

微服务时代下崛起的 TestOps 工程师

前言

微信中有些上次参加会议的很多朋友，希望整理下关于PPT的演讲内容，然后发表一篇文章：**关于微服务下TestOps的工作和未来**。然后我也正有这样的想法，但是可能单方面的自我总结比较片面，希望帮助很多还在迷茫的探索的QA工程师寻找未来之路。

文章的大纲：

1. 微服务和DevOps
2. DevOps孵化下的TestOps
3. TestOps在未来

9月24日源创会微服务专场重庆站

GitChat

TestOps很新鲜(内心认为你们是这样想的)，也可以说是衍生的新型职位，**维基百科**甚至没有收录关于**TestOps**的词条。谷歌（Google）上的关于**TestOps**只有寥寥无几的文章，国内的**TestOps**更是一片空白，很多人停在理论上，没机会去实践。因为机缘我在一家新型互联网创业公司，公司没有运维的同时，又是在微服务的架构下，我们又在做敏捷...所以有幸改变了之前的工作方式和内容。

开始前，有个段子。刚进入公司的时候内心还在想：xx(自动幻想屏蔽词)，现在测试要求会coding就算了，怎么还要会运维，而且不是简单的linux命令，搭建服务器、管理服务

微服务时代下的 TestOps 工程师

这次文章的主题是关于新型的工程师TestOps。

容易消失的 测试工程师

为什么会出现测试容易消失掉？因为测试攻城狮的职业定位缘故，很多人看来测试的工作轻松，技术含量没有那么多高，所以导致存在感降低。

如何做一个有存在感的测试工程师？

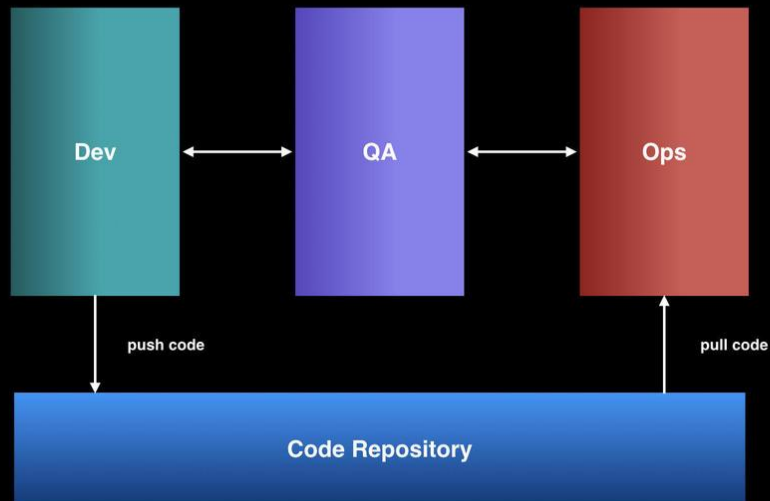
3

如何做一个有存在感的设计师工程师呢，咱们开始前规避这个话题，文章最后来回答这个问题。

GitChat

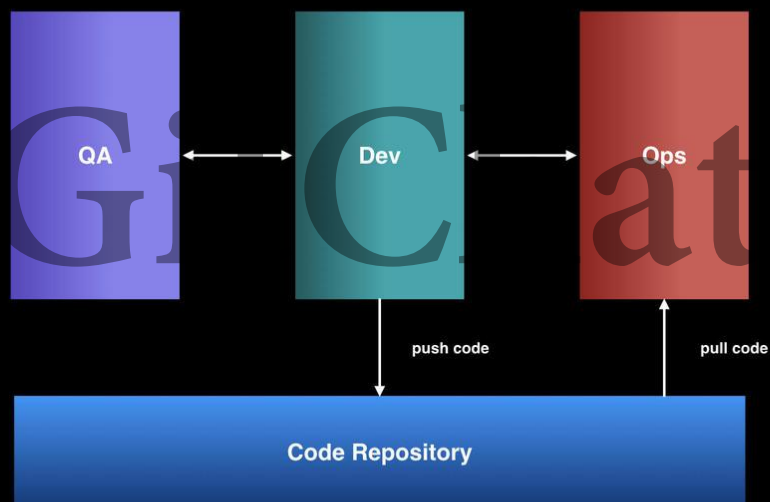
1 - 微服务 和 DevOps

传统行业



5

传统行业



5

咱们先看看传统的几个 workflow，询问过一些朋友网友和同事（第一种比较主流，第二种

场景优劣

	优点	缺点
测试为中心点	1.测试：多领域技能的提升 2.开发：专注于代码	1.测试：大量沟通成本 2.开发：专注于代码
开发为中心点	1.测试：专注于质量 2.开发：多领域视角	1.开发：沟通成本增加 2.测试：局限于质量

