Software Engineering Design & Construction

Dr. Michael Eichberg
Fachgebiet Softwaretechnik
Technische Universität Darmstadt

Design Patterns

A design pattern describes:
- *A problem that occurs over and over again in our environment.*
- *The core of the solution to that problem,*
- *in such a way that you can use this solution a million times over, without ever doing it the same way twice.*

– Christopher Alexander, professor of architecture.

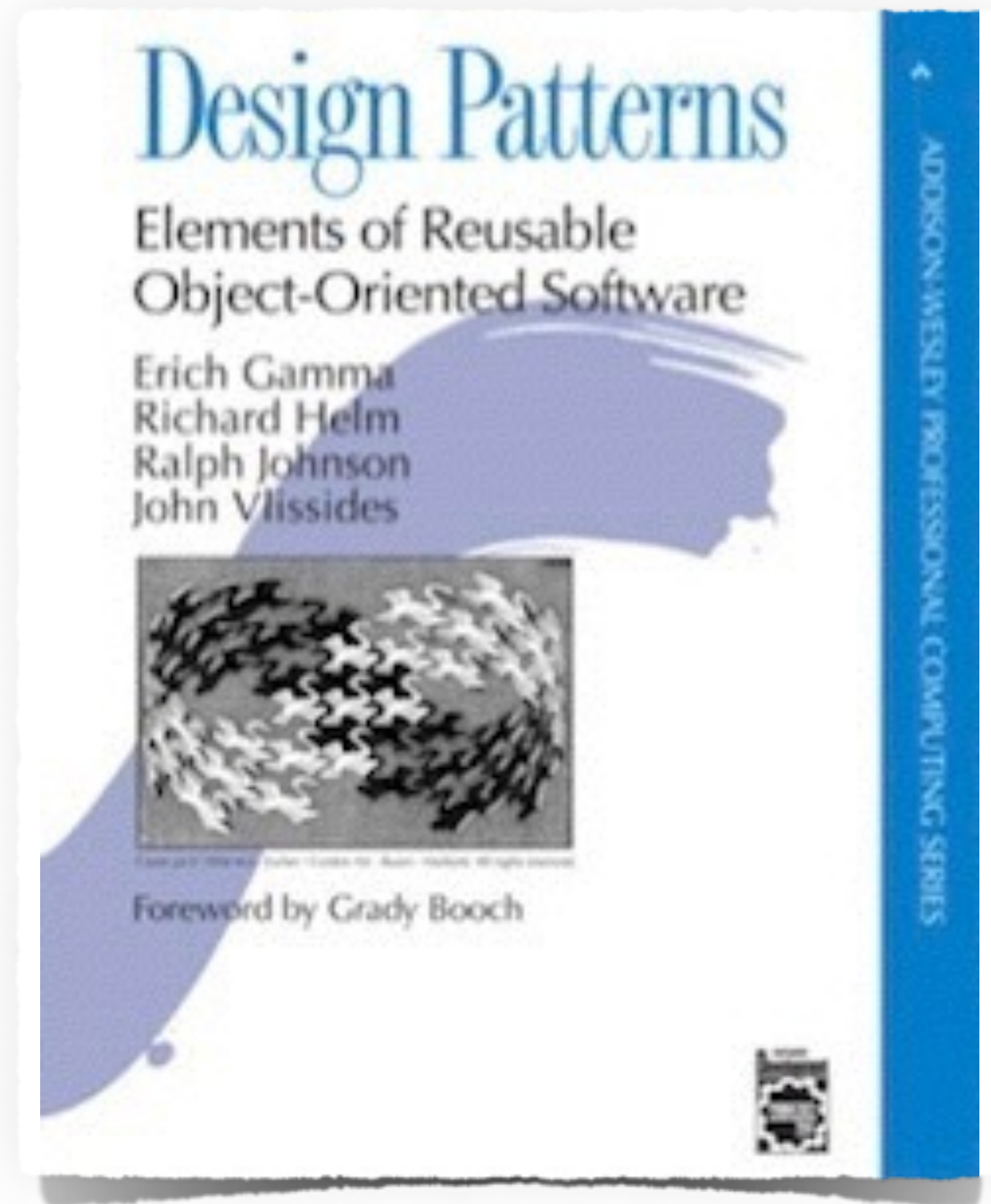# What is a Design Pattern?

