

# 软件体系结构设计模式的分析研究

## Analytic Investigation Of software architecture design patterns

楚荣珍 刘建国 (邯郸河北工程学院 056038)

李顺刚 (邯郸河北冶金建设集团有限责任公司 056038)

**摘要:**介绍了以面向对象的技术为基础,以设计模式为指导,软件体系结构为核心的软件开发过程,并针对当前软件体系结构描述语言和工具普遍存在的通用性较差,提出使用设计模式来表示软件系统的体系结构的观点,讨论了使用设计模式的一些原则,以达到最大限度的设计重用。

**关键词:**软件体系结构 设计模式 面向对象

### 1 前言

面向对象技术可用于描述不同类型的软件系统,但目前面向对象建模技术的研究很少是专注于对象之间的动态关系,也就是我们所说的“对象相互作用”。这是一个很明显的缺陷。因为面向对象程序由大量的初始化方法激活所组成,但我们讨论面向对象系统的设计时却倾向于避免描述这些动态关系的细节。为解决这一问题,我们用软件设计模式来描述面向对象软件系统的设计。

### 2 设计模式的概述

设计模式的概念最早是由名为 Christopher Alexander 的建筑师提出来的,他最早把注意力放在建筑物和城镇的设计和结构上,逐渐地他的思想影响了软件研究,并在最近流行起来。随着面向对象技术的出现和广泛使用,一方面软件的可重用性在一定程度上已经有所解决,另一方面对软件可重用性的要求也越来越高,设计面向对象的软件很难,而设计可重复使用的面向对象的软件难度更大,利用设计模式可方便地重用成功的设计和结构。设计模式是为实现高质量软件设计重用而提出的,一个设计模式系统性地命名、促成和解释一个可通用的设计,该设计着重于在面向对象系统解决一个重现的设计问题。它描述这个问题及其解决方案,当应用该解决方案时,它同时给出实现的方法和示例。该解决方案是一系列解决此问题的对象和类的组合。面向对象的设计模式是解决如何

在面向对象软件开发中完成一定的任务。

所谓设计模式(Design Patterns)简单地理解为面向对象软件的设计经验,可以作为设计模式记录下来。“一个设计模式命名、抽象和确定了一个通用软件结构的主要方面,这些设计结构能被用来构造可复用的面向对象设计”。而受到普遍认可的设计模式的定义是由 Drik Riehle 和 Heinz Zullighoven 给出的:“模式是指从某个具体模式的形式中得到的一种抽象,在特殊的非任意性的环境中,该形式不断地重复出现。”模式的概念是随设计中要解决的问题的变化而变化,每一个模式描述了一个在我们周围不断重复发生的问题,以及该问题的解决方案的核心。这样,你就能一次又一次地使用该方案而不必做重复劳动。

关于设计模式,目前的研究方向主要有:设计模式与其他面向对象设计方法(如特定领域的框架)的关系,它们各自的优劣和适应范围。除此之外,人们还在各个方面总结设计模式(如通信领域等),以及研究如何让设计模式的使用更加自动化等。

### 3 设计模式的组成

#### 3.1 设计模式的基本成分

一个设计模式被划分为四个基本组成部分:

(1) 模式名称。模式名称通常用来描述一个设计问题、它的解法和后果,由一到两个词组成。

(2) 问题。问题告诉我们什么时候要使用设计模式、解释问题及其背景。

(3) 解决方案。解决方案描述设计的基本要素, 它们的关系、各自的责任以及相互之间的合作。

(4) 后果。后果描述应用设计模式后的结果和权衡。对软件设计而言, 通常要考虑的是空间和时间的权衡, 也会涉及语言问题和实现问题。对于一个面向对象的设计而言, 可重用性很重要, 后果还包括对系统灵活性、可扩充性及可移植性的影响。明确看出这些后果有助于理解和评价设计模式。

### 3.2 设计模式的描述

如果我们要理解和讨论模式, 就必须以适当形式描述模式。好的描述有助于我们立即抓住模式的本质, 即模式关心的问题是什么, 以及提出的解决方案是什么。模式概念的创始者 Alexander 采用下面的格式来描述设计模式:

```

IF      you find yourself in CONTEXT
        For example EXAMPLES,
        With PROBLEM,
        Entailing FORCESS
THEN   for some REASONS,
        Apply DESIGN FROM AND/OR RULE
        To construct SOLUTION
        Leading to NEW CONTEXT and OTHER PATTERNS
    
```

目前最常用的格式是 Erich Gamma 等人采用的下面固定模式描述:

