

Creating Agile Business Systems with Reusable Knowledge

Agility and innovation are necessary to achieve global excellence and customer value in twenty-first century business; yet most approaches to business process engineering in existence today sacrifice these in favor of operational efficiency and economics. Moreover, the IT systems used to automate and encapsulate business processes are inflexible and unable to respond to the constantly changing business environment. Mitra and Gupta provide insight to close this gap – they show how innovation can be systematized with normalized patterns of information, how business processes and information systems may be tightly aligned, and how these processes and systems can be designed to automatically adapt to change by re-configuring shared patterns of knowledge. The authors present a modular approach to building business systems that parallels that of object oriented software. They describe how business knowledge can be identified, encapsulated, and shared, as well as how reusable process modules can be developed to offer the systems flexibility. The book provides practical templates required for accelerating integration, analysis, and design. Mitra and Gupta lay the foundation of a new paradigm in which computers manipulate meanings, not blind symbols.

This book will appeal to consultants, analysts, and managers in IT firms looking to develop new, more flexible products for their clients. It will also be of interest to researchers and graduate students in business, management, and IT. By deeply integrating business knowledge and processes with IT systems design, this book is a valuable addition to both communities.

Amit Mitra is Managing Consultant at Headstrong LLC, in addition to President and Principal Consultant at Strybiz LLC. He is an alumnus of KPMG and former Chief Methodologist of the American International Group.

Amar Gupta holds a number of positions at the University of Arizona, Tucson. He is Professor of Entrepreneurship and MIS; Thomas R. Brown Chair in Management and Technology; and Senior Director for Research and Business Development at the Eller College of Management; and he is Professor of Computer Science in the College of Science. In addition, he serves as Visiting Professor, Engineering Systems Division, College of Engineering at the Massachusetts Institute of Technology (MIT).

Creating Agile Business Systems with Reusable Knowledge

A. Mitra and A. Gupta

CAMBRIDGE UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom
One Liberty Plaza, 20th Floor, New York, NY 10006, USA
477 Williamstown Road, Port Melbourne, VIC 3207, Australia
314-321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi - 110025, India
103 Penang Road, #05-06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org

Information on this title: www.cambridge.org/9780521851633

© A. Mitra and A. Gupta 2006

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2006

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

ISBN 978-0-521-85163-3 Hardback

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party internet websites referred to in this publication, and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

I dedicate this book to my father, Ajoy Mitra, my mother, Sevati Mitra, and my teachers, who helped shape my thoughts, to my wife Snigdha and my children Tanya and Trishna, who helped shape my life, and in turn shape this book.

Amit Mitra

I dedicate this book to my mother, my sister (Beena), my wife (Poonam), my children (Amrit and Amita), and to my teachers in India (especially at IIT, Kanpur and IIT, Delhi), the UK (at Birkbeck College and the Imperial College of Science and Technology), and the US (primarily at MIT).

Amar Gupta

Contents

<i>List of boxes</i>	<i>page</i>	ix
<i>List of figures</i>		xi
<i>List of tables</i>		xiv
<i>Foreword</i>		xv
<i>Preface</i>		xvii
<i>Acknowledgements</i>		xix
Introduction		1
1 What is this book about and who should read it?		1
2 What will the information be used for?		2
3 Technology's broken promise		2
4 Component reuse – <i>the genesis</i>		4
5 Scope of this book		6
6 Foundation of knowledge reuse: three pillars		7
7 How this book is organized		11
1 On the nature of reality and the nature of business		14
1 The nature of knowledge		16
2 Modeling the real world		20
3 Metaworld of information		21
4 Basic metaobject inventory		39
5 Metaobjects and the natural repository of knowledge		39
6 The architecture of knowledge and the scope of the metamodel in this book		43
2 The object at the root of it all		51
1 Object class versus object instance		52
2 The state of an object		55
3 Inheriting behavior – subtypes, supertypes, and partitioning of objects		79
4 The problem of perspective		94
5 Repositories of meaning		103

viii	Contents	
3	The nature of attributes	112
	1 The structure of attributes and states	112
	2 Attribute constraints	122
	3 Naming and expressing attributes	158
4	Domains and their expression	170
	1 The meaning and architecture of format	172
	2 The meaning of units of measure	234
	3 Domains of meaning and the metamodel of object	252
	4 Storing abstract meaning	324
	Appendix: Key shared components of knowledge	327
	<i>Bibliography</i>	354
	<i>Index</i>	383

