

15 | 模块导读：互联网时代架构师都面临哪些新挑战？

2022-02-22 郭东白

《郭东白的架构课》

课程介绍 >



讲述：郭东白

时长 23:00 大小 21.08M



你好，我是郭东白。上个模块我们讲了架构师的六条生存法则，提到了架构师的重要工作就是组织架构活动和制定架构方案。

那么具体来说，架构活动的完整过程是什么呢？架构师一般会面临什么样的挑战呢？又需要着重关注哪些节点呢？在这个模块里，我们就来回答一下这些问题。



这节课是整个模块的导读，我就先来介绍模块的整体背景，然后再来介绍具体内容。

架构师在互联网时代面临的新挑战



架构活动，简单来说就是围绕一个架构目标而采取的行动。一个架构活动可能有上百人甚至上千人的参与。在这么大规模的人员协同中，架构师就需要找准自己在其中的定位：明

确什么事情是自己应该做的，什么事情是其他参与者应该做的。

需要说明的是，针对这些，我会提出非常具体的行动建议。上个模块讲的基本法则，相对来说，能适应的时间跨度和技术体系跨度会更长一些。但在这个模块里，这些行动建议有着非常强的时效性，主要针对现在主流的计算方式——以分布式技术为主的互联网研发活动。

每个时代的计算方式相去甚远。在企业软件时代，软件的分发靠光碟等传统介质，平均发布的间隔是每半年一次，甚至更长。数据迁移也非常复杂。自然，企业用户对软件的质量要求就非常高。

2005 年我在甲骨文数据库内核部门工作的时候，代码发布前的测试覆盖率底线是 95%，一个大版本的研发周期是一年半。另外，产品文档在研发开始的时候就基本成型了，甚至已经跟大客户完成了多轮的调研和交互稿反馈。

而到了互联网时代，多数企业是边打边学。别说产品文档了，连商业模式都很难保证能持续一年。团队每天有上千次发布，有的模块代码的测试覆盖率还不到 25%。

时代的变化导致了研发方式的变化。在互联网时代，敏捷开发成了开发模式的主流。不过敏捷的行为方式在帮助业务高速迭代的同时，也导致了大量短期行为，也就是所谓的技术债。这些技术债，就构成了架构活动所面临的主要技术挑战。具体而言，有如下五种。

第一，**反射式研发行为**。不论是初创公司还是大企业里的初创团队，往往持续面临着巨大的交付压力，导致一种被我称之为“反射式研发”的行为：写代码就像膝跳反射一样。

研发人员的日常工作都是在接需求、写代码、上线、修故障之间循环，很少有时间去思考和追求长远的设计。这种短期行为虽然不影响业务迭代，但如果在一个大型架构活动中出现频繁，造成的后果将是灾难性的。

第二，**大规模活动**。分布式研发模式往往采用微服务架构。微服务架构有一个优势，就是服务之间可以独立做变更，在保持 API 稳定的情况下，团队之间很少需要协同。所以每个服务的维护者可以独立决定自己的发布流程和交付节奏。

但是在互联网时代，为了最大化流量的传播，我们往往会通过一个大型商业活动来聚合和放大营销效果，比如 618 大促、双十一大促、春节红包、新品线上发布会等等。那么微服务相对独立自主的研发模式，对于制定跨团队统一流程和交付节奏就构成了一个巨大的挑战。

第三，**分布式研发中心**。互联网企业往往有多个分布在全球各地的研发中心。举个例子，我在 Lazada 做 CTO 时，我们团队分布在新加坡、印度的班加罗尔、越南的胡志明市，以及中国的北京、深圳与杭州。如果再加上需要协作的团队，分布就更广了。

在这种跨国、跨地域、跨语言的研发模式之下，每个研发中心内部可能有非常多的交流，但是各个研发中心之间的沟通，相比之下就是互相隔离的。时间长了，康威定律就开始发挥作用：“系统的结构和产生这些设计的组织的沟通结构是同构的”。也就是说，在我们不能改变一个组织的沟通结构的时候，架构活动产生的设计就会有很大的局限性。

第四，**普遍存在认知差异**：每个团队，甚至每个研发，平时都在自己的隔离环境下高强度地开发代码，所以大家一般没有统一的语言和全局的认知。这种情况在业务和产品团队也同样存在。因此，达成一个宏观的、完整的、可行的、被普遍认可的规划就变得非常困难。

