

PATTERN-ORIENTED SOFTWARE ARCHITECTURE

A Pattern Language for Distributed Computing

Volume 4

Frank Buschmann,
Siemens, Munich, Germany

Kevlin Henney,
Curbralan, Bristol, UK

Douglas C. Schmidt,
Vanderbilt University, Tennessee, USA



John Wiley & Sons, Ltd

PATTERN-ORIENTED SOFTWARE ARCHITECTURE

PATTERN-ORIENTED SOFTWARE ARCHITECTURE

A Pattern Language for Distributed Computing

Volume 4

Frank Buschmann,
Siemens, Munich, Germany

Kevlin Henney,
Curbralan, Bristol, UK

Douglas C. Schmidt,
Vanderbilt University, Tennessee, USA



John Wiley & Sons, Ltd

Copyright © 2007

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester,
West Sussex PO19 8SQ, England

Telephone (+44) 1243 779777

Email (for orders and customer service enquiries): cs-books@wiley.co.uk
Visit our Home Page on www.wileyeurope.com or www.wiley.com

All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except under the terms of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd, 90 Tottenham Court Road, London W1T 4LP, UK, without the permission in writing of the Publisher. Requests to the Publisher should be addressed to the Permissions Department, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, or emailed to permreq@wiley.co.uk, or faxed to (+44) 1243 770620.

Front cover Image Copyright © 2007 Yann Arthus-Bertrand/Altitude

Designations used by companies to distinguish their products are often claimed as trademarks. All brand names and product names used in this book are trade names, service marks, trademarks or registered trademarks of their respective owners. The Publisher is not associated with any product or vendor mentioned in this book.

This publication is designed to provide accurate and authoritative information in regard to the subject matter covered. It is sold on the understanding that the Publisher is not engaged in rendering professional services. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

Other Wiley Editorial Offices

John Wiley & Sons Inc., 111 River Street, Hoboken, NJ 07030, USA

Jossey-Bass, 989 Market Street, San Francisco, CA 94103-1741, USA

Wiley-VCH Verlag GmbH, Boschstr. 12, D-69469 Weinheim, Germany

John Wiley & Sons Australia Ltd, 42 McDougall Street, Milton, Queensland 4064, Australia

John Wiley & Sons (Asia) Pte Ltd, 2 Clementi Loop #02-01, Jin Xing Distripark, Singapore 129809

John Wiley & Sons Canada Ltd, 6045 Freemont Blvd, Mississauga, Ontario, L5R 4J3, Canada

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

British Library Cataloguing in Publication Data

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

ISBN-13: 978-0-470-05902-9 (hbk)

Typeset in 10/13 Bookman-Light by Laserwords Private Limited, Chennai, India

Printed and bound in Great Britain by Bell & Bain, Glasgow

This book is printed on acid-free paper responsibly manufactured from sustainable forestry in which at least two trees are planted for each one used for paper production.

For Anna, Bebé[†], and Martina

Frank Buschmann

For Carolyn, Stefan, and Yannick

Kevlin Henney

For Lori, Bronson, Mom, and Dad

Douglas C. Schmidt

[†] Bebé, July 3, 1999

Table of Contents

	Foreword	xv
	About This Book	xvii
	About The Authors	xxiii
	Guide To The Reader	xxvii
Part I	Some Concepts	1
1	On Patterns and Pattern Languages	3
1.1	Patterns Introduced	4
1.2	Inside Patterns	6
1.3	Between Patterns	10
1.4	Into Pattern Languages	13
1.5	Patterns Connected	15
2	On Distributed Systems	17
2.1	Benefits of Distribution	18
2.2	Challenges of Distribution	20
2.3	Technologies for Supporting Distribution	22
2.4	Limitations of Middleware	32
3	On the Pattern Language	33
3.1	Intent, Scope, and Audience	34
3.2	Origins and Genesis	35

3.3	Structure and Content	36
3.4	Presentation	44
3.5	Practical Use	49
Part II	A Story	53
4	Warehouse Management Process Control . .	57
4.1	System Scope	58
4.2	Warehouse Management Process Control . . .	60
5	Baseline Architecture	65
5.1	Architecture Context	66
5.2	Partitioning the Big Ball of Mud	67
5.3	Decomposing the Layers	68
5.4	Accessing Domain Object Functionality	71
5.5	Bridging the Network	72
5.6	Separating User Interfaces	76
5.7	Distributing Functionality	79
5.8	Supporting Concurrent Domain Object Access	82
5.9	Achieving Scalable Concurrency	85
5.10	Crossing the Object-Oriented/Relational Divide	87
5.11	Configuring Domain Objects at Runtime	89
5.12	Baseline Architecture Summary	90
6	Communication Middleware	95
6.1	A Middleware Architecture for Distributed Systems	96
6.2	Structuring the Internal Design of the Middleware	100
6.3	Encapsulating Low-level System Mechanisms .	103
6.4	Demultiplexing ORB Core Events	105
6.5	Managing ORB Connections	108
6.6	Enhancing ORB Scalability	111
6.7	Implementing a Synchronized Request Queue	114
6.8	Interchangeable Internal ORB Mechanisms . .	116

