The Building Blocks

<table>
<thead>
<tr>
<th>Visual Representation</th>
<th>Textual Form</th>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Object</strong></td>
<td>Nouns; capitalized first letter in every word; if ending with “ing”, “Object” is placed as a suffix</td>
<td>An object is a thing that has the potential of stable, unconditional physical or mental existence.</td>
<td>Static things. Can be changed only by processes.</td>
</tr>
<tr>
<td><strong>Process(ing)</strong></td>
<td>Nouns in gerund form; capitalized first letter in every word; if not ending with “ing”, “Process” is placed as a suffix</td>
<td>A process is a pattern of transformation that an object undergoes.</td>
<td>Dynamic things. Are recognizable by the changes they cause to objects.</td>
</tr>
<tr>
<td><strong>Object state</strong></td>
<td>Nouns, adjectives or adverbs; non-capitalized</td>
<td>A state is a situation an object can be at.</td>
<td>States describe objects. They are attributes of objects. Processes can change an object’s state.</td>
</tr>
<tr>
<td><strong>Non-Reserved Words</strong></td>
<td>Arial bold by default</td>
<td>Names of entities.</td>
<td>Words used by the system architect, unique to the system.</td>
</tr>
<tr>
<td><strong>OPL Conventions</strong></td>
<td>Arial by default; non-capitalized</td>
<td>Object-Process Language (OPL) words.</td>
<td>Words or phrases used by OPL, the same in every sentence of a certain type.</td>
</tr>
</tbody>
</table>

Links: The Mortar

*Tagged Structural Links*
Generally used between objects, but may also be used between processes. Cannot be used to link an object to a process.

<table>
<thead>
<tr>
<th>Link Name</th>
<th>Object Process Diagram (OPD) Symbol</th>
<th>OPL Sentence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tagged</td>
<td><img src="image" alt="Tagged Diagram" /></td>
<td>R Object refers to S Object</td>
<td>Relation from source object to destination object; relation name is entered by architect, and is recorded along link.</td>
</tr>
<tr>
<td>(Null)</td>
<td><img src="image" alt="Null Diagram" /></td>
<td>R Object relates to S Object</td>
<td>Relation from source object to destination object with no tag.</td>
</tr>
<tr>
<td>Bi-directional</td>
<td><img src="image" alt="Bi-directional Diagram" /></td>
<td>R Object precedes S Object. S Object follows R Object</td>
<td>Relation between two objects; relation names are entered by architect, and are recorded along link.</td>
</tr>
<tr>
<td>Tagged</td>
<td><img src="image" alt="Tagged Bi-directional Diagram" /></td>
<td>R Object and S Object are related</td>
<td>Relation between two objects with no tag.</td>
</tr>
</tbody>
</table>
Links: The Mortar (continued)

**The Four Fundamental Structural Relations**

<table>
<thead>
<tr>
<th>Shorthand Name</th>
<th>Aggregation</th>
<th>Exhibition</th>
<th>Generalization</th>
<th>Instantiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol</td>
<td>△</td>
<td>△</td>
<td>△</td>
<td>▼</td>
</tr>
<tr>
<td>Meaning</td>
<td>Relates a whole to its parts</td>
<td>Relates an exhibitor to its attributes</td>
<td>Relates a general thing to its specializations</td>
<td>Relates a class of things to its instances</td>
</tr>
</tbody>
</table>

A fundamental structural relation can have many descendants. The different OPL sentences and OPD pictures are listed below.

<table>
<thead>
<tr>
<th>Structural Relation Name and Shorthand Name</th>
<th>One (OPD)</th>
<th>Two (OPD)</th>
<th>Three or more (OPD)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Aggregation-Participation</strong></td>
<td>A B</td>
<td>A B C</td>
<td>A B C D</td>
<td>B, C and D are parts of the whole A.</td>
</tr>
<tr>
<td><strong>Exhibition-Characterization</strong></td>
<td>A B</td>
<td>A B C</td>
<td>A B C D</td>
<td>B, C and D are attributes of A. If B is a process, it is an operation of A.</td>
</tr>
<tr>
<td><strong>Generalization-Specialization</strong></td>
<td>B A</td>
<td>B C A</td>
<td>B C A</td>
<td>B, C and D are types of A.</td>
</tr>
<tr>
<td><strong>Classification-Instantiation</strong></td>
<td>A B</td>
<td>B C A</td>
<td>B C A</td>
<td>B, C and D are unique objects of the class A.</td>
</tr>
</tbody>
</table>

The four fundamental relations are also applicable to processes. Only exhibition can link objects with processes. Instantiation cannot generate a hierarchy while the other three can. Any number of things can be linked to the root.
Object-Process Methodology
Object-Process Methodology

A Holistic Systems Paradigm

Foreword by Edward F. Crawley
With 246 Figures and CD-ROM
To my family
The contemporary technological world is made up of complex electro-mechanical-information products that intimately involve human operators. Such a description includes products that range from a simple kitchen appliance to an air-traffic control system. Increasingly, the value these products deliver is based on their system nature; they have many interacting components that combine to produce emergent features or services that are desirable to the customer.

