SOFTWARE ARCHITECTURE AND DESIGN

Trần Đình Quế

Mail: tdque@yahoo.com

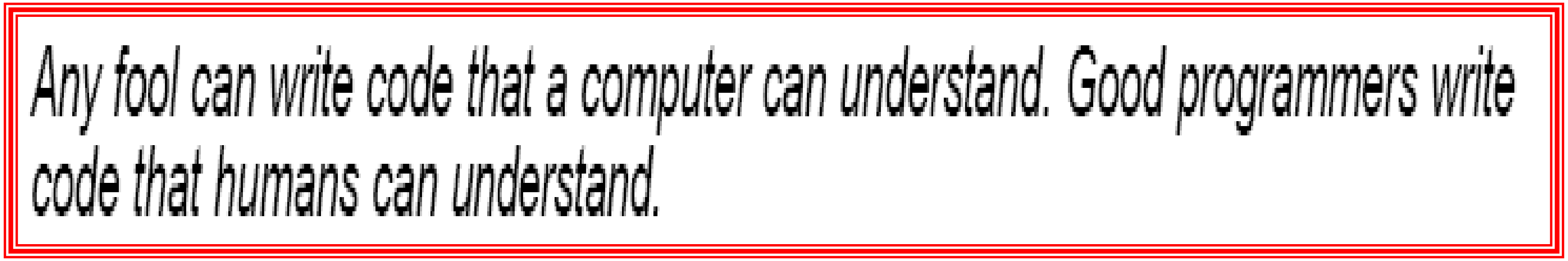
Phone: 0904066883

4/6/2020

Course Contents
General view

- **Architecture design**
- Architecture patterns
- Subsystem/detailed design [components/packages/classes]
- Design patterns
- Design intelligent systems

Questions?????



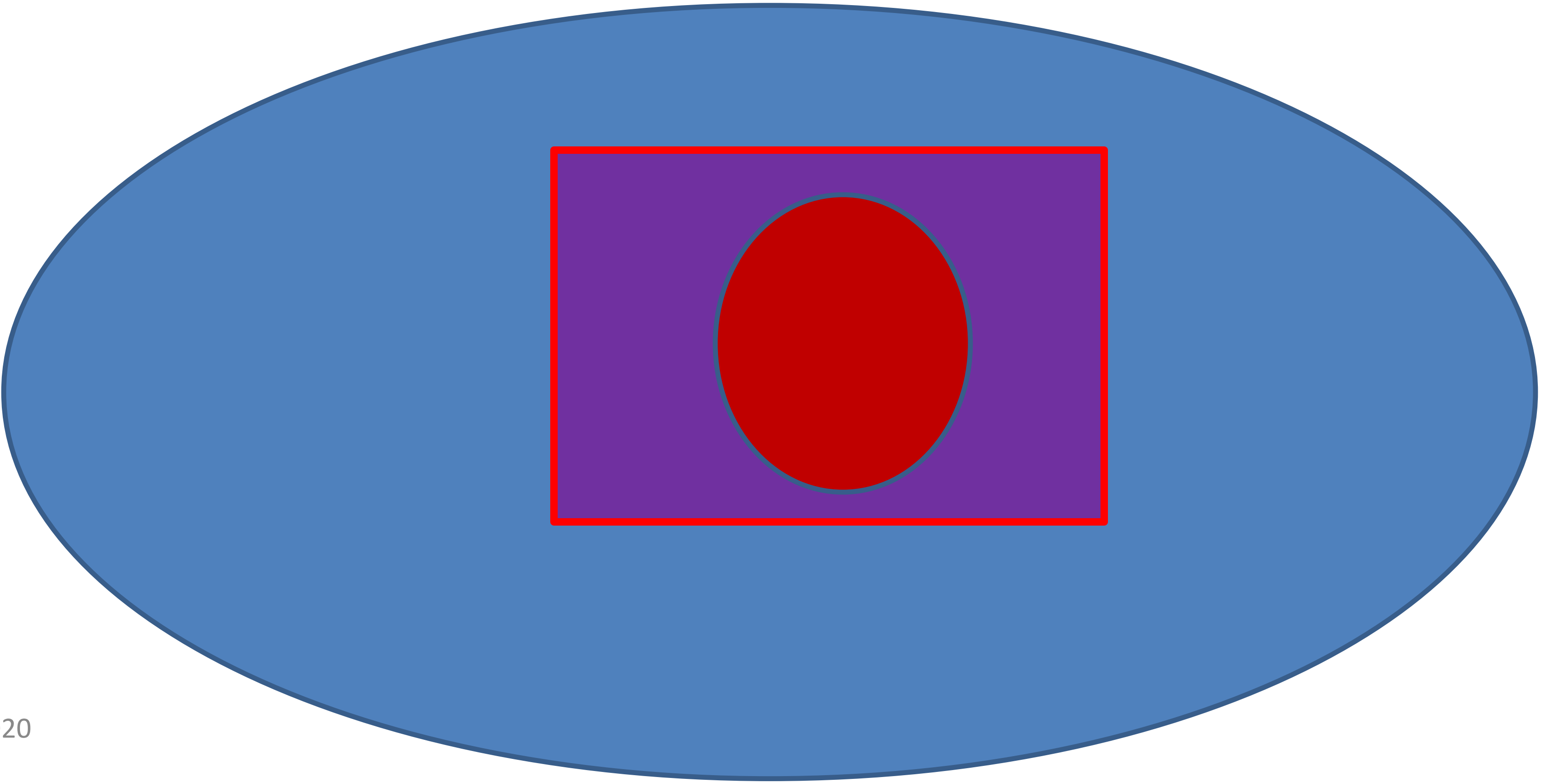
Any fool can write code that a computer can understand. Good programmers write code that humans can understand.