第五，大型架构活动本身面临的挑战：**在高风险和高回报预期的场景下，必须保障项目完成的高确定性和对目标的高保真性。**

互联网时代大多数公司都是在一个场景下通吃天下，所以一个大的架构活动背后的商业赌注一般会非常大。阿里巴巴的双十一就是典型的例子。

在双十一的第一分钟，每秒就有几十万次的交易。而第一个小时的交易额接近 1000 亿，全天的交易额共有几千亿。如果双十一这天出了什么故障，不但影响投资者的信心，有些小企业也会因为备货太多销售不出去而在一夜之间破产。而且这个架构活动的交付日期根本没有变更空间，双十一之前必须完成。所以双十一那天必须保障业务和技术目标的高度保真。

如此苛刻的要求也意味着巨大的成本。阿里相关技术团队为了准备双十一，大都从六月份就开始备战，接着是持续高强度的加班，直到双十一结束。有的方案甚至要在前一年的双十二做预演，第二年 618 时再做大规模尝试，最后到了第二年的双十一才全面铺开。

虽然很少有架构活动能像双十一这样兼具高风险和高商业价值，但是几乎每个互联网公司的架构活动都面临着超高的风险、强度和复杂度。原因也很简单。如果一个架构活动不具备这样的性质，也没有一个超高预期的回报，那它就不会被提到一个足够高的位置上，公司也不愿意为这个架构活动配置专职的架构师，更不会大范围协调团队的优先级、投入多个团队的研发资源。取而代之的将会是敏捷的开发，用最小的组织、最短的时间成本、最少的代码迅速上线，在线上看效果。

在这些挑战之下，架构师要发挥的作用其实就非常明显了。我们需要帮助团队抵抗反射式的研发行为、独立决策的研发模式、分散的研发团队、普遍存在的沟通障碍和认知差异，以及高风险、高工作强度和高复杂度的场景，最终保障架构活动以高确定性完成目标。

架构师能在架构活动中起到哪些作用？

综合上述挑战，我们可以总结出架构师在互联网时代的架构活动中所需要发挥的作用了。

第一，建设共识。架构师需要克服现有的分散的研发团队、普遍存在的认知差异、习惯于独立决策的研发团队所带来的认知局限，引导参与者在架构活动的认知上达成共识。这里的共识包括对目标的共识、对决策方法的共识，对各自责任边界的共识、对资源分配的共识，以及对交付时间、内容和质量的共识等等。

第二，控制风险。架构师需要克服互联网企业在高度不确定性的商业环境、日常工作缺乏流程、团队成员高强度工作节奏和日常反射式研发所带来的混乱与质量问题，需要在架构活动的全生命周期持续收集、发现、评估、控制和传递风险。在持续变化的目标和竞争环境中不断提升对风险的理解，逐步完善预案。并在成本可控的情况下选择合理冒险，同时随时跟踪由外部环境和事件引起的风险变化，及时传递重大风险，最终把风险控制在可接受的范围内。

第三，保障交付。架构师需要尽量降低大型架构活动的不确定性和复杂度，最小化架构方案。要做到这点，架构师不仅需要提前行动，关注用户价值；还要不断提纯目标，保障交付的价值；以及，需要反复分解架构计划，持续去除盲区，提升 API 的鲁棒性。最后，以分领域、分阶段、最小化的交付来保障项目完成。

第四，沉淀知识。任何一个大型架构活动的背后，都有着巨大的时间成本、机会成本、商业资源和人力资源的投入。从这个视角看，架构师在架构活动中必须具备区别于其他参与者的宏观视角。

一方面，需要完整记录架构活动的历史与决策逻辑；另一方面，需要通过文档来驱动理性思维，通过正式文档、Review 流程来提升企业的宏观思考与决策质量。最终，保证这些内容将会形成复盘的主要输入，为企业之后的类似活动积累经验。

