

16 | 高效上云：如何用云计算来提高效能？

2019-09-27 葛俊

研发效率破局之道

[进入课程 >](#)



讲述：葛俊

时长 14:08 大小 12.95M



你好，我是葛俊。今天，我们来聊一聊，如何使用云计算来提高研发效能。

自从 AWS 的出现，云的崛起已经成为软件开发不可阻挡的趋势。它逐渐像水和电一样，成为软件开发的一项基础设施。不容置疑的是，云极大地提升了软件研发的效能。

以我之前在 Stand 公司开发社交 App 的工作为例，项目刚开始时只有 3 个研发人员（包括两个后端和一个前端开发者）。我们使用了 AWS 的云服务，三个月就上线了第 1 个手机版本，而且是可以弹性伸缩稳定支撑百万月活的版本。在云出现之前，这是难以想象的。

所以说，怎样高效地使用云，包括公有云、私有云和混合云，对每一个团队来说都是一个绕不过去的话题。

云计算的话题很大，但我们今天重点要说的是，落实到研发效能上我们应该怎么做。我将其归纳为以下两点：

第一，深入理解在云上进行研发以及运行服务的特点，充分利用它的优势，比如服务化、自助化和弹性伸缩；

第二，了解并妥善处理它带来的挑战，比如分布式系统带来的安全和控制方面的问题。

接下来，我们分别看看这两个方面吧。

理解并利用云计算的优势

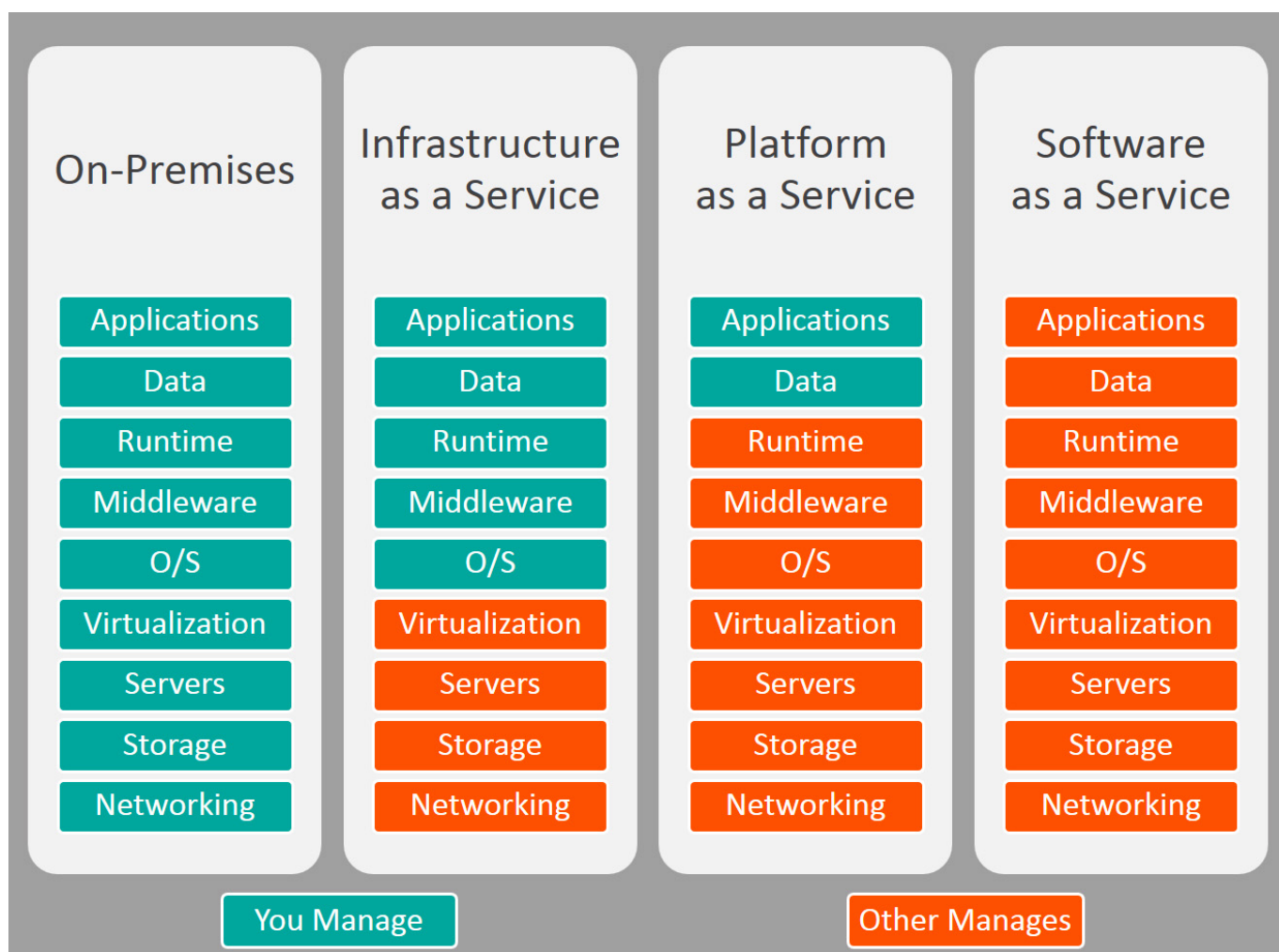
我们先简单看看云计算的定义：云计算把许多计算资源整合起来，使用软件实现自动化管理，通过网络快速提供给用户使用。其中，计算资源，包括服务器、存储、数据库、网络、软件、分析服务等。也就是说，云计算通过自动化和自助化，使得计算能力成为一种商品，在互联网上流通，就像水、电、煤气一样，可以方便地取用，且价格较为低廉。

