

浅谈系统框架与开发模式 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__E6_B5_85_E8_B0_88_E7_B3_BB_E7_c29_141270.htm 在博客中我也看到了好多关于系统框架的文章，就象有些朋友说的，这些系统框架大同小异，一般是分为数据访问层、实体层、业务逻辑层、业务外观层、表示层。然后就是这些层与层之间的调用，这些我想对于做过稍大一点的项目、或者小型以上团队开发的项目，都是会考虑到这些分层模式带来的系统扩展性优势。有些朋友都建议加一个common层，把一些共公的类与方法集中在一起，让大家一起调用，可以减少重复代码，这个我也是很支持的，也是这样做的。我相信目前摆在我们面前的，已经不是这些系统框架的问题了，而就是这些结构中体现出来的开发模式的问题。大家所说的要把公共的代码放在一起，这就是“重构”所要体现的思想，重构就是为了让代码更具的扩展性、维护性，能减少重复代码，这可以从根本上提高代码的效率并减少修改导致的BUG恶性循环。重构一般发生在什么时候？代码设计期：这要求设计人员把公用的方法总结出来，放在公用的模块中，生成文档或是通过其他方式，反正就是通知大家这些公用方法，而这是非常有限的，设计人员无法思考出所有的公用模块与方法，相反，这些只是极小的一部分，因为更多的重构应该发生在下面。代码开发时：当我们第一次写一段代码时不会注意，写第二遍时我们会想：算了再写一遍吧。准备写第三遍时，我们必须要有醒悟过来，这些相同的代码可以放在一个公用的方法中调用，对，在第三次时我想我们最好是这样做，因为你现在这样

做，不仅仅是为了现在，而是为了将来。我们就会写成公用方法，并一定要把原来的两个方法采用“调用”的方式，可千万不要偷懒，谁也无法保证今后不会修改这个函数，而如果真的修改了，你没有能力找出最先的那两段代码了，噩梦从此开始。 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