SOFTWARE ENGINEERING

DESIGN PATTERNS

Mahdi Roozbahani

Slides are based on Alex Orso.

HISTORY OF (DESIGN) PATTERNS

HISTORY OF (DESIGN) PATTERNS

1977 Christopher Alexander introduces the idea of patterns: succesfull solutions to problems

HISTORY OF (DESIGN) PATTERNS

1977 Christopher Alexander introduces the idea of patterns: succesfull solutions to problems

1987 Ward Cunningham and Kent Beck leverage Alexander's idea in the context of an OO language

HISTORY OF (DESIGN) PATTERNS

1977 Christopher Alexander introduces the idea of patterns: succesfull solutions to problems

1987 Ward Cunningham and Kent Beck leverage Alexander's idea in the context of an OO language

1987 Eric Gamma's dissertation on importance of patterns and how to capture them

HISTORY OF (DESIGN) PATTERNS

1977 Christopher Alexander introduces the idea of patterns: succesfull solutions to problems

1987 Ward Cunningham and Kent Beck leverage Alexander's idea in the context of an OO language

1987 Eric Gamma's dissertation on importance of patterns and how to capture them

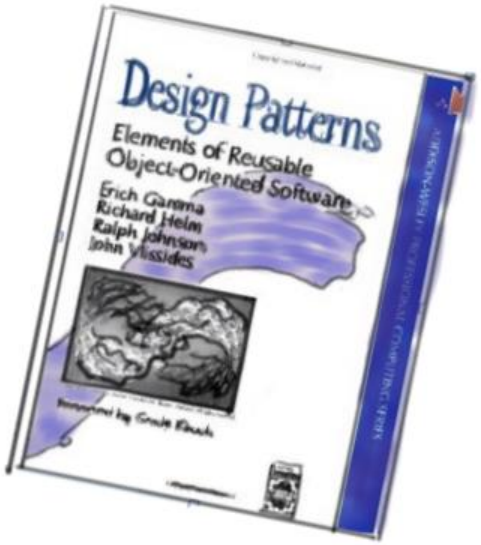
1992 Jim Coplien's book on Advanced C++ Programming Styles and Idioms

HISTORY OF (DESIGN) PATTERNS

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
(gang of four)

HISTORY OF (DESIGN) PATTERNS

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
(gang of four)



Book "Design Patterns:
Elements of Reusable
OO Software"

PATTERNS CATALOGUE

PATTERNS CATALOGUE

Fundamental patterns

PATTERNS CATALOGUE

Fundamental patterns

Creational patterns

PATTERNS CATALOGUE

Fundamental patterns

Creational patterns

Structural patterns

PATTERNS CATALOGUE

Fundamental patterns

Behavioral patterns

Creational patterns

Structural patterns

PATTERNS CATALOGUE

Fundamental patterns

Behavioral patterns

Creational patterns

Concurrency patterns

Structural patterns

PATTERNS CATALOGUE

Fundamental patterns

Delegation pattern
Interface pattern
Proxy pattern
...

Creational patterns

Abstract factory pattern
Factory method pattern
Lazy initialization pattern
Singleton pattern
...

Structural patterns

Adapter pattern
Bridge pattern
Decorator pattern
...

Behavioral patterns

Chain of responsibility pattern
Iterator pattern
Observer pattern
State pattern
Strategy pattern
Visitor pattern

Concurrency patterns

Active object
Monitor object
Thread pool pattern
...

FORMAT (SUBSET)

FACTORY METHOD PATTERN

Intent

Allows for creating objects without specifying their class, by invoking a factory method (i.e., a method whose main goal is To create class instances)

FACTORY METHOD PATTERN

Intent

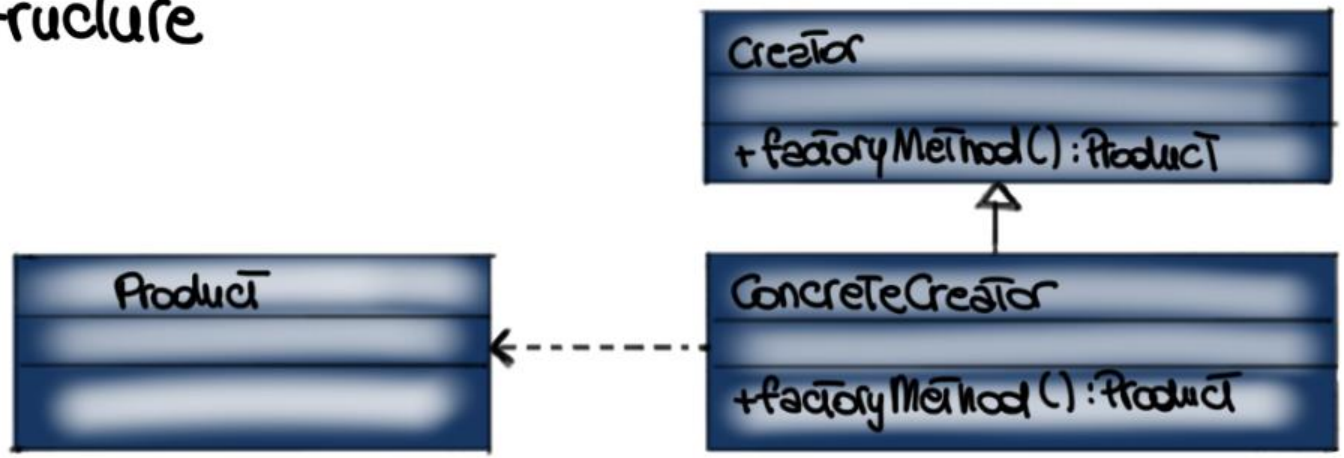
Allows for creating objects without specifying their class, by invoking a factory method (i.e., a method whose main goal is To create class instances)

