

23 | 架构师：不想当架构师的程序员不是好程序员

2019-04-20 宝玉

软件工程之美

[进入课程 >](#)



讲述：宝玉

时长 14:57 大小 13.70M



你好，我是宝玉，今天我想与你讨论一下要想成为架构师，你需要具备哪些能力。

很多程序员的梦想，就是将来能成为一名架构师。包括我刚学编程那时候，也是以当架构师为目标，觉得不想当架构师的程序员不是好程序员，希望将来能成为一个优秀的架构师。就像拿破仑那句名言：“不想当将军的士兵不是好士兵。”

随着工作经历的增多，我也开始参与到架构设计中。对架构设计了解的越多，我越发觉，其实做架构设计，并不代表一定要有一个架构师的头衔。

拿破仑那句名言，原句是“Every French soldier carries a marshal's baton in his knapsack”，意思是“每个士兵背包里都应该装有元帅的权杖”。

元帅的权杖，意味着大局观，元帅的思维方式。当士兵背包里装有元帅的权杖，就意味着士兵也能胸中有大局观，能有元帅的思维，理解元帅在特定战场上想什么，这样能更好的执行命令，提升整体的战斗力。

其实拿破仑的本意是激励每一名上战场的士兵都要有大局观，有元帅的思维，并不需要每一个人都一定去当将军、当元帅。

这也适用于技术领域，对于程序员来说，并不代表一定要有一个架构师的头衔，而是心中有大局观，有架构师的思维。从而能理解架构设计，能写出好的程序。

什么是架构师思维？

通过上一篇的学习，我们知道架构设计，是要控制技术的复杂性。对于架构师来说，要控制技术复杂性，有几种有效的方式：抽象、分治、复用和迭代。

架构师思维，其实就是这几种思维的集合。

抽象思维

抽象思维可以说是整个架构设计的基础。因为对于架构设计来说，是要为了满足业务需求的，而业务需求都是一些文字性的描述、原型、UI 设计图，这些需求要最终变成代码让机器执行，就必须先进行抽象，抽象成计算机能识别的模型。

其实抽象思维我们不陌生，因为我们从小学习的数学，就有很多抽象思维的训练。举例来说，我们小时候做的鸡兔同笼问题，看起来很复杂，但是如果我们会二元一次方程，把鸡抽象成 x ，兔子抽象成 y ，就可以用二元一次方程列出相应的方程式，从而求出解。

在软件项目中，遇到类似的场景，就会考虑抽象出来，总结一个规则和方法。有时候即使场景不同，也可以把其中有共性的内容抽象出来，可以更方便的使用。

举个例子，我们在之前文章中有对极客时间专栏做用例分析，其中有四个角色：编辑、作者、未订阅用户和订阅用户。其实这四种角色，都可以抽象成“用户”模型，然后通过对用户设置不同的角色属性，来应用成不同的角色。

还有像极客时间专栏的一篇文章稿、视频课程的一节视频课，都有标题、内容、作者、留言等信息，所以可以抽象成“文章”模型，通过文章的类型、内容来区分专栏文稿还是视频课。

在架构设计中，对需求进行抽象建模后，可以帮助我们隐藏很多无关紧要的细节，我们在高层次的架构设计时，可以关注在几个主要的模型上，而不必关心模型内的细节实现。

分治思维

架构设计的一个重点，就是要对复杂系统分而治之，分解成小的、简单的部分。但光分解还是不够的，同时还需要保证分解后的部分能够通过约定好的协议集成在一起。

分治思维在架构设计中有多经典的应用。比如说上一篇介绍的分层架构，把 UI 部分与其业务逻辑部分隔离，这样这两部分就既可以各自进行变更，又互不影响。比如说 UI 交互修改，不需要修改业务逻辑代码，业务逻辑部分对性能进行优化，不需要修改 UI 界面。而每层之间，可以通过约定好的方法或者 API 进行交互。

