by-step, based on en/ioMatrix.....	166
A final view of the hypothetical solution.....	166
enMatrix.....	168
ioMatrix	168
Contract.....	169
Transaction.....	170
So what happened to Interoperability?.....	171
The story within a story of a mislead probe	171
Analyse, design and develop a probe which will take our team to Pluto.....	171
The realisation that the solution is not perfect ... it does not meet business requirements..	171
Conclusion ... the answer to where interoperability has gone	171
So why should we care about interoperability in the first place?	172
Conclusion.....	172
Quick Reference Guidelines	173
Technology Journey Simplified.....	175
Introduction	176
.NET and Win32 Interoperability.....	177
COM, COM+ and .NET Interoperability.....	178
.NET and COM interoperability	178
Develop the COM Division Component	179
CDivision.cpp.....	179
CDivision.h	180
Use TLBIMP to generate the metadata.....	181
Develop the .NET Managed Client	182
Test.....	183
COM and .NET interoperability	184
Develop the .NET Component.....	185
Generate and register the type library in one step	186
Call .NET Managed Object from C++ based COM Component	187
Test.....	188
.NET and COM+ (Enterprise Services).....	189
Develop the .NET Component using COM+ Enterprise Services.....	189
Creating the Managed Component utilising Component Services.....	189

"Happy" and "Done" Bits.....	191
Create and register COM+ Application	191
Call COM+ Component from console test program	192
Test.....	193
Start the COM+ Application without specifying a Constructor string	193
Start the COM+ Application with a specified Constructor string.....	193
Run Test Application.....	194
Web Services and Web Service Extensions.....	195
Web Service Vanilla	195
Add Class	195
Web Service Façade	196
Console Test Client	199
But what about heterogeneous interoperability?	200
Web Service Extensions	200
What are Web Service Extensions?	200
Enhance the web service façade	200
Enhance the console client.....	205
In a nutshell	208
Database access and integration	209
Database Interoperability.....	209
Introduction	209
Overview	209
Dataset	209
Data Providers	209
Abstract Class.....	210
Factories	210
Connection String Configuration.....	210
Connection String Builder	211
Commands	211
GetSchema.....	212
Writing Database Interoperable Code.....	213
Introduction	213
Parameter Markers.....	213
Intrinsic Functions	213
Microsoft SQL Server	213
Oracle	213
Solution 1 - Stored Procedures	213
Solution 2 - Enterprise Libraries.....	214
.NET Remoting and IIOP	215
Firstly, what is CORBA / IIOP?	215
Bridging the divide	215
Looking at an example	215
Pitfalls.....	216
Conclusion	216
Conclusion.....	217
Technology Rating ... at a glance	217
Quick Reference Guidelines	218
Indigo in Easy Steps.....	219

Introduction	220
Indigo step-by-step	221
Basic Steps	221
Visual Studio Team System (VSTS) Creating Playing Ground	221
Introducing our demo solution.....	225
Calculator Web Service Façade.....	231
Basic Steps.....	231
Service Façade.....	231
1. Data, Interface and Message Contracts	231
2. Address, Binding and Contract Information (ABC).....	233
3. Service Code.....	233
4. SVC File Definition	235
5. Test the service.....	235
Test Client	236
6. Consumer test Application	236
Wrapping up by using VSTS Source Control	238
7. Secure Source Code.....	238
Addition Service.....	241
Basic Steps.....	241
Service	241
1. Update Contracts.....	242
2. Addition Service Configuration and Service Class	242
3. Update console program class	243
4. Generate proxy and configuration file	243
5. Update calculator façade	245
6. Test Addition Service Class.....	245
Division Service	247
Basic Steps.....	247
Service	247
1. Update contract, including custom fault	248
2. Create division service class.....	249
2. Update console program class.....	250
4. Generate proxy and configuration file	250
5. Update calculator facade	250
6. Test the division service.....	251
Multiplication Service.....	252
Basic Steps.....	252
Service	252
1. Update contract.....	253
2. Create multiplication service class	254
3. Update console program class.....	255
4. Generate proxy and configuration file	255
5. Update calculator facade	255
6. Test the multiplication service	257
Subtract Service.....	259
Basic Steps.....	259
Service	259
1. Update contracts	260