Aggressive disregard for originality.

Rule of three:

- Once is an event.

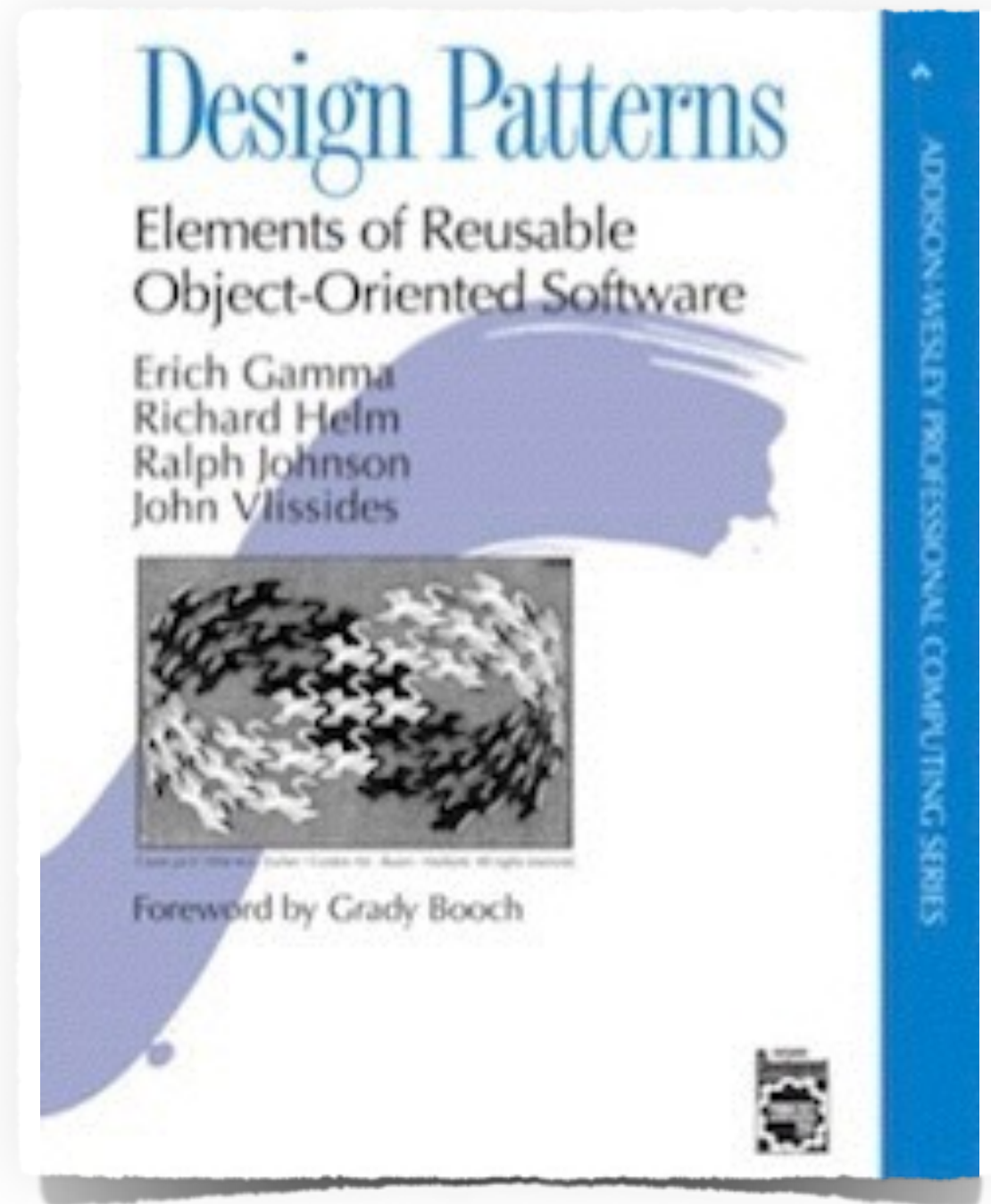- Twice is an incident.

- Thrice is a pattern.

