

# **SWE209**

# **Object Oriented Analysis and Design**

**Introduction**

# Note

- This presentation is based on the slides and content of the course main textbook.
- Bernd Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, 3rd Edition, Pearson, 2014
- [https://ase.in.tum.de/lehrstuhl\\_1/component/content/article/43-books/217](https://ase.in.tum.de/lehrstuhl_1/component/content/article/43-books/217)

# Agenda

Introduction

Why is Software Development Difficult?

What is Software Engineering?

Software Engineering Concepts

Software Engineering: A Working Definition

Software Engineering Development Activities

# Introduction

## Course Aim

- Teaching object-oriented techniques for conquering **complex** and **changing** software systems.
- Object-oriented analysis and design : Object-oriented software engineering

# Introduction

## Complex

- Software systems are complex.
- They perform many functions.
- They comprise many components.
- Many participants are required during development.
- They are difficult to understand completely by any single person.

# Introduction

## Change

- Software development projects constantly change.
- Requirements are complex.
- Staff change.
- Technologies change.
- Frozen requirements can lead to the deployment of an irrelevant system.

# Why is Software Development Difficult?

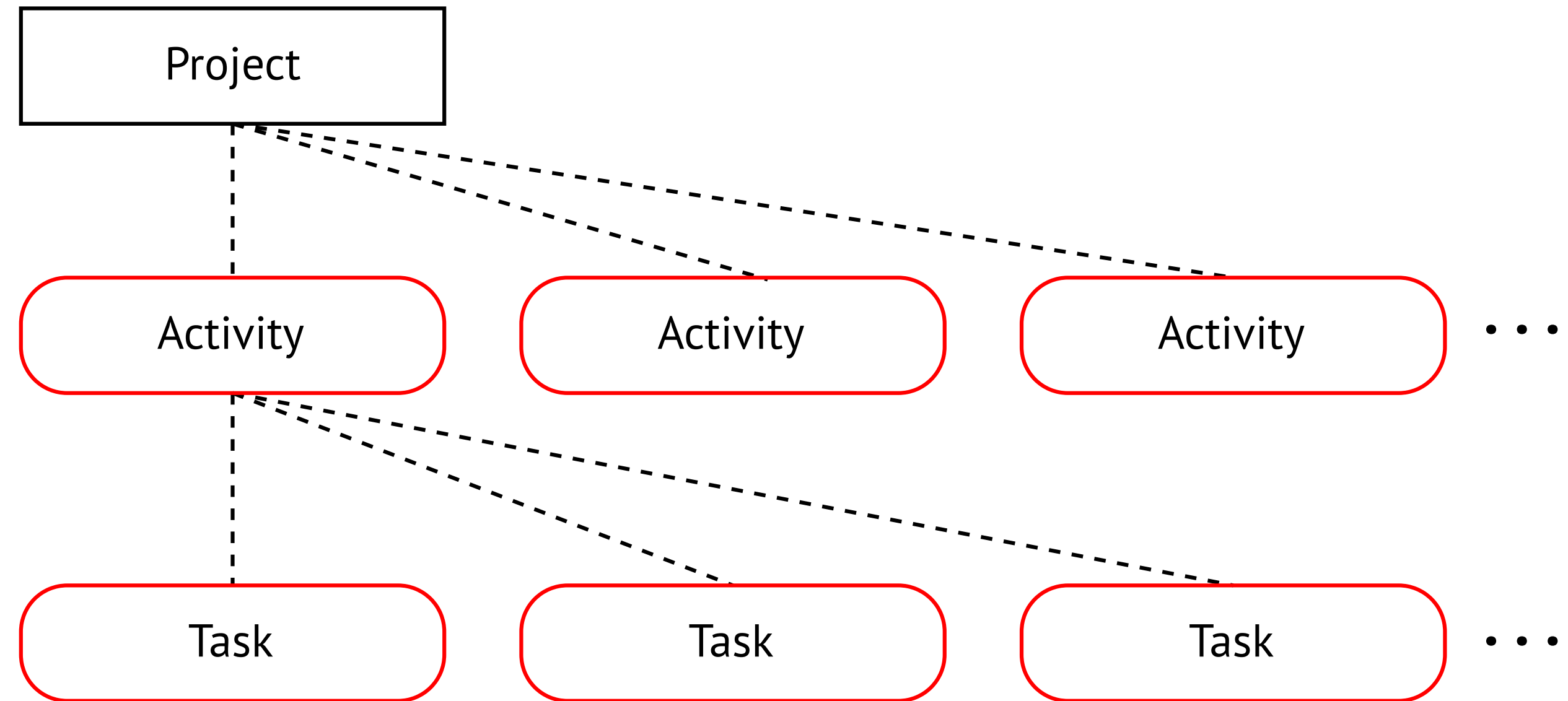
- The application domain (problem domain) is difficult.
- The solution domain is difficult.
- The development process is difficult to manage.
- Software offers extreme flexibility.
- Software is a discrete system.

