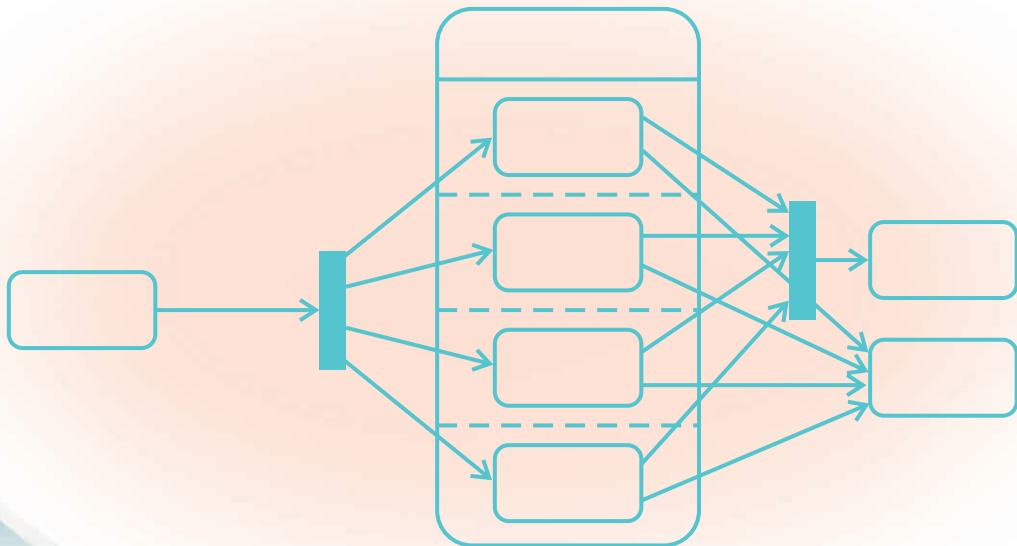


# Object-Oriented Analysis and Design Using UML

An Introduction to Unified Process  
and Design Patterns



**Mahesh P. Matha**

# Object-Oriented Analysis and Design Using UML



# Object-Oriented Analysis and Design Using UML

An Introduction to Unified Process and Design Patterns

**MAHESH P. MATHA**

*Department of Computer Science (PG Section)  
Smt. Parvatibai Chowgule College of Arts and Science  
Margao, Goa*

**PHI Learning** Private Limited

New Delhi-110001  
2010

**OBJECT-ORIENTED ANALYSIS AND DESIGN USING UML—An Introduction to Unified Process  
and Design Patterns**

Mahesh P. Matha

© 2008 by PHI Learning Private Limited, New Delhi. All rights reserved. No part of this book may be reproduced in any form, by mimeograph or any other means, without permission in writing from the publisher.

**ISBN-978-81-203-3322-2**

The export rights of this book are vested solely with the publisher.

**Second Printing**

...

...

**December, 2010**

Published by Asoke K. Ghosh, PHI Learning Private Limited, M-97, Connaught Circus,  
New Delhi-110001 and Printed by Glorious Printer, Delhi-110092.

To  
my loving Mother  
**Mrs. Kumud P. Matha**  
and my Father  
(Late) **Mr. Pandurang R. Matha**



---

---

# Contents

<i>Preface</i>	<i>xiii</i>
<i>Acknowledgements</i>	<i>xv</i>
<b>1 Introduction</b>	<b>1–10</b>
1.1 Role of Analysis and Design in Software Development	1
1.2 What is Meant by Object Orientation?	2
1.2.1 Encapsulation	2
1.2.2 Abstraction, Implementation Hiding	4
1.2.3 Inheritance, Dynamic Binding, Polymorphism	4
1.2.4 Overriding and Overloading	5
1.2.5 What is Object-Oriented Analysis and Design (OOAD)?	6
1.3 Overview of Various OOAD Methodologies	6
1.3.1 Booch Methodology	6
1.3.2 Coad and Yourdon Methodology	6
1.3.3 Fusion	7
1.3.4 Jacobson: Objectory and OOSE	7
1.3.5 LBMS SEOO	7
1.3.6 Rumbaugh OMT	7
1.3.7 Shlaer and Mellor OO Analysis	8
1.4 Goals of UML	8
<i>Summary</i>	9
<i>Review Questions</i>	9
<b>2 Use Case Modelling</b>	<b>11–40</b>
2.1 Actors and Use Cases	12
2.2 Use Case Relationships	14
2.2.1 The Include Relationship	14
2.2.2 The Extend Relationship	15
2.2.3 Use Case Generalization	17
2.3 Writing Use Cases Formally	17
2.4 How to Choose the System Boundary	19
2.5 Finding Actors	19
2.6 Finding Use Cases	19
2.7 Use of Use Cases for Validation and Verification	19