Motivation for Software Design Patterns

- Designing reusable software is hard!
(Originality is Overrated)

- Some design solutions reoccur; understanding their core is beneficial
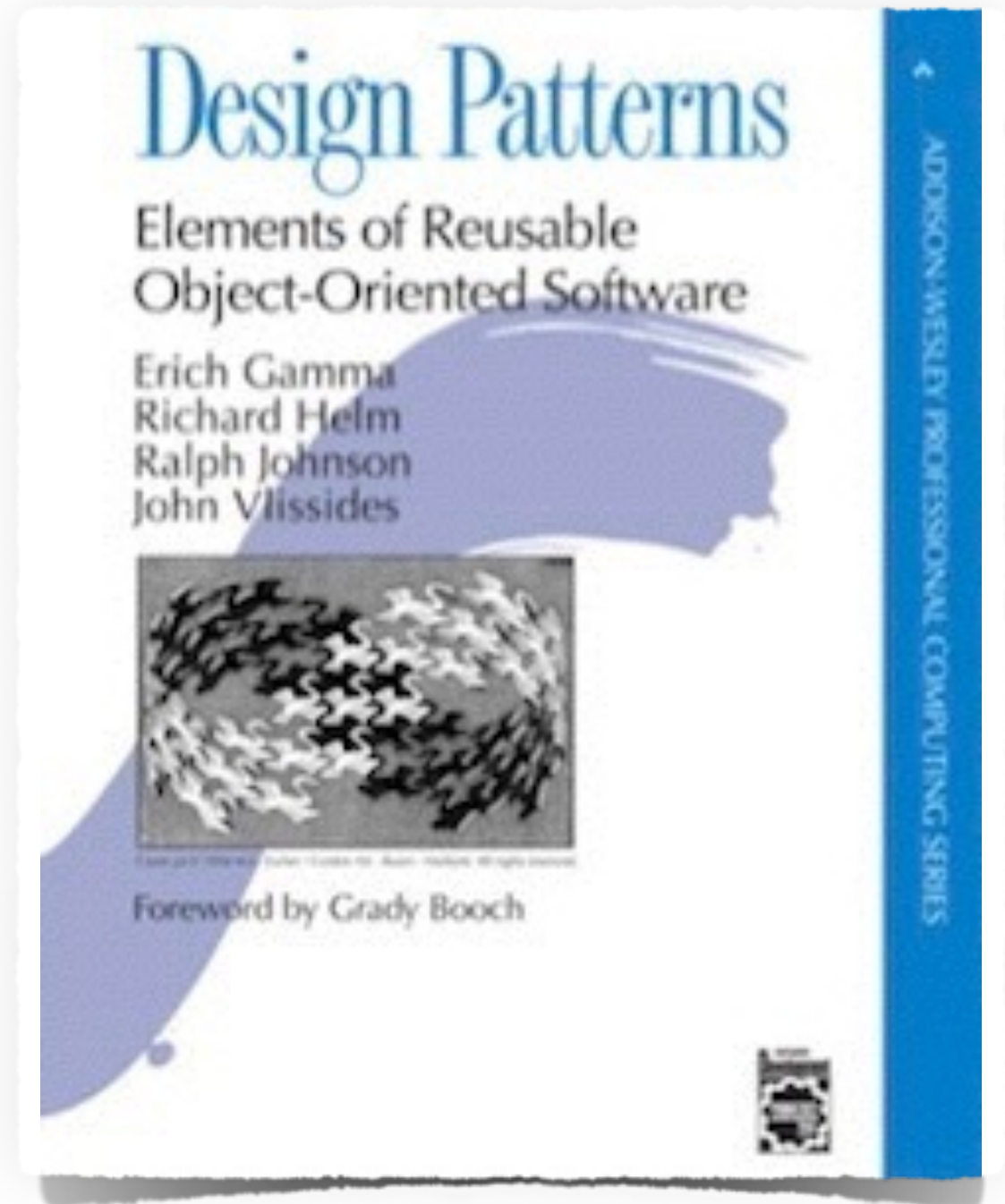
- Systematic software-development

Design Patterns and Change

- Most patterns address issues of software change/evolution.

  - Most patterns allow some part of the system to vary independent of the other parts.

  - We often try to identify what varies in a system and encapsulate it.