Boxes

1	Example of the process engineering value chain	<i>page 5</i>
2	Example of a model for baking a cookie	21
3	Objects, domains, and formats	27
4	Mr. Domain's calculations	31
5	Conversion between UOMs	35
6	The architecture of knowledge	44
7	The architecture of knowledge reuse can help make information systems flexible and scalable	47
8	How rules shift between business process automation and technology layers	48
9	Business definition of an object	51
10	Properties of objects	60
11	States, attributes, state variables, and type indicators: much ado about nothing	60
12	The state of a system is the collective state of the objects it involves	66
13	Set membership	67
14	Object instances must exist for a finite period of time once they are created (often forever after they reach some terminal state)	68
15	Mathematical definition of a business object	71
16	Domains and measures of distance	72
17	State spaces with qualitative attributes	75
18	Chaotic behavior	78
19	Set operations	81
20	Inheritance, state space, and polymorphism	82
21	Exclusion partitions, variation inheritance, and polymorphism	86
22	UML syntax for partitions and subtypes	89
23	Subtyping criteria	93
24	Perspective is an object	96
25	An example of the problem of perspective	99
26	Synonyms and homonyms	101
27	The OAV model and the structure of attributes	114
28	Components of inclusion and exclusion sets	128
29	The structure of bounds and ranges	133
30	Set versus list	134
31	Dormant ranges and containers for value sets	135

x	List of boxes	
32	Cardinality, cardinality ratio, and object counts	141
33	Relationships between attributes, meanings, and expressions	146
34	Identifying domains	160
35	The many faces of meaning	164
36	When one object represents another	173
37	Location in space	187
38	Metamodels of format, format conversion, encryption, and formatting constraint	234
39	Full formats of values	243
40	Measure conversion	248
41	Conflicting subtypes, state spaces, perspectives, and polymorphisms of metaobjects	251
42	Domain analysis and primary physical domains	264
43	The principle of subtyping by adding information	269
44	Domain analysis for components of knowledge	281
45	Partial order, fuzzy meaning, and the scaling of derived domains	284
46	“Softness” of information	293
47	Domains, relationships, and the Cartesian product	296
48	Domain rules ready reckoner	314
49	The information content of domains – new learning and changing perspectives – an example in color	322

Figures

1	Information is a key resource, but investment in information systems is fraught with risk	page 3
2	Reusing business knowledge: the three pillars	8
3	Analysis paralysis: only a few critical rules, reused often, connect the business of the enterprise but they are lost in a tangled web of minutiae	10
4	More and more, past experience is under pressure from the demands of scale, scope, and agility that businesses are placing on systems	15
5	Knowledge is the meaning of business practices, rules, goals, and guidelines and how they fit into an integrated whole	17
6	Processes are a special kind of relationship: they carry information on “before and after” effects between objects	23
7	How is information naturally manifest in the real world?	24
8	(Partial) metamodel of domain	34
9	Basic metaobject inventory: <i>kinds of rules each metaobject normalizes</i>	40
10	A rule: organization ships product assembled from objects	41
11	Adding components to assemble configurations of rules	42
12	The black box perspective of behavior	44
13	Process decomposition	45
14	Node–branch representation	45
15	The architecture of knowledge	46
16	Business process automation is only one of several mechanisms that implement abstract business rules in the physical world	48
17	The architecture of reusable knowledge components	49
18	The scope of the metamodel of knowledge in this book is focused on pure business rules	49
19	Object classes are collections of object instances with common properties	53
20	Object classes are collections of object instances with common properties (state chart perspective)	56
21	Disallowed effects, default, and “maybe” states	57
22	The state of an object changes in response to discrete events	70
23	Example of two-dimensional state space	73
24	Example of a three-dimensional state space	74
25	Example of trajectory through state space	78

