

Liang-Jie Zhang  
Jia Zhang  
Hong Cai

**Services Computing**

Liang-Jie Zhang  
Jia Zhang  
Hong Cai

# Services Computing

With 218 figures



Liang-Jie Zhang  
Ph.D., Research Staff Member  
IBM T.J. Watson Research Center  
19 Skyline Drive, Hawthorne,  
NY 10532, USA  
Email: zhanglj@us.ibm.com  
<http://www.research.ibm.com/people/z/zhanglj8>

Jia Zhang  
Ph.D., Assistant Professor  
Department of Computer Science  
Northern Illinois University  
DeKalb, IL 60115, USA  
Email: jiazhang@cs.niu.edu  
<http://www.cs.niu.edu/~jiazhang>

Hong Cai  
Ph.D., Research Staff Member  
Services Research  
IBM China Research Lab  
Zhongguancun Software Park  
Haidian District  
Beijing 100094, P.R. China  
Email: caihong@cn.ibm.com

---

ISBN 978-7-302-15075-6 **Tsinghua University Press, Beijing**  
ISBN 978-3-540-38281-2 **Springer Berlin Heidelberg New York**

---

Library of Congress Control Number: 2007930066

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable to prosecution under the German Copyright Law.

© 2007 Tsinghua University Press, Beijing and Springer-Verlag GmbH Berlin Heidelberg  
**Co-published by Tsinghua University Press, Beijing and Springer-Verlag GmbH Berlin Heidelberg**

**Springer is a part of Springer Science+Business Media**  
springer.com

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover design: Frido Steinen-Broo, EStudio Calamar, Spain  
Printed on acid-free paper

---

# Preface

---

Services Computing has become a cross-discipline subject that covers the science and technology of bridging the gap between Business Services and IT Services. Its most recent enabling technology is Web services centered on Service-Oriented Architecture (SOA). This book depicts an overall picture of the state-of-the-art of the field. A comprehensive set of innovative research results and solution methods is described and discussed, including business componentization, services modeling, services creation, services realization, services annotation, services deployment, services discovery, services composition, services delivery, service-to-service collaboration, services monitoring, services optimization, services management, business consulting methodology and utilities, business process modeling, transformation, integration, and management.

## What Is the Uniqueness of This Book?

This book introduces innovative ideas and solutions based on existing key techniques and industry standards in the field of Services Computing. In particular, this book illustrates Services Computing as an emerging interdisciplinary science and engineering subject bridging the gap between business/application services and IT services. It presents a lifecycle view of modern services industry, discusses up-to-date innovative research directions and industry solutions in the domain of Services Computing. It explains how to effectively and efficiently establish, operate, and manage business and application services using Services Computing, and it also guides research directions of Services Computing.

There have been numerous publications in the market regarding Web services and Service-Oriented Architecture from specifications and standards perspectives. To our knowledge, however, this book is the first that provides a systematic view of SOA solutions and SOA services to enable the lifecycle of modernized services businesses and applications. In our view, Services Computing is not merely a technical direction;

instead, it is an interdisciplinary area aiming to bridge the gap between IT and business. Therefore, it is not only necessary, but also critical to consider Services Computing from strategic point of view. Moreover, Service-Oriented Computing is just a pure technology fraction of Services Computing that also includes services consulting methodologies, services design and service delivery, as well as services maintenance and management.

As for implementation, there have emerged a number of industry standards and specifications for Web services, such as Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP), Universal Description, Discovery, and Integration (UDDI), Business Process Execution Language for Web Services (BPEL4WS). However, from our point of view, these existing specifications are just examples of infrastructure enabling technologies for Services Computing environment. With the development of Services Computing, these specifications and technologies will continuously be evolving into their next generation or be replaced by new technologies and standards.

Throughout this book, instead of repeating the existing specifications, we concentrate on introducing innovative frameworks and methods on how to leverage related technologies to address real business challenges. The existing technologies are used as examples to study the state-of-the-art of the field and can be used as starting points for further innovations. Along with the newly introduced ideas in this book, the present enabling technologies provide a comprehensive framework that can be used to construct domain-specific SOA solutions. It should be noted that the existing technologies may have to be adjusted, extended, and customized in accordance with particular execution contexts and business requirements.

Finally, this is a foresighted book intended to spur researchers, practitioners, and students into further explorations and investigations in the field of Services Computing. As SOA and services engineering become mainstream, there are numerous efforts underway in both academia and industry, all of which deliver concepts and technologies in the same or similar fashion. This book aims to guide readers to grasp the foundations and state-of-the-art developments in the field of Services Computing.

## **Who Should Read This Book?**

### **Researchers and students**

The audience first includes researchers, graduate students, and senior undergraduates who seek a systemic introduction to the key technologies and research innovations in the field of Services Computing. This book can be used as an introductory textbook, advanced undergraduate textbook, graduate textbook, continuing education textbook (e.g., for executive MBA), or supplemental reading materials in classrooms.

In addition, this book can be used as a reference book on advanced technologies for a set of existing courses such as Modern Software Engineering, Web Engineering,

Web Technologies, Advanced Software Engineering Methodologies, Advanced Software Architecture, and so on. Targeting departments include Department of Computer Science, Department of Industrial Engineering, Department of Business Management, Department of Automation, and Department of Management of Information System.

This book is organized in a way that is suitable for students to learn the Services Computing concepts and technologies step by step. It is written in a way that it can be used as a classroom textbook, as well as a self-study reference book.

### **Engineers and managers**

As Services Computing is being widely accepted by the business world, practitioners who are interested in building value-added services or solutions based on SOA will become suitable audiences. Companies that either develop software using SOA or intend to introduce SOA and Web services in business could use this book as a reference book for their software engineers, IT managers, business managers, salesmen, and IT and business executives.

## **Outline of This Book**

### **Part 1: Foundations of Services Computing**

This part introduces core techniques of Web services modeling, registry, and discovery. SOA paradigm is discussed, along with SOA solution architecture based on industry best practices. Current SOA and Web services standard stack is also presented. Advanced techniques are introduced including multi-dimensional services modeling, dynamic services invocation, federated services discovery, services relationship modeling, and solution-level Quality of Service (QoS) in SOA.