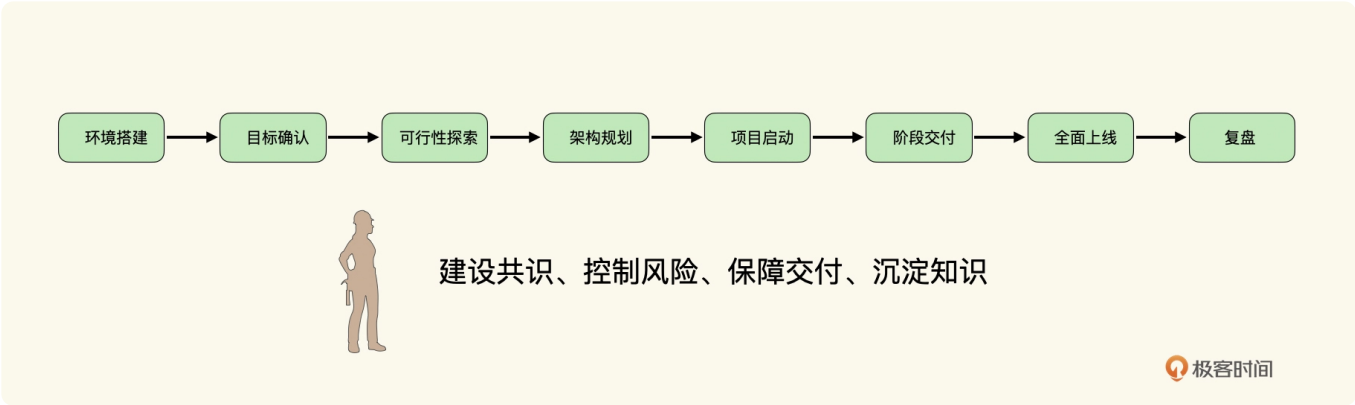
架构师所起的这四个作用，建设共识、控制风险、保障交付和沉淀知识，其实在前互联网时代也一样重要。只不过在互联网时代，有着不同的特性、具体的挑战和应对措施。我将在 16、17 讲做详细拆解。

需要特别强调的是，这四个作用中，并没有包括写代码、项目管理、培训交流等工作。事实上，这些工作往往占据了架构师的大部分时间。不过在我看来，这些工作都是为了服务于这四个作用。如果只写代码、只做项目管理，或者只做培训交流，在架构活动中都有专业人员可以轻松替代。但是通过这四个作用，架构师却可以创造出不可替代的价值。

总的来说，这四个作用是从价值创造的维度来思考架构师的工作。除此之外，我们还需要从架构活动全生命周期的维度，去挖掘架构师在每个生命周期节点的具体工作。

架构师应该关注哪些关键节点？

如下图所示，展示了架构师视角下的架构活动全生命周期。在整个周期中，一共有八个需要关注的节点。在 18-28 讲，我将会通过案例来描述每个节点要达到的目标，并提出具体的行动建议。那么这节课，我就先来大致描述下每个节点的背景与内涵。



节点一：搭建架构环境

架构师在架构活动的第一步就是为它搭建一个架构环境。遗憾的是，很多架构师由于没有认识到这个环节的内在价值，所以常常忽略了这一步。

打个比方，架构环境是架构活动在企业真实物理环境中运行的虚拟机。我们的研发是在企业这个大的物理环境下进行的。这个物理环境包括决策环境、商业环境、软件研发环境、文化环境和有限的商业和研发资源等等。在模块一里，我们有系统的介绍。这个环境是全集团共享的一个物理机，而你需要给架构活动搭建一个它自己的虚拟机，来模拟这些完整的环境。而你这个虚拟机宕机了，架构活动也就一块儿宕掉了。

其中决策环境是更为重要的一环，也将是我在这个模块要重点介绍的内容，看看怎么在一个不那么友善的环境中搭建一个相对安全的架构环境。

节点二：确认目标

我们在模块一就提到过：**架构师需要保障整个架构活动有且仅有一个正确的目标**。在这个节点上，架构师不仅需要确认这个目标的赞助人，还需要确定目标是正确、合理、可达的，也就是一个定义清晰、满足 SMART 原则的目标。

在这个过程中，切忌简单相信，而是要通过事实、数据和严密逻辑来保障决策者、赞助者和执行者等干系各方的利益，为企业找到一个正确、合理、可达的目标。在这个过程中，我们也要试图从中发现一些具有长期价值的行动点，从而吸引那些坚持长期主义的同事也投入到我们的架构活动中来。

