

李智慧（[微博](#)、[LinkedIn](#)），目前任职宅米网CTO，曾供职阿里巴巴与英特尔亚太研发中心，从事大型网站、分布式系统、大数据平台方面的研发工作。著有《[大型网站技术架构 核心原理与案例分析](#)》一书。

**CSDN：**你是如何走上技术这条路的？谈谈毕业这些年来在工作中的收获和体验。

**李智慧：**我大学学工业自动化，在校的时候一个偶然的原因在图书馆看到一本C语言编程的书，从此迷上编程，但是毕业后从事的却是仪表工程师的工作。工作四年以后，难以割舍对计算机的热爱，报考了北京工业大学计算机专业的硕士研究生，然而，录取的时候却被调剂到热能工程专业，于是又稀里糊涂读了三年汽车内燃机硕士。幸运的是，硕士毕业后我找到了一份软件工程师的工作，开始了自己的职业编程生涯，而这时我已年近30，距离大学毕业也已经7年了。

所以我的工作体验就是：[如果你真的喜欢一样东西，你不会在乎为她经历多少曲折，蹉跎多少岁月。你不会盲从所谓过来人的经验之谈「30岁以后就不能编程了」，你会心甘情愿把青春献给她，无怨无悔，无畏于任何结局。](#)

**CSDN：**你目前最关注哪些技术领域？

**李智慧：**我目前比较关注的是云计算与大数据方面的技术。

- 一方面，云计算那种为整个星球提供计算能力的愿景让我痴迷；
- 另一方面，我比较喜欢研究云计算技术中用到的各种分布式技术，在阿里巴巴工作期间也申请过几项分布式计算的发明专利。

我在Intel工作期间开发过一些大数据相关的产品，也为开源大数据平台Apache Spark贡献过代码。大数据技术虽然现在火了又火，但是我认为[大数据能提供的价值还没有爆发出来，未来计算机与互联网可以为人类提供更多个性化服务和主动服务，实现某种意义上的人工智能，现在做的还远远不够。](#)

**CSDN：**你曾在13年写了《大型网站技术架构 核心原理与案例分析》一书，能分享下写书的原因？以及简单介绍下这本书的特色等。

**李智慧：**我开始写作《大型网站技术架构：核心原理与案例分析》这本书是在2012年，那一年中国互联网出了几件比较有影响的事，京东的促销宕机事件，12306的春运宕机事件，以及多个重要网站的泄密事件等等，因此我觉得有必要普及下互联网技术架构的核心原理与基本方法，让新入行的互联网工程师们少走些弯路。

因为定位在互联网技术知识普及，所以整本书的行文都比较浅显易懂，只需要一些基础的专业知识就可以顺利阅读。这本书上市以后也得到广大读者的认可，很多读者私信我说受益良多，也有些学校选用这本书做教科书，还获得过CSDN 2013最受读者欢迎原创图书奖、电子工业出版社2014最畅销图书奖。我自己很高

兴这本书能为中国互联网技术进步贡献一点力量。

**CSDN：你个人对架构/软件架构的理解是？**

**李智慧：**关于架构的定义有很多，我自己比较喜欢这个：**「架构是最高层次的规划和难以改变的决定」**。

软件架构设计应该在一个很高层面上进行，在做架构设计的时候：

- 考虑产品需求和功能特性；
- 考虑性能、可用性、伸缩性、扩展性、安全性、运维监控、集成发布等非功能特性；
- 开发团队经验技能、组织分工、进度安排等项目管理方法；
- 产品运营目标、业务组织架构，使技术架构和产品特点与运营方案更好匹配；
- 甚至要结合技术发展趋势、公司战略目标、个人和团队发展方向，为产品、个人、团队、公司未来可持续发展去设计一个坚实而又有弹性的软件架构。

