2018302100026 龙晓怡 第四章作业

4.1 (1) 软件架构风格

软件架构风格又称软件架构惯用范型,是描述某一特定应用领域中系统组织方式的惯用范型，作为“可复用的组织模式和习语”，为设计人员的交流提供公共的术语空间，促进了设计复用与代码复用。

（2）软件架构模式

Dwayne E. Perry和Alexander L. Wolf从组件的角度给出了软件架构模式的定义:根据系统的结构组织定义了软件系统族，以及构成系统族的组件之间的关系。它们是通过组件应用的限制和组件的组织与设计规则来确定和表现的。它代表模式系统中的最高等级模式，它确定了一个应用的基本结构，并在后期的每个开发活动中都遵循这种结构。

（3）二者的区别和联系

它们的区别和联系主要在于:

1)架构风格主要描述应用系统的总体结构框架。架构模式可存在于各种应用系统规模和抽象层次上，包括从定义应用系统的总体结构到描述怎样用给定的编程语言实现特定问题的惯用法模式。

2)架构风格相对独立。各应用系统采用不同的风格后，与由其他风格构成的系统联系较少。而模式往往依赖于它所包含的较小的模式或者与它相互作用的模式。

3)架构模式比架构风格更加面向问题。架构风格侧重于从应用系统中抽取出它们的总体组织结构，而较少从实际设计环境来考虑设计的技术。而架构模式通常由问题出现的语境、解决方案和适用场景组成。

总而言之，软件架构风格是对软件架构整体方案的展现形式，反映的是整体方案实施之后的效果;软件架构模式仍然是软件设计模式的一种，也是一种问题-解决方案，反映的是针对某个需求问题，至今为止最佳的解决方案是什么。虽然软件架构风格也与解决方案有关，但它并没有强调一定是最佳方案，而软件架构模式具有这种潜在要求。

4.3 软件架构模式和设计模式的区别和联系分别是什么

区别：软件架构模式定义了系统的结构，并隐藏了实现细节，重点在于系统组件之间如何交互。它代表着模式系统中的最高等级模式，确定了一个应用的基本结构，并在后期的每个开发活动中都遵循这种结构。

另一方面，软件设计模式则集中在系统的实现上，通常会涉及很多细节。软件设计模式是一套被反复使用、多数人知晓的、经过分类编目的、代码设计经验的总结，使用设计模式是为了可重用代码、让代码更容易被他人理解、保证代码可靠性。设计模式一般有如下四个基本要素：模式名称、问题、目的、解决方案、效果。设计模式确定了所包含的类和实例，它们的角色、协作方式以及职责分配。每一个 设计模式都集中于一个特定的面向对象设计问题或设计要点，描述了什么时候使用它，在另一些设计约束条件下是否还能使用，以及使用的效果和如何取舍。

联系：软件设计模式和架构模式的关注点往往重叠。虽然一般的架构将设计包含在其范围内，但对于设计却不能。都注重复用性，并且使得系统的组织接哦古清晰，易于被理解。针对不同的系统需求，使用这些模式可以避免软件开发人员做一些重复性的工作，设计出高质量的软件系统。

4.5软件架构模式的作用是什么

软件架构模式确定了应用的基本结构，并在后期的每个开发活动中都遵循这种结构。极大地促进了设计的重用性和代码的重用性，并且使得软件的组织结构易被理解。

软件架构模式有助于定义应用程序的基本特征和行为。例如，某些架构模式适合于高度可扩展的应用程序，而其他架构模式适合于高度敏捷的应用程序。它们可以较好地支持系统内部的互相操作、协调统一。