节点三：可行性探索

接下来是可行性探索。架构师必须要确定最终的目标可达。

很多团队会将这一步放置在架构规划之后，我们会在这节课程里详细解释为什么不应该这样做。反过来，在这个节点，我们要最大程度地发现那些有可能叫停整个项目或迫使整个项目目标发生改变的那些重大风险。

那么如何在最短的时间内识别尽可能多的风险？如何从全局视角锁定重大风险呢？这就是我在这部分内容要着重阐释的问题。这个环节的目标是，向所有合作方提前传递重大风险，并准备合适的预案，从而得到决策者的支持，以继续全力推进整个项目。

当然，如果风险不可控，我们也可以选择叫停，避免重大损失。

节点四：架构规划

有了架构环境、确定目标和可行性探索的基础，我们就可以为架构活动做宏观规划了。这里有四个关键动作，分别是统一语义、执行域映射、任务边界划分和规划确认。而这些动作的目的，就是进一步提升架构活动的高确定性和对目标的高保真性。

架构规划之统一语义。统一语义的过程是一个循环往复的识别不同角色、不同场景、不同语境，然后逐渐建模、整合、修正语义的过程。直到所有参与者的需求能够被准确无损地表达、记录和传递，最终通过架构活动实现出来。

架构规划之需求确认和执行域映射。在统一语义的前提下，我们就可以确认架构活动的核心需求了。在架构规划之初，我们要梳理架构活动的用户角色，发掘每个角色的核心诉求，并从活动目标出发确认需求的正确性与合理性。同时，还要在统一语境下建立问题域模型，与执行者一起推导出从问题域到执行域的映射。在这个过程中，许多领域映射的冲突可能会被暴露，这个时候，我们必须及时将冲突上升，然后尽快解决，尽量避免将冲突带到任务边界划分中去。

架构规划之任务边界划分。任务边界划分是真正体现架构师思考力的地方，也是很多棘手问题集中爆发的地方。在这个环节，我们需要依照搭建架构环境的方法，先为团队建立一组任务边界划分的决策信条。接着，再进行用户需求和任务分解，尤其是对任务颗粒度的判断。最后，我们要确保任务相关的有限资源被提前锁定。

架构规划之任务规划确认。架构规划的最后一个环节是规划确认。在这个过程，我们需要把输入转成一个不但可行而且要有约束力的规划。除了要最小化项目交付风险外，还要确保所有参与者有能力、有意愿履行各自的责任，从而提升交付的确定性。

节点五：正式启动

很多人误以为启动会只是一个仪式性的工作，走个过场而已。这是对“启动”的一大误解。

启动，代表着合同正式签约生效。那么在正式签约之前，我们还有机会将之前未暴露的重大风险公之于众。这也是最后一个暂停或延后架构活动启动时间的机会。

所以在这个环节，我们需要与执行团队完成一次深度握手，为接下来的实施环节提供问题预警和冲突解决的机制，确保执行过程中可能发生的问题能够被及时解决，团队间的冲突

能够被及时化解。有了这些准备，我们才能以高质量的技术内容和充分的信心来开启项目。

节点六：阶段性价值交付

活动启动后，就进入了阶段性交付这一环。过去我们经常见到的都是项目按照时间段，被划分成多个里程碑。这种划分方式虽然可以降低整体目标交付的不确定性，却不能保障架构活动最终交付预期的增量价值。

我将提出一种基于最小增量价值单元的交付策略，以保障架构活动的最终产出。这样做的目的是把问题尽早提给市场，让市场给我们指点迷津。此外，我们还可以根据线上效果的评估和分析，来调整阶段性目标，甚至是整个架构活动的目标。

节点七：全面上线

最后就是全面验收后的庆祝环节了。如果前面这些工作都做到位，毋庸置疑，这将是轻松的一个环节了。值得一说的是，在这个隆重的、最具仪式感的环节，我们可以把高光时刻留给项目经理和各个产品、研发和运营等团队的一线人员。我们架构师反倒要站在幕后。