还有像我们平时说的大数据，高并发这些复杂问题，也是通过分治来解决的。要知道单台机器，无论你性能如何优化，都是有其极限的。而像“双十一”这种高峰时刻，瞬间的流量可能是几百、几千万，就需要通过设计合理的策略，分化到不同的服务器，让每个服务器的流量不至于太大。参考：《[秒杀系统优化思路](#)》。

这种分治的思维其实不仅适用于架构上，也适用于平时程序员写代码。比如说有些程序员写代码，喜欢把大量的逻辑放在一个方法或者一个类里面，最后极其难以理解和维护，如果能分拆成几个小的方法或者小的类，不仅结构更清晰，也更容易理解和维护。

复用思维

复用是一种非常简单的提升开发效率的方法，通过对相同内容的抽象，让其能复用于不同的场景。

举例来说，我们前面提到极客时间的专栏和视频课程，可以作为两个不同的模块进行开发，但是实际上内容差不多，如果能抽象成同一个“课程”模块，这样专栏和视频课程的模块就可以复用“课程”模块，不需要维护两份相似的代码，进而提升开发和代码维护的效率。后面如果要增加每日一课和微课，也不需要重新开发，只要复用之前的“课程”模块即可。

以前我在 DePaul 读书时，要给学校做一个教学播放的软件，由于当时技术框架选的是 React，而 React 没有合适的视频播放组件，于是我只好自己实现了一个。实现完成之后，我觉得这个视频播放功能肯定有很多人也需要，如果能复用的话会很实用。于是我把它封装

后放到 GitHub 上，解决了很多人在 React 中播放视频的需求。到现在已经有超过 1000 个 Star。

复用思维在日常写程序的时候也很常用，比如有的程序员喜欢复制粘贴代码，所以经常看到很多重复的代码，如果要修改，得修改好几个地方。如果能把这些重复的代码提取成公共的类或者方法，就可以减少很多重复，让代码更简洁和易于维护。

迭代思维

好的架构设计，通常不是一步到位，而是先满足好当前业务需求，然后随着业务的变化而逐步演进。

就像淘宝这样的业务，它背后的架构设计也不是一步到位成现在这样，拆分成好多微服务。最开始，它也只是个普通的分层架构，后来随着业务不断扩展，逐步迭代成今天这样复杂的架构。

这种迭代的思维，在写程序时也很重要。因为很多程序员喜欢追求完美，期望能一步到位，然而这样带来的问题是开发成本会大量增加，导致进度延误。另一方面，如果对需求的变化预测不正确，就会有很多冗余的代码，后面难以维护。

其实，开发人员对以上提到的这些思维模式都不陌生，只是在实践的时候，总是有意无意地忽略了。

好的架构师什么样？

对于程序员来说，培养架构师思维，并不是很难的事情。然而要成为好的架构师，光有架构师思维还不够。

一个好的架构师，不仅技术要好，还要懂业务；能从整体设计架构，也能在局部实现功能。

比如说一个做互联网软件架构设计有丰富经验的架构师，要去做建筑行业软件的架构设计，短时间内一定是很难设计出好的架构，因为他需要先熟悉建筑行业软件的业务，才能设计出符合业务特点的架构。

有一种架构师叫“PPT 架构师”，也就是说擅长写 PPT，画架构图。对各种热门的名词如数家珍。但是脱离一线开发，对业务和底层基础知识知之甚少。这样的架构师设计出来的架