The development of more rigorous approaches to the engineering of these systems is seen by industry as a key area for process improvement, and should be recognized by academia as an important area of scholarly development. Since the engineering process dominates development time and cost, and the system engineering process defines value delivered by a product, improvements in system engineering and product modeling will have direct influence on product utility and enterprise competitiveness. The task of engineering a new system has become more complicated by the vastly increasing number of components involved, the number of disparate disciplines needed to undertake the task, and the growing size of the organizations involved. Despite the common experience that members of many organizations share, they often lack a common product development vocabulary or modeling framework. Such a framework should be rigorously based on system science; be able to represent all the important interactions in a system; and be broadly applicable to electrical, informational, mechanical, optical, thermal, and human components.

Object Process Methodology provides such a framework. OPM includes a clear and concise set of symbols that form a language enabling the expression of the system’s building blocks and how they relate to each other. It is a symbolic representation of the objects in a system and the processes they enable. Considering the historical development of engineering disciplines, it is an appropriate time for such a rigorous framework to emerge. Disciplines often move through a progressive maturation. Early in the history of an intellectual discipline, we find observation of nature or practice, which quickly evolves through a period in which things are classified. A breakthrough often occurs when classified observations are abstracted and quantified. These phases characterize much of the work done to date in system engineering and product/system development.

We are now entering a phase, in which symbolic representation and manipulation can be developed. Such a symbolic system can lay the foundation for the ultimate step in the evolution of a discipline, the ability to predict the outcome beforehand. Mature disciplines, such as mechanics, are well into the era of symbolic manipulation and prediction. Maturing disciplines, such as human genomics, are in the phase of symbolic representation. OPM is a parallel development in symbolic representation of systems.

OPM represents the two things that are inherent in a system: its objects and its processes. This duality is recognized throughout the community that studies sys-
tems, and sometimes goes by labels such as form/function, structure/function, and functional requirements/design parameters. Objects are what a system or product is. Processes are what a system does. Yet, it is remarkable that so few modeling frameworks explicitly recognize this duality. As a result, designers and engineers try to jump from the goals of a system (the requirements or the "program") immediately to the objects. Serious theory in such disparate disciplines as software design, mechanical design and civil architectural design recognizes the value of thinking about processes in parallel with objects. Not only does OPM represent both objects and processes, but it also explicitly shows the connections between them.

Object Process Methodology has another fundamental advantage – it represents the system simultaneously in a graphic representation and in a natural language. The two are completely interchangeable, and represent the same information. The advantage in this approach lies in appreciating the human limitation to the understanding of complexity. As systems become more complex, the primary barrier to success is the ability of the human designers and analysts to understand the complexity of the interrelationships. By representing the system in both textual and graphical form, the power of "both sides of the brain" – the visual interpreter and the language interpreter – is engaged. These are two of the strongest processing capabilities hard-wired into the human brain.

OPM allows a clear representation of the many important features of a system: its topological connections, its decomposition into elements and subelements, the interfaces among elements, and the emergence of function from elements. The builder or viewer of the model can view abstractions or zoom into some detail. One can see how specification migrates to implementation. These various views are invaluable when pondering the complexity of a real modern product system.

I have used OPM in my System Architecture course at MIT. It has proved an invaluable tool to professional learners in developing models of complex technical systems, such as automobiles, spacecraft and software systems. It allows an explicit representation of the form/function duality, and provides an environment in which various architectural options can be examined. Incorporating OPM into my subject has added the degree of rigor of analysis necessary to move the study of technical system architecture towards that of an engineering discipline.

One can anticipate that there will be many academic applications of OPM. I would consider using it in intermediate or advanced subjects in system engineering, product development, engineering design and software engineering. It is ideal for courses that demonstrate how various disciplines come together to form a multi-disciplinary product.

Likewise, OPM can form the backbone of a corporate or enterprise modeling system for technical products. Such a representation would be especially valuable in conceptual and preliminary design, when much of the value, cost and risk of a product are established and only a few other modeling frameworks are available for decision support.

Massachusetts Institute of Technology
Cambridge, Massachusetts
May 2001

Edward F. Crawley
Before Fortran, putting a human problem ... on a computer was an arduous and arcane task. ... Like high priests in a primitive society, only a small group of people knew how to speak to the machine. Yet there were some heretics in the priesthood, and Mr. Backus was one of them. “I figured there had to be a better way,” he recalled nearly five decades later, “you simply had to make it easier for people to program.”