### **Part 2: Realization of Services Computing**

This part introduces services realization technologies from four perspectives: requirements-driven services composition, services value chain collaboration, business process management and integration, as well as business grid.

### **Part 3: Services Delivery and Services Engineering**

This part introduces technologies and methodologies for services delivery and engineering from four perspectives: project-based business performance management, service-oriented business consulting methodology, end-to-end services delivery

platform and methodology, as well as software as services and services as software.

## **Acknowledgements**

All the authors would like to acknowledge the invaluable cooperation of many collaborators who have been involved in the research projects in the past years. Those projects served as a basis for some research findings that underlie several of the ideas discussed in this book.

Liang-Jie (LJ) Zhang would like to send his sincere appreciation to Fausto Bernardini for his sturdy support and insightful directions. Special thanks also go to his management team: Robert Morris, Alain Azagury, Manoj Kumar, and David Cohn.

LJ would like to send special thanks to his previous collaborators in the past years: Henry Chang, Tian Chao, Jen-Yao Chung, Bing Li, Haifei Li, Yu Long, John Y. Sayah, Jing Min Xu, Shun Xiang Yang, Qun Zhou, and Ying Nan Zuo.

LJ would also like to appreciate his IBM SOA Solution Stack core team members for their insightful discussions: Ali Arsanjani, Abdul Allam, Kishore Channabasavaiah, and Michael Ellis. He would like to acknowledge the technical advice from Kerrie Holley, Ray Harishankar, George Galambos, and Shankar Kalyana during the past SOA projects, as well as valuable technical discussions with Sridhar Iyengar and Ed Kahan.

Finally, LJ would like to send his special thanks to Cesar A. Gonzales, Maria Azua, and Daniel Sturman for their invaluable encouragement and advice over the years.

Jia Zhang would like to send special thanks to Dr. Raimund K. Ege for his invaluable support.

The authors would also like to thank Annie Wang for helping proofreading the book.

Of course, our most profound thanks go to our families for their continuous love and encouragement.

*To my wife, Qun, and my two daughters, Lan and Anna, with all my love.*

Liang-Jie Zhang

*To my dearest parents for their constant love and inspiration in my life.*

Jia Zhang

*I would like to express my sincere thanks to my dear wife Yi for her love and patience over the many evenings and weekends when I was completing this book and was unable to spend time with her and little daughter LuYi.*

Hong Cai

# Contents

## Part 1 Foundations of Services Computing

<b>1</b>	<b>The Principle of Services and Services Computing</b>	<b>3</b>
1.1	Introduction of Services	3
1.1.1	Definition of Services	3
1.1.2	Definition of Services System	4
1.2	Perspectives of Services Systems	6
1.2.1	Model	6
1.2.2	Technology	7
1.2.3	Architecture	7
1.2.4	Optimization	7
1.3	Services Lifecycle	7
1.3.1	Phase 1: Consulting and Strategic Planning	8
1.3.2	Phase 2: Services Engagement	8
1.3.3	Phase 3: Services Delivery	8
1.3.4	Phase 4: Services Operation	9
1.3.5	Phase 5: Services Billing	9
1.3.6	Phase 6: Services Management	9
1.4	Key Factors in a Services Lifecycle	9
1.4.1	Data/Information	9
1.4.2	Processes	10
1.4.3	People	10
1.4.4	Resources	10
1.4.5	Finance Factors	11
1.4.6	Knowledge and Skills	11
1.4.7	Innovation and Technology	11
1.5	Comparisons Between Services and Manufacturing Models	11
1.6	Business Services Lifecycle in Telecommunication Industry	13
1.7	Relationship Between IT and Non-IT Services in a Services Ecosystem	14
1.7.1	Overview of a Services Ecosystem	14
1.7.2	Comparisons Between IT and Non-IT Services	15
1.8	Emergence of Services Computing Technology	17
1.9	Summary	18
	References	19

<b>2</b>	<b>e-Business Evolution</b> .....	20
2.1	Phases of e-Business Adoption.....	20
2.2	Top Trends of e-Business .....	22
2.3	IT Innovations to Flatten the World .....	22
2.3.1	Interactive Multimedia Services.....	22
2.3.2	Office Online Services.....	25
2.3.3	Globalization of Businesses.....	26
2.4	“Open” Trends for Technologies and Services Ecosystems.....	27
2.4.1	Open Standards .....	28
2.4.2	Open Sources.....	28
2.4.3	Open Architecture: Service-Oriented Architecture (SOA).....	29
2.5	Debuts of New Service-Oriented Business Models .....	31
2.5.1	Services Modernization.....	31
2.5.2	Software as Services.....	32
2.5.3	Services as Software .....	33
2.6	New Discipline: Services Computing.....	34
2.7	Summary .....	35
	References .....	36
<b>3</b>	<b>Web Services Modeling</b> .....	37
3.1	Basic Concept of Web Services.....	37
3.2	Modeling of a Web Service.....	38
3.2.1	Basic Concepts of WSDL Modeling .....	38
3.2.2	Web Services Communication Protocol: SOAP .....	41
3.2.3	Binding of WSDL to SOAP .....	44
3.2.4	Publishing a Web Service in Registry.....	45
3.2.5	Stateful Web Services Modeling.....	46
3.2.6	Web Services Interoperability.....	50
3.3	Modeling a Composite Web Service.....	50
3.3.1	Basic Concepts of BPEL.....	51
3.3.2	BPEL Basic Structure: via an Example .....	51
3.3.3	BPEL Key Elements .....	57
3.4	Three-Dimensional Web Services Modeling.....	59
3.5	Discussions on Web Services Modeling.....	60
3.6	Summary .....	62
	References .....	62
<b>4</b>	<b>Web Services Publishing and Discovery</b> .....	64
4.1	Web Services Publishing .....	64
4.1.1	Public/Private UDDI Publishing.....	64
4.1.2	WSIL Publishing.....	65
4.1.3	UDDI Publishing vs. WSIL Publishing .....	67