构，通常是不接地气的，实现起来会非常困难，成本也高。

因为作为架构师，如果不写代码，是不能体会出设计不好带来的问题，无法及时地对架构中的问题做出调整。

所以好的架构师，一定要是程序员出身，并且能坚持做一线程序员。也许他不需要写大量的业务代码，但至少要参与一部分编码工作，以及代码审查工作，以保证架构的正确执行。

好的架构师，不仅要有技术深度，还要有一定的技术广度。因为技术的选型，通常不能局限于一种技术，需要根据业务特点和团队特点灵活地选择。

好的架构师还有一个能力就是沟通能力。作为程序员，可能把自己的模块开发好就不错了，相对不需要太多的沟通工作。但是架构师就不一样，除了架构设计，还有大量沟通工作。

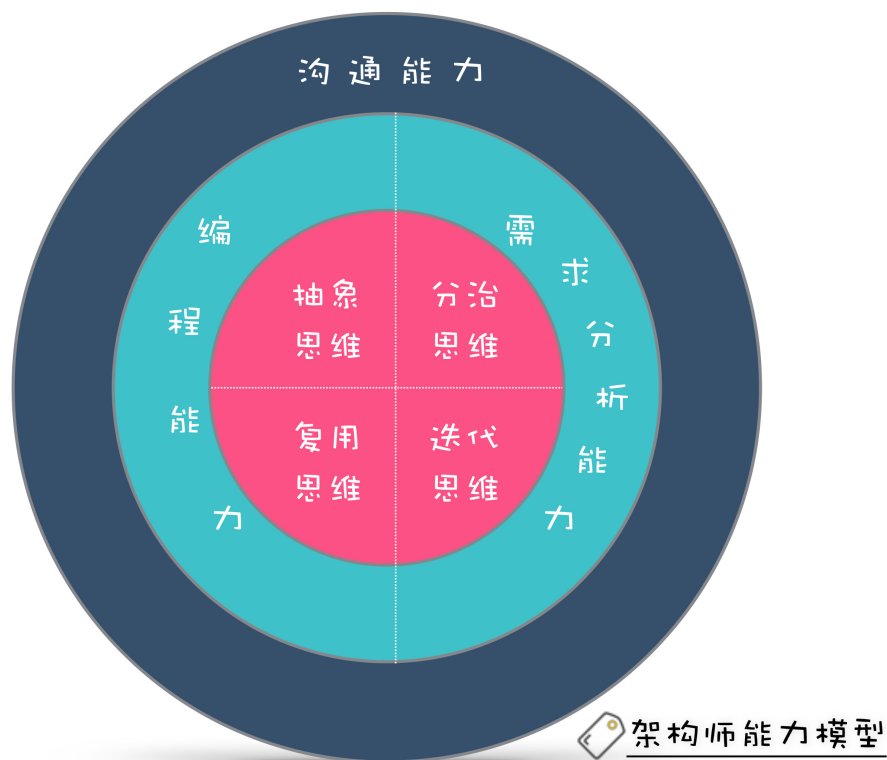
首先架构师要经常和产品经理打交道，反复确认需求，了解需求细节，只有这样才能分析清楚需求，了解各种用户场景。

然后架构师设计出来的架构，要通过文档、会议来讲给其他人听，能让其他人理解架构，用好架构。

所以要成为好的架构师，需要具备几个条件。

1. 有架构师思维：具备良好的抽象思维、分治思维、复用思维和迭代思维；
2. 懂业务需求：能很好地理解业务需求，能针对业务特点设计好的架构；
3. 有丰富的编码经验：像抽象、分治、复用这些能力，都需要大量的编码练习才能掌握；另外保持一定量的编码经验也有助于验证架构设计；
4. 良好的沟通能力：架构师需要沟通确认需求，需要让团队理解架构设计。

具备了这些条件，就可以成为很好的架构师，设计出好的架构，组织好人员和技术，低成本的满足好需求和需求变化，以及系统的运行。



如何成为好的架构师？

想要成为好的架构师，没有什么捷径，需要比普通程序员更多的努力才行。如果你有志向成为架构师的话，我的建议是：

要成为一个优秀的程序员

技术好是成为架构师的基础条件。需要让你的代码容易读，容易扩展，能重用。这样通过大量的编码实践，才能逐步地培养出好的架构师思维。

多模仿多学习

在刚开始的时候，不用想着闭门造车，想出一个特别牛的架构。反倒不如先把业界成熟的流行的架构吃透，用好。

现在网络上也有很多好的开源项目，这些开源项目都有良好的架构设计，可以找几个跟你研究方向相关的项目，本地搭建一下，然后自己试一下，最好能弄一个自己的项目二次开发或者模仿一遍，做中学，是最简单有效的。

