

CircleCI每月处理450万次构建背后的技术栈解析

原创：Nick 滴滴云 今天

本文是【他山之石】系列文章的第1篇，主要介绍国内外热门应用（网站）的技术栈及其考量原因，旨在为大家做技术选型提供一些借鉴和参考。

背景

CircleCI是一个持续集成和交付的 (CI/CD) 平台，每月处理450万次+的构建系统。成千上万的工程师在上面测试并部署代码，这样他们就可以专注于构建优秀的软件。这些能力依赖于其稳健的技术栈，本文为大家展示CircleCI的技术栈。

作为CircleCI的CTO，我帮助团队做重大的技术决策，保持团队的凝聚力，让团队摆脱困境。在之前，我曾是Copious的CTO。

团队

工程师们往往在小的团队中工作效率更高，所以我们将技术人员划分为几个功能单元团队，灵感来自于Spotify的pods（技术组织架构）。但我们的团队规模要更小，同时保持着核心原则，即每个团队拥有在整个堆栈中实现功能的所需资源。

灵活性是CircleCI的关键价值：我们的大多数工程师在多个时区远程工作。为了让每个人都在同一层面交流，使用Zoom进行视频会议、屏幕共享、以及更新Pingboard中的状态，这种远程的方式更像是建立了一个虚拟的“办公室”。

我们使用JIRA在各个团队的流程中创建一致性。如果工程师需要或想要切换团队，这种一致性能够保证更加灵活。使用GitHub进行版本控制，使用Slack进行Giphy控制。除了聊天，还使用基于Slack的集成工具，如Hubot、PagerDuty和Looker，可以集中访问许多日常任务。

技术堆栈

语言

CircleCI的大部分都是用Clojure编写。从一开始就这样。虽然早期也用到了Rails，但是对开发的狂热让我们最终选择了Clojure；当CircleCI上线时，它完全是用Clojure编写的，从那时起Clojure就一直是平台的核心开发语言。

CircleCI的前端曾用CoffeeScript开发，但当

Om (ClojureScript interface to Facebook's React) 单页的ClojureScript应用程序可行时，虽然我们更喜欢使用Clojure，但还是选择了一致性和统一性。当工程师想在不同堆栈之间切换时，通用开发语言也有助于减少开销。

这意味着，我们会在有保障的前提下尝试其他（语言）工具。最近推出的 2.0版的构建代理是用Go编写的。我们还使用Go for CLI工具，其中静态依赖项编译和快速启动能力，比单纯对Clojure的热爱更重要。

在技术架构中创建微服务时，Clojure仍是首选武器。我们已经有了10多个微服务，这个数字还在快速增长。能达到这种速度的一个重要原因是使用了Clojure，它确保开发人员可以在团队和项目间快速切换，而不会面临越来越大的学习挑战。

前端

CircleCI Web应用程序的UI是用ClojureScript编写的。具体来说，使用了框架Om，这是Facebook React的ClojureScript接口。目前正在升级到Om Next，Om重启可以修复很多问题。

后端

两组机器池

有两组主要的机器池：第一组用于托管自己的服务 - 为站点提供系统服务、作业管理、通知发送等。这些服务部署在Kubernetes编排的Docker容器中。

使用Kubernetes在Docker中进行滚动部署效率很高，它的生态系统和工具链显然成为了我们流程中的一个选择：毕竟任务类型的变动以及内部所需的堆栈数量还是相对较少的。

另一方面，客户的需求也在频繁变化中。这就要求做到动态预测，包括预测将来运行的任务类型及每个任务所需要的资源数量，这很有挑战性。Nomad在这方面表现出色，凭借着快速、灵活的内置集群调度系统，Nomad在第二组机器池中为客户分配任务，这是专门为调度目的保留的。

虽然认真评估了Kubernetes和Nomad做的所有这些事情，但这两种工具单方面看都没能针对以上问题实现全程优化。我们将Nomad的集群调度角色视为软件堆栈的一部分，而不是管理或操作层的一部分。因此，使用Kubernetes来管理Nomad。

最近也开始使用Helm，它能更容易地将新服务分发、部署到Kubernetes中。在建立一些小型服务的时候，将CD流程与Helm结合在一起，同时保持Kubernetes的锁定，结果非常好。为每个服务创建一个图表，这使得可以轻松回滚新软件，并提供安装或升级内容的审计跟踪。

基础设施

过去五年，在AWS上运行基础架构。选择AWS是因为CircleCI的架构相对简单，但也有复杂业务所必需的链接帐户、VPC、安全组、并用到了AWS提供的用于分区和限制的其他所有资源。我们在多个地区开展业务，对AWS的深度投入使得从代码层面如何管理软件的思考越来越多。

当提出CircleCI Enterprise（我们的本地产品）的时候，已开始支持多种不同的部署模型。通过将代码打包在Docker容器中并使用和云无关的Kubernetes来管理资源和分发，从而进一步将业务与系统分离。