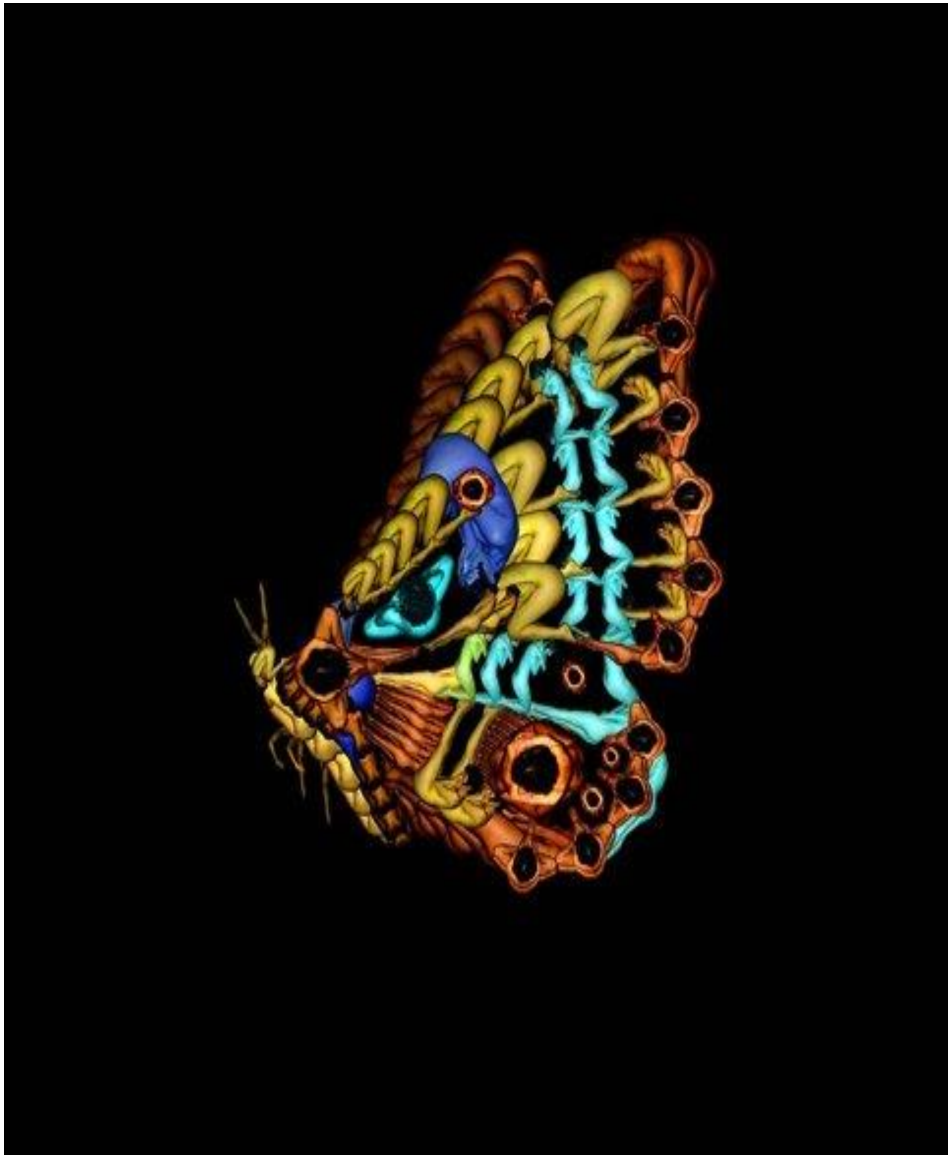
- Philosophy of architecture design: what is basis for define, determine software architecture?
- View?? Software intensive systems
- Static/dynamic, inside/outside?
- Stakeholders?