6

第一种情况：测试会花费大量的时间在沟通，但是有多领域技能的提升。开发人员专注力提升。缺点是：测试人员无法专注于质量掌控，开发局限于coding。

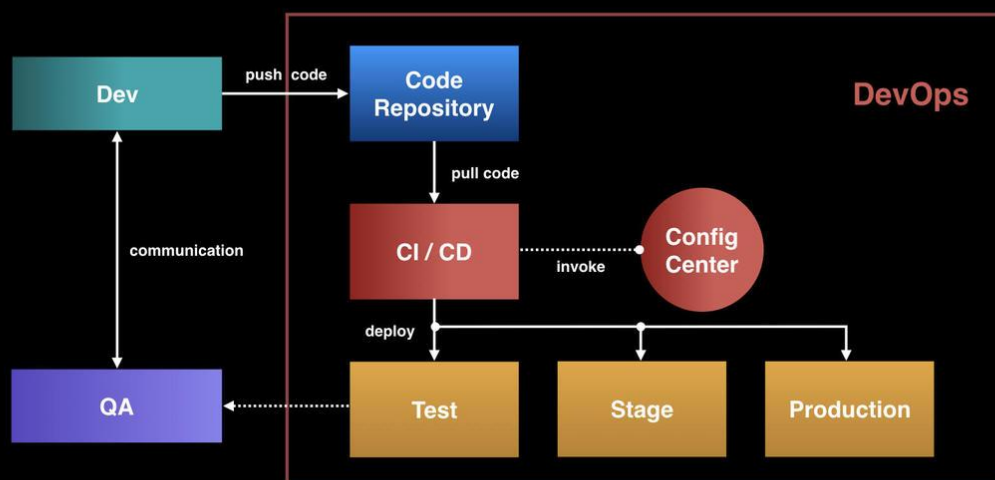
第二种情况：开发会花费大量的时间在沟通，但是有多角度的视野。测试人员专注力提升。缺点是：开发人员无法专注于业务开发和技术专研，测试局限于测试。

他们有一个共同的缺点：

无论是开发和运维还是测试和运维，他们之间总有无尽的黑暗的墙阻隔。运维像个黑盒被吞没。

那么微服务如何解决这些场景呢？

解决方案



7

这是个微服务较为主流的工作流程：

开发人员提交代码到代码仓库，微服务所独有的持续集成CI(Continuous Integration)和持续交付工具CD(Continuous Delivery)自助拉取代码调取一个配置中心，ssh连接对应远程服务器将代码部署到服务器上启动服务。通过工具通知或者开发测试沟通通知测试人员进行测试。测试通过后，部署到预生产环境和生产环境。

红色框内的工作流我们称之为 **DevOps**。这个名词最近越来越火。简单解释下 **DevOps**（来自可爱的wiki百科）：

DevOps是软件开发、运维和质量保证三个部门之间的沟通、协作和集成所采用的流程、方法和体系的一个集合。它是人们为了及时生产软件产品或服务，以满足某个业务目标，对开发与运维之间相互依存关系的一种新的理解。

技术选型

代码仓库	Gitee	
持续集成和交付	Jenkins	
配置中心	Ansible	
服务启动	Docker	
注册发现中心	Zookeeper	
日志系统	ElasticSearch + Logstash + Kibana	
监控系统	Prometheus + Grafana	

8

这些都是比较主流的微服务设施，当时现场有人提问：像京东有一整套的设施为何不去使用？

我们需要的设施是服务于最适合当前的业务和架构场景的，所以我们的设施都是经过调研斟酌的。筛选组合成了一套最适宜于自己架构的设施，经历过很多次失败。并且我们不断完善我们的架构，最近在做高可用服务，欢迎志同道合的伙伴加入我们一起玩微服务。