由于没有被供应商轻易锁定，可以灵活地将部分工作负载推送到适合于我们的Google Cloud Platform（GCP）。之所以选择GCP，是因为它特别适合短期虚拟机。今天，如果您使用我们的机器执行器来运行作业，它将在GCP中运行，此执行器为需要它的任务分配完整VM。

与前端通信

当前端需要与后端通信时，通过专用的API实现。这些API主机也由Kubernetes管理，尽管为了隔离而将其放在一个单独的集群中，但几乎所有的API都是公开的，这意味着我们使用的是与客户相同的接口。内部对这些API做足测试的意义不容小觑，它使我们能够保证API的准确性并在用户找到错误之前发现它。

如果您正在与我们的Web应用程序进行交互，那么所有的请求都会访问API主机。大多数身份验证都是通过GitHub或Bitbucket的OAuth处理的。经过身份验证后，还可以生成API令牌，以便以编程方式访问在UI系统中公开的所有内容。

数据！数据！数据！

主要数据存储是用MongoDB，这在CircleCI早期就决定了。通过简单的“无模式”数据存储和快速迭代来吸引用户。在MMAP中，已有了超过10TB的不断增长的数据存储，虽然架构方面有了很大改进，但仍然面临着因历史遗留问题所导致的数据集太大而无法有效治理的情况。

所以，我们正在退回到PostgreSQL架构。当使用自己的数据存储来构建微服务时，便有了很好的迁移机会。我们还使用Redis来缓存不会永久存储的数据，并且会对合作伙伴的API（如GitHub）请求进行速率限制。

我们处理的大量不可变数据（日志、工件和测试结果），将它们存储在Amazon S3中。

构建Build

当来自GitHub/Bitbucket的webhook告诉系统，用户推送了一些新代码时，我们把这些信息在数据存储中创建一个新的构建或工作流，然后将其排队等待处理。为了从第一个队列中升级，需要在其

计划中有足够的容量来运行构建或工作流。

如果是使用容器的客户，则在足够的容器释放之前不会有新的构建或工作流可运行。当发生这种情况时，我们把将要执行的工作定义传递给Nomad，由Nomad来按工期分配硬件。

运行Build

大多数构建都在Docker容器或容器集中运行，这些容器由客户定义为完全定制的构建环境。构建代理将其工作结果通过gRPC流式传输到输出处理器。

为了将实时流数据传输到浏览器，我们使用Pusher管理的WebSockets，还使用此通道向浏览器提供状态更改通知。

Hubot Postscript

在CoffeeScript Hubot应用程序中只添加了很少的东西 – 仅仅让它与Hubot工作人员对话。Hubot工作人员实施运营管理功能并将其暴露给Hubot，以便可以免费进行聊天集成。此外，还定制了Hubot的身份验证和授权代码，以满足团队中的角色需求。

对于更大任务，有一个用Go编写的内部CLI，它与Hubot使用相同的API，可以访问Slack中的相同功能、添加脚本、管道和所有喜欢的Unix工具。Hubot工作程序识别出CLI正在使用时，它会将命令记录到Slack以保持对操作更改的可见性。

分析和监控

监控和警报的主要来源是Datadog。为每个场景提供了预构建的仪表板，并与PagerDuty集成来管理警报的路由。

使用Rollbar来捕获未处理的异常，如果预见到异常会继续发生，会快速转换指标并指回Datadog，以尽可能保持Rollbar的清洁。

使用Segment来整合所有跟踪器，其中最重要的是用于分析用户模式的Amplitude。但是，如果需要更加统一的视图，会将所有数据推送到运行Postgres的数据仓库中，可以通过Looker进行快速分析和仪表板创建。

许多工程师想要打造自己的分析使用工具，用起来更加习惯，也会包括sed和awk，以及Pandas和R。

基于滴滴出行的业务技术和经验积累，滴滴云为开发者提供简单快捷、高效稳定、高性价比、安全可靠的IT基础设施云服务。关于滴滴云的产品和优惠信息，以及更丰富的云计算行业动态、产品资讯、

活动报名、技术干货及案例分析，请关注「滴滴云」微信号！



相关文章：

DC2 云服务器预装工具介绍（二）网络分析篇

关于滴滴云负载均衡 开发者需要知道的九件事

DC2 云服务器预装工具介绍（一）性能监控篇

滴滴云 LSTM 优化之路

滴滴云 GPU 服务器深度学习实践篇—— Tensorflow Object Detect 对象检测

滴滴云GPU机型性能评测（一）：GPU计算

滴滴云GPU机型性能评测（二）：CPU计算

滴滴云GPU机型性能评测（三）：内存带宽

更安全！滴滴云专有网络 VPC 浅析

滴滴云：同等性能下，比其他云厂商价格低40%

一起玩转滴滴云盘（EBS）