Questions ?????

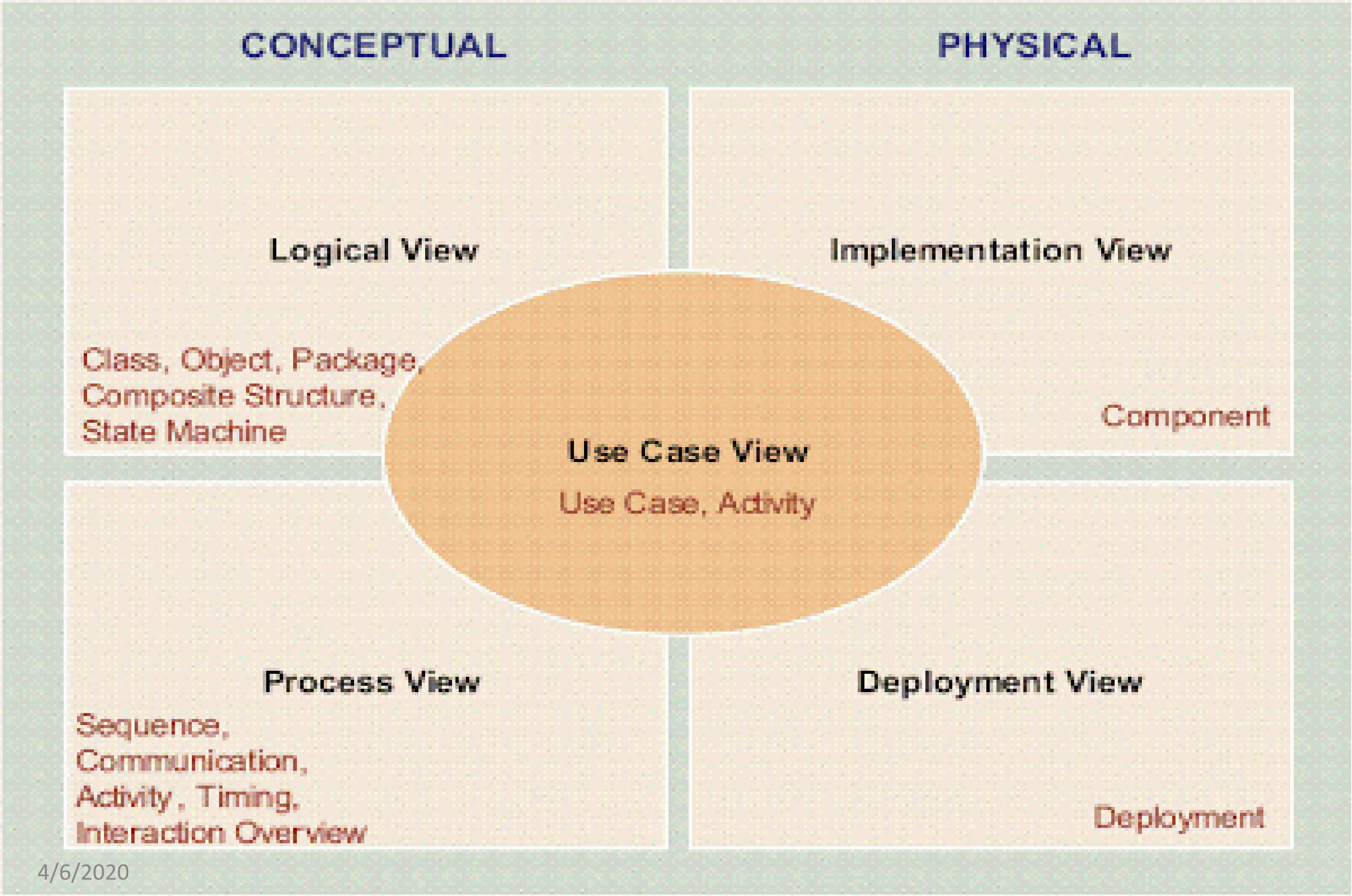
- The more you know, the more you don’t know