介绍完云计算的定义，我们再来看看它的特点。云计算的特点包括，大规模、分布式、虚拟化、按需服务、高可用、可扩展等。其对研发效能的提升，我觉得主要可以归结为服务化、自助化和弹性。下面，我们分别来看看这三个方面吧。

服务化

在服务化方面，云服务按照抽象程度分为 3 类，包括：基础设施即服务 ([IaaS](#))、平台即服务 ([PaaS](#)) 和软件即服务 ([SaaS](#))。它们也被称为云计算栈，最下层是 IaaS，最上层是 SaaS，越上层的服务越抽象，也就是把下层细节封装的越多，由云平台处理的越多。当然，如果不使用云的话，这些所有的服务和资源都需要我们自己管理。

如下图所示，分别列举了这三种服务方式中，哪些服务和资源由云服务商提供，哪些由我们自己处理。



备注：图片引自<https://www.bmc.com/blogs/saas-vs-paas-vs-iaas-whats-the-difference-and-how-to-choose/>

可以看出，细节抽象得越多，云服务商负责的部分就越多，我们就越能够聚焦自己的业务，从而提高研发效能。所以，在业务开发中，我们应该尽量使用抽象的服务，也就是说尽量使用 SaaS，不行就使用 PaaS，还不行再使用 IaaS。

再以 Stand 的 App 开发为例。因为那款社交 App 与金融相关，同时客户增长速度还不确定，所以我们对数据库的要求包括高可用和可扩展两个方面。如果我们自己来维护数据库的话，需要招一个专门的 DBA，同时还需要一定的时间去搭建。但，如果使用 PaaS 服务提供的 RDS（关系型数据库）的话，高可用和可扩展都可以由 AWS 承担，我们可以全身心地投入到业务开发中，快速上线产品。

当然，正是因为 RDS 这个 PaaS 服务给用户提供了非常大的价值，所以它的收费也很高。在我们使用的所有 AWS 服务中，数据库花费大概占到了 70%。我推荐你在权衡使用 IaaS、PaaS 或者是 SaaS 产品的时候，综合考虑其提供的价值以及价格。一个常见的模式

是，初创公司在业务刚起步时，使用 SaaS 或者 PaaS 快速开发业务；业务成长到一定规模之后，再逐步转到 IaaS 以及私有云降低成本。

自助化

云服务的第 2 个特点，也是它的重要优势，是自助化。也就是说，用户在使用云上的资源和服务时，尽量减少人工参与，主要采用自动化的方式，比如通过工具或者 API 调用来完成工作。这样一来，既降低了成本，也提高了使用的灵活性和方便性。比如，使用云计算我们可以用一个 API call 去获得一个环境，再用另一个 API call 就可以部署一个版本。这相比传统的、用手工填表的方式，效率不知道要高出几个数量级。

总结来讲，自助化在软件开发中的价值主要表现在开发环境的获取和 CI/CD 流程两个方面。

开发环境的获取，包括开发机器环境、联调环境的获取，你可以再回顾下[第 11 篇文章](#)中的相关内容。这是自助化最典型的价值。

方便实现 CI/CD 流程是自助化提供的第二大优势。比如，在持续发布方法中有一个很重要的评判标准，就是看你的团队能够在多长时间内把一行代码的改动部署到生产环境上去。现在做得比较好的互联网公司，大都能够做到分钟级。这一点之所以重要，是因为只有能够快速部署，才能快速修复错误，从而才敢于快速试错，实现快速创新，在竞争中获得优势。