**CSDN：如何最大限度避免一个架构设计的失败？**

**李智慧：**我个人感觉无所谓失败的架构，正如没有完美的架构。只要产品还能继续开发、维护、使用，再糟糕的架构也不能称为失败的架构；而一个产品如果开发中难以为继，交付后难以使用，也绝不会仅仅是因为架构设计的失败。

而要避免糟糕的架构设计，以及避免糟糕的架构设计给项目带来更多的伤害，我认为最好的办法就是**让项目组全体成员共同拥有架构；而不是架构师把架构当做个人的私有财产，拒绝他人染指**。架构师设计架构，然后项目组共同讨论修正，通过讨论让每个工程师都参与到架构设计的过程中，让每个工程师都理解架构并遵从架构设计去开发。这样即使架构设计存在问题，也会在开发过程中暴露出来，发现问题的工程师会积极和架构师讨论进一步修正完善架构。

**CSDN：大型网站有哪些典型的故障以及通常有哪些解决之道？**

**李智慧：**网站的故障千奇百怪，各种不可思议，但是大多数故障都是处理高并发请求而直接或间接导致的。很多问题在用户访问量小的时候，根本不会成为问题，比如线程锁的问题，磁盘争用的问题，数据库负载高的问题等，但是用户访问量一高，就会爆发出各种故障。高并发场景下的故障很难通过一般的功能测试发现，性能测试可以发现一部分问题，但是一定有一些问题只有在实际的高并发场景中才会出现。

淘宝的技术堪称国内翘楚，但是每年双十一为全球电商网站交易额创出新高的同时，也是淘宝技术部压力最大的时候，所有工程师那一天无不战战兢兢、提心吊胆，不到狂欢结束，谁也不知道问题会不会出在自己写的某一行代码上。我在

《大型网站技术架构：核心原理与案例分析》中专门有一章讲「大型网站典型故障案例分析」，有兴趣的同学可以参考。

**CSDN：如何学习大型网站的架构技术？**

**李智慧：**学习大型网站架构技术一个比较好的办法是去大型网站工作。但是，大型网站进入门槛很高，而且内部分工明确，只有少数人能完整了解网站整体架构，你很难成为那少数的一些人。

**学习大型网站架构技术一个更好的办法是去一家创业公司**，伴随着这家创业公司逐步成长，成为大型网站，你参与其中，自然就学习了大型网站的架构技术。可是，创业公司千千万万，哪一家才能成长为大型网站？

所以，学习大型网站架构技术最好的办法就是阅读《大型网站技术架构 核心原理与案例分析》，（笑），然后在自己的工作中实践和提高。

**CSDN：一名优秀的网站架构师必须具备哪些能力？**

**李智慧：**良好的编程能力，虽然架构设计不需要编程，虽然很多架构师日常工作也不需要编程，但是如果你的编程能力不过硬，你几乎无法设计一个良好的架构，设计出来也难以在开发过程中落实。最主要的，和一群挑剔的工程师合作，如果架构师缺乏良好的编程能力，几乎没有容身之地，遑论开展工作。

良好的沟通能力，架构师不是把架构设计出来就完事了，必须要让项目组成员认可架构设计并在开发过程中落实架构设计，必须要和项目组成员频繁沟通，在各种质疑、建议、疑惑中推进工作。同时，架构设计需要综合考虑产品、业务以及公司各方面的情况，需要和相关团队密切合作。总之需要处理大量的沟通协调工作。

此外，有一些不是必须具备，但是具备了会如虎添翼的能力

- 战略眼光，跳出架构和产品，在一个更高的层面审视架构和产品，会获得一种更开阔的胸怀和格局去看待自己的工作。
- 英文能力，紧跟世界潮流，获得技术领域最新发展动态的第一手资料，对自己的工作有更多自信和思考。
- 项目管理，虽然有项目经理去管理项目，但是架构师懂项目管理，可以更好推动项目健康发展。

**CSDN：在你的书中曾提到，「先成就他人，再成就自己」、「刚开始加入的时候不要急于证明自己，要先融入」、「最好的奖励就是目标的达成，最大的惩罚就是目标没实现」、「技术是要解决问题，但是我们要关心的是解决问题的人」、「学会妥协」、「越激烈的争辩代表越关心这个问题」，对于入门或初级的架构师你有什么心得和经验可分享？**