Applicability

- Class can't anticipate the Type of objects it must create
- Class wants its subclasses to specify the type of objects it creates
- Class needs control over the creation of its objects

FACTORY METHOD PATTERN

Structure



Participants

Creator : provides interface for factory method

ConcreteCreator : provides method for creating actual object

Product : object created by the factory method

FACTORY METHOD PATTERN

```
public class ImageReaderFactory {
  public static ImageReader createImageReader(InputStream is) {
      int imageType = getImageType(is);
      switch(imageType){
        case ImageReaderFactory.GIF
            return new GifReader(is);
        case ImageReaderFactory.JPEG
            return new JpegReader(is);
        //...
      }
  }
}
```

STRATEGY PATTERN

Intent

Allows for switching between different algorithms for accomplishing a task

STRATEGY PATTERN

Intent

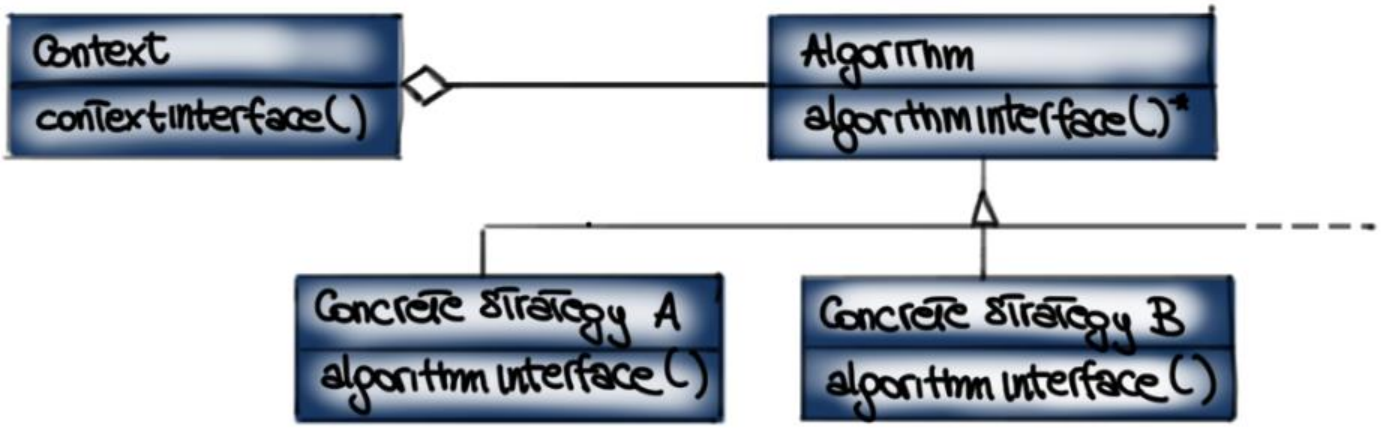
Allows for switching between different algorithms for accomplishing a task

Applicability

- Different variants of an algorithm
- Many related classes differ only in their behavior

STRATEGY PATTERN

Structure



Participants

Context : interface to outside world

Algorithm (strategy) : common interface for the different algorithms

Concrete Strategy : actual implementation of the algorithm

STRATEGY PATTERN: EXAMPLE

Program

- Input: text file
- Output: filtered file

Four filters

- No filtering
- Only words that start with "t"
- Only words longer than 5 characters
- Only words that are palindromes


DEMO

OTHER COMMON PATTERNS

OTHER COMMON PATTERNS

Visitor A way of separating an algorithm from an object structure on which it operates

[Visitor pattern example]

OTHER COMMON PATTERNS

Visitor A way of separating an algorithm from an object structure on which it operates

Decorator A wrapper That adds functionality to a class : stackable

Decorator Example

OTHER COMMON PATTERNS

Visitor A way of separating an algorithm from an object structure on which it operates

Decorator A wrapper That adds functionality to a class : stackable

Iterator Access elements of a collection without knowing underlying representation

[Iterator Example]

OTHER COMMON PATTERNS

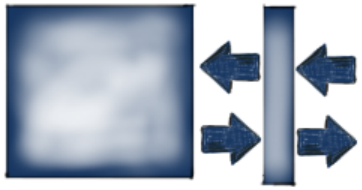
OTHER COMMON PATTERNS

Observer

Notify dependents when object changes

OTHER COMMON PATTERNS

Observer
Notify dependents when object changes



Proxy
Surrogate controls access to an object

CHOOSING A PATTERN

CHOOSING A PATTERN

Approach

- Understand your design context
- Examine the patterns catalogue
- Identify and study related patterns
- Apply suitable pattern

CHOOSING A PATTERN

Approach

- Understand your design context
- Examine the patterns catalogue
- Identify and study related patterns
- Apply suitable pattern

Pitfalls

- Selecting wrong patterns
- Abusing patterns

Imagine that you have to write a class that can have one instance only. Using one of the design patterns that we discussed in this lesson, write the code of a class with only one method (except for possible constructors) that satisfies this requirement.

NEGATIVE DESIGN PATTERNS

NEGATIVE DESIGN PATTERNS

Also in Christopher Alexander's book

NEGATIVE DESIGN PATTERNS

Also in Christopher Alexander's book

How not to (design, manage, etc.)

NEGATIVE DESIGN PATTERNS

Also in Christopher Alexander's book

How not to (design, manage, etc.)

Also called anti-patterns and bad smells