2. Create service framework	261
3. Add subtraction service class	261
4. Register certificates	262
5. Create server configuration	263
6. Install Service	264
7. Proxy and Calculation Client	264
8. Update calculator facade	265
Logger Service	267
Basic Steps.....	267
Service	267
1. Update contract.....	268
2. Create logger service class and configuration.....	268
3. Generate proxy and configuration code	270
4. Update calculator facade	274
5. Test logger service	274
Square Service	276
Basic Steps.....	276
Service	276
1. Create and configure COM+ Application.....	276
2. Create COM+ Indigo Façade	277
3. Create COM+ Indigo Proxy	279
4. Update calculator façade.....	279
5. Test service.....	280
Indigo Fx Tools.....	281
Overview of main tools.....	281
ComSvcConfig.exe	281
COM+ hosting mode	281
Web hosting mode.....	281
SvcConfigEditor.exe.....	282
SvcTraceViewer.exe	283
Basic Bindings Summary.....	285
Can we host two service interfaces at the same time?.....	286
Conclusion.....	287
Technology Rating ... at a glance	287
Quick Reference Guidelines.....	288
Reference Blueprints & Building Blocks	289
Introduction	290
EDRA	291
Introduction	291
Background.....	291
Installing and using EDRA.....	293
EDRA and Application blocks.....	293
Setting up a Simple Authentication Web Application.....	293
Conclusion	296
Connected Services Framework (CSF)	297
Overview	297
The Basics	297
BSS Integration.....	297

OSS Integration.....	298
Supporting products and technologies	298
CSF Core Architecture.....	299
E.L. Services	299
CSF Core Services	299
Session Management	299
Identity Management.....	300
Service Catalogue.....	300
Profile Management	300
Resource Management	300
Standard Business Events.....	300
An overview of CSF Framework development	300
The SDK.....	300
Step-by-step Web Service consumer calling a CSF Service	301
Enterprise Libraries	302
Introduction	302
History of application blocks and the road to the Enterprise Library	302
Configuration Application Block.....	305
Summary of Block	305
What does the Configuration Application Block do	305
MVP Serialiser Transformer	305
Coding the Configuration Block.....	307
Concerns and discussion on the Configuration Block	308
Benefits of Configuration Application Block in Integration	308
Logging and Instrumentation Application Block	309
Summary of Block	309
What does the Logging and Instrumentation Application Block do	309
Coding the Logging and Instrumentation Block.....	311
Configuring Asynchronous Logging.....	311
Tracing	312
Concerns and discussion on the Logging and Instrumentation Block.....	313
Benefits of Logging and Instrumentation Application Block in Integration.....	313
Exception Handling Application Block.....	314
Summary of Block	314
What does the Exception Handling Application Block do.....	314
Coding Exception Handling	314
Concerns and discussion on Exception Handling.....	317
Benefits of Exception Handling Block in Integration	317
Other application blocks in the Enterprise Library	317
The future of the Enterprise Library	317
Integration Patterns by Microsoft Platform Architecture Guidance (PAG).....	318
1 - Integration Layer	318
2 - Systems Connections.....	318
3 - Integration Topology	318
Creating an integration architecture using patterns step-by-step	318
Solution Overview.....	319
Integrating Layer	320
Systems Connections.....	321

Integration Topologies.....	322
The final picture	325
Conclusion.....	326
The end is near	326
Quick Reference Guidelines.....	327
Code Snippets	329
Introduction.....	330
Service Factory	331
Why this code.....	331
Base Factory Code	331
Factory Configuration	335
Resource Assemblies	336
Why this code.....	336
Resource File.....	336
Source Snippet – The new old World.....	336
Old World.....	336
New World.....	336
Debugger Attribute	337
Why this code.....	337
Code Snippet	337
Simple Polymorphism.....	339
Service Security.....	340
Why this code.....	340
Code Snippet	340
Appendix	343
Bibliography & References.....	344
Cool Links	345
Cool Shortcuts.....	346
Quick Reference & Poster Map.....	347
Poster Extracts.....	348
Glossary	365
Meet the Authors	369
Willy-Peter Schaub.....	369
James Pereira.....	370
Meet the Team.....	371
Tables & Figures	373
Figures.....	373
Tables.....	377
Index	379

