

**Diagrams**

**Links, Associations, and Inheritance**

**UML 2.0**

**Model Systems**

**COMPLETE STARTER KIT**  
CD-ROM Includes  
Possidon for UML  
Community Edition 2.2

**SAMS**  
**Teach Yourself**

**UML**

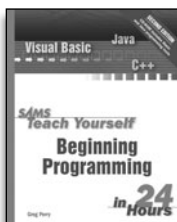
Joseph Schmuller

**in 24**  
**Hours**

## What you should already have to get the most out of this book...

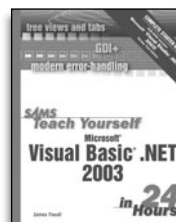
- An understanding of basic programming concepts
- A healthy curiosity about how to model programs and systems

Some books  
that may  
help...



### Sams Teach Yourself Beginning Programming in 24 Hours

*Learn the basics of  
programming*



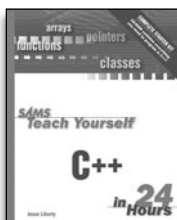
### Sams Teach Yourself Visual Basic .NET 2003 in 24 Hours

*Learn more about  
programming using  
Microsoft Visual Basic*

## What this book will help you learn...

- How to use the Unified Modeling Language to model systems
- Fundamental object-oriented programming concepts such as inheritance and encapsulation
- Fundamental use-case concepts such as inclusion and extension
- New concepts in the latest version of UML (UML 2.0)

Related  
titles...



### Sams Teach Yourself C++ in 24 Hours

*Learn more about OOP  
using C++*



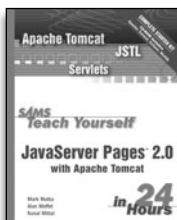
### Sams Teach Yourself Java 2 in 24 Hours

*Learn more about  
programming with the  
Java language*

## Where you may want to go from here...

- Use UML to design more complicated systems
- Develop Web applications with Java or .NET

Possible  
titles to  
look for...



### Sams Teach Yourself JavaServer Pages 2.0 with Apache Tomcat in 24 Hours

*Learn to build dynamic  
Web sites with JSP 2.0*



### Sams Teach Yourself ASP.NET in 24 Hours

*Learn more about creating  
.NET Web applications*

# Sams Teach Yourself UML in 24 Hours, Complete Starter Kit

## Table of Contents

### Table of Contents

#### Introduction

What's New in This Edition

Who Should Read This Book?

Organization of This Book

Conventions Used Throughout This Book

#### Part I: Getting Started

##### HOUR 1: Introducing the UML

Adding a Method to the Madness

How the UML Came to Be

Components of the UML

Some Other Features

New Diagrams in UML 2.0

Why So Many Diagrams?

But Isn't It Just a Bunch of Pictures?

Summary

Q&A

Workshop

##### HOUR 2: Understanding Object-Orientation

Objects, Objects Everywhere

Some Object-Oriented Concepts

The Payoff

# Table of Contents

Summary

Q&A

Workshop

## HOUR 3: Working with Object-Orientation

Visualizing a Class

Attributes

Operations

Attributes, Operations, and Visualization

Responsibilities and Constraints

Attached Notes

Classes What They Do and How to Find Them

Summary

Q&A

Workshop

## HOUR 4: Working with Relationships

Associations

Multiplicity

Qualified Associations

Reflexive Associations

Inheritance and Generalization

Dependencies

Class Diagrams and Object Diagrams

Summary

Q&A

Workshop

## HOUR 5: Understanding Aggregations, Composites, Interfaces, and Realizations

Aggregations

Composites

# Table of Contents

Composite Structure Diagram

Interfaces and Realizations

Interfaces and Ports

Summary

Q&A

Workshop

## HOUR 6: Introducing Use Cases

Use Cases: What They Are

Use Cases: Why They're Important

An Example: The Soda Machine

Including a Use Case

Extending a Use Case

Starting a Use Case Analysis

Summary

Q&A

Workshop

## HOUR 7: Working with Use Case Diagrams

Representing a Use Case Model

Visualizing Relationships Among Use Cases

Use Case Diagrams in the Analysis Process

Applying Use Case Models: An Example

Taking Stock of Where We Are

The Big Picture

Summary

Q&A

Workshop

## HOUR 8: Working with State Diagrams

What Is a State Diagram?

Substates

# Table of Contents

History States

New in UML 2.0

Why Are State Diagrams Important?

Building the Big Picture

Summary

Q&A

Workshop

## HOUR 9: Working with Sequence Diagrams

What Is a Sequence Diagram?

Cars and Car Keys

The Soda Machine

Sequence Diagrams: The Generic Sequence Diagram

Creating an Object in the Sequence

Framing a Sequence: Sequence Diagramming in UML 2.0

Building the Big Picture

Summary

Q&A

Workshop

## HOUR 10: Working with Communication Diagrams

What Is a Communication Diagram?