我们比较一下传统行业下和微服务下的变化吧：

场景改变

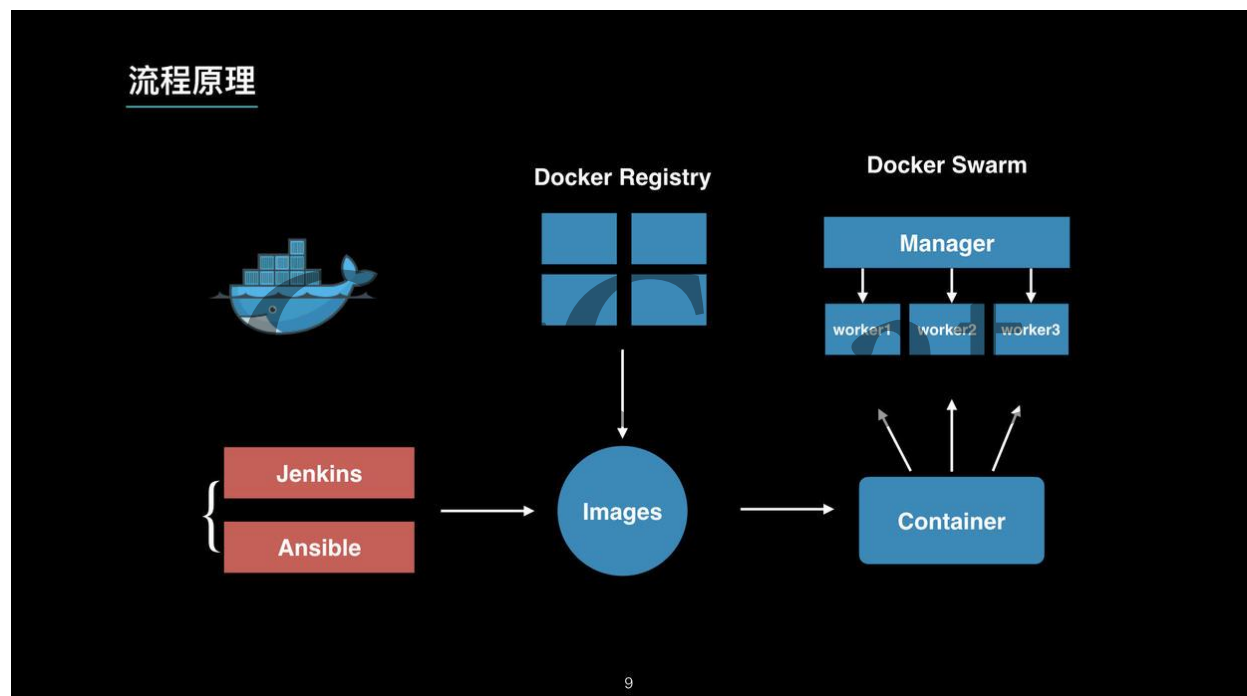
2. 测试由代码测试变成镜像测试
3. 处理线上故障和上线宕机机制改变

说到镜像这里有一个单词不得不提了：**Docker**，**Docker**和微**服务**相辅相成。如果说微服务没使用docker甚至都是个伪命题。

Docker 有三个大要素：**镜像仓库、镜像、容器**

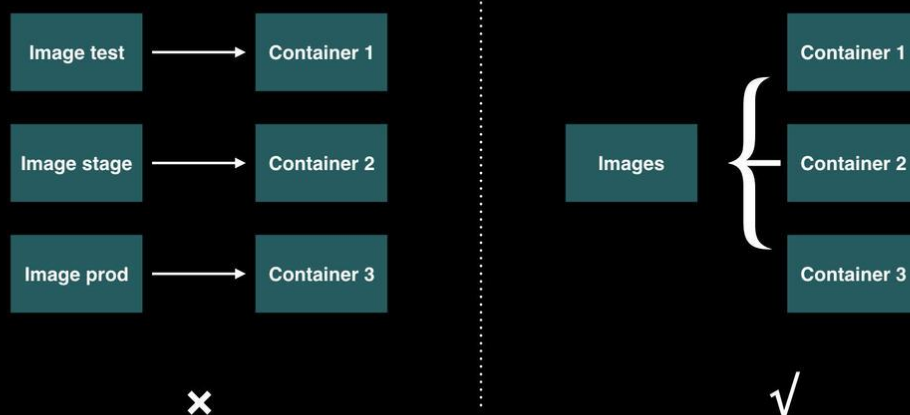
下面总结了Docker的简易工作流：

Jenkins和Ansible设施把开发人员的代码通过读取dockerfile文件的方式生成一个镜像。这个镜像扔进镜像仓库中，可以是个可视化的页面进行管理。当容器启动时从对应的镜像仓库拉取分发到被docker swarm集群下的一个worker服务中就被运行起来了。



