



Practical Design Patterns



Eamonn Faherty

About Me

Blog - <http://blog.eamonnfaherty.co.uk>



Twitter - @eamonnfaherty



Object Orientated Programming

Code organized into groups called classes which represent things in the real world.

Classes are nouns which have behaviors which are called methods.

Methods are verbs.



Object Orientated Programming - Problems

Different programming paradigm

- Different grouping
- Where code is?

Objects? Customer, transaction, product?

- Lots of them?
- Reusable?



Object Orientated Programming : Reuse

- Verbs and nouns in the same class
- (Re)use some behaviors across objects but not all
- Share some attributes of objects but not all
- Composite objects - coupling issues

Object Orientated Programming

The title is centered at the top of the slide. It is flanked by five circles: a solid light purple circle on the far left, a hollow light purple circle, a solid light purple circle, a hollow light purple circle, and a solid light purple circle on the far right.

- Interfaces
- Inheritance
- Polymorphism
 - Sub-typing
 - Method Overloading
 - Method Overriding

Object Orientated Programming : Reuse again!

- Example:

Car
numberOfDoors:uint engine:Engine
drive() reverse()

PoliceCar
numberOfDoors:uint engine:Engine callSign:String
drive() reverse() turnOnSiren() turnOffSiren()

MiniCab
numberOfDoors:uint engine:Engine callSign:String
drive() reverse() pickUp(customer) dropOff(customer)



Object Orientated Programming : Reuse again!

- Verbs and nouns in the same class - **STRUCTURAL**
- Reuse some behaviors across objects but not all - **BEHAVIORAL**
- Share some attributes of objects but not all - **STRUCTURAL**
- Composite objects - coupling issues - **CREATIONAL**

Design Patterns - Types

The title is centered at the top of the slide. It is flanked by five circles: a solid light purple circle on the far left, a hollow light purple circle, a solid light purple circle, a hollow light purple circle, and a solid light purple circle on the far right.

- Creational Patterns

- Create objects for you so that you don't have to.

- Structural Patterns

- Composing objects together so that you can (re)use parts or all of them.

- Behavioral Patterns

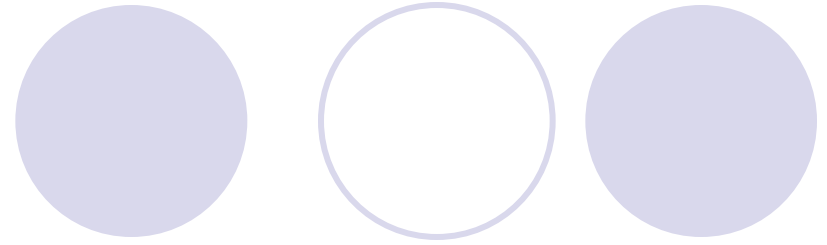
- Define communication between objects to help simplify complicated systems.

Design Patterns - Other types

The title is centered at the top of the slide. It is flanked by five circles: a solid light purple circle on the far left, a hollow light purple circle, a solid light purple circle, a hollow light purple circle, and a solid light purple circle on the far right.

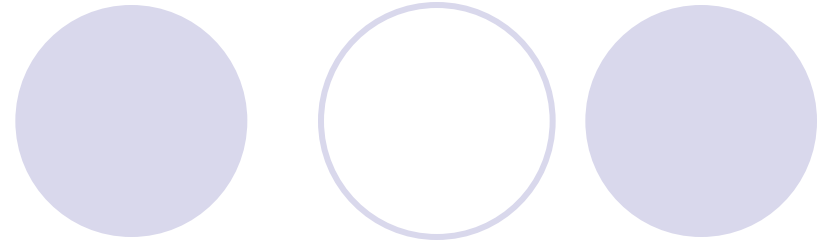
- Concurrency patterns
- Enterprise
- More...

Creational Patterns



- Singleton
- Multiton
- Abstract Factory
- Factory Method
- Builder
- Object Pool

Structural Patterns



- Adapter
- Bridge
- Composite
- Decorator
- Façade
- Proxy

Behavioral Patterns



- Chain of Responsibility
- Command
- Iterator
- Mediator
- Memento
- Observer
- Template



