

软件架构简介

2017141493004 常家奇

软件架构是设计软件的蓝图.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。

我认为软件架构是宏观的,是针对整个系统的,要考虑全局.根据需求来设计合适的架构,一般好的（静态）架构可以尽量使变化发生在局部（模块内）而不影响整个系统。架构上的变化往往成本会非常高。架构的责任还有保证软件的可用性，可扩展性，可伸缩性，安全等等。如: 系统运行在什么平台？用什么语言开发？采用什么样的交互方式？用什么来存储数据？如何保证过程安全性等等。

一个优秀的架构:

(1) 高性能

系统必须满足预期的性能目标，在并发用户数（Concurrent Users）、并发事务数（Transactions per Second, TPS）、吞吐量（Throughout）等指标方面达到预估值，支撑使用人群的正常操作。

(2) 可靠性

系统直接影响到用户的经营和管理，因此必须是可靠的。

(3) 稳定性

软件系统必须是能够在用户的使用周期内长期稳定运行的。

(4) 可用性

可用性是指系统在指定时间内的提供服务能力的概率值。

(5) 安全性

用户的业务数据是具有非常高的商业价值，如果被泄露或篡改将会带来重大损失。安全性是软件系统的一个重要的指标，也是架构设计的一个重要目标。

(6) 灵活性

软件系统应该具备满足不同特点的用户群和目标市场的能力，更灵活。

(7) 易用性

软件系统必须拥有较好的用户体验，便于用户使用。

(8) 可扩展性

业务和技术都在不断的发展变化，软件系统需要随时根据变化扩展改造的能力。

(9) 可维护性

软件系统的维护包括修复现有的错误，以及将新的需求和改进添加到已有系统。

基于这些目标，经常有人说：“架构是系统非功能性需求的解决办法的集合”。