Docker不是虚拟机，它类似于虚拟机。但是它更轻量，容器只用启动开发者软件所需要的依赖，不像虚拟机启动重要的依赖系统和软硬件件。

镜像误区

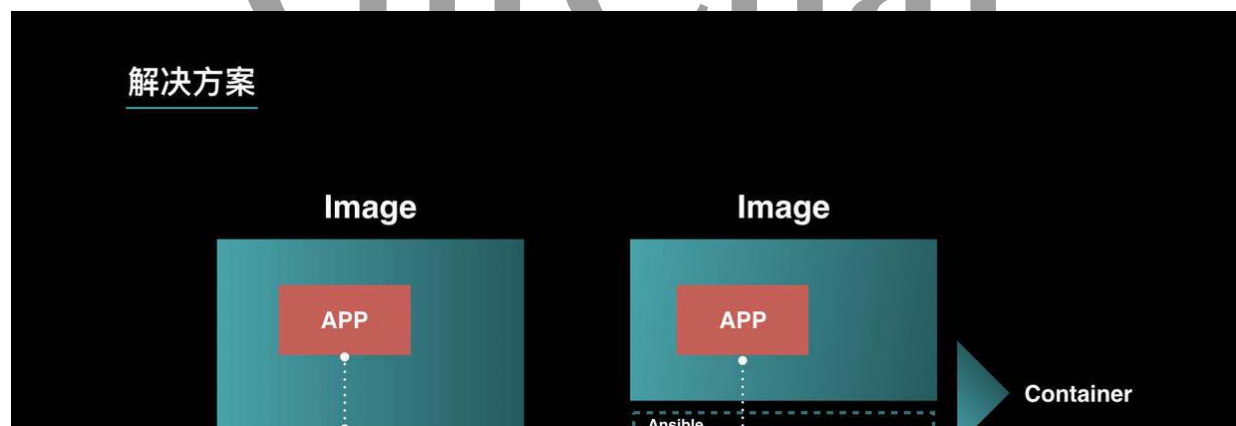


11

之前我们一直以为一个镜像对应一个容器，所以不同环境启动不同的容器。我们发现其实镜像中依赖和代码都是一样，除了数据库的地址也就是数据源不同。如果一个镜像对应一个容器我完全可以按照传统的做法将代码部署到不同的服务器上就可以了。所以我们认为真正的镜像应该只有一个，所有不同的服务器上的容器启动的镜像也应该是一个。

我们如何解决呢？

解决方案



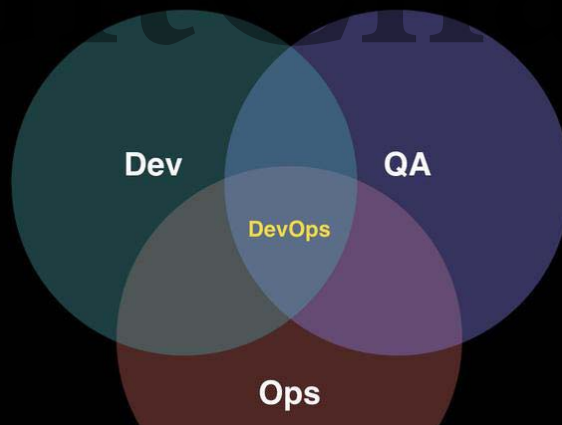
基本到这第一节的内容就结束，这次主题是Testops。很多人肯定会奇怪为什么要将微服务和DevOps，TestOps是被孵化出的衍生职位，让微服务飞一会儿。我们好登机！让我们进入正题吧：