//the more you learn, the more you are stupid





4+1 View


CONCEPTUAL
PHYSICAL
Logical View
Class, Object, Package, Composite Structure, State Machine
Implementation View
Component
Use Case View
Use Case, Activity
Process View
Sequence, Communication, Activity , Timing, Interaction Overview
Deployment View
Deployment

4+1 view

- **4+1** is a view model used to describe the architecture of software-intensive systems, based on the use of multiple, concurrent views.
- The views are used to describe the system from the viewpoint of different **stakeholders**, such as end-users, developers, system engineer, and project managers.

4+1 view

- The four views of the model are logical, development, process and physical view.
- In addition, selected use cases or scenarios are used to illustrate the architecture serving as the 'plus one' view. Hence, the model contains 4+1 views

4+1 view

- ***Logical view*:** The logical view is concerned with the functionality that the system provides to end-users. UML diagrams are used to represent the logical view, and include class diagrams, and state diagrams.
- ***Process view*:** The process view deals with the dynamic aspects of the system, explains the system processes and how they communicate, and focuses on the run time behavior of the system. The process view addresses concurrency, distribution, integrator, performance, and scalability, etc. UML diagrams to represent process view include the sequence diagram, communication diagram, activity diagram.

4+1 view

- ***Development* /implementation *view*:** This view illustrates a system from a programmer's perspective and is concerned with software management. It also is named view. It uses the Component diagram to describe system components and the Package diagram to represent the development view.
- ***Physical /Deployment view***: The physical view depicts the system from a system engineer's point of view. It is concerned with the topology of software components on the physical layer as well as the physical connections between these components. UML diagrams used to represent this view include the deployment diagram.

4+1 view

- ***Scenarios/Use cases***: a small set of use cases or scenarios, which become a fifth view, is used to illustrate the description of an architecture .
- The scenarios describe sequences of interactions between objects and between processes. They are used to identify architectural elements and to illustrate and validate the architecture design. They also serve as a starting point for tests of an architecture prototype. This view is also known as the **use case view**.