xii **List of figures**

26	Example of object partitions	84
27	Constraints across partitions	88
28	Perspective is a point of view	95
29	Name is an object class linked to perspective	103
30	Primary names, perspectives, and aliases	103
31	Fundamental metamodel of object	105
32	Expanded inventory of metaobjects	106
33	Views are aggregate objects	109
34	An attribute is the overlap between an object and a domain	113
35	The structure of attributes and metamodel of state	117
36	Attributes can represent an object's participation in relationships	119
37	Attributes and values inherit properties of domains	120
38	Attributes values may be constrained	124
39	Inclusion and exclusion sets are mutually exclusive partitions	125
40	Subtypes of ordinal domains may contain both discrete values and ranges	130
41	Metamodel of value set	133
42	Metamodel of value constraint	139
43	The state of the check is constrained jointly by two other attributes	143
44	A quantitative relationship relates check amount, monthly rental, and energy charges	143
45	A rule constraint is an aggregate object	145
46	Intermediate rules for mapping object instances to object population	148
47	Constrain is a recursive relationship	155
48	Metamodel of "rule constrain"	156
49	Integrated metamodel of value constraint	157
50	Patterns are symbols that consist of symbols	177
51	Examples of finite delimited and undelimited patterns	185
52	Metamodel of measure of similarity	201
53	Kinds of patterns	207
54	The metamodel of pattern is the source of universal properties, subtypes and partitions of pattern	211
55	Perspectives of object – two-dimensional cross sections of a three-dimensional array	217
56	Candu Compoot's three-dimensional array	218
57	Integrated metamodel of pattern	222
58	Structure of language	225
59	Format conversion is a recursive aggregate relationship	230
60	The metamodel of format maps values to symbols	231
61	The metamodel of format conversion maps symbols to symbols	231
62	Metamodel of representation	233
63	Two maps working in tandem map quantitative values to physical symbols	236
64	Metamodel of unit of measure	244

xiii **List of figures**

65	Structure of unit of measure conversion rules	247
66	Conflicting subtypes?	249
67	Information hierarchy of domains	259
68	The metamodel of domain	312
69	Shifting perspectives of color	319

Tables

1	Proximity metrics in spaces of different kinds	<i>page</i> 200
2	Valid measures of proximity in mixed spaces	203
3	A two-dimensional slice of Candu's nine-dimensional array	218

Foreword

The health and well-being, and today the very survival, of an enterprise depend on its ability to respond and adapt in *timely, innovative, and effective* manners. The relatively static behaviors of the past have been changed unalterably by the explosion of *telecommunication* and *information technologies/capabilities* as typified through the emergence of the World-Wide Web (W³). Enterprises are learning to adapt to the challenges of the new global business and national security environment by exploiting the same capabilities that are driving the *dynamic* environment, telecommunications and IT. In essence, *information* and the *knowledge* derived therefrom have emerged as key *assets of the enterprise* in responding and adapting to the demands of the global environment.

The experiences over the past decade for a wide variety of enterprises, including both governmental and commercial entities, are reflected by *more failures* than successes in embracing successful strategies and solutions for creating, engineering, and evolving the *knowledge system* that serves the enterprise most effectively. Certainly, problems have arisen through failures of *leadership and management*, who have been unable to break the static behaviors and narrow organizational views that served them well in the past. On the other hand, *engineers*, given responsibility for exploiting information technologies to meet the information needs of leadership and management for knowledgeable decision making, have found the challenges of dealing with the *complex event-driven environments* and the *complex* array of enterprise *stakeholders* and *systems* vastly more difficult than the systems engineering problems of the past.

From both successful and failed efforts, there is an ever-growing body of knowledge about some of the keys to successfully reengineering the global enterprise as a flexible and adaptive entity. The concepts that are seeing increasing attention include enterprise architecting, service-oriented architectures, business process modeling, enterprise and e-business patterns, enterprise systems engineering, and agile development methods. All of these approaches and methods contribute to one or more of the fundamental advantages that are driving the developments. These advantages span a number of enterprise dimensions and can be summarized in the following way:

- *Strategic focus*: provides a basis for understanding the contributions of complex, large, distributed information systems in achieving enterprise goals and missions.
- *Broadened communications*: enhances communication across the enterprise community from leaders/managers/users to engineers/developers/testers.

xvi **Foreword**

- *Performance and QoS*: improves performance and quality of service (QoS) for decision support and knowledge-based decision making.
- *Timely and flexible response*: enables flexibility for timely and effective response to new and unexpected situations.
- *Integration and interoperable operations*: enables mechanisms for assuring integrated and interoperable applications, both among legacy and new systems.
- *Commercial technology evolution*: facilitates the introduction and effective use of rapidly changing commercial information systems and technology.
- *Cost-effective migration*: establishes a foundation for value-based thinking, analysis of alternatives, and investment planning to establish cost-effective system evolution.
- *Organizational efficiencies*: allows organizational efficiencies due to reduced staffing requirements, easier system evolution process, etc.