2 - DevOps孵化下的TestOps

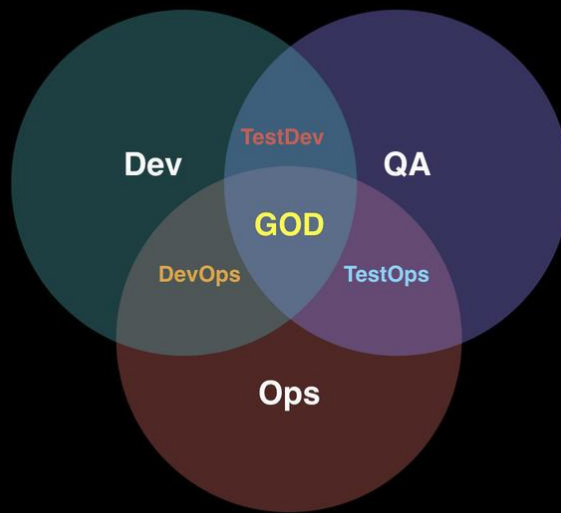
13

一张wiki百科的维恩图：

维恩图



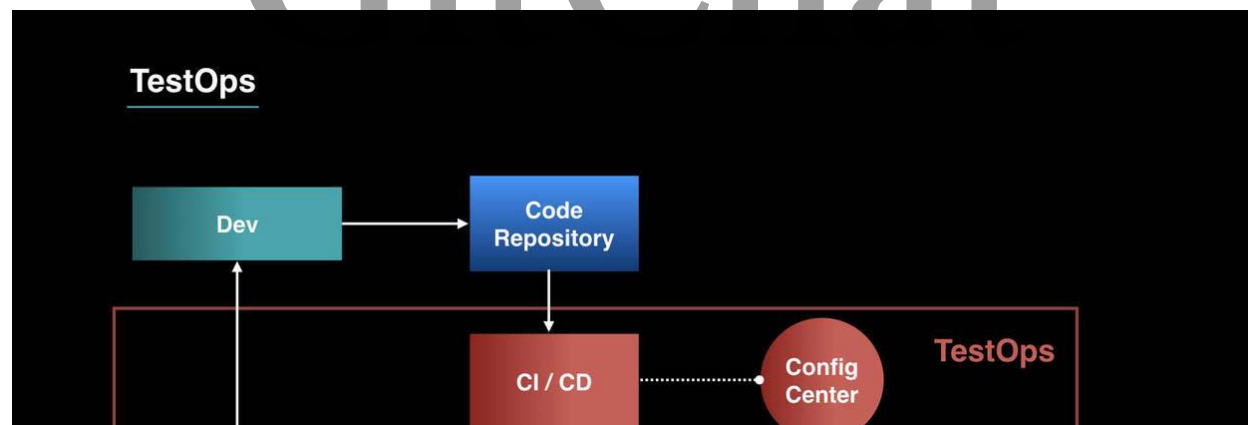
趋势图



15

之前那副图有讲过DevOps workflow。以我为例，公司Testops主要涉及的工作内容，如图所示的所有的范围：

包括持续集成工具的搭建和维护，配置中心的代码编写和维护。服务相关的处理和和维护。测试的本职工作。



定义

1.测试运维工程师

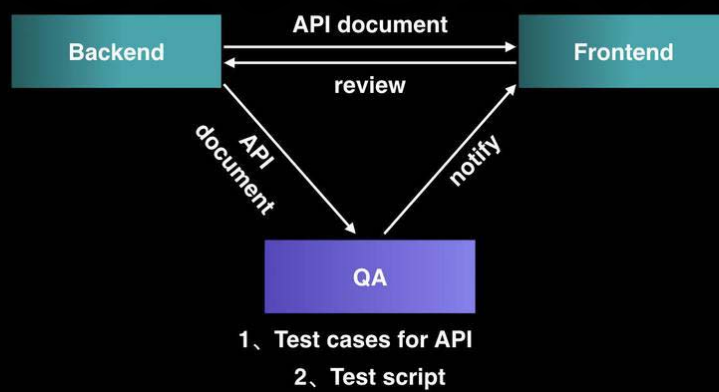
2.测试角度推动研发、运维

3.持续测试到持续集成

17

有一个简易开发团队中Testops的工作流：

工作流程



技能掌握



19

TestOps在DevOps中的价值体现于团队价值和个人价值：

团队价值

1. 推动先进的团队协作方式
2. 持续交付生命周期的把控
3. 持续集成的高质量要求

20

个人价值

1.DevOps和TestOps技能

2.自动化测试推动持续交付

3.主导开发流程的生命周期

21

TestOps应该有怎样的未来？

3 - TestOps 在未来

专家名言

At Facebook and Microsoft Bing,
they practice combined engineering,
with no separate test discipline.

The goal is for developers to not only develop code
but to also do all of the TestOps tasks above.