2.8 Use Case Realization	20
CASE STUDY: A Career Guidance Portal	21
<i>Summary</i>	38
<i>Review Questions</i>	39

### 3 Static Modelling Using Class Diagrams

41–73

3.1 Classes and Objects	42
3.2 Attributes and Operations	43
3.3 Visibility of Attributes and Operations	44
3.3.1 Private Visibility	44
3.3.2 Public Visibility	46
3.3.3 Protected Visibility	46
3.4 Class Scope Attribute	46
3.5 Mapping Class to Java Code	47
3.6 Attributes with Default Values	48
3.7 Associations	48
3.7.1 The Java Code for Unidirectional Association	49
3.7.2 The Java Code for Bidirectional Association	51
3.8 Role Names	52
3.9 Qualified Association	52
3.10 Association Class	54
3.11 Ternary Association	54
3.12 Recursive Association	55
3.12.1 The Java Code for Recursive Association	56
3.13 Multiple Associations between Two Classes	56
3.13.1 The Java Code for Multiple Associations between Two Classes	56
3.14 Aggregation	56
3.14.1 Composite Aggregation	57
3.14.2 Shared Aggregation	57
3.15 Generalization	58
3.15.1 The Java Code for Generalization	58
3.16 Abstract Class	59
3.17 Subclass Partitioning	60
3.18 Generalization Set	61
3.19 Interfaces	62
3.19.1 The Java Code for Interface Realization	63
3.20 Packages and Grouping of Classes into Packages	63
3.20.1 Graphical Display of Owned Elements	64
3.20.2 Import, Access and Export	65
3.21 Parameterized Classes	68
3.22 Identification of Classes for Career Guidance Portal	69
<i>Summary</i>	72
<i>Review Questions</i>	72

## 4 Interaction Diagrams

74–95

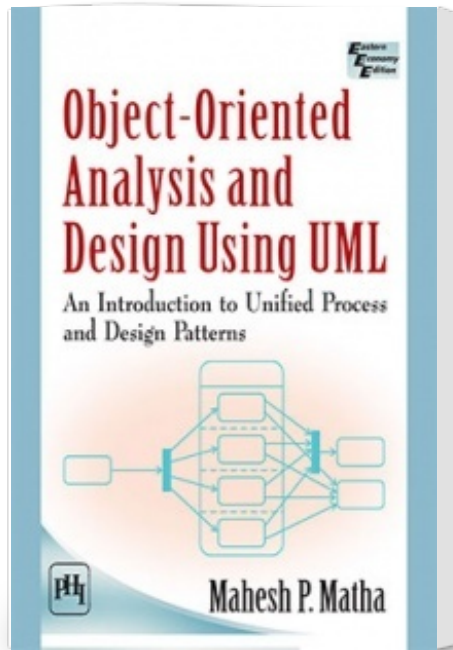
- 4.1 Introduction to Interaction Diagrams 74
  - 4.1.1 Sequence Diagrams 75
- 4.2 Creating New Objects 78
- 4.3 Combining Fragments (Condition and Loop) 78
  - 4.3.1 The Option Combination Fragment 80
  - 4.3.2 The Break Combination Fragment 80
  - 4.3.3 The Parallel Combination Fragment 82
  - 4.3.4 The Ref Combination Fragment 83
  - 4.3.5 Gates 84
  - 4.3.6 Recursion 86
  - 4.3.7 Iteration over a Collection 86
  - 4.3.8 Invoking Static Messages 88
- 4.4 Communication Diagrams 88
  - 4.4.1 Basic Notation for Communication Diagrams 88
  - 4.4.2 Labelling Messages in Communication Diagrams 89
  - 4.4.3 Messages to Self 90
  - 4.4.4 Creating Instances 90
  - 4.4.5 Mutually Exclusive Conditional Messages 91
  - 4.4.6 Iteration for a Single Message 92
  - 4.4.7 Iteration over a Collection 92
  - 4.4.8 Invoking Static Messages 93
- Summary* 93
- Review Questions* 94

## 5 Dynamic Modelling Using State and Activity Diagrams

96–132

- 5.1 What are State Machines? 97
- 5.2 Events 98
  - 5.2.1 Time Event 98
  - 5.2.2 A Guard Condition Becoming True 98
  - 5.2.3 Change Event 98
  - 5.2.4 Call Event 98
  - 5.2.5 Signal Event 99
- 5.3 States and Transitions 99
- 5.4 Formal Syntax for Specifying a State Transition 100
- 5.5 Substates 101
- 5.6 Modelling Complex Transitions 102
- 5.7 The History Indicator (Shallow and Deep History) 103
- 5.8 The Junction State 104
- 5.9 The Synch State 105
- 5.10 The Submachine Reference and the Stub States 106
  - 5.10.1 Stub State 107

# Object-oriented Analysis And Design Using UML An Introduction To Unified Process And Design Patterns



Publisher : **PHI Learning**

ISBN : 9788120333222

Author : **Mahesh P Matha**

Type the URL : <http://www.kopykitab.com/product/6183>



**Get this eBook**