**李智慧：**要勇于承担责任，不要害怕失败。不要怕自己的能力不足以胜任工作，

既然公司安排了你做架构师，就是经过反复权衡慎重考虑的，你就是目前最适合这个职位的人。不要害怕自己犯错给公司带来损失，为员工犯错承担必要的损失是公司应付的成本之一。相对这些损失，锻炼骨干、磨合团队、打造敢拼敢闯的企业文化对公司的长远发展更重要。

要虚心学习，不要不懂装懂。作为新晋架构师，一定有许多不会的技能技巧，不要因为害怕被同事质疑自己的能力而不懂装懂。不懂装懂即使能糊弄过现在，你也不会有更加美好的将来。

**CSDN：**请谈一下当前架构师所面临的挑战。

**李智慧：**架构师是一个非常有挑战的工作：

- 一方面，架构师需要有足够的阅历和经验，在某些领域有多年的积淀和训练，才能对业务问题有深入的理解、对技术方案有充分的把握、对开发过程有娴熟的掌控；
- 另一方面，架构师又必须对新技术保持足够的敏锐和快速学习的能力，这样在快速发展的技术变革面前才能保持持续创新能力。

这两方面的要求对多数人而言是冲突的，很多人年轻的时候擅长学习新知识而缺乏经验，年长后擅长总结经验而对新知识反应迟钝。架构师必须要克服这种冲突，在拥有经验的同时不停更新自己的知识。Stay hungry, Stay foolish。

**CSDN：**身为CTO您从技术人员转到管理人员，有什么技巧？

**李智慧：**我做架构师的时候，发现如果开始能设计一个较好匹配业务的架构，并能和项目组成员就系统架构达成共识，整个项目开发过程并不需要投入太多精力，只要关注项目不要偏离目标即可。我相信大道应该至简，触类可以旁通。我做CTO也沿用这一思路，和技术部同学们就组织架构、技术体系、工作流程、业务重点达成一致并推行实践，而不是企图将个人意志加诸整个团队身上。

**CSDN：**您现在每天还编程吗？

**李智慧：**我在做CTO之前，在离开Intel前的最后三个月大概写了一万行代码。但是做CTO之后没有为公司写过一行代码，我会review代码，review架构设计，但是我不写代码，我觉得每个人都应该也有责任把自己的工作做好，CTO的工作职责不是写代码，CTO写代码是一种越俎代庖。

但是CTO应该编程以保持技术敏感性，他可以写一些小项目做一些技术验证，可以参与某个开源项目的开发。甚至也可以为公司写一些代码，但是最好不要将这些代码提交到公司的代码库。

**CSDN：**您作为技术人员，可否分享下您的学习知识或技能的方法？

**李智慧：**软件开发领域是一个变化特别快的领域，新的技术新的工具层出不穷。如果工程师不能快速学习新知识就会被淘汰，如果技术领导者不能快速学习新知

识甚至会成为团队进步的障碍。

学习知识技能首先需要**时间**，工程师应该用工作时间的20%去学习，如果工作时间抽不出空学习，可以用加班时间学习，如果加班时间也被安排满满的，至少能忙完这一阵（自己熟悉了情况后）可以有机会安排时间学习。如果这种机会和希望都没有的，如果你对自己负责的话，我建议你还是辞职吧。

学习知识技能其次需要**方法**，工程师应该建立一套自己的知识体系，然后不断把新知识新技能纳入这套体系中。这套体系成熟以后，读一本自己领域内的技术书，听一次自己领域内的技术讲座，需要学习的新知识不会超过20%，这样读书听讲座就会很轻松，也很容易就能吸收这些新知识，进入良性循环。至于如何建立自己的知识体系，每个人都有不同方法，但是有一点是共同的，那就是学校里的基础专业知识一定要掌握好。