

计算机软件水平考试：开源框架思索[2] 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__E8_AE_A1_E7_AE_97_E6_9C_BA_E8_c29_141461.htm 虽然，我们一直强调整体大于单个的总和，但是如何将单个“个体”正确的组合成发挥更大效应的“整体”却并非易事。因为这些单独的框架都由不同的团队开发，框架与框架之间存在天然的阻抗，这种框架和框架之间的“代沟”需要额外配置和编码才能弥合。每个框架都拥有自己的配置文件，框架的整合经常带来配置的灾难，如将Spring和Struts整合时，不仅Struts本身的配置文件一个不能少，在Spring中还需要每个Action提供配置信息，而且两者需要遵守一定的契约。相互搭配的框架和框架之间经常会出现相似的或重复的功能，如何取舍，如何使用往往让开发者们为难。如Spring本身提供了AOP方法返回结果的缓存功能，而Hibernate本身也提供二级缓存，究竟两者都使用呢，还是择一而从？往往中间又会引出很多争论。来源：www.examda.com 框架整合的问题已经日益突出，我们可以在各开源论坛或社区发现大量有关讨论的主题。目前也出现了一些试图解决的框架整合问题的开源项目，如国外的AppFuse，国内的SpringSide，为框架的整合提供了专业的指导。但是并没有很好的解决现实开发中的实际需要。这些整合框架为了增加通用性，网都撒得太大，导致整合框架本身象一个庞然大物，让人望而生畏，定制性和灵巧性上都存在不足，降低了它们的实用性，所以这些整合性的开源项目往往降格为指导性的实例。升级的困扰 活跃的框架每天都在升级改造，丰富功能。其次由于开源框架在一定程度上存在

随意性，往往导致框架在实际使用后，发现大量隐含的Bug，所以有时对某个框架的升级变得不可避免。开源框架比之Sun正规的规范有着更加灵活的升级方式，高低版本不兼容的问题已经成为司空见惯的事情。如著名的Hibernate，其3.0版本和2.0版本的包名都发生了彻底的变化，刚发布的Acegi和低版本也存在很大的差异，无法兼容。一个整合性的框架由多个出自于不同团队的框架组成，整合框架在这些组合框架之上高位运行，底层框架的升级变化就造成了组合框架水涨船高的局面，整合框架脆弱的稳定性很容易被打破。组合框架的升级还直接带来了开发团队学习的压力，为了熟悉框架新功能和改进，在开发工作之余，他们不得不努力压榨自己的业余时间不断地充电学习。总是某个框架新功能学习还未完成，另一个框架的新版本又在一阵欢呼声中闪亮登场，让开发人员发现自己所有的努力只是一场骑牛追马游戏。

3、开发者如何走出迷局

框架的爆炸性增长和技术更替一日千里的速度，让刚刚从传统J2EE迷局中走出来的开发者重新堕入了新的困境之中。有许多切身体验的开发者在网上大倒苦水，甚至有许多声音在呐喊，希望重新回到JSP JavaBean + JDBC那个纯真的年代中去。框架的作者们本想还软件开发一个清新美满的世界，不想个体性的良性企盼变成了一种整体性的混乱纷争。在纷繁复杂的开源世界如何走出迷局和困境，把握自己技术航船的方向，是每个开发者们冥思遐想的事情。

重点学习 触类旁通

每个人的时间是有限的，对于周期紧，进度急，加班赶的开发者来说更加如此，使得开发者不可能“识遍天下字，读尽人间书”逐个学习框架。选择好适合自己、适合项目的框架进行重点学习尤为重要。不但要掌握技术细节

，更要理解框架的原理和思想，这样在接触相关框架时，我们才能触类旁通，慧眼识真。如果你深入理解了Struts框架的MVC的原理和思想，在接触Tapestry，Spring MVC等框架时，你会发现两者只是形上的区别，而非质上的差异，即使因现实需要确实要转换框架时，也可以轻松平滑地过渡。

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