Elements of Design Patterns

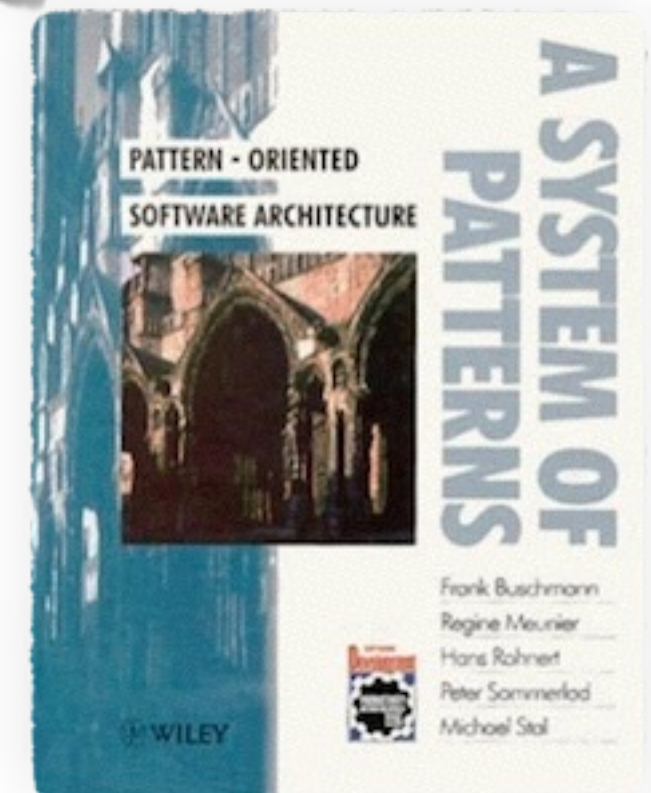- Pattern Name

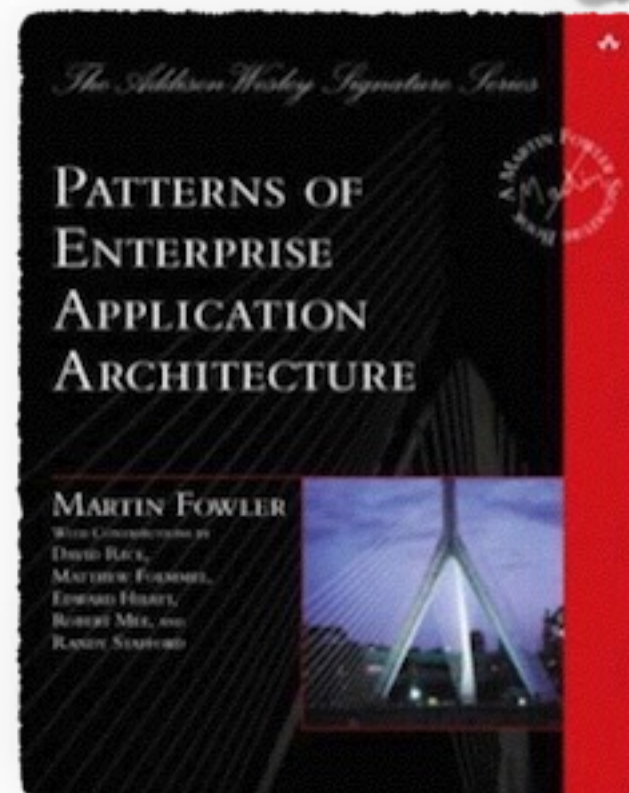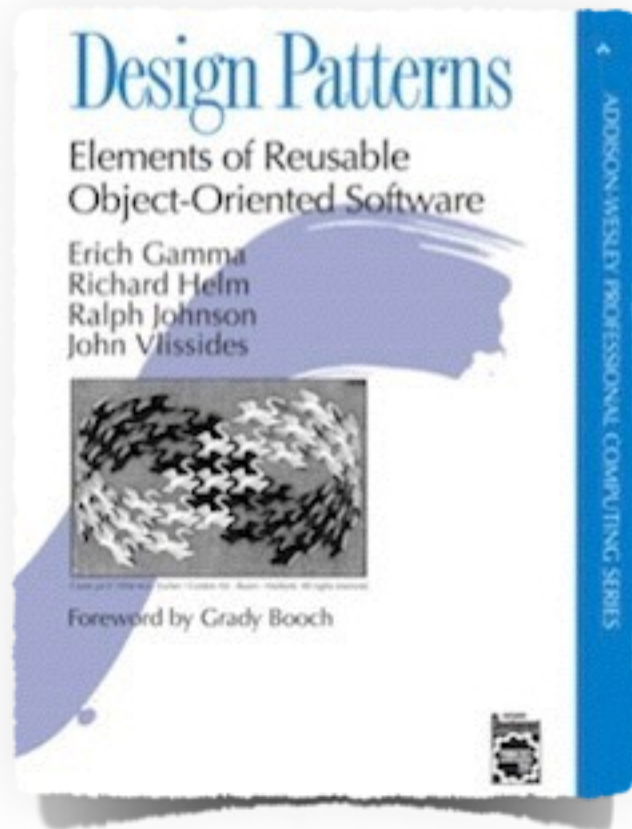- Intent

