

第一周：架构方法学习总结

首先是主要重要内容的摘录

1.1 大厂架构师JD解读：这里解读了架构师需要什么技能，学完以后可以回过头来再看看，对照一些自己有了什么又缺了什么

1.2 架构师面试常见面试题解读

1. 解读了架构师面试里有哪些元素（这些东西我现在基本上不知道，到时候要回头来复习啊）
2. 什么是架构设计？——要能写出架构文档（清楚架构文档所包含的元素），给不同的人看的文档不一样
3. 什么是架构师：对系统负责的人。
4. 关于架构师的成长：要主动去思考，分析团队当前的架构状况，描述，改进等，在一个领域有了相当的深度以后，就会很容易在其他领域有所建树

1.3 4+1视图模型：软件开发的本质是什么？

1. 通过架构文档呈现相关方的利益点，关注点
2. 架构 = {元素，形式，关系/约束}；逻辑视图，过程视图，物理视图，开发视图，场景视图
3. UML：unified modeling language，用来表达架构，意图的一种方式

1.4 UML：软件建模的一般方法和工具

1. 需求分析的时候使用用例图，从宏观上给出用例的总体轮廓。用例内部可以不断细化：顶层用例图，细节用例图。这些用例图共同构成了用例文档
2. 类图：静态建模。详细设计的时候画的图，类名，方法名都定义了。为了代码落地和实现
3. 包图：有点像类图，也是静态建模，用在组件开发中，用的不多
4. 动态模型：状态图，活动图，时序图，合作图
5. UML中的消息：分同步异步，方法调用就是一种同步消息
6. 时序图：不同纬度，粒度对象的交互。类之间一般是同步消息，组建图之间才会开始使用异步消息
7. 活动图：主要是用来描述处理流程。一个业务流程跨越了多个领域就是跨越了多个泳道。
8. 状态图：订单的状态描述，电梯的状态描述等。在描述复杂状态变迁的时候是一张很重要的图
9. 合作图：可以由时序图生成，表述对象之间的交互关系
10. 组建图：物理组建，逻辑组建之间的关系。模块的依赖和交互开发。组建之间可以依赖，关联，聚合，组合等。
11. 部署图：最大的蓝图，有哪些组建，部署在哪些服务器中。
12. 一次完整的设计：最初是部署图，哪些服务器中放哪些组件，然后对于服务器内部的组件，画出其具体的组件图，组件之间的关系，再细化出类图时序图等等

13. 产品开发流程：需求设计阶段：[产品经理/架构师]用例图，时序图，活动图，状态图；概要设计阶段：[架构师]部署图，组建图，类图(高阶)，时序图，活动图等；详细设计阶段：[架构师]类图，时序图，活动图，状态图，写作图

14. 画图很重要！

1.5 架构设计文档：软件架构设计文档的写作模式。

1. 李老师讲了自己的经历，刚毕业的时候是边缘人物，凭借几个UML图得到认可，通过评审，成为架构师，每周新进来五六十的开发都有他指挥

2. 在一家日企工作，已经成功研究清楚一个件事情该怎么做了，为了和总公司说清楚，做了逆向设计，通过代码还原出UML设计图

3. UML是一种很好的通过模型表达自己，并获得别人认可的方式

4. 架构师的真正输出是要提供一个设计文档，把设计固化下来，以便于日后的维护

5. 拆解了一个架构设计文档。详细的可以参见架构设计文档模版。

感悟：

工作有差不多3年多一些了，在团队中承担的职责开始变得比以前重要，过去这段时间可以说，主要是把精力花在了技术水平的提升上面，在项目中使用了各种各样的技术，认识范围较广泛，也有了一定的架构设计能力。

但是我并不是很重视写文档，这一方面当然也是由于工作实在是太忙，另一方面是没有意识到文档的设计对理清思路的帮助，以及它在获取决策者的支持，获取周围人掌声过程中的重要作用。

可以说第一周的训练课程就多多稍稍有点颠覆了我以往的认知，从画UML开始，从此重视对系统的设计和表达，以及与各方的探讨，做好点滴的积累。