6.9	Consolidating ORB Strategies	118
6.10	Dynamic Configuration of ORBs	121
6.11	Communication Middleware Summary	124
7	Warehouse Topology	129
7.1	Warehouse Topology Baseline	130
7.2	Representing Hierarchical Storage	131
7.3	Navigating the Storage Hierarchy	133
7.4	Modeling Storage Properties	135
7.5	Varying Storage Behavior	137
7.6	Realizing Global Functionality	140
7.7	Traversing the Warehouse Topology	142
7.8	Supporting Control Flow Extensions	144
7.9	Connecting to the Database	146
7.10	Maintaining In-Memory Storage Data	147
7.11	Configuring the Warehouse Topology	149
7.12	Detailing the Explicit Interface	151
7.13	Warehouse Topology Summary	153
8	The Story Behind The Pattern Story	157
Part III	The Language	163
9	From Mud To Structure	167
	Domain Model **	182
	Layers **	185
	Model-View-Controller **	188
	Presentation-Abstraction-Control	191
	Microkernel **	194
	Reflection *	197
	Pipes and Filters **	200
	Shared Repository **	202
	Blackboard	205
	Domain Object **	208

10	Distribution Infrastructure	211
	Messaging **	221
	Message Channel **	224
	Message Endpoint **	227
	Message Translator **	229
	Message Router **	231
	Publisher-Subscriber **	234
	Broker **	237
	Client Proxy **	240
	Requestor **	242
	Invoker **	244
	Client Request Handler **	246
	Server Request Handler **	249
11	Event Demultiplexing and Dispatching . . .	253
	Reactor **	259
	Proactor *	262
	Acceptor-Connector **	265
	Asynchronous Completion Token **	268
12	Interface Partitioning	271
	Explicit Interface **	281
	Extension Interface **	284
	Introspective Interface **	286
	Dynamic Invocation Interface *	288
	Proxy **	290
	Business Delegate **	292
	Facade **	294
	Combined Method **	296
	Iterator **	298
	Enumeration Method **	300
	Batch Method **	302

13	Component Partitioning	305
	Encapsulated Implementation **	313
	Whole-Part **	317
	Composite **	319
	Master-Slave *	321
	Half-Object plus Protocol **	324
	Replicated Component Group *	326
14	Application Control	329
	Page Controller **	337
	Front Controller **	339
	Application Controller **	341
	Command Processor **	343
	Template View **	345
	Transform View **	347
	Firewall Proxy **	349
	Authorization **	351
15	Concurrency	353
	Half-Sync/Half-Async **	359
	Leader/Followers **	362
	Active Object **	365
	Monitor Object **	368
16	Synchronization	371
	Guarded Suspension **	380
	Future **	382
	Thread-Safe Interface *	384
	Double-Checked Locking	386
	Strategized Locking **	388
	Scoped Locking **	390
	Thread-Specific Storage	392

	Copied Value **	394
	Immutable Value **	396
17	Object Interaction	399
	Observer **	405
	Double Dispatch **	408
	Mediator *	410
	Command **	412
	Memento **	414
	Context Object **	416
	Data Transfer Object **	418
	Message **	420
18	Adaptation and Extension	423
	Bridge **	436
	Object Adapter **	438
	Chain of Responsibility *	440
	Interpreter	442
	Interceptor **	444
	Visitor **	447
	Decorator	449
	Execute-Around Object **	451
	Template Method *	453
	Strategy **	455
	Null Object **	457
	Wrapper Facade **	459
	Declarative Component Configuration *	461
19	Modal Behavior	463
	Objects for States *	467
	Methods for States *	469
	Collections for States **	471

20	Resource Management	473
	Container *	488
	Component Configurator *	490
	Object Manager **	492
	Lookup **	495
	Virtual Proxy **	497
	Lifecycle Callback **	499
	Task Coordinator *	501
	Resource Pool **	503
	Resource Cache **	505
	Lazy Acquisition **	507
	Eager Acquisition **	509
	Partial Acquisition *	511
	Activator **	513
	Evictor **	515
	Leasing **	517
	Automated Garbage Collection **	519
	Counting Handle **	522
	Abstract Factory **	525
	Builder *	527
	Factory Method **	529
	Disposal Method **	531
21	Database Access	533
	Database Access Layer **	538
	Data Mapper **	540
	Row Data Gateway **	542
	Table Data Gateway **	544
	Active Record	546
22	A Departing Thought	549