1



Readiness- Take your marks. Get set and keep on going

Introduction

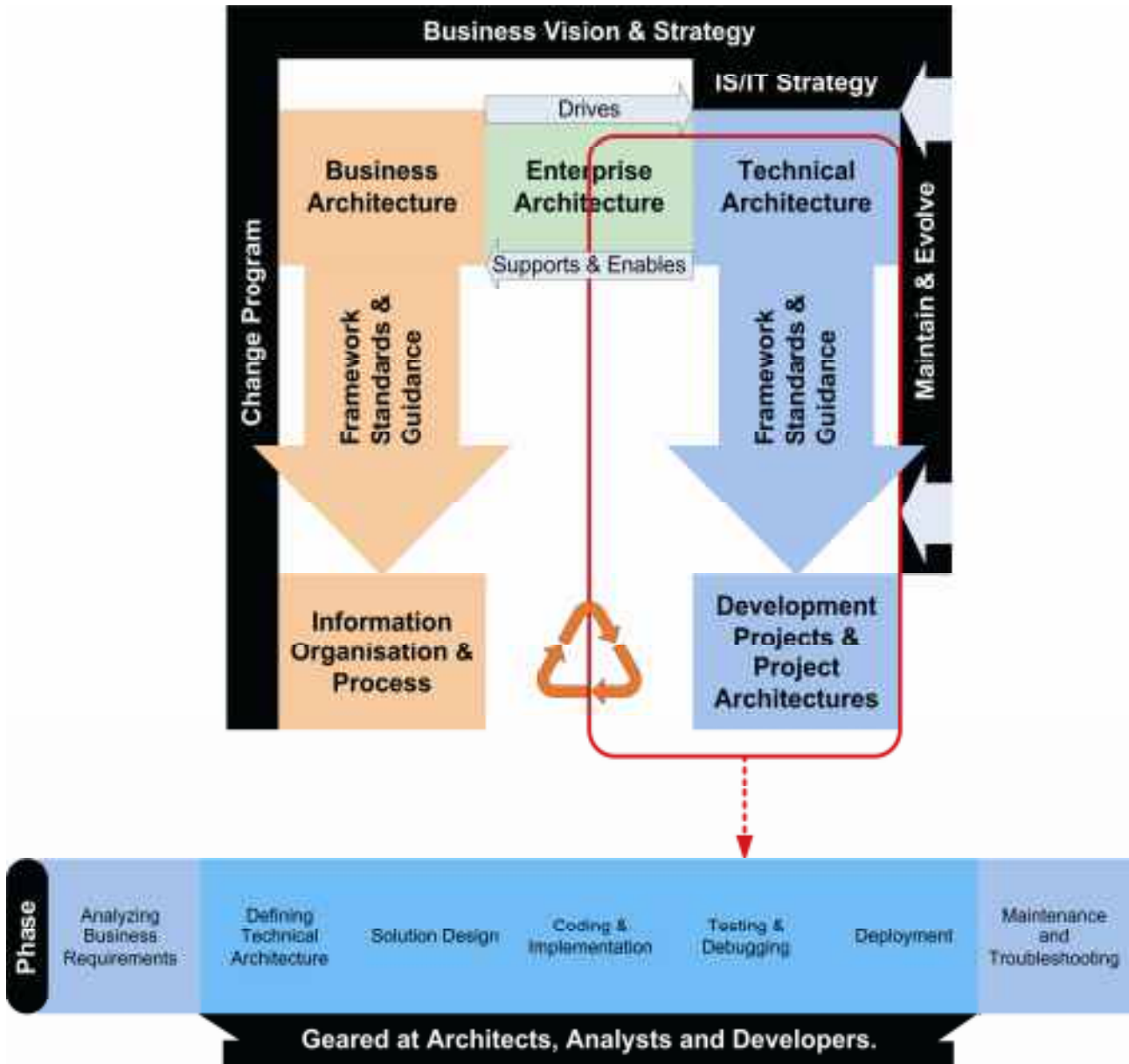
Objectives

- ✓ ... Introduce the team that created this handbook.
- ✓ ... Introduce the system requirements for solutions on the companion solution.
- ✓ ... Introduce the structure and contents of this handbook.

Audience and Prerequisites

Technology is like an express train that never slows down. Come up to speed using this book. It is intended to help with just that by providing an information exchange for developers by developers. It primarily covers the terrain of Interoperability and Integration through the core phases as shown below, and in doing so it touches on various architectures to some extent with the world of Enterprise Architecture being central.

The content is relevant to most stakeholders of a typical enterprise solution shown below. (The relevance of the sections is outlined in "Figure 2 - Structure of this handbook" on page 7.)



