

从建筑角度来看软件体系结构 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\\_\\_E4\\_BB\\_8E\\_E5\\_BB\\_BA\\_E7\\_AD\\_91\\_E8\\_c29\\_141258.htm](https://www.100test.com/kao_ti2020/141/2021_2022__E4_BB_8E_E5_BB_BA_E7_AD_91_E8_c29_141258.htm) 在业界，软件体系结构和建筑学的设计框架可以类比。如果把软件体系结构类比做建筑学的蓝图，那构件就可以比作一砖一瓦，或者更大概念如：庭院，花园等。软件体系结构之所以可以独立于软件的数据结构和软件的算法，是因为业界把软件的数据结构和算法看做了烧砖的方法，或设计庭院的思想。但没有扩展到整个软件系统。一个软件系统是庞杂的。但是分治之后的东西仍然用到了最基本的软件算法和数据结构思想。之所以也有些人说SA可以不用，是因为算法和数据结构的扩张有代替现在的SA的可能。完美的构件是可以进行黑箱复用的。不太完美的复用构件就是白盒复用，要在原来代码的基础上做些必要的改动。软件工程的思想正是体现在软件流程、软件模块和软件文档的复用之上。否则也就没有工程的说法。但是又有人提出软件体系结构复用的思想。这应该被认为是迄今为止最大粒度的复用。如果建筑师可以把自己以前的建筑蓝图拿来稍作修改就可以进行多次施工，那么软件架构师们为什么不能拿过去的系统设计方案稍作修改并实施到新的项目中去？但是很可惜，没有一个企业的业务流程和另一个企业是完全相同的。所以软件体系结构的复用起来比建筑蓝图的复用要复杂的多。软件体系结构直接决定了软件系统的运行框架,其优劣不但决定了软件系统是否可以满足针对此软件的功能需求,而且还决定了这些功能需求是否能被合理、高效地实现,即也决定了软件系统基本的非功能属性。 [1]、每个

用户的非功能属性多多少少的有所不同，也就决定着软件体系结构的复用的复杂性。软件体系结构是描述软件单元(element)、软件单元的属性(property)以及这些单元之间关系(relationship)的结构 [2]、这里的软件单元应该就是构件。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。没有蓝图就不能构建出雄伟的大厦。没有好的体系结构也就不能构造出庞大的系统。但是对于小型的系统，体系结构的思想似乎是一种多余。对于大厦，我们应该有建筑蓝图，但是对于建造小茅屋，似乎再画草图就是一种多余。软件体系结构表示了一个软件系统的高层结构，主要特点有： 软件体系结构是一个高层次上的抽象，它并不涉及具体的系统结构（比如B/S还是C/S），也不关心具体的实现。 软件体系结构必须支持系统所要求的功能，在设计软件体系结构的时候，必须考虑系统的动态行为。 在设计软件体系结构的时候，必须考虑有现有系统的兼容性、安全性和可靠性。同时还要考虑系统以后的扩展性和伸缩性。所以有时候必须在多个不同方向的目标中进行决策。 [3]、抽象的东西之所以对我们有指导的意义。是因为我们可以运用这个抽象的东西对具体问题做具体分析。法无定法，也就是这个道理。 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](http://www.100test.com)