Many in 1

https://www.visual-paradigm.com/guide/

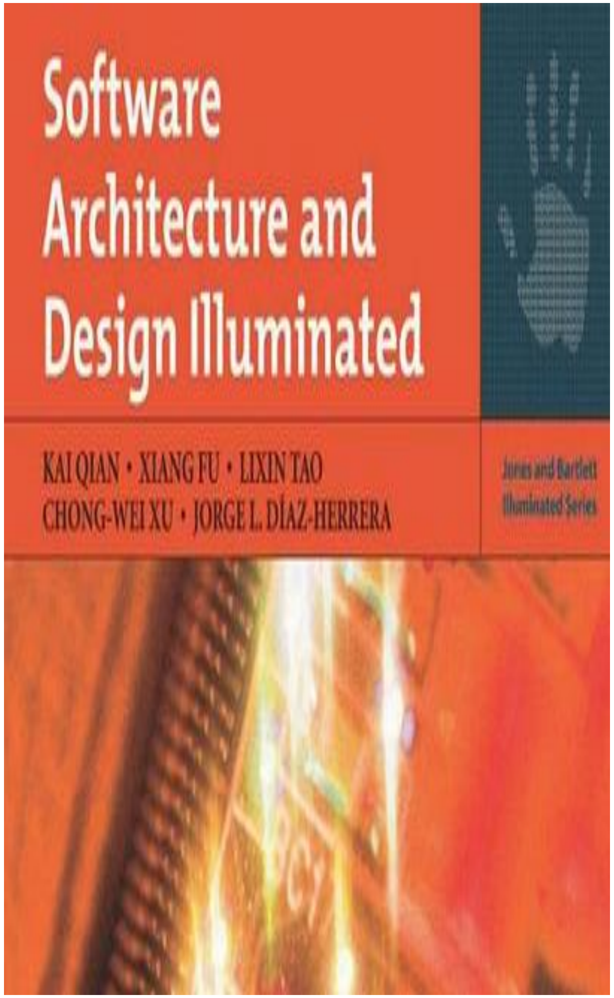
Part I: Design architecture

TRẦN ĐÌNH QUẾ

GIÁO TRÌNH

KIẾN TRÚC

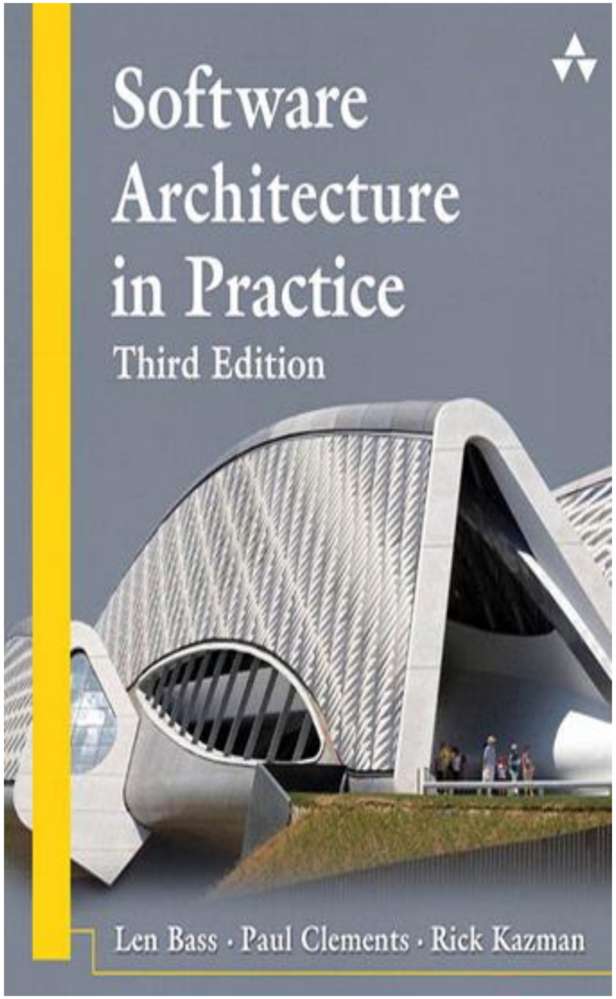
VÀ THIẾT KẾ PHẦN MỀM



Software Architecture and Design Illuminated