Cars and Car Keys

The Soda Machine

Creating an Object

One More Point About Numbering

A Few More Concepts

Building the Big Picture

Summary

Q&A

Workshop

# Table of Contents

## HOUR 11: Working with Activity Diagrams

The Basics: What Is an Activity Diagram?

Applying Activity Diagrams

Swimlanes

Hybrid Diagrams

New Concepts from UML 2.0

An Overview of an Interaction

Building the Big Picture

Summary

Q&A

Workshop

## HOUR 12: Working with Component Diagrams

What Is (and What Isn't) a Component?

Components and Interfaces

What Is a Component Diagram?

Applying Component Diagrams

Component Diagrams in the Big Picture

Summary

Q&A

Workshop

## HOUR 13: Working with Deployment Diagrams

What Is a Deployment Diagram?

Applying Deployment Diagrams

Deployment Diagrams in the Big Picture

Summary

Q&A

Workshop

## HOUR 14: Understanding Packages and Foundations

Package Diagrams

# Table of Contents

A Hierarchy

To Boldly Go . . .

Packaging the Infrastructure of UML

And Now At Last . . . the UML!

Extending the UML

Summary

Q&A

Workshop

## HOUR 15: Fitting the UML into a Development Process

Methodologies: Old and New

What a Development Process Must Do

GRAPPLE

RAD3: The Structure of GRAPPLE

The GRAPPLE Wrap-up

Summary

Q&A

Workshop

## Part II: A Case Study

### HOUR 16: Introducing the Case Study

Getting Down to Business

GRAPPLEing with the Problem

Discovering Business Processes

Lessons Learned

Summary

Q&A

Workshop

### HOUR 17: Performing a Domain Analysis

Analyzing the Business Process Interview

Developing the Initial Class Diagram

# Table of Contents

Grouping the Classes  
Forming Associations  
Forming Aggregates and Composites  
Filling Out the Classes  
General Issues About Models  
Lessons Learned  
Summary  
Q&A  
Workshop

## HOUR 18: Gathering System Requirements

Developing the Vision  
Setting Up for Requirements Gathering  
The Requirements JAD Session  
The Outcome  
Now What?  
Summary  
Q&A  
Workshop

## HOUR 19: Developing the Use Cases

The Care and Feeding of Use Cases  
The Use Case Analysis  
The Server Package  
Components of the System  
Summary  
Q&A  
Workshop

## HOUR 20: Getting into Interactions

The Working Parts of the System  
Interactions in the System

# Table of Contents

Implications

Summary

Q&A

Workshop

## HOUR 21: Designing Look, Feel, and Deployment

Some General Principles of GUI Design

The GUI JAD Session

From Use Cases to User Interfaces

UML Diagrams for GUI Design

Mapping Out System Deployment

Next Steps

And Now a Word from Our Sponsor

Summary

Q&A

Workshop

## HOUR 22: Understanding Design Patterns

Parameterization

Design Patterns

Chain of Responsibility

Your Own Design Patterns

The Advantages of Design Patterns

Summary

Q&A

Workshop

## Part III: Looking Ahead

### HOUR 23: Modeling Embedded Systems

Back to the Restaurant

The Mother of Invention

Fleshing Out the GetAGrip

# Table of Contents

What Is an Embedded System?

Embedded Systems Concepts

Modeling the GetAGrip

Flexing Their Muscles

Summary

Q&A

Workshop

## HOUR 24: Shaping the Future of the UML

Extensions for Business

Lessons from the Business Extensions

Graphic User Interfaces

Expert Systems

Web Applications

That's All, Folks

Summary

Q&A

Workshop

## Part IV: Appendixes

### APPENDIX A: Quiz Answers

Hour 1

Hour 2

Hour 3

Hour 4

Hour 5

Hour 6

Hour 7

Hour 8

Hour 9

Hour 10

# Table of Contents

Hour 11

Hour 12

Hour 13

Hour 14

Hour 15

Hour 16

Hour 17

Hour 18

Hour 19

Hour 20

Hour 21

Hour 22

Hour 23

Hour 24

## APPENDIX B: Working with a UML Modeling Tool

What You Should Find in a Modeling Tool

Working with UML in Visio Professional Edition

A Few Words About a Few Tools

## APPENDIX C: A Summary in Pictures

Activity Diagram

Class Diagram

Communication Diagram

Component Diagram

Composite Structure Diagram

Deployment Diagram

Object Diagram

Package Diagram

Parameterized Collaboration

Sequence Diagram

# Table of Contents

State Diagram

Timing Diagram

Use Case Diagram

Index