Developers should have at least one year of design and programming experience in VB, C# , C++ or Java and be familiar with traditional and object-oriented programming concepts such as user-defined structures, classes, inheritance and interfaces. Experience with COM, Win32 and ASP is a benefit, but is not required. Sample code and labs will primarily be in C# , so preliminary experience with C# is an advantage.

Architects will find the content valuable in all chapters except perhaps the last few with the 'How To' and Code Snippets.

Foreword

Following on from .NET Enterprise solutions – Best Practices for the Connoisseur, this next book again represents a wealth of practical experience and presents an excellent insight into the issues of interoperability and integration using current technologies.

The debate of how best to achieve interoperability and integration will continue to rage on in the world of technology with ever changing and competing technologies from major vendors and varying approaches from the software architects and designers of the world. With this rapid change it is vital that organizations achieve interoperability between new and old systems and technologies.

Willy-Peter Schaub, James Pereira and the supporting authors have taken the current topics of interoperability and integration and made sense of both the technologies and the philosophy behind it all. It has been fascinating to discuss these issues with James and Willy and to see their thinking and knowledge come alive in the pages of this book.

Peter Searle
M•TU Executive and Director of Barone, Budge and Dominick



Acknowledgements

We thank Barone, Budge and Dominick (BB&D) for providing an environment that made this book possible, from committing dedicated time, booking a boardroom for us for nearly an entire year in which we could conveniently hide and escape from the daily IT responsibilities and for allowing the usage of the ASTRA research laboratory for the evaluation and experimentation of development and interoperability technologies described in this book.

We thank the following content contributors:

Hennie Walters
David Morrison
Ernst Marais
Peter Scheffel

IOP and Remoting
MQ-Series Overview
MSMQ Overview
Enterprise Libraries, EDRA

We thank the following artists:

Tony Hulsenbeck
Alexander Schaub

Cartoonist
Shark artwork on pages 22 and 175

We thank our "patient" editor:

Geoff Bruce

Content & continuation editing

We also thank the following people for their input. Any errors or omissions in the pages to follow are most certainly not their doing ... Carola Schaub, Charles Klein, Charles Verschuur, Clemens Vasters, Craig McMurtry, Dave Svab, Dr. Lucas Dreyer, Gavin Warrener, Jan van der Eecken, Jean-Pierre Fouche, Johan Drenth, Maxime Papazian, Neville Wright, Peter Searle, Ralf Dominick, Richard Oates, Robert McMenamin, Ruari Plint, Salim Naim, and Tarin Blankfield.

We thank our spouses and children for their efforts, support and immense sacrifices in allowing us to conclude this project against all odds, battling immense time management and technology monsters on the long, yet exciting journey.

A special thanks also goes to the BB&D Microsoft Technology Unit (M•TU), the best team in this universe, which supported us throughout this adventure by proof reading, organising food, printing numerous draft copies and even family/house sitting when some of us were working far away from home base.

We welcome your comments and ideas towards improving this material. This book contains a vast amount of experience gathered over the years by the authors and contributors.

Willy-Peter, James

Introduction

This book is predominantly intended for Developer Readiness Program (DRP)¹ engagements and for IT professionals who read and research on their own. It is organized as a reference and self-study guide, with objectives, guidelines and introductions to more advanced material and topics.

Our previous book “.NET Enterprise Solutions ... Best Practices for the connoisseur, ISBN: 0-620-33013-9” was focused primarily on the smartProcess SDLC and best practices applicable to each stage of a solution.

In this book we are continuing the discussion and sharing of best practices, but are focusing specifically on interoperability and integration of similar and dissimilar (heterogeneous) solutions. Rather than specialising on any one concept or technology, this book intentionally presents a “broad scan” introduction to a variety of technologies within that scope.



Figure 1 – Solution stakeholders at work

History seems to be repeating itself. In the mid 1980's there was a huge growth of remote terminal to distant host communications, with the associated demand for fundamental techniques of integration and interoperability. Various protocols and standards emerged or gained further popularity -- for example, RS232, RS449, RS422, NetBIOS and TCP/IP. Now twenty years later we are faced with similar challenges, except that the growth is more explosive, resulting in solutions that are technically more complex and challenging to develop, interconnect and maintain.