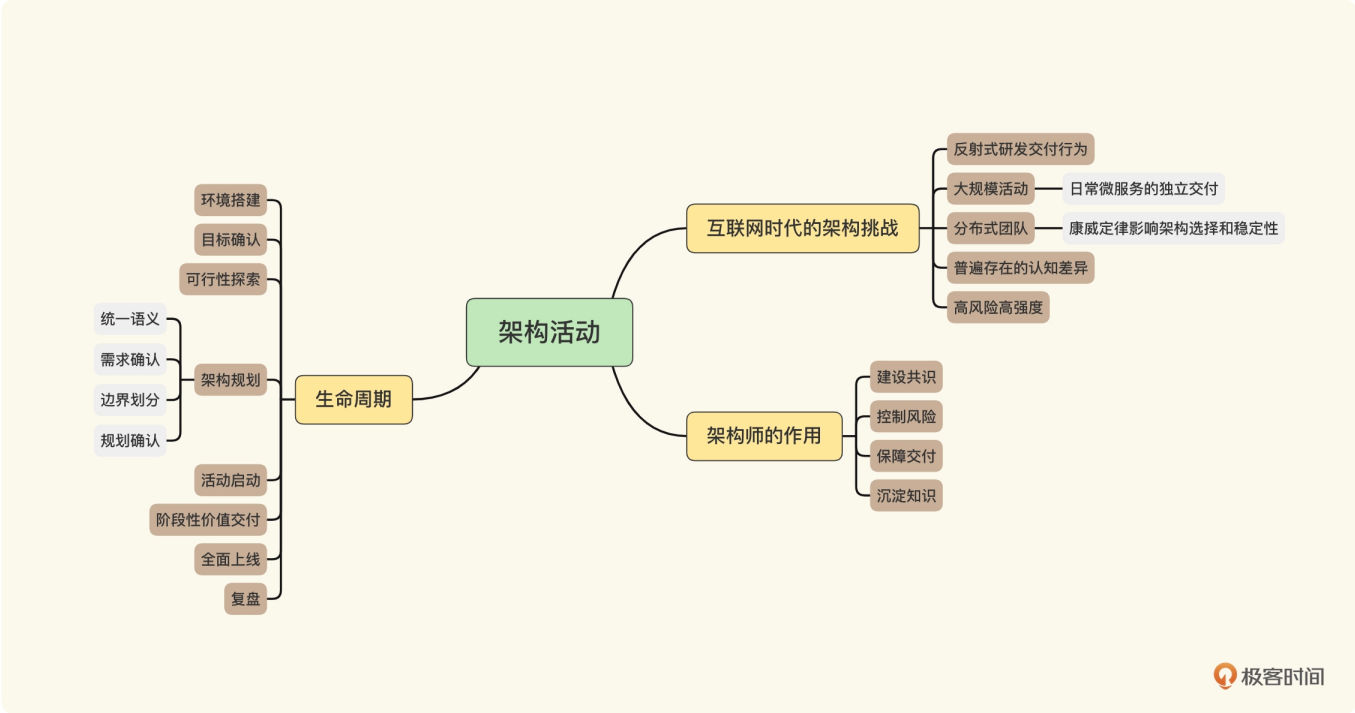
这个环节没有其他要赘述的了，课程里也不会再单独讲解。

节点八：活动复盘和机会发现

复盘是被最频繁提起但却很少被认真执行的话题。不过在我看来，这是一个架构师成长的关键。

对于架构活动而言，复盘的真正目的是为企业未来的架构活动提升成功概率。那么在复盘工作中，架构师需要平衡好不同的审查视角和分析维度。通过搭建安全的复盘氛围、充足的前期准备、对复盘过程针对性的控制、对机会点的梳理与把控，以及讨论过程中的深度剖析，从而发现深藏在流程和假设中的问题，并找到有效的行动点，确保之后的架构活动能够吸取之前失败的教训。

好了，这就是互联网时代一个大型架构活动的全过程，以及架构师在每个节点上应该关注的事情。思维导图如下：



两个小的番外

第一个番外，是关于这个模块的学习建议。

你可能会问，这些步骤似乎有点多、有点繁琐啊。在敏捷开发时代，难道我不应该小步快跑吗？的确，在实际的架构活动中，我并没有按照步骤一个一个执行。

不过在初学时期，我会想办法把完整的流程多跑几遍，将每个节点及其底层逻辑烂熟于心。然后再根据具体项目、工作环境和参与团队来做精简。不要连基本的招数都没学会，一上来就想着无招胜有招。