S. Lohr (2001)

This book is about how to make it easier for people to understand and develop systems. Systems are all around us. Understanding and developing complex systems involve a host of disciplines that need to be brought together in a unifying framework. Systems evolve over time, and they keep growing and getting more complex. Hundreds, perhaps thousands, of individuals are required to maintain and operate systems such as electrical power grids or globally interconnected telecommunication networks. Nature’s global climate, the solar system and biological systems are at least as intricate. Systems science and engineering are emerging as new interdisciplinary fields of study that identify and utilize important commonalties among systems of all types.

A fundamental requirement of any science and engineering domain is that its intellectual underpinnings be formulated and well grounded. Such formulation, in turn, mandates that a set of elementary building blocks is agreed upon and relations among these building blocks are studied and understood. A unifying approach is indispensable for developing, communicating, supporting, and evolving systems of various domains, types, magnitudes and complexities. To this end, a clear and concise language must be developed. Pioneers in established science and engineering disciplines, like physics, chemistry, mechanical engineering, and electrical engineering, have long agreed on such languages. More recently, biology has joined this list and is now establishing the molecular foundations of life. Being a young field, systems science and engineering is just beginning to contemplate this problem and sort it out. This book will hopefully advance the state of affairs regarding our ability to model and understand systems.

The book is intended for people interested in modeling systems, and reading it does not require special background in either mathematics or programming. Specifically, system integrators, analysts, designers, modelers, executives, and project leaders in a variety of industry and service domains, private as well as public, will benefit from reading the book and applying Object-Process Methodology (OPM) for the purpose of developing better systems faster and more reliably. An equally
important point is that developing systems with OPM is fun. The combination of graphics and natural language that automatically complement each other when an automated tool, such as Systemantica® (Sight Code, 2001) is employed, is an enjoyable way of analysis and design that increases one's confidence in the quality of the resulting system specification. Information technology professionals, including computer scientists, software engineers, information system managers, and database administrators will gain insight into possible extensions of Object technology. Scientists and engineers, along with science and engineering educators and students of all ages will hopefully find OPM useful to model, communicate, and explain systems they research and design.

The book, which consists of 15 chapters arranged in three parts, is designed as a textbook for graduate or advanced undergraduate courses. Typical course titles include engineering systems, specification and analysis of information systems, systems architecture, information systems engineering, systems design and management, or Object-Process Methodology. Indeed, various versions of this book have been used and tested in graduate and undergraduate courses at the Technion and MIT during the past five years. Each chapter concludes with a summary of the chapter's highlights and problems that enable hands-on experience, elaborate on concepts discussed in the chapter and provide food for further thought.

Part I, *Foundations of Object-Process Methodology*, is a gradual exposure to OPM. Chapter 1 is a gentle introduction to basic principles of OPM through a simple example of a wedding. Walking through a comprehensive case study of an automated teller machine (ATM), Chapter 2 exposes the reader to Object-Process Diagrams as the graphic representation of the single object-process model. Continuing with the ATM case study, Chapter 3 introduces Object-Process Language as the complementary modality to the graphic one and shows how the two synergistically reinforce each other. Chapter 4 discusses in depth the two basic building blocks of OPM, objects and processes. In Part II, *Concepts of OPM Systems Modeling*, the dynamic and static system aspects are the focus of Chapters 5 and 6, respectively. The next two chapters elaborate on the four OPM fundamental structural relations. Aggregation and Characterization are discussed in Chapter 7; Generalization and Classification in Chapter 8. Chapter 9, which concludes Part II, is devoted to complexity management. Part III, *Building Systems with OPM*, shows how OPM is used to develop systems, products and projects. Systems and modeling is the topic of Chapter 10. Chapter 11 provides a comprehensive OPM model of system evolution and lifecycle, and discusses how this model can be used to develop systems in an OPM-based environment. The next two chapters expand upon issues presented in Parts I and II. Chapter 12 elaborates on states, while Chapter 13 presents advanced OPM concepts. Chapter 14 discusses systems theory and Chapter 15 concludes with a survey of object-oriented (OO) methods and their relation to OPM.

I thank Professor Ed Crawley for his encouragement and support throughout my stay at MIT and, in particular, for his enthusiastic Foreword to this book. OPM has become and integral part of the course "Systems Architecture," which he and Pro-
Professor Olivier de Weck teach at MIT. Thanks to Robert M. Haralick whose draft, written at my request, was the basis for Section 4.1. Thanks for the professional editing as well as the detailed and helpful comments to Dr. Hans Wössner of Springer-Verlag. Thanks to Idan Ginsburg and William Litant for their meticulous proofreading. Thanks to my wife Judy and my daughters, Limor, Tlalit, Shiri and Inbal, who have been really helpful and considerate. Special thanks to Shiri Dori, whose thorough proofreading and wise comments, at both the editorial and content level, as well as the problems she composed, were extremely helpful.