But from reviewing and assessing a myriad of enterprise developments, it is apparent that a much deeper understanding is needed to increase the probability of success for enterprises working to meet the challenges of the global environment. In this book, complemented by their earlier book, *Agile Systems with Reusable Patterns of Business Knowledge*, the authors provide a very comprehensive and integrated perspective on the range of topics mentioned above. Starting from basic principles, the book presents an approach to enterprise reengineering that merits careful attention and thoughtful application. As they say in the Preface, the book provides a description of a “hidden and elegant theoretical framework: a framework that is a direct bridge between business process engineering and systems engineering.” The approach that is presented is ambitious and provocative, and commands thoughtful consideration from developers and researchers in this field of ever-increasing importance.

Harold W. Sorenson

*Professor of Engineering Systems
Faculty Director
Graduate Program in Architecture-based
Engineering of Enterprise Systems
Jacobs School of Engineering
Rady School of Management
University of California, San Diego
La Jolla, CA 92037*

*[Former Chief Scientist, US Air Force;
Former Senior Vice President and General
Manager for Air Force Systems at MITRE;
Former Chief Engineer for AF Electronic
Systems Center]*

Preface

Why this book? – Because it is a book *begging* to be written. The real world is chaotic and never stands still. Businesses constantly strive to re-invent themselves under continual, and often intense, pressures of competitive, regulatory, and technological change. The pivotal issue in business computing lies in incorporating new learning in automated information systems; adding to what is already known and adapting automatically as perspectives and priorities continually change. This is the challenge for which we have sought the answers presented in this book.

Where did it all start? It was 1992. One of the authors was the Chief Methodologist for the American International Group, an unusual global corporation that believed in turning on a dime. The firm needed a systems development discipline to facilitate nimble and innovative business practices. Thus a truly exciting and wonderful journey began – a journey we want to share with you. In this book, you will find readily usable patterns and models you can leverage to establish business requirements, object models, and knowledge bases to support the agile and exacting business needs of the twenty-first century. You will also find the exciting and simple beauty of a framework that is the direct bridge between business process engineering and systems engineering. Yes, it is a proven framework that works for every industry and business application we have tested it on – from telecommunications to insurance, from financial services to manufacturing.

In the chapters that follow, we will share with you not only how this framework works in practice, but also how it actually anticipates key requirements even before users articulate them, such as those that flowed from strategic shifts in the regulatory bedrock of the US telecommunications industry. The theoretical foundation of the approach is not only deep, but also elegant in its simplicity.

Where will it eventually lead? In the short term, it can make your business more agile. It can provide reusable models, processes, and business knowledge components to compress your time to market new or improved products, services, and processes. It can also show you how you can compress systems development and integration times. However, it is the vision at the end of this journey that is the most fascinating of all. The concepts in this book can provide the foundation of disciplines that can make business systems truly maintenance free – systems based on software that can automatically adapt to change and chaos. These systems can be supported by automated intelligent agents¹ that will, some day, maintain

¹ The “Intelligent agents” section of the Bibliography at the end of this book lists papers that describe agent technology and the-state-of-the-art.

xviii **Preface**

software and respond to environmental change at the speed of thought, a vision we will share with you at the end of this book.

As practitioners, managers, and teachers in the field of information systems, we often talk of change control. Change plays havoc with our plans and products. However, the wealth in the knowledge economy will flow from global excellence, thriving on change and innovation. The only justification for technology will, and must be, change facilitation, not change control. Are we ready?

“Wouldst thou,” – so the helmsman answered

“Learn the secret of the sea?

Only those who brave its dangers

Comprehend its Mystery!”

(Henry Wadsworth Longfellow,

The *Galley of Count Arnaldos*)

Acknowledgements

We deeply appreciate the time that Dr. E. C. Subbarao, our former teacher, spared from his busy schedule to help shape this book. We thank Srividhya Subramanian, Yan An Surendra Sarnikar, Shivram Mani, and Prithi Avanavadi for going through this entire manuscript carefully. We are grateful to Jayant Pal and Raghunandan Dhat of Setu Inc. for their enthusiasm in instantiating the concepts in software, and to Pat Fayad, for keeping the faith, and her unflagging encouragement as this book was being written. Finally, one of the authors (Amar Gupta) would like to thank the Engineering Systems Division of the Massachusetts Institute of Technology (MIT), especially Professor Daniel Hastings and Professor Joel Moses, for providing a Visiting Professorship and other support in 2005 and 2006 when this book was being finalized.