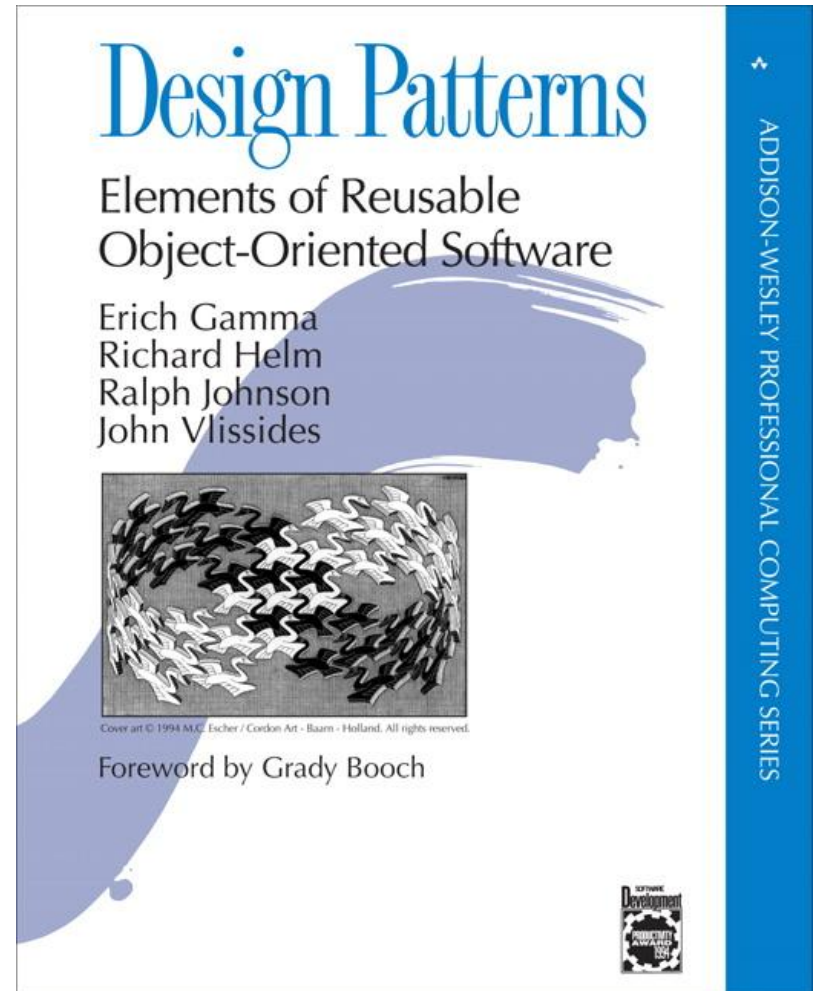
Summary / Conclusion

- Design patterns are not a silver bullet
 - Sometimes makes code more complex - too many abstraction layers
 - Sometimes confusing
- Design patterns are great for communication when talking about designs

Reading: Object Orientated Programming

“Design Patterns”

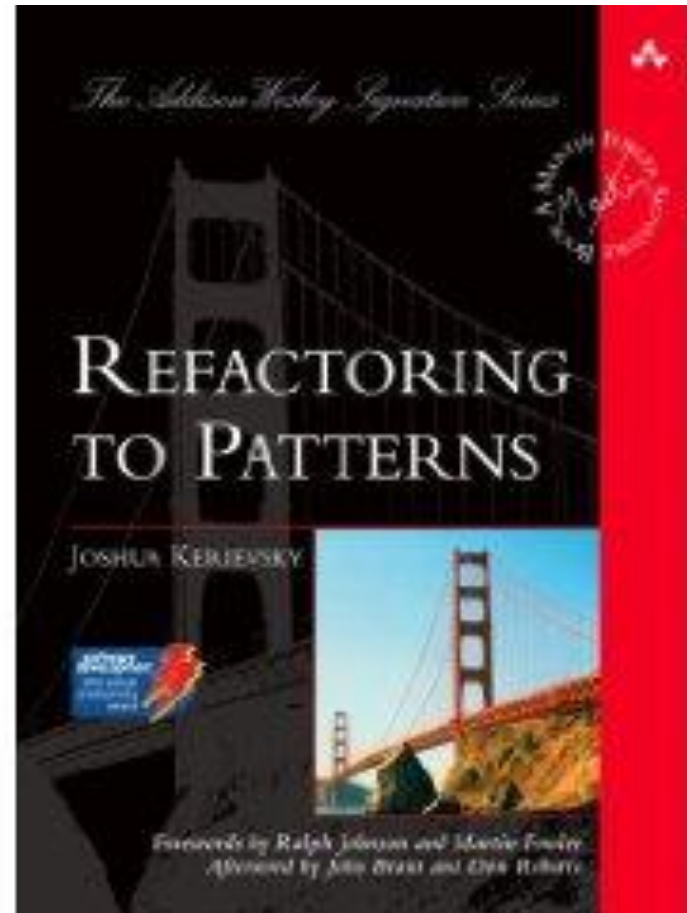
- straight to the point
- covers a lot of patterns
- not in AS3
- languages used are very different to AS3
- it is a tough read



Reading: Object Orientated Programming

“Refactoring to Patterns”

- Good advice on keeping code simple and clean
- Good compliment to Gang of Four book
- Good example uses of design patterns



Reading: General

“Refactoring”

- One of the best technical books I have read!
- Great descriptions of code smells
- Clear steps on how to improve design