Haifa, Israel, and Cambridge, Massachusetts

January 2002

Dov Dori
Contents Overview

Part I  Foundations of Object-Process Methodology ...... 1
Chapter 1  A Taste of OPM ................................................................. 3
Chapter 2  Object-Process Diagrams ............................................. 13
Chapter 3  Object-Process Language .............................................. 33
Chapter 4  Objects and Processes .................................................... 55

Part II  Concepts of OPM Systems Modeling ............... 81
Chapter 5  Dynamics ................................................................. 83
Chapter 6  Structure .............................................................. 105
Chapter 7  Aggregation and Exhibition ................................. 133
Chapter 8  Generalization and Instantiation ....................... 171
Chapter 9  Managing Systems’ Complexity ............................ 207

Part III  Building Systems with OPM ....................... 247
Chapter 10  Systems and Modeling ........................................... 249
Chapter 11  System Lifecycle and Evolution ....................... 289
Chapter 12  States and Values .................................................... 311
Chapter 13  Advanced OPM Concepts ..................................... 339
Chapter 14  Systems Theory ...................................................... 379
Chapter 15  Object-Oriented Modeling ................................. 401
# Table of Contents

**Part I  Foundations of Object-Process Methodology ........................ 1**

**Chapter 1  A Taste of OPM .......................................................... 3**

1.1 The Wedding Example: A Sneak Preview of OPM ..................... 5
1.2 OPM Building Blocks: Objects, Processes, and States .............. 5
1.3 Specialization and Inheritance .................................................. 7
1.4 Aggregation and the Result Link ............................................... 8
Summary .............................................................................................. 10
Problems ............................................................................................. 11

**Chapter 2  Object-Process Diagrams ............................................. 13**

2.1 Objects and Aggregation ............................................................. 13
2.2 Structural Relations and Structural Links ................................. 15
2.3 Processes and Procedural Links .................................................. 16
2.4 System Diagram: The Top-Level OPD ...................................... 18
2.5 Zooming into the Transaction Executing Process .................... 20
2.6 The OPD Set .............................................................................. 21
2.7 How to Read an OPD ................................................................. 22
   2.7.1 Flow of Control ................................................................. 22
   2.7.2 The Timeline in OPDs ....................................................... 23
   2.7.3 Object States and Conditions ............................................. 23
2.8 Completing the In-Zoomed Transaction Executing OPD ........... 24
   2.8.1 Logical XOR, AND, and OR Operators ............................. 25
   2.8.2 The System Map ............................................................. 26
   2.8.3 The Ultimate OPD ........................................................... 27
   2.8.4 Zooming Out of Transaction Executing ......................... 28
Summary .............................................................................................. 29
Problems ............................................................................................. 30
Chapter 3  Object-Process Language ........................................... 33
  3.1 Motivation for a Language .................................................. 33
    3.1.1 Real-Time Textual Feedback ........................................ 34
    3.1.2 Closing the Requirements-Implementation Gap ................. 35
  3.2 Structural Links and Structure Sentences .............................. 35
    3.2.1 The First OPL Sentence ............................................. 36
    3.2.2 The First OPL Aggregation Sentence ............................. 37
  3.3 The OPL Paragraph and the Graphics-Text Principle ................ 38
    3.3.1 Extending the OPL Paragraph ....................................... 40
    3.3.2 Enabling Sentences .................................................. 40
    3.3.3 Transformation Sentences .......................................... 41
    3.3.4 The SD Paragraph .................................................... 43
  3.4 More OPL Sentence Types ................................................ 43
    3.4.1 State Enumeration and Condition Sentences ..................... 44
    3.4.2 AND, XOR, and OR Logical Operators ............................ 46
    3.4.3 The SD1 Paragraph ................................................... 47
    3.4.4 In-Zooming and Out-Zooming Sentences .......................... 48
  3.5 Boolean Objects and Determination Sentence ........................ 49
    3.5.1 Boolean Condition Sentences ....................................... 50
    3.5.2 Compound Condition Sentences .................................... 50
    3.5.3 State-Specified Generation Sentence ............................. 50
    3.5.4 Converting a Dual-State Object into a Boolean Object ......... 50
  3.6 OPD-OPL Item Pairs and Synergy ....................................... 52
  Summary .................................................................................. 52
  Problems ................................................................................ 53

Chapter 4  Objects and Processes ........................................... 55
  4.1 Existence, Things, and Transformations ............................... 55
    4.1.1 Objects ....................................................................... 56
    4.1.2 Transformation and Processes ...................................... 57
  4.2 Processes and Time ............................................................ 59
    4.2.1 Cause and Effect ....................................................... 59
    4.2.2 Syntactic vs. Semantic Sentence Analysis ...................... 60
    4.2.3 The Process Test ....................................................... 61
  4.3 Things .............................................................................. 62
    4.3.1 Things and Entities .................................................... 63
    4.3.2 The Perseverance of Things ....................................... 64
    4.3.3 The Essence of Things ............................................... 66
    4.3.4 Symbolizing Physical Things ....................................... 68
    4.3.5 The Origin of Things .................................................. 70
    4.3.6 The Complexity of Things ........................................... 70
# Table of Contents