4.2	Simple Web Services Discovery.....	67
4.2.1	Simple UDDI Search.....	68
4.2.2	Simple WSIL Search.....	70
4.2.3	Issues of the Simple UDDI/WSIL Search.....	70
4.3	UDDI Search Markup Language.....	71
4.3.1	USML Schema.....	72
4.3.2	Composite Search Options.....	74
4.3.3	Aggregation Operators.....	75
4.4	USML-Based Advanced UDDI Search Engine (AUSE).....	76
4.4.1	AUSE Structure.....	77
4.4.2	AUSE-Based Search Process.....	78
4.5	WSIL-Oriented Dynamic Services Discovery Framework (DSDF) ....	80
4.5.1	Architecture of DSDF.....	80
4.5.2	DSDF Implementation.....	83
4.6	Federated Web Services Discovery Framework.....	83
4.6.1	Basic Ideas.....	83
4.6.2	Search Language.....	85
4.6.3	Comparison with Generic Web Search Engine.....	86
4.7	Discussions on Web Services Publishing and Discovery.....	87
4.8	Summary.....	88
	References.....	88
<b>5</b>	<b>Service-Oriented Architecture.....</b>	<b>89</b>
5.1	Concept of Service-Oriented Architecture (SOA).....	89
5.1.1	Triangular SOA Operational Model.....	89
5.1.2	Web Services-Based SOA.....	90
5.2	Services Invocation.....	91
5.2.1	Simple Services Invocation.....	91
5.2.2	Introduction to MetaWSDL.....	93
5.2.3	MetaWSDL Publishing.....	97
5.2.4	MetaWSDL-Based Advanced Services Invocation Framework.....	98
5.3	SOA: Bridging Business and IT Architecture.....	99
5.4	SOA Solution Lifecycle.....	101
5.4.1	Modeling.....	102
5.4.2	Development.....	103
5.4.3	Deployment.....	103
5.4.4	Publishing.....	103
5.4.5	Discovery.....	104
5.4.6	Invocation.....	104
5.4.7	Composition.....	104
5.4.8	Collaboration.....	105
5.4.9	Monitoring and Management.....	105

5.5	Enterprise Service Bus (ESB).....	106
5.6	SOA Reference Architecture (SOA-RA).....	107
5.7	Discussions on SOA.....	111
5.8	Summary.....	111
	References.....	112
<b>6</b>	<b>Services Relationship Modeling</b> .....	<b>114</b>
6.1	Introduction to Services Relationship Modeling.....	114
6.1.1	UDDI Specifications on Simple Relationships.....	115
6.1.2	Other Relationship Specification Languages.....	115
6.2	Web Services Relationship Language (WSRL).....	117
6.2.1	Structure of a WSRL Document.....	117
6.2.2	WSRL Discussions.....	117
6.3	Layered Services Relationship Modeling.....	119
6.4	SOA-Based Relationship Modeling Language (SOA-RML).....	120
6.4.1	Business Services Relationships at Business Entity Level....	120
6.4.2	Business Services Relationships at Business Service Level...	125
6.4.3	Layered Relationships Summary.....	127
6.5	SOA-RML-Enriched Services Registry.....	130
6.5.1	SOA-RML Schema.....	131
6.6	Discussions on SOA-Based Relationship Modeling.....	132
6.7	Summary.....	132
	References.....	133
<b>7</b>	<b>SOA and Web Services Standards</b> .....	<b>134</b>
7.1	Introduction.....	134
7.2	Web Services Standard Stack.....	134
7.2.1	Transport.....	135
7.2.2	Messaging.....	137
7.2.3	Description/Publishing/Discovery.....	138
7.2.4	Quality of Service (QoS).....	141
7.2.5	Service Composition.....	144
7.3	Industry-Specific Service-Oriented Standards.....	145
7.3.1	Electronics Industry.....	145
7.3.2	Insurance Industry.....	146
7.3.3	Telecommunication Industry.....	147
7.4	Generic SOA Solution Standards are Evolving.....	148
7.5	Discussions on SOA and Web Services Standards.....	149
7.6	Summary.....	149
	References.....	150
<b>8</b>	<b>Solution-Level Quality of Service in SOA</b> .....	<b>152</b>
8.1	State-of-the-art of QoS on Web Services.....	152

8.2	SOA-QoS.....	153
8.2.1	Context-Aware QoS Model.....	153
8.2.2	Representation of QoS Model.....	154
8.2.3	QoS Data Management .....	157
8.2.4	Business Relationship Model.....	157
8.3	QoS Framework in an SOA Solution .....	158
8.3.1	QoS Framework Descriptions.....	158
8.3.2	Relationships Between Constructs in QoS Framework .....	161
8.4	Data Architecture Framework .....	162
8.4.1	Data Architecture Framework Descriptions .....	162
8.4.2	Relationships Between Constructs in Data Architecture.....	164
8.5	Modeling the Key Elements in QoS Management .....	165
8.5.1	Modeling of Resources .....	165
8.5.2	Modeling the QoS Assurance Process .....	167
8.6	Discussions on QoS in SOA.....	169
8.7	Summary .....	169
	References .....	169

## **Part 2 Realization of Services Computing**

<b>9</b>	<b>Requirements Driven Services Composition .....</b>	<b>173</b>
9.1	Introduction .....	173
9.2	Business Requirements Modeling.....	174
9.2.1	Target Components and Environment .....	175
9.2.2	Asset Lifecycle Management.....	177
9.2.3	Project Management .....	177
9.2.4	Finance Management .....	178
9.2.5	Representation of Business Requirements Modeling .....	178
9.3	Requirements Driven Services Discovery.....	180
9.4	Optimization for Business Services Composition.....	183
9.4.1	Formalization of Business Services Composition.....	183
9.4.2	Optimization Algorithms for Business Services Composition .....	188
9.5	Service Integration Framework .....	190
9.5.1	Services Integration Procedure .....	191
9.6	Discussions on Services Composition.....	192
9.7	Summary .....	193
	References .....	194
<b>10</b>	<b>Services Value Chain Collaboration .....</b>	<b>195</b>
10.1	Value Chain Collaboration .....	195
10.1.1	Example of Business Collaboration.....	195