# What is Software Engineering?

- Software engineering is more than writing code.
- Software engineering is;
  - Modeling
    - Model: An abstract representation of a system that enables us to answer questions about the system.
  - Problem solving
  - Knowledge acquisition
  - Rationale management



# Software Engineering Concepts



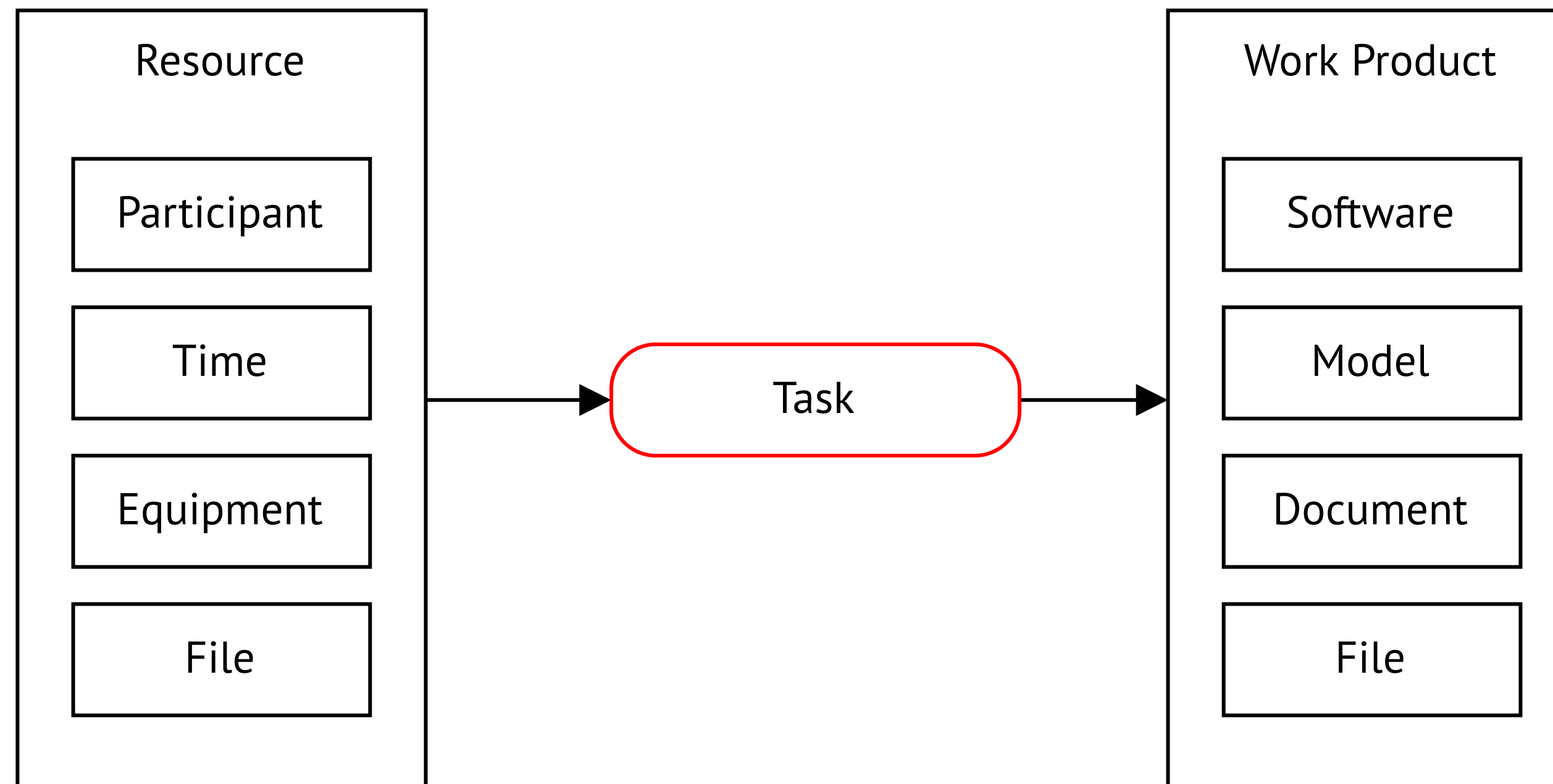
# Software Engineering Concepts

- Project
  - An entity whose purpose is to develop a software system.
  - Project is composed of a number of activities.
- Activity
  - Also called phase.
  - An entity that is composed of a number of tasks or other activities.

# Software Engineering Concepts

- Task
  - Atomic unit of work that can be managed.
  - Task produces work product.
  - Task consumes resources.

# Software Engineering Concepts



# Software Engineering Concepts

- Work Product
  - An artifact produced during the development.
  - Artifact: An object produced by a human.
  - A work product can be a software, a model, a document or a file.
- Resource
  - An asset that is used to accomplish a work.
  - A resources can be a participant, time, equipment or a file.

# Software Engineering Concepts

- Work Product
  - Software
    - A software (system), a piece of software, a library.
  - Model
    - An abstraction of the system.
  - Document
    - A piece of information about the project.
  - File
    - An entity that stores different types of information.

# Software Engineering Concepts

- Resource
  - Participant
    - All the people involved in the project.
  - Time
    - The time spend between the start and the end of the project.
  - Equipment
    - Tools used during the development of the project.
  - File
    - An entity that stores different types of information.

# Software Engineering Concepts

- A project is build around a set of requirements.
- Requirement
  - Requirements specify a set of features the system must have.
  - Two types:
    - Functional requirement
    - Nonfunctional requirement



# Software Engineering Concepts

- Three main instruments of a project: notation, method, and methodology.
- Notation
  - Graphical or textual set of rules for representing a model.
  - Unified Modeling Language (UML)
- Method
  - A repeatable technique that specifies the steps involved in solving a specific problem.

# Software Engineering Concepts

- Methodology
  - A collection of methods for solving a class of problems.
  - It also specifies how and when each method should be used.

# Software Engineering: A Working Definition

- Software Engineering is a collection of **notations**, **methods**, and **methodologies** that helps to the production of
  - A **high quality** software system developed with a given **budget** before a given **deadline** while **change** occurs.

# Software Engineering Development Activities

- Requirements Elicitation
- Analysis
- System Design
- Object Design
- Implementation
- Testing
- Maintenance

# References

- Bernd Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, 3rd Edition, Pearson, 2014.
- Object Management Group, OMG Unified Modeling Language Superstructure, Version 2.2., <http://www.omg.org/2009>.

**Thank you.**