4.3.7 Thing Types ......................................................... 70
4.3.8 The Relativity of Object and Process Importance .......... 71
4.3.9 Object and Process Naming ........................................ 71

4.4 Informatical Objects ..................................................... 72
4.4.1 Telling Informatical and Physical Objects Apart .......... 72
4.4.2 Systems and Information Systems .............................. 73
4.4.3 Translation of Informatical Objects ............................ 74
4.4.4 Toward “Pure” Informatical Objects ......................... 74

4.5 Object Identity ............................................................ 75
4.5.1 Change of State or Change of Identity? ....................... 76
4.5.2 Classes and Instances of Objects and Processes .......... 77

Summary ............................................................................. 78
Problems ............................................................................. 78

**Part II** Concepts of OPM Systems Modeling ....................... 81

**Chapter 5** Dynamics ........................................................ 83

5.1 States ............................................................................ 83
5.1.1 Object States and Status ............................................ 83
5.1.2 Change and Effect ...................................................... 84
5.1.3 Explicit and Implicit Status Representations .............. 85
5.1.4 The Input, Output, and Effect Links ......................... 87
5.1.5 State Suppression and the Effect Link ..................... 88
5.1.6 State Expression ....................................................... 89

5.2 Existence and Transformation ......................................... 90
5.2.1 Result and Consumption Links .................................. 92
5.2.2 Procedural Links, Enablers, and Transformees ............ 93
5.2.3 Enablers .................................................................... 93
5.2.4 Agents ..................................................................... 94
5.2.5 Instruments ............................................................ 94
5.2.6 Enabling Links ......................................................... 95
5.2.7 Transformees .......................................................... 96
5.2.8 Odd Man Out: The Invocation Link ......................... 97

5.3 Object Roles with Respect to a Process ......................... 98
5.3.1 Enablers and Affectees .............................................. 99
5.3.2 The Involved, Preprocess, and Postprocess Object Sets .. 99
5.3.3 Condition and Agent Condition Links ..................... 100
5.3.4 Operator, Operand, and Transform? ....................... 101

Summary ............................................................................. 102
Problems ............................................................................. 102
Chapter 6  Structure ................................................................. 105
  6.1 Structural Relations ........................................................... 105
    6.1.1 Structural Links ........................................................... 106
    6.1.2 Structural Relation Directions ......................................... 108
    6.1.3 Unidirectional Structural Link ......................................... 109
    6.1.4 OPD Sentences ............................................................. 109
    6.1.5 The Reciprocity of a Structural Relation ............................ 110
    6.1.6 Null Tags and Their Default OPL Reserved Phrases ................ 111
    6.1.7 Structural Relations as Static Verbs .................................. 112
  6.2 Participation Constraints and Cardinality .............................. 113
    6.2.1 Participation Constraints ............................................... 113
    6.2.2 Parameterized Participation Constraints .............................. 114
    6.2.3 Range Participation Constraints ...................................... 115
    6.2.4 Shorthand Notations and Reserved Phrases ........................... 116
    6.2.5 Cardinality ................................................................. 117
    6.2.6 Participation Constraints in Procedural Relations ................ 119
  6.3 The Distributive Law and Forks .......................................... 120
    6.3.1 Forks ............................................................................ 121
    6.3.2 Fork Degree ..................................................................... 123
    6.3.3 Fork Comprehensiveness .................................................. 124
  6.4 The Transitivity of Structural Relations .................................. 125
  6.5 The Four Fundamental Structural Relations .............................. 126
  Summary .................................................................................... 129
  Problems .................................................................................... 130

Chapter 7  Aggregation and Exhibition ......................................... 133
  7.1 Aggregation-Participation: Underlying Concepts ...................... 133
    7.1.1 Aggregation-Participation as a Tagged Structural Relation ....... 135
    7.1.2 The Aggregation-Participation Symbol .................................. 136
    7.1.3 Sets and Order ................................................................. 137
    7.1.4 Aggregate Naming ........................................................... 138
    7.1.5 Aggregating Processes ...................................................... 139
  7.2 Aggregation Hierarchy and Comprehensiveness ....................... 141
    7.2.1 Aggregation Hierarchy ..................................................... 141
    7.2.2 Aggregation Comprehensiveness ......................................... 142
    7.2.3 Parameterized Participation Constraints .............................. 142
    7.2.4 Participation Level and Aggregational Complexity ................ 143
  7.3 Exhibition-Characterization: Underlying Concepts .................. 144
    7.3.1 The Name Exhibition-Characterization .................................. 145
    7.3.2 The Exhibition-Characterization Symbol ................................ 145
    7.3.3 Attribute and Operation Are Features .................................. 146
    7.3.4 Exhibition Complexity ....................................................... 147
### Table of Contents