KAI QIAN • XIANG FU • LIXIN TAO
CHONG-WEI XU • JORGE L. DÍAZ-HERRERA

Jones and Bartlett Illuminated Series



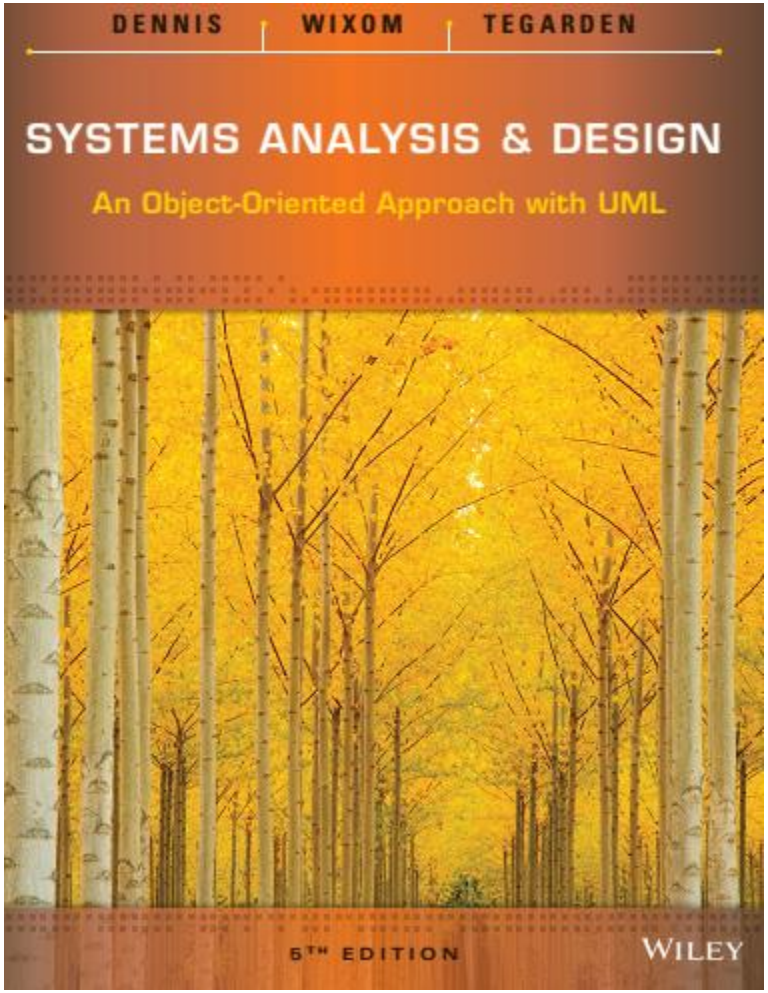
Software Architecture in Practice

Third Edition

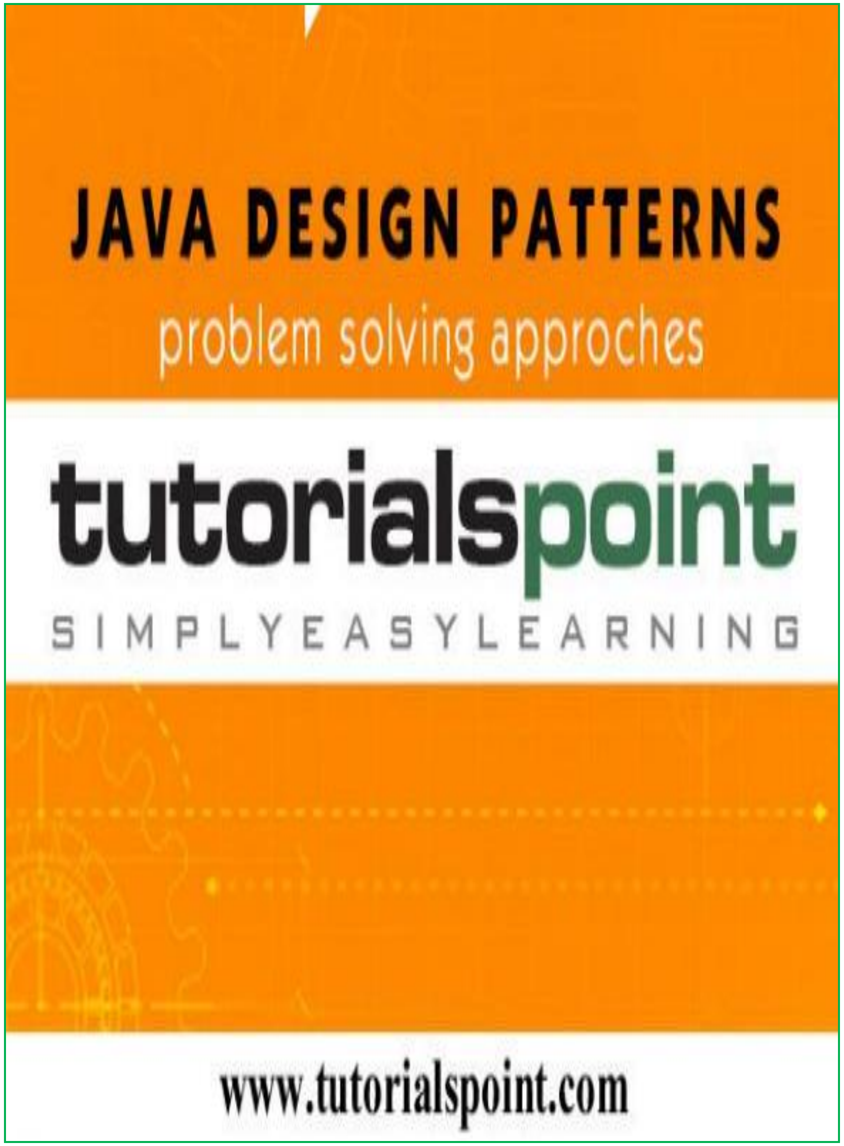
Len Bass · Paul Clements · Rick Kazman

Part II: Design Layers

- Design layers
- Design package
- Design class
- Design method


DENNIS
WIXOM
TEGARDEN
SYSTEMS ANALYSIS & DESIGN
An Object-Oriented Approach with UML
5TH EDITION
WILEY