“微软测试专家 TestOps之父”

Seth Eliot

23

未来价值：

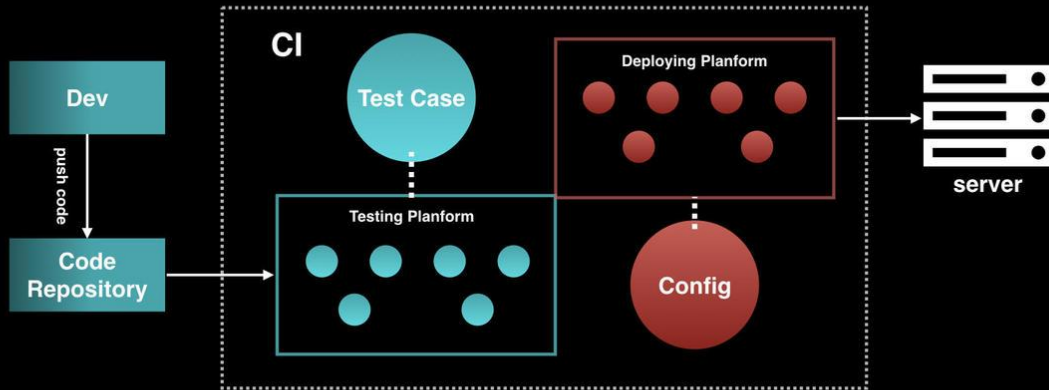
DevOps能推动整个测试和运维团队统一整个研发流程，帮助团队更敏捷的提交产品。他能解决流程问题，但是无法发现开发过程中测试的缺陷。只有专业的**TestOps**站在专业的测试角度推动开发和运维一起进行。**TestOps**和**DevOps**形成一个完整的**持续集成**和**持续交付**体系，才是完善了整个微服务下的工程师架构了。

未来价值



开发人员提交代码到代码仓库，CI工具会有**持续测试平台**和**持续部署平台**。持续测试平台包含：代码质检工具类似sonar，接口测试工具，UI测试工具...测试人员只用编辑测试场景和用例来帮助工具执行用例。如果有了人工智能AI那么很有可能功能测试人员将会失业哦。