#### 7.4 Features in OO vs. OPM

7.5 The Four Thing-Feature Combinations

7.5.1 The Object-Attribute Combination

7.5.2 The Object-Operation Combination

7.5.3 The Process-Attribute Combination

7.5.4 Process-Operation Combination

7.6 The Feature Hierarchy

7.7 Feature-Related Natural Language Issues

7.7.1 Attribute Naming Dilemmas

7.7.2 Reserved Objects and the Measurement Unit Reserved Object

7.7.3 Continuous Values and Multi-Valued Attributes

7.7.4 Mathematical Inequalities in OPM

7.8 Reflective Metamodeling of an Attribute

7.8.1 The Size of an Attribute

7.8.2 The Mode of an Attribute

7.8.3 The Touch of an Attribute

7.8.4 The Source of a Feature

7.8.5 The Operation a Feature Carries

Summary

Problems

---

**Chapter 8 Generalization and Instantiation**

8.1 Generalization-Specialization: Introduction

8.1.1 Specialization Symbol and Sentence

8.1.2 Process Specialization

8.2 Inheritance

8.2.1 Feature Inheritance

8.2.2 Structural Relations Inheritance

8.2.3 Procedural Link Inheritance

8.2.4 State Inheritance

8.2.5 State Specialization

8.2.6 Process Specialization

8.2.7 Generalization Complexity

8.3 Qualification

8.3.1 Qualification Inheritance

8.3.2 Multiple Qualification Inheritance

8.4 Classification-Instantiation

8.4.1 Classes and Instances

8.4.2 The Relation Between Instantiation and Specialization

8.4.3 The Relativity of Instance

8.4.4 Instance Qualification

8.4.5 Process Instances

8.4.6 Classification Complexity
8.5 Modifiers and Instances ............................................................. 198
  8.5.1 Natural Language Modifiers and Shortcuts .......................... 198
  8.5.2 Adjectives and Attributes ................................................... 199
  8.5.3 Adverbs and Operations ..................................................... 201
8.6 Specializations of the Involved Object Set Members .................. 202
8.7 Non-Comprehensiveness ......................................................... 203
  8.7.1 Non-Comprehensiveness of Fundamental Structural Relations .... 203
  8.7.2 Non-Comprehensiveness of States and Values ..................... 204
Summary ......................................................................................... 205
Problems ......................................................................................... 205

Chapter 9 Managing Systems’ Complexity ................................. 207
9.1 The Need for Complexity Management ..................................... 207
  9.1.1 Middle-Out as the De-Facto Architecting Practice ................. 208
  9.1.2 Determining the Extent of Refinement ................................. 210
  9.1.3 Towards Quantifying Complexity ........................................ 211
9.2 Divide and Conquer: By Aspects or by Details? ........................ 212
  9.2.1 Why is Detail Decomposition Good? .................................... 215
  9.2.2 When Should a New OPD Be Created? ................................. 217
9.3 The Attributes of Scaling .......................................................... 217
  9.3.1 The Purpose of Scaling ....................................................... 217
  9.3.2 The Mode of Scaling .......................................................... 218
  9.3.3 Controlling Visibility by In- and Out-Zooming ...................... 221
  9.3.4 The Distributivity of Procedural Links .................................. 223
  9.3.5 Unfolding and Folding ....................................................... 224
  9.3.6 State Expressing and Suppressing ...................................... 227
  9.3.7 Primary and Secondary Operands ...................................... 230
9.4 Abstracting ............................................................................. 230
  9.4.1 Consolidating ................................................................. 230
  9.4.2 Zoom consolidating ......................................................... 231
  9.4.3 Paths and Path Labels ....................................................... 231
  9.4.4 Zoom Consolidating Pitfalls ............................................... 233
  9.4.5 Zoom Consolidating Conditions ...................................... 234
  9.4.6 Fold Consolidating ........................................................... 235
9.5 What Happens to Procedural Links During Abstracting? .......... 235
  9.5.1 Procedural Link Precedence .............................................. 236
  9.5.2 Semi-Folding and Semi-Unfolding .................................... 238
  9.5.3 Selective Semi-Folding and Semi-Unfolding ...................... 240
9.6 Looking at the Big Picture: The System Map .......................... 241
  and the OPM Construct Pairs ..................................................... 241
Summary ......................................................................................... 244
Problems ......................................................................................... 245
# Table of Contents

**Part III** Building Systems with OPM ................................. 247

**Chapter 10** Systems and Modeling ................................. 249

10.1 Defining Systems ...................................................... 249  
   10.1.1 Some Existing Definitions ....................................... 249  
   10.1.2 Function ........................................................... 251  
   10.1.3 The Various Functions of Stone ................................. 251  