我以前在用 Asp.Net 的时候，就基于一个开源的 Asp.Net 项目 Community Server 做了大量的二次开发工作，这对我后来做架构设计帮助非常大，因为我从里面学习和实践了很多非常好的架构设计思想。

选择好行业 and 平台

软件其实下面细分了很多行业领域，大类有像互联网应用、企业应用、游戏应用，大类下面又有细分的小类。比如说企业应用又和各行各业的业务结合在一起的，像建筑行业软件，就需要有建筑行业的专业知识。

前面我说过，架构师要同时懂业务和技术，而这些行业知识，也不是短时间内能积累起来的。所以如果想当架构师，最好能选择一个合适的行业，能在一个行业里面早点积累足够的行业知识，后面做架构设计的时候，就能更好地设计出符合业务特点的架构。

同时，这些行业领域的业务经验，和技术结合的架构经验，也会成为你个人独特的优势，不容易被替代。

还有平台也很重要，好的平台，能给你更多的实践机会。所以你看极客时间上那些开课讲架构、微服务的，无一例外都是大厂出来的，因为只有大厂，才有机会去实践这种高并发大数据的架构设计。

如果你有志成为架构师，不能光埋头写程序，也要早做打算，选择适合你自己的行业 and 平台，少走弯路。

总结

今天，我们谈了“不想当架构师的程序员不是好程序员”这个话题。其实对于程序员来说，并不代表一定要有一个架构师的头衔，而是心中有大局观，有架构师的思维。从而能理解架构设计，能写出好的程序。

架构师思维，指的是要具备良好的抽象思维、分治思维、复用思维和迭代思维。

另外没有架构师的头衔，也一样可以做架构设计，只要你有架构师的能力就可以了。而好的架构师，需要具备：

有架构师思维；

懂业务需求；

有丰富的编码经验；

良好的沟通能力。

要想成为好的架构师，没有什么捷径可以走，首先需要成为一个优秀的程序员，然后多模仿、多学习好的架构设计，最后还要早点选择好行业 and 平台，积累好行业的业务知识，借助平台获得大量的实践机会。

课后思考

互联网架构师和企业架构师有什么不同？你有没有成为架构师的梦想，有什么打算？欢迎在留言区与我分享讨论。

感谢阅读，如果你觉得这篇文章对你有一些启发，也欢迎把它分享给你的朋友。

 极客时间

软件工程之美

重新理解软件工程

宝玉

Groupon 资深工程师
微软最有价值专家



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 22 | 如何为项目做好技术选型？

下一篇 24 | 技术债务：是继续修修补补凑合着用，还是推翻重来？

精选留言 (10)

写留言



kirogiyi

2019-04-20

4

架构师是一个概念性职位，没有明确的界定，需要具备的能力和素质也是千差万别，每个开发人员心目中的架构师画像也都不一样，神秘的IT牛人，高级的保姆，无休的恶魔...

在我看来，一名优秀的架构师应该具备良好的技术思维、产品思维和项目管理思维。技术思维是基础，评估技术难度、分析技术复杂度、准确把握技术方向，这些都是架构师在...
展开

作者回复: 严重认同。虽然说并不需要每个人都去当架构师当CTO，但是把架构师当职业目标，并且按照架构师的要求去努力，对职业成长的帮助是非常大的，今后可以选择的路会很宽。



青石

2019-04-21

3

经验积累就是个过程，付出的越多，努力的越多，收获自然越多。看源码学设计是件很不错的事情，当你思考为什么这么设计的时候，同样问题未来就很有可能按此设计。

成功的道路没有捷径，前面弯弯曲曲的道路，只是让你更容易面对后面的严峻陡峭。
展开

作者回复: 感谢分享



alva_xu

2019-04-22

2

讲到架构，我想先得谈一下康威定律。康威在1967年曾说过，“Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.” 通俗地说，就是组织形式等同于系统设计。所以系统架构设计的进化，是和组织形式的变化结合的。从ITIL来说，BIA, (business IT alignment) 是IT的核心，所以充分认识组织的业务模式和运营方式，才...
展开