10.1.2	Inter- and Intra-Enterprise Collaboration.....	197
10.1.3	Web Services Based Value Chain Collaboration.....	199
10.2	Extended Business Collaboration (eBC) Model.....	199
10.2.1	Introduction to Business Resources .....	199
10.2.2	Annotated Business HyperChain Technique.....	201
10.2.3	eBC to WS-Collab .....	201
10.3	Web Services Collaboration (WS-Collab) Resources.....	202
10.3.1	WS-Collab Resources.....	202
10.3.2	WS-Collab Resource Specifications .....	204
10.3.3	WS-Collab Ontology on Relationships Between Resources.....	205
10.4	Web Services Collaboration Message Primitives.....	211
10.4.1	WS-Collab Primitive.....	211
10.4.2	WS-Collab Message Structure .....	213
10.5	Web Services Collaboration Construct .....	215
10.6	Web Services Collaborative Exchange Protocol.....	217
10.7	WS-Collaboration Realization.....	219
10.7.1	Annotation Data Generation Process .....	219
10.7.2	HyperChain Manager.....	220
10.8	Relationships with Industry Standards.....	221
10.9	Discussions on Service-Based Business Collaboration.....	222
10.10	Summary .....	222
	References .....	222
<b>11</b>	<b>Business Process Management and Integration .....</b>	<b>224</b>
11.1	Business Process Modeling .....	224
11.2	SOA-Based Business Process Management .....	225
11.2.1	Top-down Business Process Management.....	226
11.2.2	Bottom-up Business Process Management .....	227
11.3	Bridging Gap Between Business Modeling and IT Implementation .....	228
11.3.1	Business Process Modeling from Business Analysts .....	228
11.3.2	Business Process Re-engineering in SOA .....	229
11.3.3	Generic Guidance for Re-engineering Business Process Modeling in SOA.....	230
11.3.4	Methodology for Re-engineering Business Process Models in SOA .....	230
11.4	Flexible Business Process Integration in SOA.....	234
11.4.1	Integration Ontology .....	234
11.4.2	Integration Manager.....	238
11.4.3	Lifecycle of an Integration Activity .....	239
11.4.4	Business Process Monitoring .....	240
11.5	Discussions on Business Process Management and Integration ....	241

11.6	Summary .....	242
	References .....	242
<b>12</b>	<b>Business Grid.....</b>	<b>243</b>
12.1	Grid Computing .....	243
12.2	Open Grid Services Architecture (OGSA).....	244
12.2.1	Distributed Resource Sharing Using OGSA .....	245
12.3	Business Grid.....	248
12.3.1	Enhancing OGSA with Advanced Web Services Technologies .....	248
12.3.2	Concept of Business Grid.....	249
12.3.3	Business Grid Solution Framework .....	250
12.4	Logical Grid Infrastructure .....	251
12.4.1	Packaged Application Grid.....	251
12.4.2	Business Grid Middleware.....	252
12.4.3	Business Process Grid.....	253
12.5	Business Grid Service Development and Invocation.....	254
12.6	Discussions on Business Grid.....	255
12.7	Summary .....	255
	References .....	255
<b>Part 3 Service Delivery and Services Engineering</b>		
<b>13</b>	<b>Enterprise Modeling .....</b>	<b>259</b>
13.1	Introduction .....	259
13.1.1	Dynamics of Services Ecosystem.....	259
13.1.2	Requirements from Decision Makers.....	260
13.2	Methodologies for Enterprise Modeling .....	261
13.2.1	Balanced Scorecard and Strategy Map .....	261
13.2.2	Component Business Modeling Circle.....	264
13.2.3	Enterprise Architecture.....	269
13.2.4	Relationships Between Enterprise Models and Business Process Transformation Model.....	272
13.3	Discussions on Enterprise Modeling .....	273
13.4	Summary .....	273
	References .....	274
<b>14</b>	<b>Project Based Enterprise Performance Management.....</b>	<b>275</b>
14.1	Changes of Enterprise Operational Views.....	275
14.2	Overview of Project Management .....	277
14.3	Enterprise Performance Management (EPM) .....	279
14.3.1	Concept of EPM .....	279

14.3.2	EPM Framework.....	279
14.3.3	From Project Management to Enterprise Portfolio Management .....	280
14.4	Service-Oriented Enterprise Project Management .....	281
14.4.1	EPM toward SOA .....	281
14.4.2	WS-EPM Framework.....	282
14.4.3	WS-EPM Operations .....	284
14.4.4	Formalization of WS-EPM Methodology.....	285
14.5	WS-EPM Common Services .....	286
14.5.1	WS-EPM Resource Management Facility .....	286
14.5.2	WS-EPM Utilities .....	290
14.6	WS-EPM Workspace.....	292
14.7	Discussions on SOA-Based EPM.....	294
14.8	Discussions on Enterprise Portfolio Management.....	294
14.9	Summary.....	295
	References.....	295
<b>15</b>	<b>Service-Oriented Business Consulting Methodology .....</b>	<b>296</b>
15.1	Vision of Services System .....	296
15.2	The Traditional Business Consulting Methods.....	298
15.2.1	Traditional Consulting Method for Strategic Change.....	298
15.2.2	Traditional Consulting Method for IT Strategic Plan .....	298
15.2.3	Shortcomings of Traditional Methods .....	299
15.3	Modeling of Services Ecosystem.....	299
15.4	Service-Oriented Business Consulting Method.....	301
15.4.1	Gap Analysis over SOA.....	301
15.4.2	Identification of Transformation Initiatives.....	303
15.4.3	Value Chain Analysis .....	304
15.4.4	Business Case Analysis.....	305
15.4.5	Portfolio Analysis and Transition Planning.....	306
15.4.6	Service-Oriented Project Management and Collaboration .....	307
15.4.7	IT Service Management.....	307
15.5	Discussion on SO-BCM .....	308
15.6	Summary.....	308
	Reference .....	308
<b>16</b>	<b>End-to-End Services Delivery Platform and Methodology .....</b>	<b>310</b>
16.1	Introduction to Services Delivery .....	310
16.2	Changes of Services Delivery Mechanisms .....	310
16.3	An SOA-Based Services Delivery Platform .....	312
16.3.1	Layered View of the Services Delivery Platform .....	312
16.3.2	Collaboration View of the Services Delivery Platform .....	314
16.3.3	Key Services Needed in the Services Delivery Platform.....	317

