

卡内基梅隆大学软件研究所关于软件架构的定义[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__E5_8D_A1_E5_86_85_E5_9F_BA_E6_c29_141464.htm 软件架构 (software architecture) 是一系列相关的抽象模式，用于指导大型软件系统各个方面的设计。软件架构是一个系统的草图。软件架构描述的对象是直接构成系统的抽象组件。各个组件之间的连接则明确和相对细致地描述组件之间的通讯。在实现阶段，这些抽象组件被细化为实际的组件，比如具体某个类或者对象。在面向对象领域中，组件之间的连接通常用接口_(计算机科学)来实现。软件体系结构是构建计算机软件实践的基础。与建筑师设定建筑项目的设计原则和目标，作为绘图员画图的基础一样，一个软件架构师或者系统架构师陈述软件构架以作为满足不同客户需求的实际系统设计方案的基础。来源：www.examda.com 软件构架是一个容易理解的概念，多数工程师（尤其是经验不多的工程师）会从直觉上来认识它，但要给出精确的定义很困难。特别是，很难明确地区分设计和构架：构架属于设计的一方面，它集中于某些具体的特征。在“软件构架简介”中，David Garlan 和 Mary Shaw 认为软件构架是有关如下问题的设计层次：“在计算的算法和数据结构之外，设计并确定系统整体结构成为了新的问题。结构问题包括总体组织结构和全局控制结构；通信、同步和数据访问的协议；设计元素的功能分配；物理分布；设计元素的组成；定标与性能；备选设计的选择。” [GS93] 但构架不仅是结构；IEEE Working Group on Architecture 把其定义为“系统在其环境中的最高层概念” [IEEE98]。构架还包括“符合

”系统完整性、经济约束条件、审美需求和样式。它并不仅注重对内部的考虑，而且还在系统的用户环境和开发环境中对系统进行整体考虑，即同时注重对外部的考虑。在 Rational Unified Process 中，软件系统的构架（在某一给定点）是指系统重要构件的组织或结构，这些重要构件通过接口与不断减小的构件与接口所组成的构件进行交互。从和目的、主题、材料和结构的联系上来说，软件架构可以和建筑物的架构相比拟。一个软件架构师需要有广泛的软件理论知识和相应的经验来事实和管理软件产品的高级设计。软件架构师定义和设计软件的模块化，模块之间的交互，用户界面风格，对外接口方法，创新的设计特性，以及高层事物的对象操作、逻辑和流程。一般而言，软件系统的架构

（Architecture）有两个要素：它是一个软件系统从整体到部分的最高层次的划分。一个系统通常是由元件组成的，而这些元件如何形成、相互之间如何发生作用，则是关于这个系统本身结构的重要信息。详细地说，就是要包括架构元件（Architecture Component）、联结器（Connector）、任务流（Task-flow）。所谓架构元素，也就是组成系统的核心“砖瓦”，而联结器则描述这些元件之间通讯的路径、通讯的机制、通讯的预期结果，任务流则描述系统如何使用这些元件和联结器完成某一项需求。建造一个系统所作出的最高层次的、以后难以更改的，商业的和技术的决定。在建造一个系统之前会有很多的重要决定需要事先作出，而一旦系统开始进行详细设计甚至建造，这些决定就很难更改甚至无法更改。显然，这样的决定必定是有关系统设计成败的最重要决定，必须经过非常慎重的研究和考察。历史早在1960年代，诸

如EW戴克斯特拉就已经涉及软件架构这个概念了。自1990年代以来，部分由于在Rational Software Corporation 和Microsoft 内部的相关活动，软件架构这个概念开始越来越流行起来。卡内基梅隆大学和加州大学埃尔文分校在这个领域作了很多研究。卡内基梅隆大学的Mary Shaw和David Garlan于1996年写了一本叫做 Software Architecture perspective on an emerging discipline的书，提出了软件架构中的很多概念，例如软件组件、连接器、风格等等。加州大学埃尔文分校的软件研究院所做的工作则主要集中于架构风格、架构描述语言以及动态架构。计算机软件的历史开始于五十年代，历史非常短暂，而相比之下建筑工程则从石器时代就开始了，人类在几千年的建筑设计实践中积累了大量的经验和教训。建筑设计基本上包含两点，一是建筑风格，二是建筑模式。独特的建筑风格和恰当选择的建筑模式，可以使一个独一无二。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