Part III: Design patterns



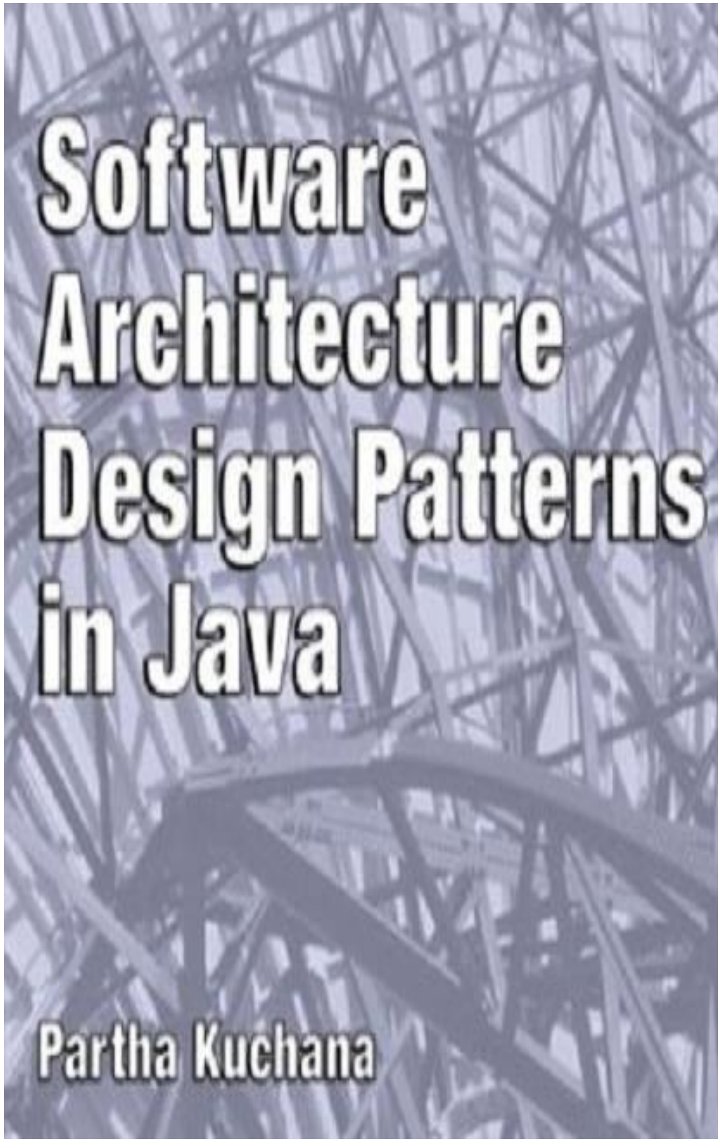
JAVA DESIGN PATTERNS

problem solving approches

tutorialspoint

SIMPLYEASYLEARNING

www.tutorialspoint.com

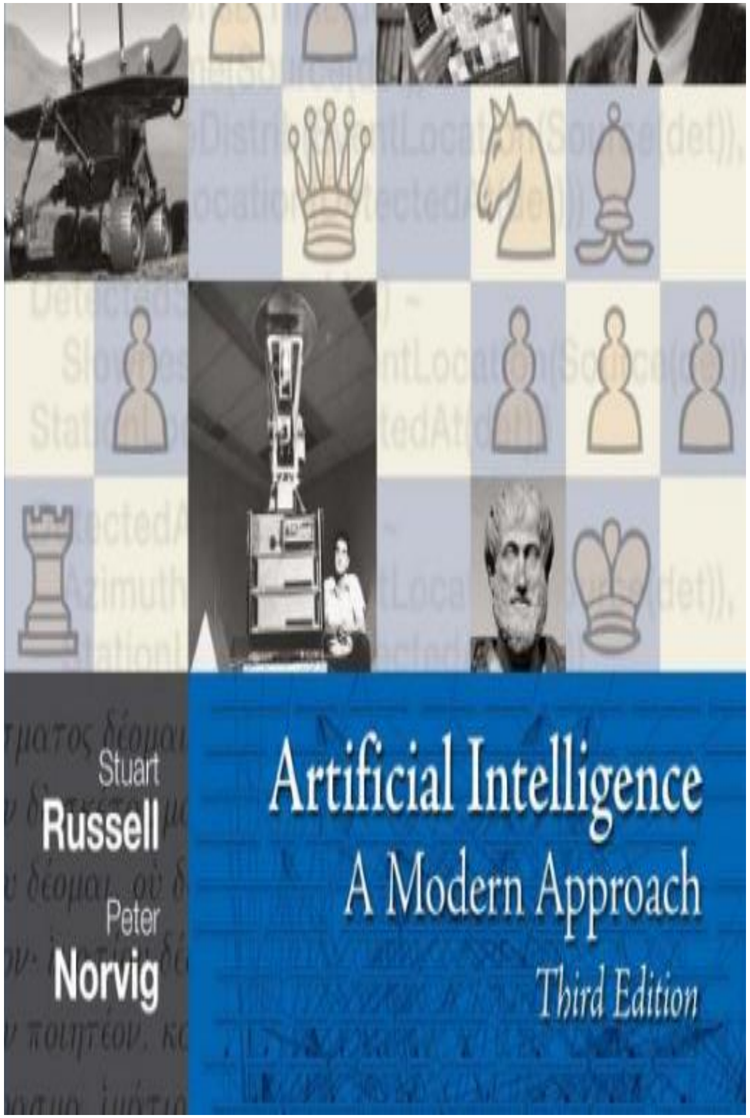


Software Architecture Design Patterns in Java

Partha Kuchana

Part IV: Intelligent systems

- Artificial intelligence
- Machine learning
- Design intelligent systems



Assessment

- 20%: 4 Assignments
- 20%: Big Project (group working)
- 60%: Final examination (writing)