16.4	The End-to-End Services Delivery Methodology .....	322
16.4.1	Services Delivery Readiness Phase .....	323
16.4.2	Services Delivery Creation Phase .....	324
16.4.3	Services Delivery Operation Phase.....	326
16.5	Discussions on the Services Delivery Methodology and Platform .....	327
16.6	Summary.....	328
	References.....	328
<b>17</b>	<b>Software as Services and Services as Software .....</b>	<b>330</b>
17.1	Software as Services .....	330
17.1.1	Next Generation of Internet: Web 2.0.....	331
17.1.2	A Case Study of Web 2.0 Service Model—Service Mash-up .....	334
17.1.3	New Business Models Through Software as Services.....	335
17.1.4	Tips for Software as Services Model.....	336
17.2	Services as Software .....	336
17.3	Successful Business Cases.....	337
17.3.1	Healthcare Under Transformation .....	338
17.3.2	Innovative Store .....	340
17.3.3	Personalized Insurance Premiums .....	340
17.3.4	Business Performance Transformation Services .....	342
17.4	Summary.....	343
	References.....	343
<b>Index</b>	.....	<b>345</b>

# **Part 1 Foundations of Services Computing**

# 1 The Principle of Services and Services Computing

## 1.1 Introduction of Services

### 1.1.1 Definition of Services

The term “service” has existed for thousands of years along human history. When a person or a group performs some work to benefit another, it becomes a service. Many versions of definitions exist for the term “service”. For example, James Fitzimmons<sup>[1]</sup> defines a service as follows:

*“A service is a time-perishable, intangible experience performed for a customer acting in the role of co-producer.”*

Christian Gronroos defines a service from the perspective of management and marketing as follows<sup>[2]</sup>:

*“A service is an activity or series of activities of more or less intangible nature that normally, but not necessarily, take place in interactions between customer and service employees and/or physical resources or goods and/or systems of the service provider, which are provided as solutions to customer problems.”*

Although these definitions look different, they all indicate a fact that each service involves two inevitable sides: service provider and service consumer. A service provider offers the service, and a service consumer utilizes the service. The interaction between a service consumer and a service provider may happen in real-time or off-line. Focusing on IT-enabled business services, this book defines the term “services” as follows:

*Services represent a type of relationships-based interactions (activities) between at least one service provider and one service consumer to achieve a certain business goal or solution objective.*

A service provider commits to complete the tasks and provide values to a service consumer during the service’s lifecycle. Both sides share a common goal of keeping a healthy, long-term trust with efficient and valuable services.

An IT-enabled business service is typically characterized by two features: its service operation model and its service charge model. A service operation model defines how the service is to be delivered; a service charge model specifies how

## Services Computing

the delivered service is to be charged.

Services can be realized in different ways, represented by corresponding service operation models. Traditionally, services are typically provided in an end-to-end service operation model, meaning that service providers deliver services directly to their end users (i.e., service consumers). Leveraging the recent Information Technologies, services can now be delivered in several novel approaches, such as hosted service model, business process outsourcing, data-centered outsourcing, and services through online broker agency. These approaches intend to enhance customers' service experiences and enhance service providers' productivity.

Based on different business models, services can be charged in different ways. In general, there are three categories of service charge model: free-of-charge models, fee-based models, and government service models. Adopting a free-of-charge model, a service provider offers free services. Nowadays, free services are common, such as free email services, free Voice over IP (VoIP) services, and free instant message services. Adopting a fee-based model, a service consumer needs to pay a pre-announced fee to use the service. For example, an online payment service charges transaction-based service fee. Between free services and fee-based services, there are public services provided by governments. They are free-of-charge to use; however, they are actually "paid" (funded) by citizens' tax money.

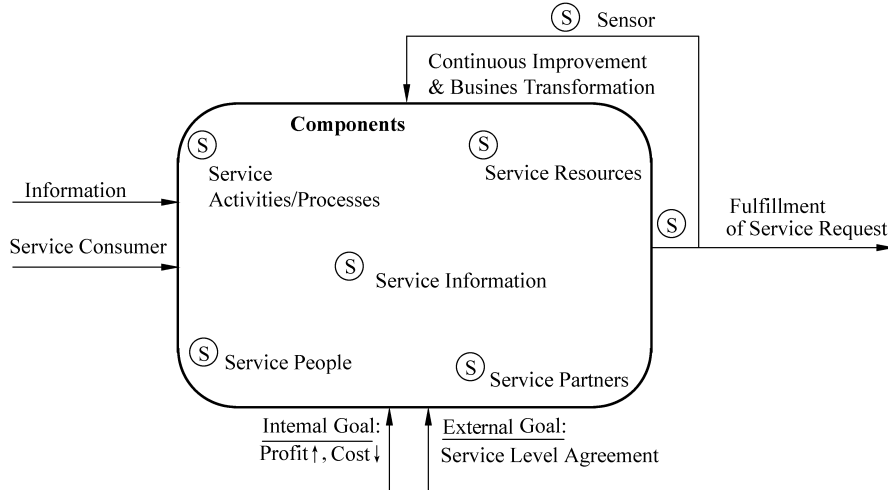
One service may be further divided into different service levels, each being associated with different service fees. A service consumer may choose to pay higher service fees to obtain higher-quality services with advanced capabilities, or pay lower fees to obtain services with less functionality. For example, a Web hosting service provider may provide three levels of services: gold Web plan, silver Web plan, and bronze Web plan. Each plan offers different storage space sizes and file transfer rates with different monthly charges.