作者回复: 感谢你对组织架构维度的补充，组织架构和系统架构确实是相辅相成的关系，例如像微服务，一个拆分的原则就是看组织架构要不要细分，否则并不一定要分拆成微服务的架构。



Charles

2019-04-20

👍 2

初浅理解：

互联网架构师无论是b应用还是c应用，目标是希望更多用户使用，所以必须要考虑到网络、并发性能、可用性、安全性已经各种存储的横向扩展等架构问题

企业架构师更多的是如何针对一个行业深度挖掘需求并抽象，把复杂问题简单化，最短...
展开 ▾

作者回复: 我觉得你对互联网架构师和企业架构师很到位。互联网产品特点是用户多，企业产品特点业务复杂，所以架构的侧重点不一样。

架构师在架构设计后，产出首先是架构设计文档，让大家理解架构。

然后还要写架构开发的文档，比如如何基于这个架构开发功能模块，有哪些公共API可以调用，怎么样是最佳实践，要遵守哪些规范等。

再要帮助搭脚手架和基础模块或示例项目，也就是要搭建一个最基础的可运行项目，通过这个项目，大家可以直观的理解你的架构是怎么落地的，通过基础模块或者示例项目，可以知道如何基于框架开发，后面就也可以照葫芦画瓢照着实现。

还有就是在开发过程中，要答疑、解决架构中存在的问题，对架构做优化，还要做代码审查，对于不符合架构规范要指出和修正。



dancer

2019-04-23

👍 1

个人觉得架构师这个称谓在国内用过了。。。其中不乏一些PPT架构师。。

展开 ▾

作者回复: 是的，头衔不重要，还是看有没有架构师的能力水平。



小先生

2019-04-20

👍 1

抽象，分治，复用，迭代思维。是架构师必备。

展开 ▾



Dora

2019-04-20

👍 1

关于最后提到的问题，我想到一点，不知是否正确。互联网架构，要考虑互联网很快的迭代速度，所以对于扩展等特别注意。企业架构，内部IT系统相对稳定，对比互联网架构，更简单？

展开 ▾

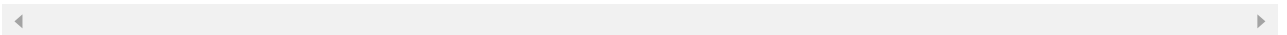
作者回复: 🐼 挺好的分析。

帮你补充几点：

互联网架构不仅迭代会快一些，用户规模通常更大，但业务也会单一一些。

企业应用通常业务比较复杂，尤其是和行业会有一些结合，但是用户规模要小很多。

这些特点，都会影响架构设计的选择。



bearlu

2019-04-20

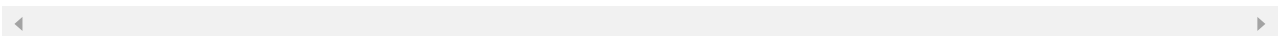
👍 1

其实最难是选择行业，请问老师有什么建议？

展开 ▾

作者回复: 抱歉这个我还真没啥好的建议，通常第一份工作会对从事的行业有很大影响。

我建议你可以请教下你身边的朋友，同时结合下你自己的资源、特长、兴趣爱好等综合选择一下。



hua168

2019-04-20

👍 1

老师，我发现市面上很少有架构类的书，有几本感觉OK，

张开涛《亿级流量网站架构核心技术》、陈康贤的《大型分布式网站架构设计与实践》、李智慧的《大型网站技术架构：核心原理与案例分析》
感觉只是讲了一个大概而已，具体深入都是靠自己。

...

展开 ▾

作者回复: 谢谢分享 📬



hua168

2019-04-20

👍 1

老师，你可以继续写个专栏叫《架构师之路》 😊

展开 ▾

作者回复: 七牛CEO许式伟已经写了一门课《许式伟的架构课》，应该不错：)

