

# USING UML

SOFTWARE ENGINEERING  
WITH OBJECTS AND COMPONENTS

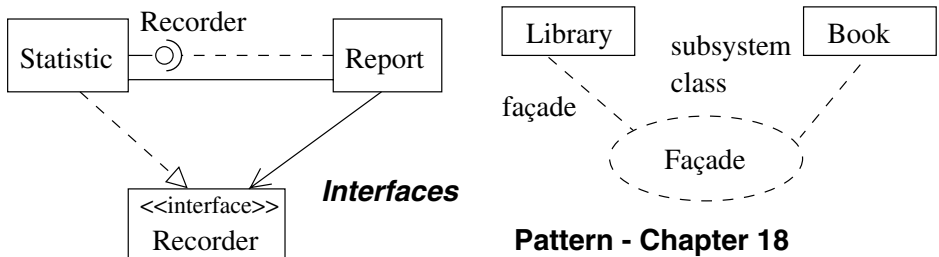
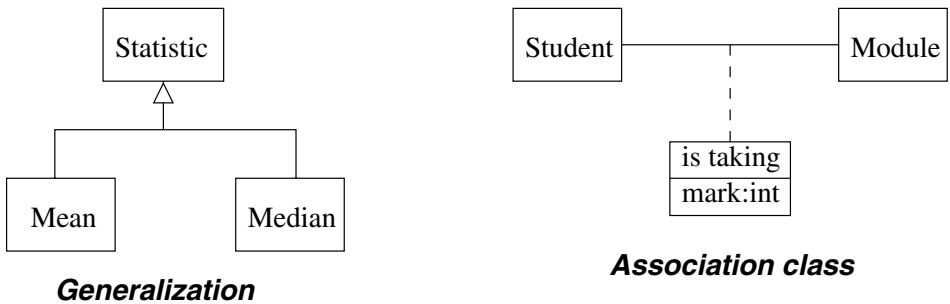
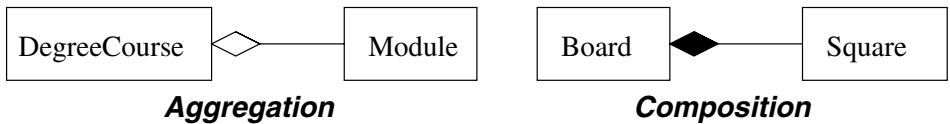
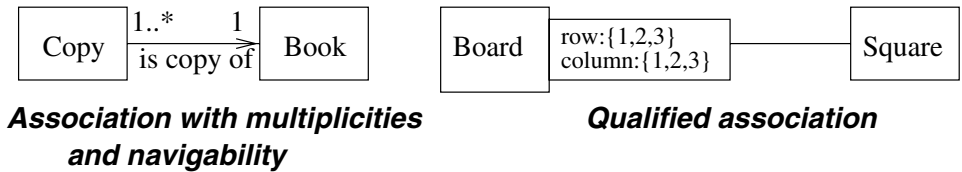
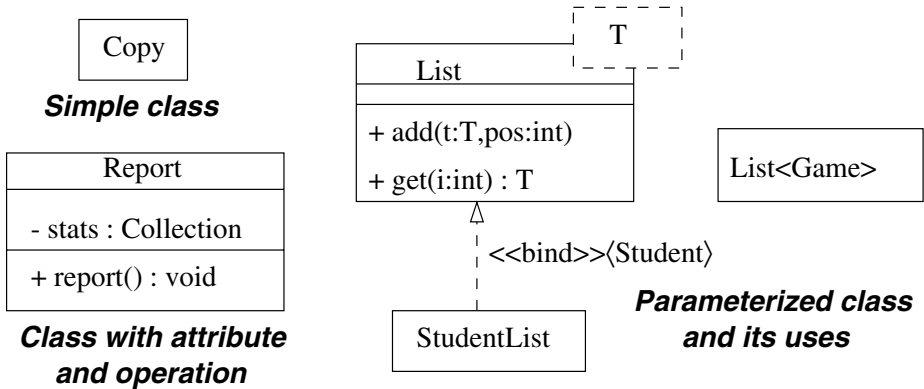
PERDITA STEVENS

WITH ROB POOLEY

SECOND EDITION



## Classes : Chapters 5 and 6



# Using UML

## Table of Contents

Cover

Using UML 2nd edition

Contents

Preface

First edition acknowledgments

List of Figures

part I Conceptualbackground

Software engineering with components

What is a good system?

Do we have good systems?

What are good systems like?

How are good systems built?

Object concepts

What is an object?

How does this relate to the aims of the previous chapter?

Inheritance

Polymorphism and dynamic binding

Introductory case study

The problem

Scope and iterations

Identifying classes

Relations between classes

The system in action

Design by Contract 1

# Table of Contents

Persistence

## The development process

Defining terms

The development process

System, design, model, diagram

## part II The Unified Modeling Language

### Essentials of class models

Identifying objects and classes

Associations

Attributes and operations

Generalization

Design by Contract 2 Substitutivity

The class model during the development

CRC cards

### More on class models

More about associations

OCL, the Object Constraint Language

More about classes

Stereotypes

Properties and Tagged Values

Parameterized classes

Dependency

Components and packages

Visibility, protection

### Essentials of use case models

Actors in detail

Use cases in detail

System boundary

Using use cases

# **Table of Contents**

Possible problems with use cases

Use Case Driven Development?

## **More on usecase models**

Relationships between use cases

Generalizations

Actors and classes

## **Essentials of interaction diagrams**

Collaborations

Communication diagrams

Sequence diagrams

Where Should Messages Go? Law of Demeter

More advanced features

Interaction diagrams for other purposes

## **More on interaction diagrams**

Beyond simple sequences of messages

Concurrency

## **Essentials of state and activity diagrams**

State diagrams

Designing Classes with State Diagrams

Activity diagrams

## **More on state diagrams**

Other kinds of events

Other kinds of actions

Looking inside states

Concurrency within states

## **Architectural and implementation diagrams**

Component structure diagrams

Deployment model

The Deployment Model in the Project

# Table of Contents

## Packages and models

- Packages

- Models

## part III Case studies

### CS4 administration

- The case study

- Discussion

### Board games

- Scope and preliminary analysis

- Interaction

- Back to the framework

- States

### Discrete event simulation

- Requirements

- Outline class model

- Use cases

- Standard mechanism for process-based simulation

- Associations and navigability

- Classes in detail

- Class Report

- Class Statistic

- Building a complete simulation model

- The dining philosophers

## part IV Towards practice

### Reuse: components, patterns

- Practicalities of reuse

- Design patterns

- Frameworks

### Product quality: verification, validation, testing

# **Table of Contents**

Quality review

How can high quality be achieved?

Verification

Validation

Testing

Reviews and inspections

Process quality:management,teams, QA

Management

Teams

Leadership

Quality assurance

Quality Assurance: The Case Against

Further reading

**Bibliography**

**Index**