We are convinced, especially with the recent drive of service oriented architectures, that the challenges in the immediate and long-term future will be the areas of interoperability and integration. There is a vast amount of existing and emerging solutions that need to communicate and benefit from each other seamlessly, with minimal effort or overhead on the solution teams who are better off focusing on business functionality, than battling with plumbing issues which will only get worse.

¹ "So what is the DRP?" You are probably asking yourself. Well, it is an initiative we started together with Microsoft South-Africa to get developers together for a few days of information sharing and experimenting with new technologies. While it was initially intended to focus on the area of training, it has fortunately dominated the proof of concept environments, where rapid technology skills acquisition and information/experience sharing are crucial.

We explore Microsoft Indigo² as part of the journey and share our experiences gathered with the technology while researching it during the Microsoft Indigo early technology adoption and evaluation program. The technology is of great interest to us as it promises unification of several existing Microsoft technologies, support for cross-vendor interoperability, and explicit service-orientation.

It must be emphasised that the best practices and concepts covered in this book are technology agnostic and are therefore relevant in most technological environments of today.

As shown the book is split into three main areas, namely (1) the introduction of concepts and technologies, (2) a review of challenges and experiences in the interoperability and integration space today and tomorrow and (3) how to use some of the technologies. Great care has been taken to include the entire "how to", the best practices, the "gotchas" and experience we have gathered during early adoption programs and solution development projects.

Conventions in this book



This symbol is used to highlight recommendations and guidelines.



This symbol is used to highlight reference material.



This symbol is used to highlight alerts.



This symbol is used to highlight concepts.

This book includes the following features in every chapter, which are both conducive to learning and quick reference of information.

- **Objectives** offer an outline of the concepts discussed in the chapter.
- **Notes**, indicated by the symbols above, highlight important information regarding the concepts introduced in the chapter.
- **Code Examples** within most chapters explain key concepts with relevant code. Each line of the example code is numbered for easy reference.
- **Quick reference summary** of recommendations ending with a brief review of the information covered in the chapter.

² Indigo is the product code name and is used as the product had not been released at the time this book was created.

