



日调度 5 万亿次，腾讯云微服务架构体系 TSF 深度解读

作者：江柳
阅读数：3345 | 2018 年 2 月 5 日 | 话题：架构 语言 & 开发 腾讯云 微服务

看新闻很累？看技术新闻更累？试试[下载 InfoQ 手机客户端](#)，每天上下班路上听新闻，有趣还有料！

写在前面

当前，传统企业的 IT 系统以单体架构为主，在面对互联网业务的冲击时，系统架构的性能瓶颈逐渐显现。云计算、Docker、DevOps、持续交付等概念的深入人心，以 Spring Cloud 为代表的微服务框架日渐兴起，微服务架构成为传统 IT 架构转型的集中趋势。

在微服务化的行业汹涌浪潮里，腾讯云历经五年磨砺，整合外部开源框架和内部 PaaS 平台，完成了王者荣耀全球同服的毫秒级延时和春节红包的高并发交易等性能需求，以日 5 万亿次的惊人调度次数，支撑腾讯内部海量业务的构建与发展。

微服务改造的核心思想，指通过 IT 架构的微服务化，将复杂的单体架构，重组为小而美的独立服务，从而降低系统的复杂性，让企业更便捷的构建基于云计算的大规模分布式架构。

本文结合腾讯云微服务架构体系的构建原理、技术选型和改造实践，为你讲讲如何解决微服务部署、实施、监控余位中面临的难题。

传统企业 IT 架构面临的痛点

单体架构通常在一个归档包里容纳了所有功能的应用程序，整个项目包含的模块种类繁多，模块边界界定模糊，每个模块之间具有强耦合性，项目复杂。大多数传统企业在上云的过程中，由于单体架构的固定属性，会面临着 IT 系统复杂、升级迭代慢、运维扩展性差、海量用户支撑能力薄弱、数据孤岛等一系列问题。

如传统企业在做电子政务、智能零售、工业 4.0 等智能化转型，或者想要开发人脸识别 / 支付系统、关联小程序等热门应用时，应用体系的改变以及用户量级的爆发式增长，都会对单体系统的性能瓶颈会提出极大的挑战。

不同于构建单一、庞大的应用，微服务架构以小型服务的方式开发独立应用系统，将应用拆分为一套小且互相关联的服务，每个小型服务都运行在自己的进程中，各服务之间采用 HTTP 资源 API 轻量的机制进行通信。相对于单体架构，微服务体系在迭代速度、系统吞吐量、扩展性以及技术栈的多样性上均有明显的优势。

- 0
- 喜欢
- 收藏
- 评论
- 微信
- 微博



| | | |
|--------|----------|---------|
| 迭代速度 | 较慢 | 快 |
| 部署频率 | 不经常部署 | 经常发布 |
| 系统性能 | 吞吐量小 | 吞吐量大 |
| 系统扩展性 | 差 | 较好 |
| 技术栈多样性 | 单一、封闭 | 多样、无缝 |
| 运维 | 简单 | 运维复杂 |
| 部署难度 | 容易部署 | 较难部署 |
| 架构复杂度 | 较小 | 复杂度高 |
| 差错 | 简单 | 定位问题困难 |
| 管理成本 | 主要在于开发成本 | 服务治理与运维 |

由于单体架构的缺陷日益明显，越来越多的公司采用微服务架构范式构建复杂应用。但在微服务的部署实施过程中，亦存在着架构与运维复杂、部署实施难度较高、问题定位困难等挑战。

- 运维要求较高。更多的服务意味着更多的运维投入，单体架构中只需要保证一个应用的正常运行；而在微服务中，需要保证几十甚至几百个服务的正常运行与协作；
- 分布式固有的复杂性。使用微服务构建的是分布式系统，对于一个分布式系统，系统容错、网络延迟、分布式事务等对于开发者而言都是巨大的挑战；
- 接口调整成本高。微服务之间通过接口进行通信，如果修改某个微服务的 API，可能所有使用了该接口的微服务都需要做调整；
- 重复劳动。很多服务可能都会使用到相同的功能。而这个功能并没有达到分解为一个微服务的程度，这个时候，可能各个服务都会开发这一功能，导致代码重复。

腾讯云微服务架构体系及其技术选型策略

互联网企业的微服务转型经验，可为传统行业开发者提供经验参考。腾讯云基于自身的海量业务需求，通过 IT 架构的微服务化，将紧耦合的塔式架构，重组为小而简的独立服务系统，打磨出一套可用的微服务架构平台和微服务构建方法论，实现业务之间解耦和技术栈独立，允许低成本试错，使团队迭代更敏捷。

微服务构建的五大核心要素

微服务构建的初衷不外乎实现敏捷迭代、灵活扩展、服务复用三大功能，腾讯云在构建微服务的过程提炼出了以下五大方法论：

- 在线协同：对外的 API 文档就是一份公共的说明，常有发布新的不兼容接口，因此，如何跨团队协作、通知至关重要，这方面需要通过 swagger UI 做在线的接口定义，以此公共契约；
- 部署原则：在真实环境里，Docker 应用还未完全普及，服务的部署耗时费力。可以尝试提供批量的自动化工具，做微服务独立打包，批量的部署，包括启动、停止、升级、回滚、下线等操作；
- 拆分原则：如常见的线上房地产交易门户商品中心，包含运营相关的子系统如二手房推荐系统等，服务拆分得过细会带来不必要的分布式事务、调用环节冗长等问题。各系统的拆分原则上可强调“抓大放小”；
- 数据扁平化：服务升级过程中，要注重数据模型的统一。首先，各微服务的数据层要允许有完善的权限管理系统，支持多种数据格式转换、数据清洗、数据同步等，便于业务高效地挖掘数据的价值；
- 渐进性架构：大多企业和开发者很难从一开始高瞻远瞩，规划处 3、5 年不变的微服务架构，因此，需要有长期的演进迭代以及小规模团队维护，允许小团队技术栈独立，来拥抱业务团队的快速试错。