这其中的工作很多，包括拉分支、合并代码、构建、测试验证、部署、监控，以及可能的回滚。要实现分钟级的部署，我们必须要把服务化和自助化用到极致。而云服务提供的各种自助化，就大大降低了 CI/CD 流程自助化的实现难度。

弹性和共享

云服务的一个基本理念是，把大量的资源集成起来，然后共享给多个客户使用。正是因为多个客户共享资源，所以可以实现单个客户的按需使用。也就是说，客户 A 不使用资源的时候，可以释放出来供其他用户使用，A 也不必对这部分资源付费。同时，A 在需要大量资源的时候，资源池里通常可以有空闲的资源供它使用。这也就是我们所说的弹性。云服务的弹性，是可以用来提高研发效能的第三大特性。

接下来，我与你分享下具体**如何利用云服务的弹性**。比如，业务量的大小不确定，对很多公司的技术和基础设施都是一个巨大的挑战。如果我们使用自己的机房，不可能非常灵活地添

加、删除资源，所以需要投入大量精力进行容量规划，以防止后期业务超过系统承载量而导致宕机。但这样的容量规划，在业务还不够大的时候必然会造成资源浪费。而使用云计算，我们就可以很好地解决这个问题。

如果使用的是公有云，那么你使用的服务本身就可以自动扩容，实现弹性伸缩，所以一定要在可以使用弹性伸缩的地方充分利用它。比如，我在 Stand 公司的时候，我们的网站后端服务器、消息队列中间件、数据库、压测环境等，都使用了弹性伸缩，而且是尽量使用自动弹性伸缩的功能。

比如，后端服务器可以根据 CPU 的负载量来调节服务节点数量，达到一定阈值之后，就自动产生新的服务器并添加到集群中；当负载量降低到某一个阈值时，又会自动释放一些机器。从而，既实现了在业务量突增的时候，能够保证业务的正常运行，又能最大限度地降低成本。

如果使用的是私有云，你的服务和其他公司就不能共享。所以，从共享的角度看，不如使用公有云的好处大。不过在这种情况下，公司内部的各种不同服务、不同团队之间的资源共享，也可以给你带来收益。

比如，你可以让开发环境、测试环境、类生产环境、生产环境、内部工具系统环境都使用同一套 Kubernetes 集群，在其中使用不同的 namespace 进行隔离。这样就可以实现这几个环境的资源共享。又比如，你可以在公司内部署一套 OpenStack 私有云环境，让公司的各个产品线都在这个私有云上工作，从而实现团队之间、产品之间的资源共享，享受弹性和共享的好处。

处理云带来的分布式计算的挑战

云计算在给我们带来巨大方便的同时，也因为它的新特性以及开发模式带来了相当大的挑战。在我看来，**云计算带来的最大挑战在于**，为了使用云的弹性伸缩能力，我们的软件架构必须是分布式的，支持水平扩展。

而这种分布式的架构和传统的单体架构区别很大，如果处理不好，会给我们带来很大麻烦，最典型的例子当属微服务。微服务架构非常适合云计算，能够充分利用它的弹性伸缩能力。但如果使用不当、管理不好的话，就会出现调用混乱、依赖不清晰、难以维护的问题。

分布式计算有很多挑战，从我的经验看来，做好以下两点特别重要：

1. 自治和集中管理相结合。要做好分布式计算，我们首先要让解耦的产品团队能够独立进行产品的设计、开发、测试和上线，这样才能真正利用解耦带来的灵活性。但同时，我们必须要有集中式的管理，这样才能把控全局。

这其中最重要的是信息可视化，比如系统整体的质量看板，能让大家一眼看到整个系统中各部分的运行状况；又比如，针对调用链复杂难以调测的情况，建设微服务调用链追踪系统，收集每一个客户端请求从发出到被响应经历了哪些组件、哪些微服务、请求总时长、每个组件所花时长等信息，以帮助我们定位性能瓶颈，进行性能调优。

2. 错误处理。分布式系统，因为有很多组件同时运行，很多组件都有可能会出错。所以，我们必须要对这些错误进行处理，保证局部错误不会对全局带来非常大的影响。具体来说，可以使用的办法有：

