

# Object Oriented Analysis & Design

## 面向对象分析与设计

### Lecture\_09 GOF 设计模式 (一)

1) 单实例      2) 适配器      3) 外观      4) 观察者

**主讲: 姜宁康 博士**



# ■ 1、关于设计模式...Design Pattern

- 追求永恒的美

# 1.1 “模式”一词的起源

“每个模式描述了:

一个在我们周围反复出现的问题,  
然后是针对这个问题的解决方案。  
这样, 其他人可以无数次地反复  
使用这种解决方案, 不需要第二  
次再去思考该如何解决这类问题”

--- Christopher Alexander,  
1979



Christopher Alexander 最初是在建筑行业、小镇建设中提出来的  
在软件工程中, 设计模式是建立在类、对象、接口这些概念的上  
面, 而不是砖块、门  
In SE, design patterns are in terms of objects and interfaces, not walls and doors

# 1.2 GoF (Gang of Four) 设计模式

Design Patterns – Elements of Reusable Object-Oriented Software by Erich Gamma, Richard Helm , Ralph Johnson & John Vlissides

Addison-Wesley, 1995. (As CD, 1998)

简称 **GOF**, “四人帮”

First systematic software pattern description

第一本系统性引入**软件设计模式**

注意: 设计模式不是他们“**创建**”出来的, 而是他们在大量软件中“**提炼**”出来的

Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma  
Richard Helm  
Ralph Johnson  
John Vlissides



Foreword by Grady Booch



ADDISON WESLEY PROFESSIONAL COMPUTING SERIES

# 1.3 Essential Elements of Design Patterns

## ■ 设计模式的特点

- 描述了一个反复出现的问题 describes a problem which occurs over
- 描述了核心的解决方案 describes the core of the solution
- 其他人可以无数次地使用这个方案解决类似的问题

## ■ 模式的定义，四个本质的构成 four essential elements

- 模式的名字 Name of the Pattern
- 模式解决的问题 The Problem
- 模式提出的解决方案 The Solution
- 应用模式的后果、折衷考虑的问题 Consequences, tradeoffs

## ■ 注意点

- 解决方案针对反复出现的问题 a solution to a recurrent problem
- 不是一个“具体”的解决方案，而是一个抽象的方案 not a “concrete” solution, but an abstract version of it

# 1.3 Essential Elements of Design Patterns

Item	Description
Name	All patterns have a unique name that identifies them.
Intent	The purpose of the pattern.
Problem	The problem that the pattern is trying to solve.
Solution	How the pattern provides a solution to the problem in the context in which it shows up.
Participants and Collaborators	The entities involved in the pattern.
Consequences	The consequences of using the pattern. Investigates the forces at play in the pattern.
Implementation	How the pattern can be implemented. <i>Note:</i> Implementations are just concrete manifestations of the pattern and should not be construed as the pattern itself.
GoF Reference	Where to look in the Gang of Four text to get more information.

## 1.4 模式的作用 Patterns help to

- **解决某些特殊的设计问题** solve specific design problems
- **减少重复设计的工作量** reduce the need for redesign
  - 重用设计方案比重用代码更有意义，它会自动带来代码重用
  - 模式在实践中可作为模板使用 act as templates
- **把专家的知识、经验传递给新手** pass on knowledge from experts to novices
- **为设计提供共同的词汇** Common Vocabulary
  - 每个模式名就是一个设计词汇，其概念使得程序员间的交流更加方便
- **编写开发文档更加容易** Easy Documentation
  - 在开发文档中采用模式名称可以让其他人更容易理解设计师的想法，为什么这么做？做了些什么？
- **应用设计模式可以让系统重构变得容易** Easy refactor

## 1.5 关于模式的误解

### ■ 模式不是 Patterns are not ...

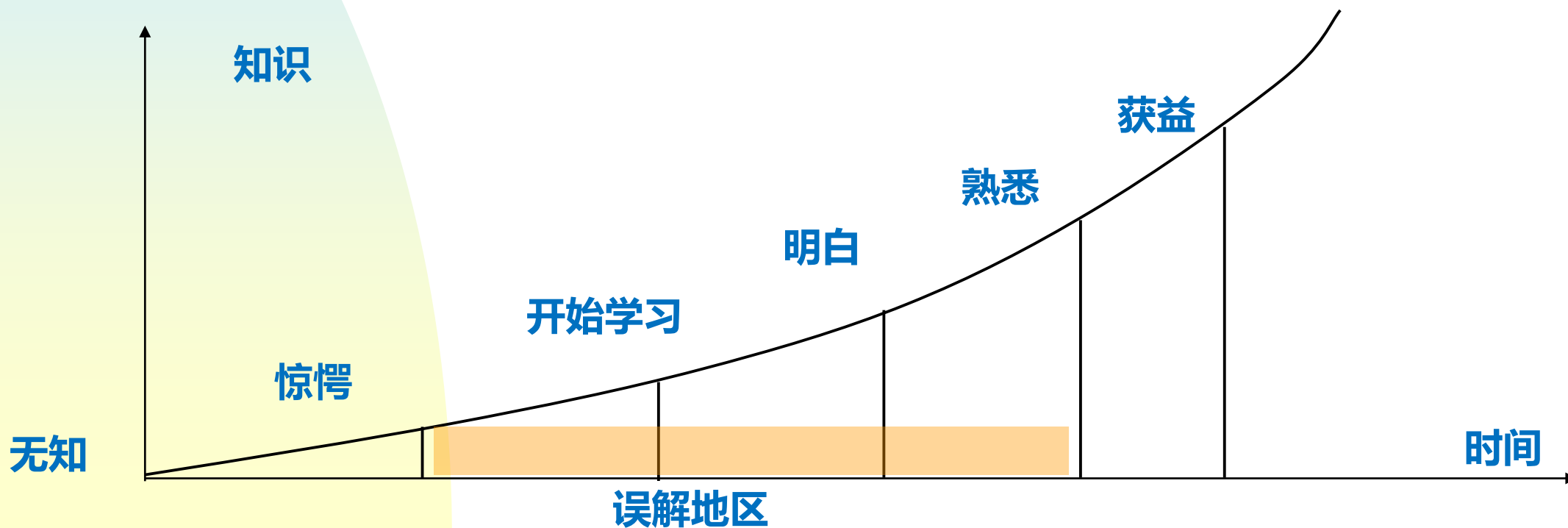
- 类 Classes
- 函数库 Libraries
- 功能包 Packages
- 宏定义 Macros
- 高阶函数 Higher-order functions
  - 一个高阶函数可以将函数作为参数，也可以返回一个函数
- 模板类 Template classes

### ■ 但是，模式确实是由这些元素构成的 However, some of these could conceivably(令人信服地) capture some design patterns



# 1.6 模式的学习进阶

- 模式的学习是一个渐进的过程
  - 误解了模式是什么
  - 误解了模式能做什么
- 注意：
  - 下决心学习模式、弄明白模式的本质，并不断地应用熟悉，以后的收益很大
  - 模式不会为你提供什么保证，它只能提高你的潜能，让你的软件设计更加成熟





■ **本讲结束**