

系统构架设计时应考虑的一些必要性因素[1] PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/141/2021_2022__E7_B3_BB_E7_BB_9F_E6_9E_84_E6_c29_141405.htm

本文从程序的运行时结构和源代码的组织结构两个方面探讨了系统构架设计应考虑的各种因素，列举了系统构架设计文档应考虑的一些问题。关键字：系统构架、设计、考虑、因素 本文目录：一、与构架有关的几个基本概念；二、构架设计应考虑的因素概览；三、程序的运行时结构方面的考虑；四、源代码的组织结构方面的考虑；五、写系统构架设计文档应考虑的问题 六、结语 一、与构架有关的几个基本概念：1、模块（module）：一组完成指定功能的语句，包括：输入、输出、逻辑处理功能、内部信息、运行环境（与功能对应但不是一对一关系）。2、组件（component）：系统中相当重要的、几乎是独立的、可替换部分，它在明确定义的构架环境中实现确切的功能。3、模式（pattern）：指经过验证，至少适用于一种实用环境（更多时候是好几种环境）的解决方案模板（用于结构和行为。在UML中：模式由参数化的协作来表示，但UML不直接对模式的其他方面（如使用结果列表、使用示例等，它们可由文本来表示）进行建模。存在各种范围和抽象程度的模式，例如，构架模式、分析模式、设计模式和代码模式或实施模式。模式将可以帮助我们抓住重点。构架也是存在模式的。比如，对于系统结构设计，我们使用层模式；对于分布式系统，我们使用代理模式（通过使用代理来替代实际的对象，使程序能够控制对该对象的访问）；对于交互系统，我们使用MVC（M模型(对象) / V视图(输出管理) / C控制

器(输入处理)) 模式。模式是针对特定问题的解，因此，我们也可以针对需求的特点采用相应的模式来设计构架。

4、构架模式（architectural pattern）：表示软件系统的基本结构组织方案。它提供了一组预定义的子系统、指定它们的职责，并且包括用于组织其间关系的规则和指导。

5、层（layer）：对模型中同一抽象层次上的包进行分组的一种特定方式。通过分层，从逻辑上将子系统划分成许多集合，而层间关系的形成要遵循一定的规则。通过分层，可以限制子系统间的依赖关系，使系统以更松散的方式耦合，从而更易于维护。（层是对构架的横向划分，分区是对构架的纵向划分）。

6、系统分层的几种常用方法：1）常用三层服务：用户层、业务逻辑层、数据层；2）多层结构的技术组成模型：表现层、中间层、数据层；3）网络系统常用三层结构：核心层、汇聚层和接入层；4）RUP典型分层方法：应用层、专业业务层、中间件层、系统软件层；5）基于Java的B/S模式系统结构：浏览器端、服务器端、请求接收层、请求处理层；6）某六层结构：功能层（用户界面）、模块层、组装层（软件总线）、服务层（数据处理）、数据层、核心层；

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com