### 1.1.2 Definition of Services System

Business services are realized by IT software systems, called services systems. This section defines a services system hosted by a service provider. As shown in Fig. 1.1, a services system can be viewed as a self-contained encapsulated system providing some services to the outside world. Such a services system shows as a feedback system, meaning that it possesses internal controls and reacts to surrounding environments. Therefore, a services system  $SS$  can be informally defined as a 6-tuple:

$SS = \langle Inputs, Outputs, Goals, Transformation, Components, Sensors \rangle$ , where: *Inputs* denote input information sent from service consumers, so that the services system can provide customized and personalized services. *Outputs* denote the output of the services system, i.e., the services to be offered. *Goals* denote the objectives of the system as a set of predefined system requirements. These goals

## 1 The Principle of Services and Services Computing



**Figure 1.1** Feedback Control-based Services System

can be further divided into internal goals and external goals. Internal goals come from service providers who intend to increase their profits or decrease cost, for example. External goals come from service consumers who request certain levels of service quality. *Transformation* denotes the control or tuning activities applied to the system and its linkage with its interconnected services systems. A services system typically requires support from other services system; therefore, it needs to collaborate with other systems to fulfill the business goal. *Components* denote major elements of the services system, which will be discussed in detail later. *Sensors* denote the system elements that monitor and detect changes from surrounding environments (contexts), so that the services system can react accordingly to provide better services.

Same as a traditional software system, each services system has its proprietary lifecycle, from the time it is created until the time it is destroyed or discarded. During its lifecycle, the services system typically includes the following five major components:

*Components* =  $\langle$ service people, service partners, service information, service activities, service infrastructure resources $\rangle$ , where:

*Service people* denote the people involved in the lifecycle of the services system. These people typically act in different roles, such as system designer, system developer, and system tester. One person may act in different roles at different times or simultaneously. *Service partners* denote the business partners involved in the lifecycle of the system. *Service information* denotes the information exchanged and required in the lifecycle of the system. *Service activities* denote the activities or business processes conducted in the lifecycle of the system. *Service infrastructure resources* denote all non-people resources required for the system. Three categories can be identified: physical resources such as building offices, IT resources comprising hardware resources (e.g., servers and networks)

## Services Computing

and software resources (e.g., application servers and operating systems), and abstract resources (e.g., time).

In order to support such a comprehensive services system, innovative technologies and methodologies from cross-disciplinary subjects are required, such as computer science, management science, Information Technology (IT), organizational science, economics, and operational research.

## 1.2 Perspectives of Services Systems

With the definition of the services system, this section will discuss the essential perspectives internal of the services system. Figure 1.2 illustrates a coherent logical view of a services system, which comprises four major perspectives: model, technology, architecture, and optimization.

### 1.2.1 Model

Enablement of a services system should be guided by modeling and model-driven methodologies that help developers manage complexity at various levels of abstraction. A model is a “philosophy” of how to design a services system and its development process. Based on different phases during the lifecycle of a services system, various models may need to be adopted to guide the transformations between models (e.g., operational-level models, platform-independent models, and platform-specific models), codes, and other artifacts (e.g., business objects). These models typically define proprietary patterns, which formalize repeatable scenarios and provide tested solutions, to improve developer’s productivity.

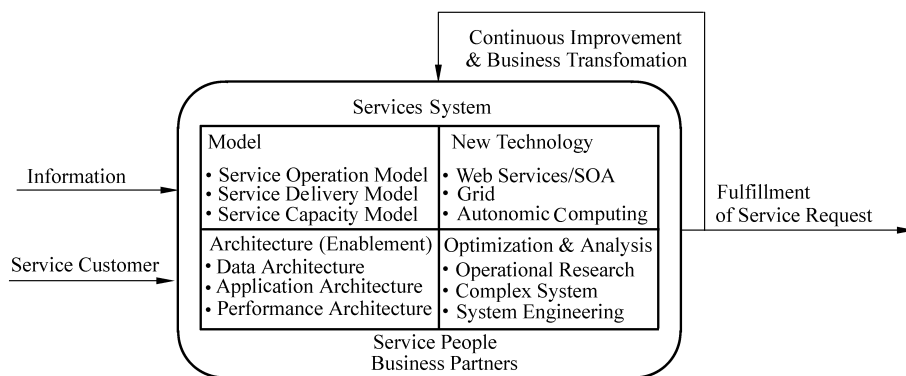


Figure 1.2 Enablement of services systems

### 1.2.2 Technology

The unique features of services and services systems demand corresponding IT support. The advancement of IT brings more opportunities for innovations. To date, several service-oriented technologies have emerged, such as Service-Oriented Architecture (SOA), Web services, Grid Computing, and Autonomic Computing. These technologies facilitate the construction of the services systems with higher reusability, flexibility, extensibility, and robustness.

### 1.2.3 Architecture

It has been well accepted that software architecture plays a key role in the design and development of a software system. A sound architectural model allows a software system to be adaptive to future changes. Similarly, software architecture is a critical aspect of a services system. It provides a guidance of designing and constructing a services system, by identifying system components along with the connections and interactions between them. It should be noted that the term “architecture” in Services Computing is not limited to IT architecture. Instead, since the tenet of Services Computing is to align IT with business in a service-oriented framework, the definition of the “architecture” becomes a manifold architectural model, including business architecture, IT architecture, data architecture, and performance architecture.

### 1.2.4 Optimization

In order to provide optimal services, a services system needs to be adaptive to ever-changing environments and business requirements. Optimization intends to select the best solution when facing multiple choices. This process may be needed during the entire lifecycle of a services system, and includes not only local optimization but also global optimization. Various related disciplines, such as operational research, complex system modeling, and system engineering, can be applied under different scenarios to facilitate the analysis and design of a services system.

## 1.3 Services Lifecycle

A typical service’s lifecycle often consists of six key phases as shown in Fig. 1.3: consulting and strategic planning, services engagement, services delivery, services operation, services billing, and services management. Using a bookstore service as an example, this section briefly summarizes the activities within each phase.

## Services Computing

In this example, a customer *C* invites a consulting service provider (*CSP*) and an IT service provider (*ITP*) to construct a set of bookstore branches locating at different places.