Structure of this handbook

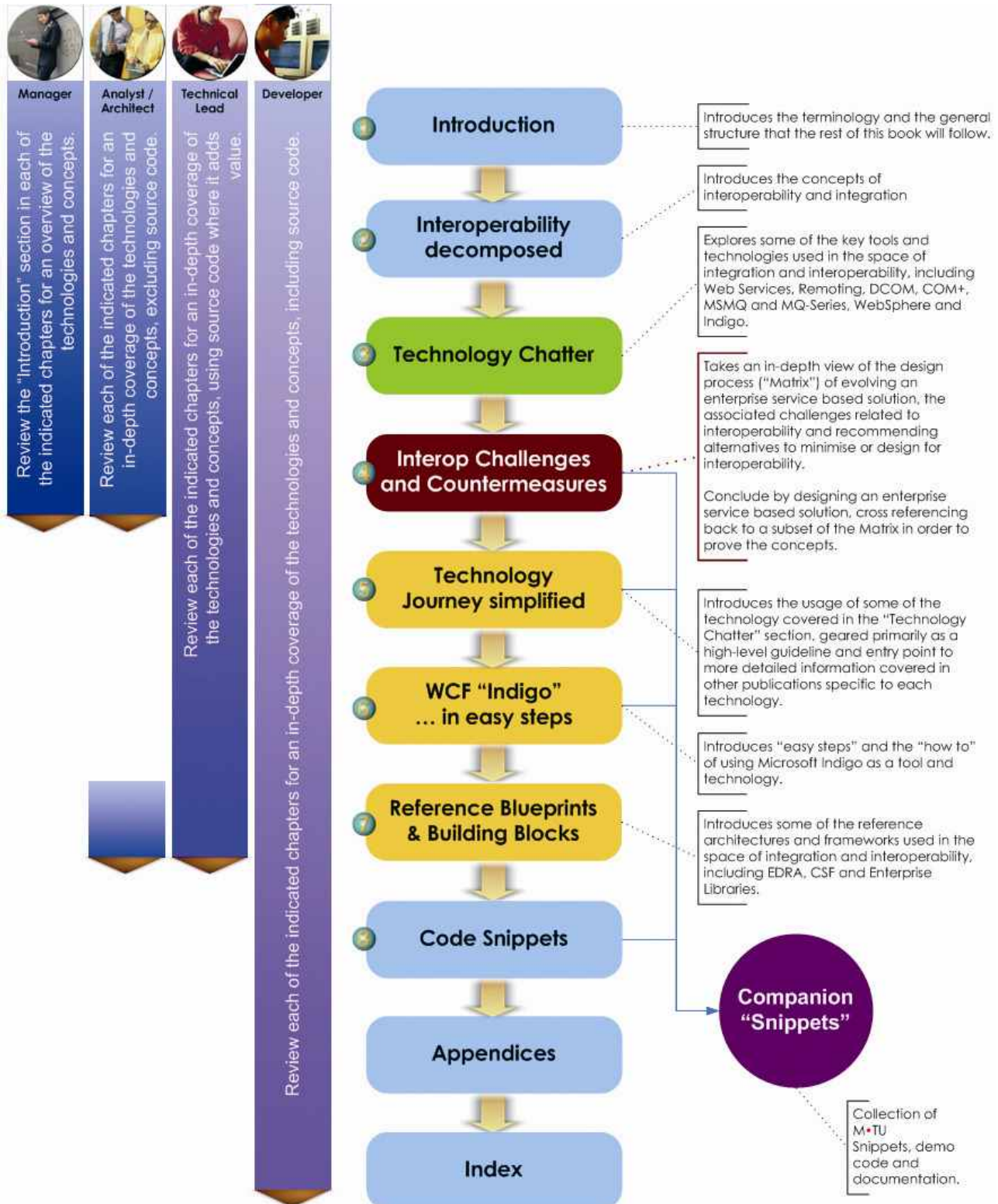


Figure 2 - Structure of this handbook

DRP Web Site

You are also invited to visit us on <http://www.drp.co.za> and <http://www.bbd.co.za>.

Corrections and Comments

Every effort has been made to ensure the accuracy of this book and of the demo and companion solutions.

If you have issues, comments or recommendations regarding this handbook, demo or the companion solutions, please email them to:

drpsupport@bbd.co.za

-or- mail them to:

Barone, Budge and Dominick (Pty) Ltd
Attention: DRP Authors
P.O. Box 72420
Parkview
2122
Republic of South Africa

Support for Microsoft .NET technologies is not offered through the preceding address. For such support please refer to <http://support.microsoft.com>.

System Requirements

You will need the following minimum hardware and software to complete the solutions described in this book:

- ☞ Microsoft Visual Studio .NET 2005 Professional, BETA 2 or higher, or Visual Studio 2005 Team Suite (recommended).
- ☞ Microsoft WinFX BETA-2.
- ☞ A computer capable of running Microsoft Visual Studio.NET 2005.
 - Computer:
 - PC with a Pentium III 1GHz processor
 - Operating System:
 - Windows 2003 Server
 - or-
 - Windows XP Home or Professional (recommended)
 - Memory:
 - 256MB (1GB recommended)
 - Hard Disk:
 - 5GB free disk space
 - Other Drives:
 - CD-ROM
 - Video:
 - VGA or higher resolution monitor
 - Input Device:
 - Keyboard and Mouse
- ☞ See <http://msdn.microsoft.com/vstudio> for more information.

Quick Reference Guidelines



- ☞ The following DRP quick reference posters are some of the posters that are relevant to this book. The latest versions of all the posters can be downloaded from www.drp.co.za:

0000 PosterMap
0000 System_Namespace
0001 Modelling Poster
0020 OO Poster
0022 Patterns 1
0024 Patterns 2
0024 smartProcess
0060 Data Access Building Block
0062 Enterprise Libraries
0064 Custom Enterprise Libraries
0090 Best Practices Interfaces
0092 Best Practices Performance
0094 Best Practices Security
0096 Best Practices SDLC
0098 Best Practices Security Checklist
0100 Best Practices Data Access
0202 Microsoft Reporting Services
0202 Microsoft Team System
0202 Microsoft Team System Security Mapping
0202 Microsoft Team System Single Server Deployment
0202 Microsoft Team System Source Control
0202 Microsoft Team System Modelling Flow
0202 Microsoft Team System Workspace
0204 Best Practices Patterns Poster
0206 Microsoft COMPlusPlus
0206 Microsoft Indigo Artefacts
0206 Microsoft Indigo Configuration
0206 Microsoft Indigo
0208 Best Practices Contract First
0208 Best Practices Summary