10.2 System Defined ....................................................... 252  
   10.2.1 System as a Relative Term ...................................... 254  
   10.2.2 System as a Subjective Term ................................... 254  
   10.2.3 The Function of Natural and Artificial Systems ............... 255

10.3 Goal, Concept, and Function ....................................... 255  
   10.3.1 The Intent and Goal of Artificial Systems ....................... 256  
   10.3.2 Telling System Function and Dynamics Apart ................... 256  
   10.3.3 Function, Structure, and Behavior ................................ 260

10.4 System Architecture .................................................. 260  
   10.4.1 Function vs. Dynamics ........................................... 260  
   10.4.2 The Concept Behind a System .................................... 261  
   10.4.3 The Origin and Essence of Systems ................................ 262

10.5 Objects, Systems, and Products .................................... 263  
   10.5.1 Product Defined ................................................ 263  
   10.5.2 The Object-System-Product Hierarchy ............................ 264  
   10.5.3 Goods, Services, and Projects .................................. 266

10.6 Documenting Functions of the System Architecture ............... 267  
   10.6.1 The Function Hierarchy ......................................... 267  
   10.6.2 Function Boxes and Function Sentences ......................... 268  
   10.6.3 Functionality ................................................... 271

10.7 From Systems to Models ............................................. 271  
   10.7.1 Some Model Definitions ......................................... 272  
   10.7.2 Model Defined ................................................ 272

10.8 Modeling Paradigms .................................................. 272  
   10.8.1 Natural Language as a Modeling Tool .......................... 273  
   10.8.2 Mathematical and Symbolic Modeling ......................... 276  
   10.8.3 Graphic Modeling and Knowledge Representation ............... 281

10.9 Reflective Metamodeling ............................................. 283

Summary ............................................................................. 285

Problems ............................................................................ 286

**Chapter 11** System Lifecycle and Evolution ........................ 289

11.1 System Lifecycle .......................................................... 289  
   11.1.1 Lifecycle of Artificial Systems .................................... 290  
   11.1.2 Software and Product Development Processes .................. 290
11.2 Systems Analysis and the Scientific Method ........................................ 291
11.3 Categorization vs. Interdisciplinarity .............................................. 293
11.4 System Engineering and the Role of the System Architect ............... 293
11.5 An OPM Model of System Lifecycle Phases ..................................... 294
  11.5.1 Top-Level Description of System Evolution .............................. 295
  11.5.2 Initiating the System .......................................................... 297
  11.5.3 Developing the System ....................................................... 297
  11.5.4 Analyzing .............................................................................. 298
  11.5.5 The Refining-Abstracting Cycles ............................................ 299
  11.5.6 Designing .............................................................................. 300
  11.5.7 The Waterfall Model vs. Iterative and Incremental Development 301
  11.5.8 Deploying the System .......................................................... 302
11.6 Zooming into Analyzing .................................................................. 304
11.7 Zooming into Designing and Implementing ..................................... 306
11.8 From Design to Implementation ..................................................... 308
Summary ............................................................................................... 308
Problems ............................................................................................... 309

Chapter 12 States and Values ............................................................. 311
12.1 State-specified Objects and Links .................................................... 311
  12.1.1 Initial, Ultimate and Default States ........................................ 313
  12.1.2 The Transformation Attribute of a Process ............................ 314
  12.1.3 Object as a Role Player for State ........................................... 315
  12.1.4 State Maintaining Processes .................................................. 316
  12.1.5 Sentences and Phrases of States and Values ............................ 317
  12.1.6 Single Value Sentence .......................................................... 322
12.2 Telling States Apart from Values ................................................... 322
12.3 Metamodeling the Attributes of Value and Their States ................. 324
  12.3.1 Numeric and Symbolic Values ................................................ 326
  12.3.2 Mapping Object States onto Attribute Values ........................ 326
12.4 Compound States and State Space .................................................. 328
  12.4.1 The Attribute Feasibility Matrix ............................................. 332
  12.4.2 Logical Compound States ..................................................... 332
Summary ............................................................................................... 335
Problems ............................................................................................... 337

Chapter 13 Advanced OPM Concepts ................................................. 339
13.1 Real-Time Issues ........................................................................... 339
  13.1.1 Sequential vs. Parallel Process Execution ............................... 339
  13.1.2 Process Synchronization ....................................................... 340
  13.1.3 Events .................................................................................. 341
Chapter 13  Systems Theory .................................................... 379