- Solution

- Consequences

R. Martin's Chess Analogy

- When people begin to play chess they learn the rules and physical requirements of the game.

- As they progress, they learn the principles.

- However, to become a master of chess, one must study games of other masters.

- So it is with software. First one learns the rules. The algorithms, data structures, and languages of software.

- Later, one learns the principles of software design.

- But to truly master software design, one must study the designs of other masters.
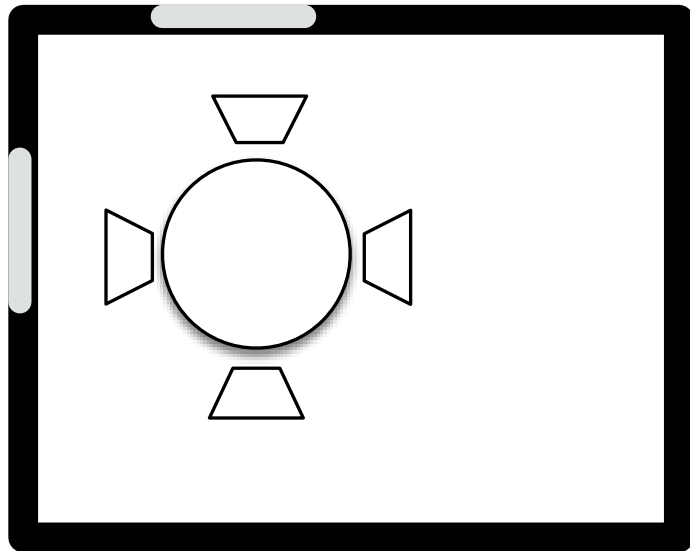
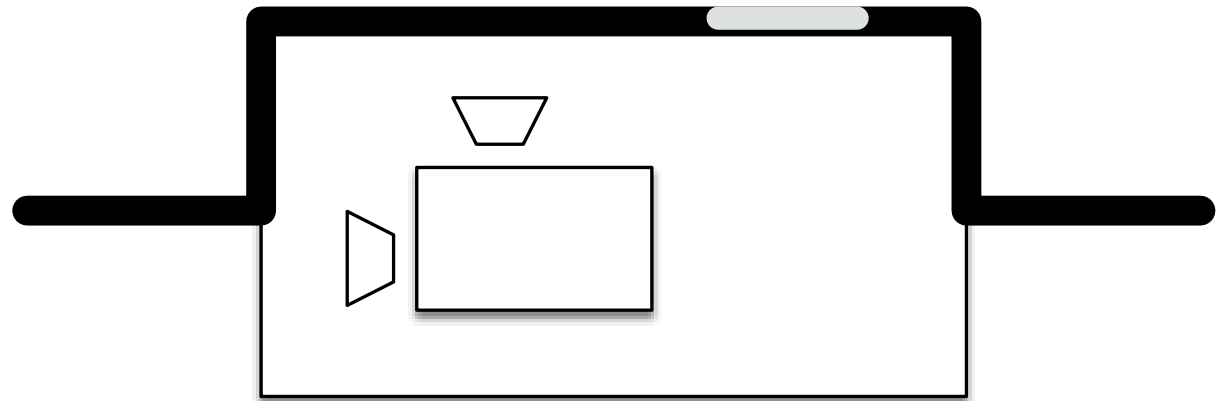# Software Design Patterns

Selected Books

Patterns in Architecture

Light from two sides

Deep terrace

Discussed Patterns

- Quick warm up with the Template Method Pattern

- The Strategy Pattern

- The Decorator Pattern

- The Proxy Pattern

- The Visitor Pattern

- The Bridge Pattern

- The Adapter Pattern

- The Builder Pattern

- The Command Pattern