[一线架构师实践指南阅读笔记3](https://www.cnblogs.com/zlj843767688/p/13095595.html)

时间： 2020-05-11

Refined Architecture阶段是细化阶段是属于架构设计，不是详细设计，这两个不能混谈

对于细节架构和逻辑架构的差异有三个典型

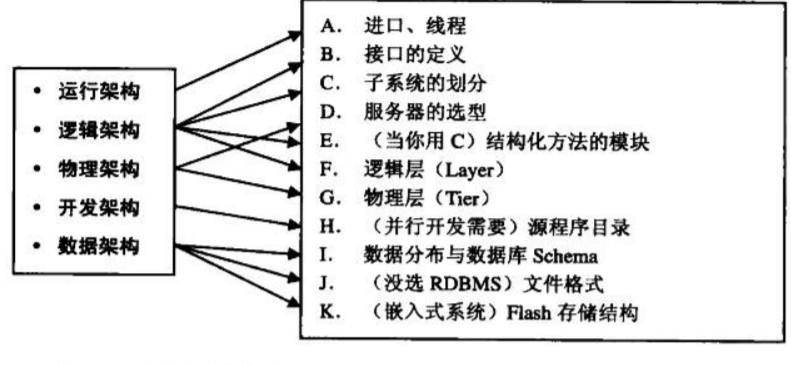
1.接口，在细节架构里，接口是核心部分，而逻辑架构并不关心接口

2.子系统，细节架构重视通过通过子系统和模块来分割整个系统，并且子系统有明确的接口，而逻辑架构只有抽象的组件

3.交互阶段，细节架构的交互阶段是‘实在的’，是基于接口方法的调用，而逻辑结构是概念化的,如A层调用了B层的服务。

架构师还应该记住 方案=项目+需求+架构 并不是方案=架构的全部。

对于架构的工作范围



多视图方法的实际意义：

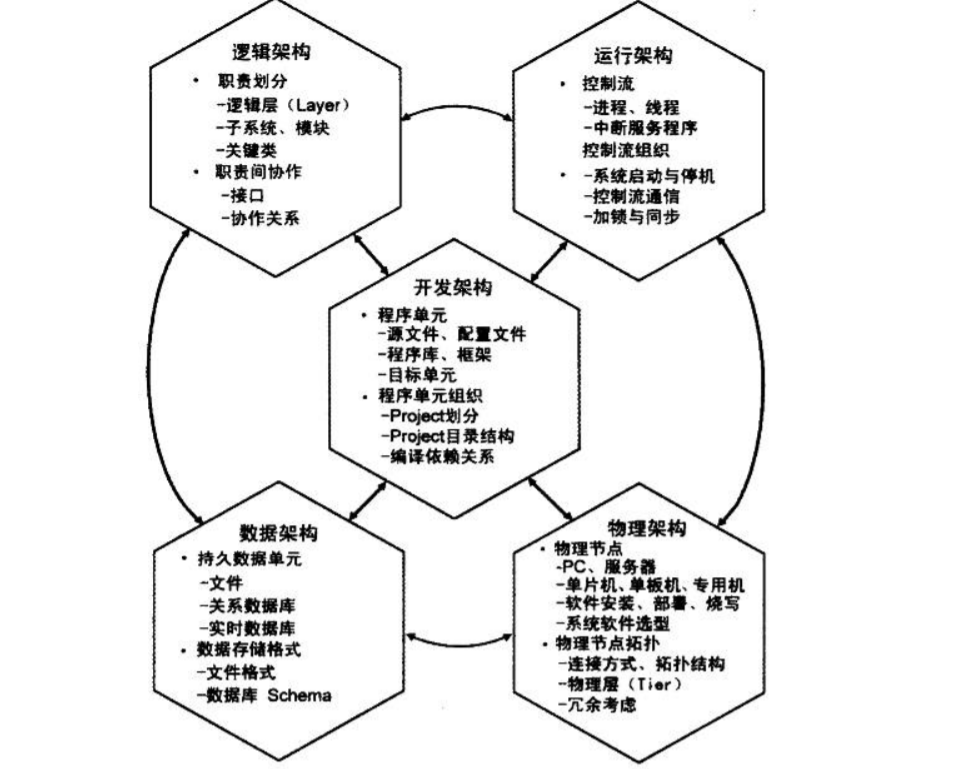
1.利于思考

2.便于交流

而对于概念架构，逻辑架构，物理架构，逻辑架构和物理架构是架构设计同一阶段要同时考虑的两个方面，是同一个阶段里的，是两个视图，并不是两个阶段。

多视图的5视图方法包含

逻辑视图，开发视图，运行视图，物理视图，数据视图

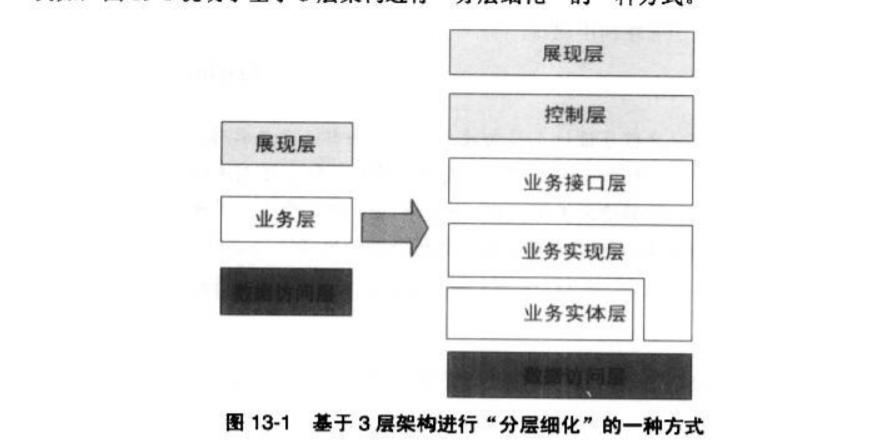
如

对于逻辑架构的划分子系统实践策略归纳为三种

1.分层的细化

2.分区的引入

3.机制的提取

**分层的细化**

 将三四层架构细化为多层的方法，分层细化是划分子系统的必用策略之一