13.1 Chronon and Event .............................................................. 342
13.1.4 Chronon and Event .............................................................. 342
13.1.5 Basic Triggering Event Types ................................................ 342
13.2 Process and State Duration ................................................... 344
13.3 Processing states .................................................................. 346
13.4 Probability in Procedural Relations ........................................ 348
13.5 Scope and Name Disambiguation .......................................... 349
13.5.1 The Fundamental DAG ...................................................... 350
13.5.2 Scope of an Object ........................................................... 351
13.6 The Reserved Words “of” and “which” ............................... 352
13.6.1 The Reserved Word “of” and the Dot Operator .................... 356
13.6.2 Using “of” with Tagged Structural Relations ..................... 357
13.6.3 The Reserved Word “which” .............................................. 357
13.6.4 Operation: A Process Without Side Effect ....................... 360
13.7 Structure-Related Issues ...................................................... 360
13.7.1 Transitivity Strength ........................................................ 361
13.7.2 Hamiltonian Distance ....................................................... 362
13.7.3 The Fractal Relation ........................................................ 362
13.7.4 Covariance and Contravariance ........................................ 364
13.8 OPM Metamodelling Issues ................................................ 365
13.8.1 A Metamodel of Thing ....................................................... 366
13.8.2 The Specialization-Specification Hierarchy ....................... 368
13.8.3 A Refined Generic Processing Model ................................. 369
13.8.4 Time Exception Handling ................................................ 371
13.9 The OPM Construct Hierarchy ............................................ 372

Summary ................................................................................. 374

Problems .................................................................................. 376

Chapter 14  Systems Theory .................................................... 379

14.1 The Informatics Hierarchy .................................................... 379
14.1.1 Computers Are Climbing the Informatics Hierarchy ............ 380
14.1.2 Knowledge and Understanding ......................................... 381
14.2 Ontology ............................................................................. 382
14.3 General Systems Theory ...................................................... 383
14.3.1 A Brief History of General Systems Theory ....................... 383
14.3.2 The Hierarchy of System Levels ........................................ 385
14.4 Autopoietic vs. Allopoietic Systems ...................................... 387
14.5 Systems and Humans .......................................................... 387
14.6 Systems Theory Characteristics .......................................... 388
14.6.1 Previously Defined Characteristics ................................. 389
14.6.2 System, Environment and Beneficiaries ............................ 389
14.6.3 Control and Feedback ..................................................... 390
14.7 Classical Physics vs. Quantum Theory ........................................... 393
  14.7.1 Visualization ................................................................. 394
  14.7.2 Causality ........................................................................... 395
  14.7.3 Locality .............................................................................. 395
  14.7.4 Self-Identity ................................................................. 395
  14.7.5 Objectivity ........................................................................ 396

14.8 Objectifying: Converting a Process into an Object ......................... 397
Summary ......................................................................................... 399
Problems ......................................................................................... 400

Chapter 15  Object-Oriented Modeling .............................................. 401

15.1 The Evolution of System Analysis Methods ....................................... 401
  15.1.1 Data Flow Diagrams ......................................................... 402
  15.1.2 Entity-Relationship Diagrams and Their Combination with DFD 402
  15.1.3 The Object-Oriented Paradigm ........................................... 403

15.2 Pre-UML Object-Oriented Methods .............................................. 403
  15.2.1 Object Modeling Technique ................................................ 404
  15.2.2 Object-Oriented Software Engineering ................................... 404
  15.2.3 Object-Oriented Analysis and Object-Oriented Design ............. 405
  15.2.4 Object-Oriented Systems Analysis ....................................... 405
  15.2.5 Object-Oriented Analysis & Design ..................................... 406
  15.2.6 Object Life-Cycles ............................................................. 406
  15.2.7 The Booch Method ............................................................ 407
  15.2.8 MOSES ............................................................................. 407
  15.2.9 The Fusion Method ............................................................ 407
  15.2.10 OPEN Modeling Language ............................................... 408

15.3 Unified Modeling Language – UML ............................................. 408

15.4 Metamodelling in OO Methods .................................................... 410

15.5 OO Methods – A Summary ......................................................... 410

15.6 Software Development Approaches and Trends .............................. 412
  15.6.1 Aspect-Oriented Programming ............................................. 412
  15.6.2 The Rational Unified Process .............................................. 412
  15.6.3 Extreme Programming ....................................................... 413
  15.6.4 Agile Modeling ................................................................. 413

15.7 Challenges for OO Methods ......................................................... 413
  15.7.1 A Historic Perspective ......................................................... 414
  15.7.2 The Encapsulation Challenge .............................................. 415
  15.7.3 The Model Multiplicity Challenge ....................................... 415
  15.7.4 Empirical Evidence of the Model Multiplicity Problem .......... 417
  15.7.5 The Complexity Management Challenge ............................ 417
## Table of Contents

15.8 OPM and OO ................................................................. 418  
15.8.1 The UML 2.0 Initiative ................................................. 419  
15.8.2 Systemantica: an OPM Supporting Tool .......................... 420  
15.8.3 OPM Applications and Research: Present and Future .......... 421  

Summary .................................................................. 423  

Problems .................................................................. 424  

Appendix A: The ATM System ..................................... 425  

References ............................................................... 435  

Index ........................................................................ 443