在我们团队做规划时，我总会给团队 Leader 们一套固定的架构规划模版，帮助他们提升架构能力。一旦我看到某个人理解得很透彻，做得很到位。我反倒劝他丢掉模版。这就是：先固化，再内化，最后才优化。

第二个番外，我想在开始模块内容之前解释一下案例的前提与背景。

模块里的案例都是以电商为背景的。主要是我们日常生活中会频繁接触电商，选择这样的案例，我不需要介绍那么多背景知识了。

案例的大背景是这样的：一个大企业有很多国际化的电商业务，各自业务形态不同，不过在细分的领域内有相似的部分。我们作为架构师，需要在三个月内交付支持多个部门的国

际化电商平台项目，让各个国家和地区的业务得到快速启动、升级或增长。

我在四家电商企业做过几十个超百人参与的大型架构项目，模块中的案例均来源于我的经历，但却和任何一个具体经历无关。我的目标是构造一个假想的案例，以覆盖更多的细分场景。否则案例覆盖的细分场景有限，就会陷入关键细节的复盘当中，不利于观点的阐释。

用计算机语言讲，我们使用的不是单个案例（Instance），而是一个类（Class）。有时候，这种在类层面和案例层面的讨论会引起一些理解上的困难。那么我建议你把这些案例，具象化作一个类下面的、有成有败的多个具体场景，这样做会容易理解得多。

如果这节课对你有帮助，欢迎你把课程转发给你的同事或朋友。顺便打个小广告，我刚开了个人抖音号，我会定期发表一些比较新、但是不一定那么成熟的观点。欢迎在抖音上搜索“郭东白”并关注，也欢迎你的批评指正。我们下节课再见！

分享给需要的人，Ta购买本课程，你将得 20 元

 生成海报并分享

 赞 4  提建议

© 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 [直播加餐02 | 阿里速卖通是如何在竞争变化中做技术战略的？](#)

更多课程推荐

李智慧 · 高并发架构实战课

“附身”大厂架构师，玩转高并发系统设计

李智慧

同程艺龙交通首席架构师
前 Intel & 阿里架构师
《大型网站技术架构》作者



新版升级：点击「 请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。

精选留言 (5)

[写留言](#)

Jxin

2022-02-22

架构活动中如何跳出招式的套路，做到无招胜有招。有什么评判的基本原则？

《生活的艺术家》中，李小龙是这么说的，招式所提供的答案，正是问题的症结所在，他为我们自然成长设置了很多限制和障碍，妨碍了我们真正理解知识。他也是主张放下招式的。而在他的只言片语中，可以看到他有一条决策依据，就是效率。他说，越接近功...
展开 ∨

作者回复：我自己是信奉达尔文主义的，就是生存第一。



3

骰子、

2022-02-22

"不过在初学时期，我会想办法把完整的流程多跑几遍，将每个节点及其底层逻辑烂熟于心。然后再根据具体项目、工作环境和参与团队来做精简。不要连基本的招数都没学会，一上来就想着无招胜有招"

我理解这和我们小时候咿呀学语一样，最开始总是模仿父母或者其他人的发音，慢慢的...

展开 ∨



👍 2



SZS

2022-02-23

听了这一课感觉架构师和项目经理的职责有重复，PMI要求在项目启动阶段和收尾阶段要做的事也包括这些，看来不会做PM的程序员做不好架构师

作者回复: 有重复的地方，但是我们这个课程会避免将项目经理的职责



👍 1



Geek_fd0943

2022-02-23

干货满满。要是再上多点表单，感觉就是麦肯锡团队的工作风



👍



术子米德

2022-02-22

☺️☺️☺️☺️☺️

* 📅：一个架构活动，可能上百甚至上千人参与，架构师要明确，什么事情自己应该做，什么事情其他参与者来做

* 😱：这个参与数量，有点惊到我，老师能否举个例子，让这个恐怖的数字，有个可想象的实际情况。...

展开 ∨

作者回复: “* 📅：一个架构活动，可能上百甚至上千人参与，架构师要明确，什么事情自己应该做，什么事情其他参与者来做”

亚马逊的本地系统--》全球化系统。我在加餐里讲的战略案例，阿里各种XX中台、大促等等。



👍

共 2 条评论 >