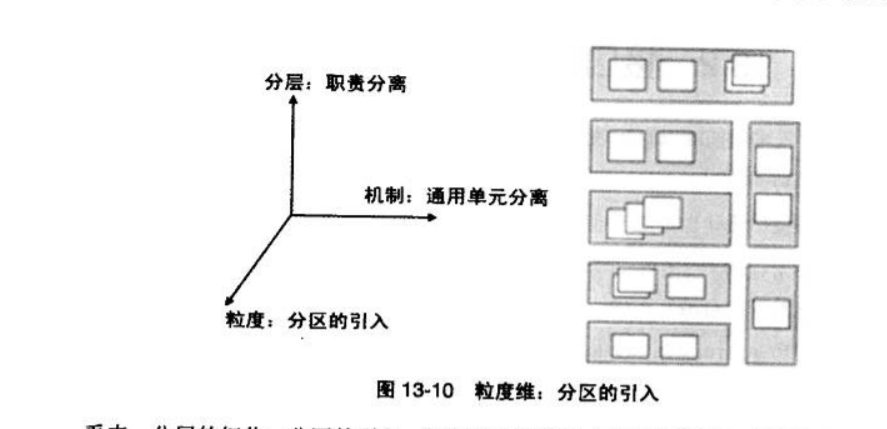
分区的引入

在架构设计中如果只有分层的话，程序员会无从下手，为了开发必须引入分区，分区是在层里进行的，单位比层要小，因此分区的引入也是划分子系统的必用策略之一。

机制的提取

基于接口或抽象类的协作是机制）（基于具体类的协作算不上机制）：  
  机制是一种特殊的子系统

所以在划分子系统必须要三管齐下，综合的运用三种手段，分层的细化，分区的引入，机制的提取



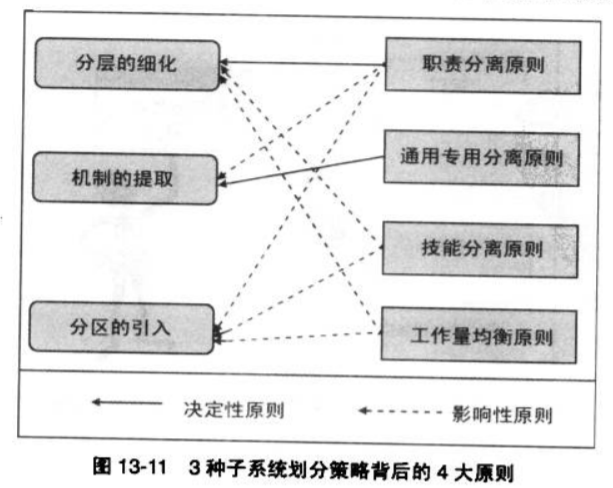
 划分子系统的原则有四种·

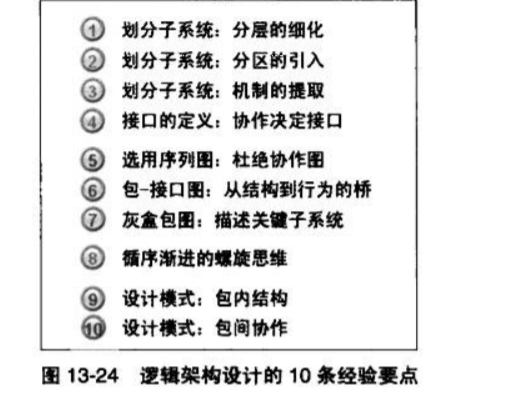
1.职责分离原则

2.通用专用分离原则

3.技能分离原则

4.工作量均衡原则





## **4.关于物理架构、运行架构、开发架构**

## 物理架构主要设计：

硬件选择与物理拓扑

软件到硬件的映射关系

方案的优化

运行架构的设计内容：（控制流图是关键）（实现控制流三种手段：进程 线程 中断服务程序）

确定引入哪些控制流；

确定每条控制流的任务；

控制流的创建、销毁、通信机制等；

控制流之间的同步关系，加锁机制等

开发架构的设计内容：

将“逻辑结构”映射为程序单元

要自主编写的源程序

可重用的库、框架

其他方式（shell脚本，平台配置文件）

开发技术选项：开发语言，平台工具

“程序单元”间关系

* Project划分
* Project目录结构
* 编译依赖关系