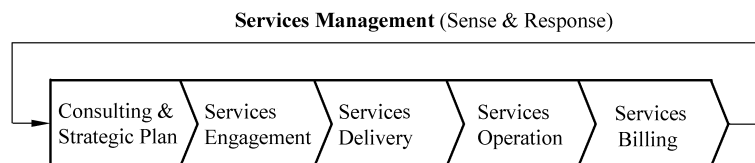


Figure 1.3 A services lifecycle

### 1.3.1 Phase 1: Consulting and Strategic Planning

The key activity in this phase is to invite third-party consulting companies to perform strategic planning. Using the bookstore example, in this phase, customer *C* invites *CSP* to analyze its potential position in the market together with an IT strategic plan for the next five years. At the end of the consulting phase, *CSP* helps *C* build a Request For Proposal (RFP).

### 1.3.2 Phase 2: Services Engagement

The key activities in this phase include: opportunity identification, RFP, negotiating service demands, forming the Statement of Work (SoW), and contracting. Using the bookstore example, in this phase, *C* distributes RFPs to multiple service providers to bid for the project; then *C* decides one service provider, say *ITP*.

### 1.3.3 Phase 3: Services Delivery

The key activities in this phase include: establishing service delivery teams and governing project management, solution creation, work breakdown, macro design, micro design, development and implementation, testing, and deployment. Using the bookstore example, in this phase, *ITP* establishes a project team and selects sub-service providers *ITP1* and *ITP2*. *ITP*'s project team comprises a project executive, an IT architect, and some senior IT experts with rich knowledge in SOA solution creation. Based on the results from the consulting company and RFP, they decide to adopt the SOA infrastructure. Part of the work is decided to be outsourced to *ITP1* and *ITP2*; the testing work is decided to be outsourced to *ITP1*.

### 1.3.4 Phase 4: Services Operation

The key activities in this phase include: call center (contact center) management, selecting tools for service operation, business and IT performance monitoring, change management and problem management, configuration management, and other IT service management functions. Using the bookstore example, in this phase, customer *C* decides to build a call center to improve customer experience. *C* may also decide to run the call center by himself/herself, and outsource other IT service management work to *ITP*.

### 1.3.5 Phase 5: Services Billing

In this phase, the service provider gathers payment based on the contracts with the customer. Using the bookstore example, in the service delivery case, *ITP* successfully completes the project for *C* by collaborating with *ITP1* and *ITP2*. The contract will be closed, and *C* will pay *ITP* according to the contract. Besides the transaction-based payment for the completed project, *C* may sign a new outsourcing contract with *ITP* and decide to pay *ITP* the IT maintenance fee annually.

### 1.3.6 Phase 6: Services Management

Using the bookstore example, in this phase, since *ITP* is in charge of the solution creation and IT service management tasks, *ITP1* could predict the traffic knowing that customer *C* wants to add a new service, and recommends that *C* increase IT resources for the new service.

## 1.4 Key Factors in a Services Lifecycle

Seven key factors need to be considered in a services lifecycle: data and information, processes, people, resources, finance factors, knowledge and skills, and innovation and technology.

### 1.4.1 Data/Information

The output of a business company is typically one of two things: either physical products/goods or business services, which can be fulfilled by manufacturing systems and services systems, respectively. The difference between a services

## **Services Computing**

system and a manufacturing system is that, a services system is a “soft” system in that it may not produce “hard” products/goods as a manufacturing system typically does; instead, it generates valuable information or manipulates information and service resources to benefit service consumers. When a customer requests a service, a business object (e.g., an order) is usually submitted; when a service provider wants to charge the customer, a bill is presented containing detailed billing information.

### **1.4.2 Processes**

A services system is typically comprised of business processes to fulfill business objectives. A process usually contains a series of organized activities with regulated inputs and outputs, while chaining different roles together. The efficiency of a business process highly depends on core competency of related service providers. A process can also be further decomposed into sub-processes.

### **1.4.3 People**

Business services typically require people involvement to leverage services, software, or other assets to deliver services to customers. Therefore, people are considered as one individual key factor in a services system. Two types of actors are identified: service customers and service consultants.

Service customers refer to the end users of a services system, whose demands need to be fully understood by corresponding service providers. They are logically considered as an integral part of a services system in that they may influence the status and operations of the system through interactions.

Service consultants refer to human beings from a service provider side who perform certain tasks. Compared with the role of human beings in a manufacturing system, service consultants are oftentimes called “human capital” because their knowledge and skills are critical to system efficiency in a services system.

### **1.4.4 Resources**

Service execution typically needs to consume various types of resources: physical resources, IT resources, information resources, and abstract resources. Examples of physical resources are estates and offices; examples of IT resources are servers and storages; examples of information resources are databases; examples of abstract resources are time and knowledge.

### 1.4.5 Finance Factors

For both service consumers and service providers, cost and value are two dual factors which should always be taken into consideration. The objective is to best leverage resources available to service providers and generate the most return.

### 1.4.6 Knowledge and Skills

The degree of the ownership of knowledge and mastering of unique skills (e.g., IT operation) often differentiate a service provider from its competitors. In a service business unit, the employees should continuously be trained to master up-to-date knowledge.

### 1.4.7 Innovation and Technology

In the present fierce service competition, innovation and technology often decide the time to market. Those having the advanced technologies may have higher opportunity to win the service market by introducing new service products / packages or improving the operations of their services systems.

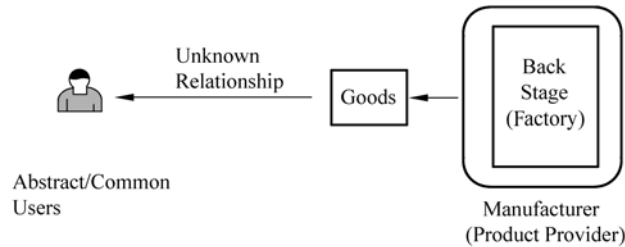
## 1.5 Comparisons Between Services and Manufacturing Models

The major differences between a services model and a manufacturing model root in their different outputs: services vs. goods, as described by Sasser<sup>[3]</sup>:

*“A precise definition of goods and services should distinguish them on the basis of their attributes. A good is a tangible physical object or product that can be created and transferred; it has an existence over time and thus can be created and used later. A service is tangible and perishable. It is an occurrence or process that is created and used simultaneously or nearly simultaneously. While the consumer cannot retain the actual service after it is produced, the effort of service can be retained.”*

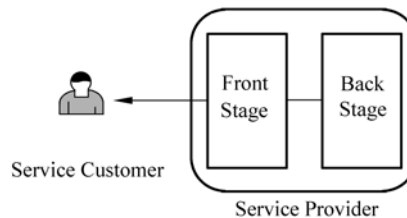
A typical manufacturing model is shown in Fig. 1.4. In this model, the goal of manufacturing activities is to produce goods that fulfill common purposes. The products (goods) should satisfy some predefined specifications; however, goods themselves may not directly solve an end user’s problem. As shown in Fig. 1.4, goods act as connections between the product provider and end users.