未来场景



25

未来TestOps应该会一直关注：

未来关注

1. 持续测试

2. 质量统计分析和回溯

居 强



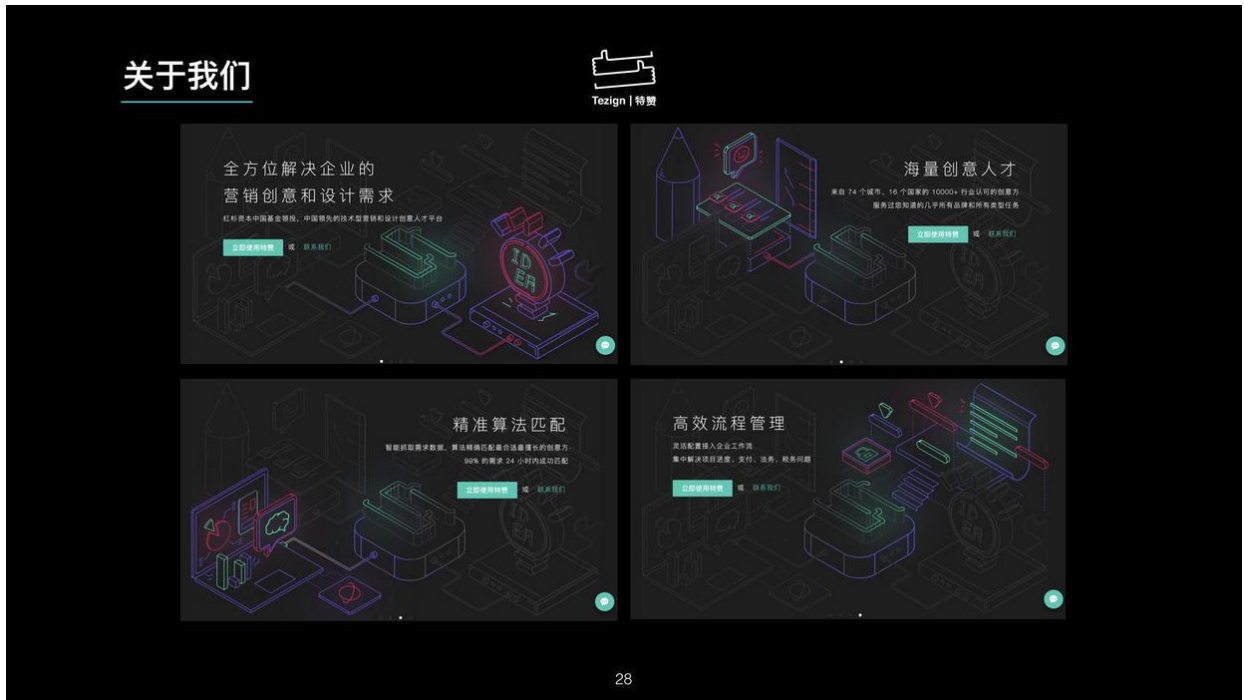
27

关于特赞：

特赞(tezign.com)是一个利用大数据和智能匹配技术为企业精确对接设计创意人的科技公司，目前开放平面设计、UI/UX设计、插画设计、动画视频设计，积累了来自16个国家、74个城市的10000+位优秀设计师，服务了4000+企业客户。特赞重新定义未来企业和创意人才的合作方式，让天下没有难做的设计。特赞在2016年4月获得红杉资本中国基金领投的数千万级人民币A轮融资。

特赞 (Tezign) 的名字是 科技 (Technology) 和 设计 (Design) 的结合，我们将科技和商业带入设计。Design Matters是特赞发起的一个社会项目，致力于通过会议、研讨和工作坊等模式把设计和人文融入科技，吸引了超过40家媒体和累积20万的观众。

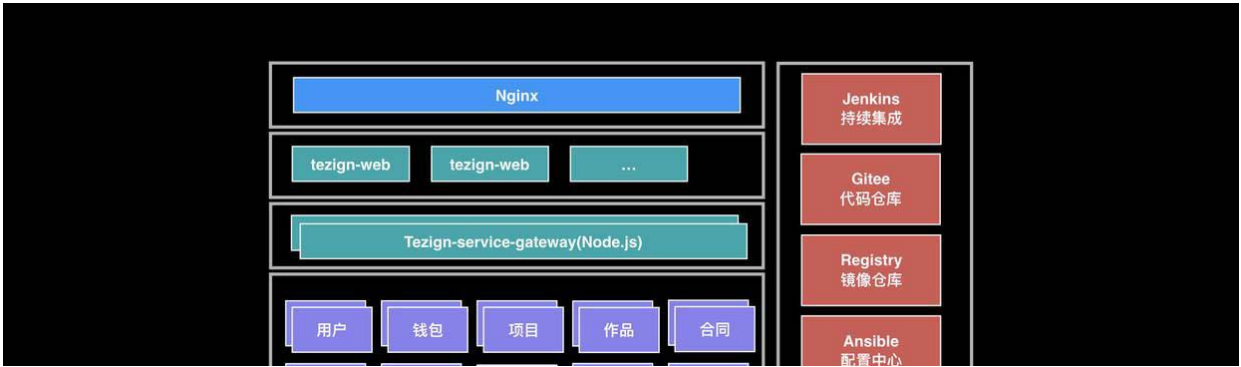
关于团队：特赞是一个具有创意基因的互联网技术团队，来自于人工智能、人机交互、大数据、SaaS软件服务化、创意管理、广告媒体等跨学科背景的成员组成。毕业于哈佛大学、普林斯顿大学、哥伦比亚大学、复旦大学、浙江大学、清华大学。



28

下面是我们比较成熟的架构了：

我们服务全部是通过Docker启动的，通过Node.js开发了一个服务网关（Tezign-service-gateway）也是通过docker启动。服务通过注册中心（Zookeeper）注册，服务网关发现服务。动态路由根据请求连通前后端。Nginx做负载均衡。红色框内为微服务所需的一些微服务设施。这套架构目前非常稳定，我们还在持续优化中。



浩瀚宇宙99%的星星不会发光，但是他们一直存在，并且永垂不朽。我坚信存在即合理。所以做最好的自己，你就最好的存在着。

浩瀚宇宙星光璀璨 99%的星星不会发光
它们依然存在 并且永垂不朽

30

最后，由于文章篇幅问题。工作内容的细节问题，我们可以线上进行交流，如何做好或者发展成为一个TestOps。**感谢大家阅读这篇冗长的理论文。**