信息可视化。通过数据可视化、监控、预警，迅速发现错误以便及时处理。

错误隔离。比如，微服务的一大好处就是，把错误限制在一个小的服务中。

提高系统容错性。只进行系统拆分还不能满足系统对容错性的要求，我们还需要确保一个服务的问题不会影响其他服务，形成所谓的“雪崩效应”。

自动修复能力。也就是说，出现问题能够自动修复，可以大大提高系统的可用性。一个最简单的办法就是，重启服务。

小结

在今天这篇文章中，我首先与你介绍了云计算对提升研发效能的作用，第一是服务化，第二是自助化，第三是弹性伸缩。

在产品设计和日常工作中，我们应该注意考虑如何利用云计算的特性，比如快速获取环境、自助化部署、在更高的抽象级别使用资源服务，来提高生产效率。

然后，我与你介绍了云计算带来的挑战以及一些应对方案。正是因为云计算带来了分布式计算，所以它的安全和控制是最大的挑战。具体的解决方法有，自治和集中管理相结合、信息可视化、错误隔离、提高系统容错性、自动修复等。

在我看来，云计算的确可以提高开发体验以及产品上线的效能。比如，在 Facebook 的时候，我们使用自研的容器及管理系统 Tupperware，极大地方便了各个环境的获取以及系统的使用效率；又比如，我们的测试、构建、分支管理都做到了很好的服务化、自助化，为产品的快速发布和高质量奠定了基础。

同时，我们也在处理分布式计算方面，投入了很多精力。举一个典型的例子，为了提高系统各个服务的容错能力，Facebook 会在某一时刻对某一个数据中心的所有服务器断电进行测试，并事前会通知每一个服务的所有者，确保自己的服务在这样的极端情况下仍能保证业务健康运行。

而我在创业公司的时候，更是充分利用了公有云提供的各种服务来快速上线产品，实现了传统基础设施环境下不可想象的高效能。虽然云计算现在还没有像水、电、煤气那样普遍，但我相信那一天不会太遥远。

思考题

在你当前的工作中，你觉得最能够使用云计算来提高研发效能的地方是什么？

感谢你的收听，欢迎你在评论区给我留言分享你的观点，也欢迎你把这篇文章分享给更多的朋友一起阅读。我们下期再见！

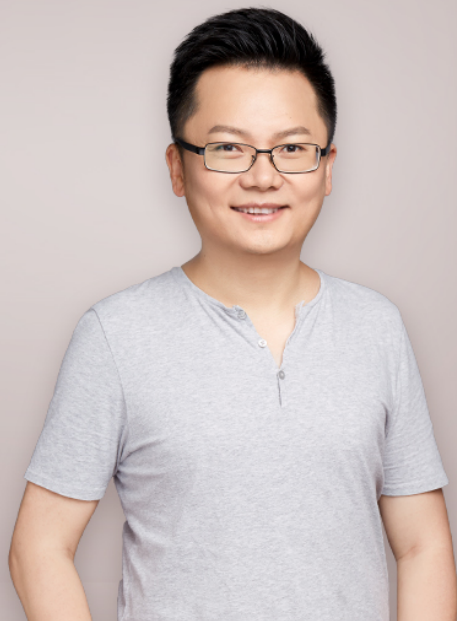


研发效率破局之道

Facebook 研发效率工作法

葛俊

前 Facebook 内部工具团队 Tech Lead



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。

© 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 15 | 开源：从Phabricator的开源历程看开源利弊

下一篇 17 | 测试左移：测试如何应对新的开发模式？

精选留言 (3)

写留言



日拱一卒

2019-09-28

云计算带给我们团队的好处有：

1. 基础设施的自动维护，降低不同环境因为底层配置不同带来的各种问题。
2. 服务的开箱即用，包括各种中间件、数据库等，可以让开发人员更集中在业务端开发，减少等待时间。
3. 各种资源的自动伸缩，降低了运维团队的压力。...

展开 ▾



3



李双

2019-09-27

讲的挺大的，能否从一些自身开发经验上，提高研发效能！例如，如何快速准确定位需求，转化技术方案，编写高质量代码！

展开 ▾

作者回复: 多谢你的反馈和建议！

第一二部分更多的是从整体上讨论团队效能，所以可能会感觉比较“大”。在第三个部分“个人效能”会比较多的讲述这一部分。



1



刘丹

2019-09-27

还可以使用云厂商提供的代码托管、CI、CD服务以及测试

作者回复: 是的。这些Hosted Service可以发挥作用，他们实际上是SaaS，云厂商承担了绝大部分职责，我们拿来用就行了：)