模式名称和分类、目的、别名、动机、应用、结构、成分、合作、后果、实现、例程代码、已知的应用、相关模式等 13 要素, 在特定的软件开发领域中, 可以使用不同的描述方法。可以忽略其中某些要素, 也可合并几个要素。

### 3.3 用设计模式描述软件体系结构风格

体系结构风格类似于前面所描述的体系结构模式。通过在体系结构风格中引入面向对象视图, 使得体系结构风格的组成部分, 如对象、连接器可视为方法激活(控制和数据的转换)。最近的主要研究将体系结构风格分为 5 类。每一类依据体系结构风格的组成要素(构件和连接器)部分、控制视图观点、数据视图和控制/数据相互作用视图等来划分:

(1) 数据流风格。数据通过系统时, 接收者是无逆向的内容控制流的软件系统, 如顺序批处理、数据流网

络、管道和过滤器等。

(2) 调用-返回风格。通常是单线程控制计算机软件系统, 如主程序/子程序、抽象数据类型、对象、基于调用的客户机/服务器及分层系统等。

(3) 独立构件风格。主要是独立的通信模式, 如并发程序、进程事件系统和通信进程。

(4) 数据集中风格。主要是复杂的集中数据存储、通过独立计算操纵数据、存储库及黑板程序等软件系统。

(5) 虚拟机风格。将一条指令翻译为其他指令, 如解释程序系统。

## 4 设计模式目录的内容

目前 Gamma 和他的同事已发布了可用于 OO 系统的 23 个设计模式。用一种类似分类目录的形式将设计模式记载下来, 我们称这种设计模式为设计模式目录。根据模式的目标(所做的事情), 可以将它们分成创建性模式(creational)、结构性模式(structural)和行为性模式(behavioral)。创建性模式处理的是对象的创建过程, 结构性模式处理的是对象/类的组合, 行为性模式处理类和对象间的交互方式和任务分布(图 1 为设计模式目录各模式间的关系示意图)。

根据它们主要的应用对象, 又可以分为主要应用于类和对象的。(表 1)

表 1 Gamma 设计模式分类表

	创建型	结构型	行为型
类	Factory Method	Adapter Class	Interpreter Template Method
对象	Abstract Factory Builder Prototype Singleton	Adapter Object Bridge Composite Decorator Facade Flyweight Proxy	Chain of Responsibility Command Iterator Mediator Memento Observer State Strategy Visitor