## Services Computing



**Figure 1.4** Manufacturing model

On the contrary, in the services field, a service provider and a service consumer have closer relationships in a services environment as shown in Fig. 1.5. A service provider's environment can often be divided into two parts: service front stage and service back stage. Interactions between the service provider and the service consumer often happen at the service front stage. The service provider intends to provide good service experiences to consumers to improve customer satisfaction and attract more customers, while keeping current customers' loyalty. Service front stage uses service desks to collect service requirements or other contents captured in pre-designed service forms to serve service consumers. Some key performance indicators include service execution time, efficiency, accuracy, and customer satisfaction.



**Figure 1.5** Model of services

At the back stage, the service provider uses factory models to seamlessly integrate sub-services to pursue high productivity. Typical activities happened at the service back stage include: making ready the service operation environment, allocating resource to fulfill service consumer requirements and avoid fraud, billing customers, running business intelligent analysis to further improve the service, and so on.

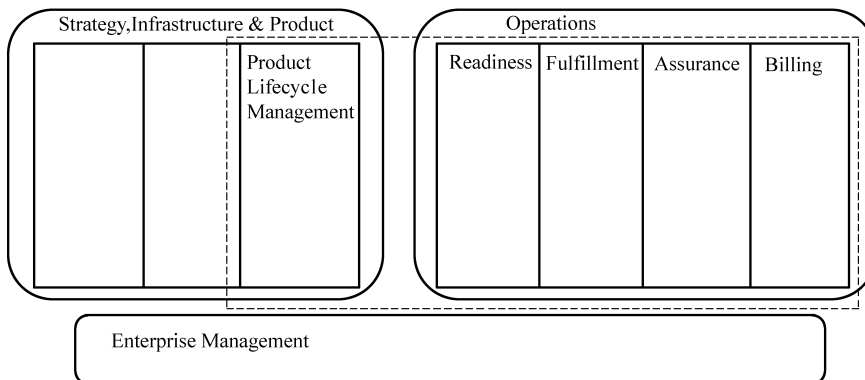
Meanwhile, from product lifecycle point of view, a service consumer may involve in all phases (pre-production, in-production, after-production), while a consumer of manufactured goods is mainly involved in the after-production phase only. In other words, a consumer of goods has little interaction with a goods provider in the goods' overall lifecycle, while a service consumer is usually fully involved in the service fulfillment process. This difference leads to highly different

approaches when one tries to improve a manufacturing or services system.

Beyond those components mentioned above, new technologies could play important roles in enhancing a services system. Among other technologies, Service-Oriented Architecture (SOA), Business Process Integration and Management (BPIM) play critical roles for business transformation in recent years.

## 1.6 Business Services Lifecycle in Telecommunication Industry

Each company typically has many business function units: some provide direct services to customers; others provide back-office support. In an enhanced Telecommunication Operations Map (eTOM)<sup>[4]</sup> shown in Fig. 1.6 (with some changes based on services industry), key service components are highlighted in the dashed rectangle.



**Figure 1.6** Typical service operation model in an enterprise

Figure 1.6 illustrates a typical view of the businesses units in a service-oriented company. As shown in Fig. 1.6, the components in an enterprise can be divided into three parts based on their objectives.

The first part includes strategies, infrastructures, and products. This part mainly solves the planning or strategy-related issues.

The second part comprises service operations related to daily activities. These operations can be divided into two categories: function-facing and people-facing. Function-facing operations include: service readiness, service fulfillment, service assurance, and service billing. People-facing operations can be further divided into four types: customer-facing (e.g., customer relationship management), internal worker-facing (e.g., service management and operation), physical and people

## Services Computing

resource-facing (e.g., resource management and operation), and partner and supplier-facing (e.g., supplier/partner relationship management).

The third part comprises enterprise management components that span the first part and the second part. This part covers most of the business administration works, such as strategic and enterprise planning, enterprise risk management, enterprise effectiveness management, knowledge and research management, financial and asset management, stakeholder and external relations management, and human resource management.

## 1.7 Relationship Between IT and Non-IT Services in a Services Ecosystem

The trend of the evolution of business ecosystems is that, businesses increasingly depend on IT technologies as a source of innovation and differentiation, while IT is moved to a higher level to align with business needs. In the past, IT people mainly focused on technology itself; at present, people have recognized that IT should be aligned with business in the whole service lifecycle. As a matter of fact, both the delivery of IT services and the enablement of IT to business changes require sound understanding of the principles of the uniqueness of IT services. This section will compare the lifecycle of an IT service with that of a known healthcare service. The goal is to identify their commonalities and illustrate why Services Computing is needed to build modern services industries.

### 1.7.1 Overview of a Services Ecosystem

A variety of service industries (a.k.a., vertical service) exist, such as aerospace and defense, automotive, banking, chemicals and petroleum, consumer products, education, electronics, energy and utilities, financial markets, government, healthcare, insurance, life sciences, media and entertainment, retail, wholesale distribution, telecommunications, and travel and transportation. All services can be divided into two basic categories: pure IT services (e.g., software as services) and IT-enabled services (e.g., Customer Relationship Management (CRM) services and some vertical services such as banking services, telecommunication services, automotive services, and chemicals services). However, no matter which category one service falls in, it shares some commonalities that are summarized in Fig. 1.7.

Figure 1.7 shows a high-level view of a simplified services ecosystem. A service consumer may enjoy services from different service industries simultaneously. Each of these services is typically constructed on top of some reusable cross-industry common services (a.k.a. horizontal services), which are in turn divided into two categories: common business services and common IT services.