## 5 软件体系结构与设计模式之间的关系

软件体系结构描述的是组件及其关系, 针对特定

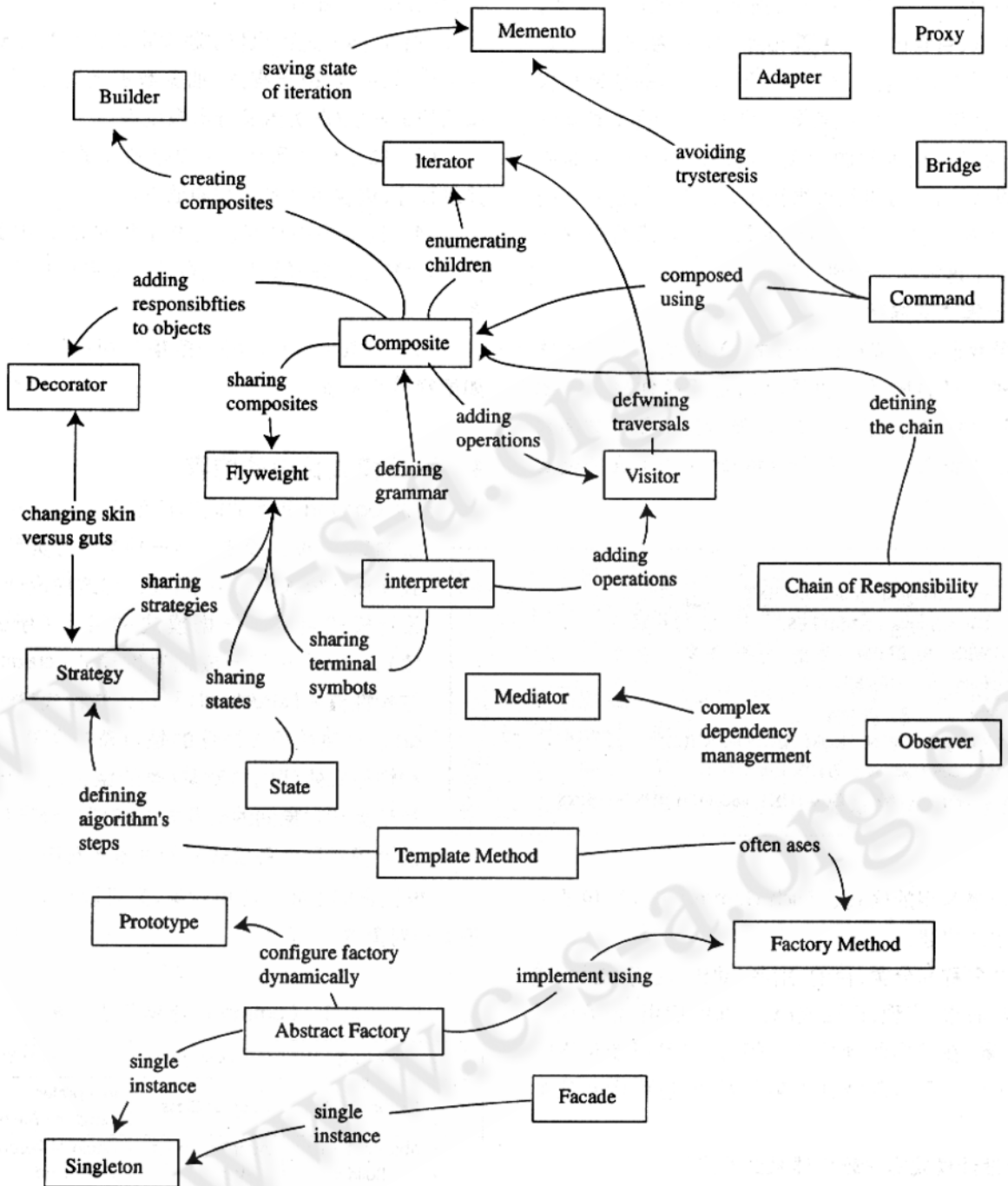


图 1 设计模式目录各模式间关系示意图

的应用领域;设计模式提供了获得高层次的可复用组件的描述方式,是复用成功设计的有效手段。如前所述,体系结构定义了软件整体的结构、组成、协作及控制参数,基于设计模式设计的软件组件可以是整个体系结构的一部分,可以帮助体系结构框架获得更高层次的设计复用和代码复用,有助于获得无须重新设计

就可适用于多种应用的框架结构体系。

## 6 存在的不足

不管采用软件体系结构还是设计模式的指导思想,在带来先进、复用设计和降低解决问题复杂度的同时,也带来了许多问题,主要表现在以下几个方面:

(1) 由于技术的不完善性,带来了许多很难预测的效果;

(2) 由于设计模式和软件体系结构的结合引起的表达上的问题;

(3) 缺乏实用工具和标准化规范的支持。几乎每种体系结构都有相应的支持工具,如 Unicon、Aesop 等体系结构支持环境,C2 的支持环境 ArchStudio,支持主动连接件的 Tracer 工具等。另外,支持体系结构分析的工具,如支持静态分析的工具、支持类型检查的工具、支持体系结构层次依赖分析的工具、支持体系结构动态特性仿真工具、体系结构性能仿真工具等。但与其他成熟的软件工程环境相比,体系结构设计的支持工具还不很成熟,难于实用化。对软件体系结构及设计模式的研究目前尚处在迅速发展中。

## 7 结论

面向对象的设计模式是解决如何在面向对象软件开发中完成一定的任务,基于软件重用可以优化软件的结构,降低软件成本,缩短产品开发周期。解决好软件的重用、质量和维护问题,是研究软件体系结构的根

本目的。以设计模式为指导,软件体系结构为核心的软件开发过程,是以面向对象的技术为基础的设计方法。本文介绍了 Gamma 的 23 个设计模式目录,在设计模式的研究中,我们要不断总结、不断回顾以前使用过的设计模式,不同的模式之间有联系,同时也有各自的优缺点,设计模式的组织和分类有待进一步优化,同时,软件体系结构也需要相应的规范和实用工具来支撑。

## 参考文献

- 1 张友生,软件体系结构,北京清华大学出版社,2004.1。
- 2 周小健、余冬梅、张聚礼,基于设计模式的软件体系结构研究。
- 3 耿刚勇、李渊明、仲萃豪,基于构件的应用软件系统的体系结构及其开发模型,计算机研究与发展,1998(7):594-598。
- 4 Perry,DE. Software engineering and software architecture. In: Feng Yu - lin, Beijing: Electronic Industry Press,2000.1-4.