腾讯云微服务中间件 TSF 框架解析

0



喜欢



收藏



评论



微信



微博

具价值的经验参考。

如下图为腾讯云微服务中间件 TSF 一站式解决方案，最底层是云基础资源平台，包括云服务器、云数据库、云存储和专线加速几大模块，用作数据的存储和调用；同时，作为围绕微服务的 PaaS 平台，TSF 服务框架底层也融合了腾讯云内部大量的中间件服务，提供企业云化架构所必需的消息队列、Kafka、负载均衡、API 网关、全局配置服务等全套中间件。



在此之上，TSF 支持应用的全生命周期管理功能，如对于虚拟机应用，提供代码包打包上传，批量发布、变更，版本切换等产品生命周期功能；对于火热的 Docker 应用，提供基于行业主流编排框架 Kubernetes 的全流程自动化持续集成和持续交付。

对于已经在使用 Dubbo 框架的用户，可以通过修改 pom.xml 中的依赖，平滑地迁移到 TSF。其中，Dubbo 存量系统迁移方式包含两种：

- 1. 将业务发布包中的 dubbo-registry-zookeeper.jar 包替换为 tsf-registry-consul.jar 包；
- 2. 修改 dubbo 配置项：adress 部分替换成腾讯云提供的 url 即可使用腾讯 TSF 服务。

```
<dubbo:registry id="xxx1" address="xxx://ip:port"/> <!-- 定义注册中心 -->
```

另外，在微服务运行与治理框架方面，TSF 为应用提供了自动注册、发现、治理，隔离，调用分析等一系列微服务治理能力，用以屏蔽微服务系统带来的复杂度。整个 TSF 框架基于 Spring Cloud，支持分布式服务发布与注册、服务调用、服务鉴权、服务降级、服务限流、配置管理、调用链跟踪等功能。

TSF 微服务框架技术选型策略

• RPC 高性能服务框架

TSF 基于 Spring Cloud 生态，提供标准 restful 服务框架，同时，鉴于分布式服务框架对性能和可靠性等能力具有极高的要求，腾讯云 TSF 在设计之初，就基于 gRPC 提供高性能 RPC 服务框架，系统地考虑了分布式服务发现、路由管理、安全、负载均衡等细节问题，满足 100 万 +QPS 的服务压力。



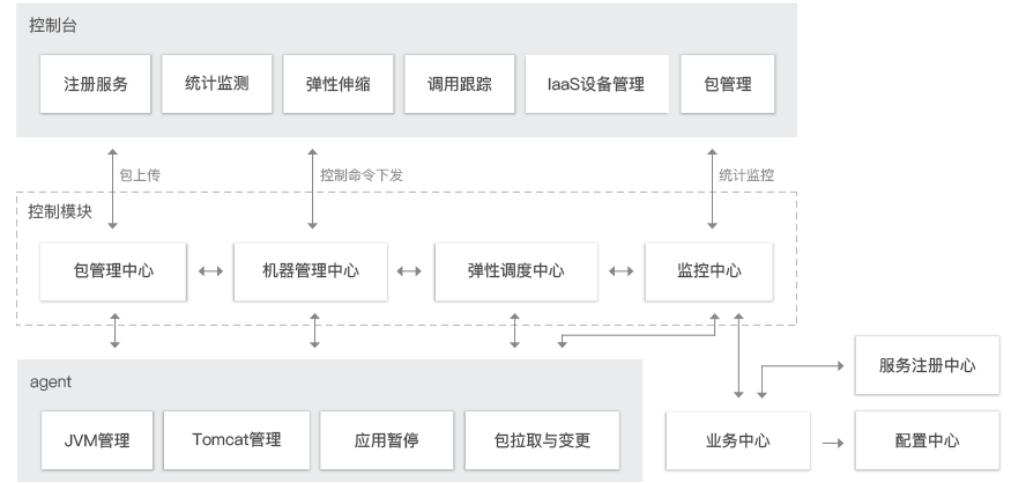
如上图所示，TSF 服务注册发现包括三个角色，服务提供者，服务调用者和服务注册中心。服务提供者和
服务调用者将地址信息注册到服务注册中心，并从服务注册中心获取所有注册服务的实例列表。TSF 使用
Consul 作为服务注册中心，提供金融级别的两地三中心容灾部署能力。

• 构建可视化控制台

应用是 TSF 管理的基本单位，支持 Java jar 包，docker 镜像及其他语言应用（通过 service mesh）等，一
个应用下面通常包含了多台机器，TSF 提供了完整的可视化控台，完善应用的生命周期管理机制，对分布式
系统应用进行发布和管理，减少部署时间，提升部署效率。

在 TSF 控制台上，可以设置自定义 JVM 参数。TSF 将每次操作记录下来，用户可以在应用的变更记录页面
中查看和搜索变更记录。

- 1. 通过控制台发布应用，包括创建、部署、启动应用，也支持查看应用的部署状态。
- 2. 通过控制台管理应用，包括回滚应用、手动扩容、缩容、删除应用。



• 全链路调用跟踪

在监控层面，TSF 采取的策略是通过全链路调用跟踪技术打造一站式监控平台，以可视化的形式，准确掌握
平台各项指标。具体实现如下图，通过对日志埋点的收集和分析，得到一次请求在各个服务间的调用链关
系，帮助梳理应用的请求入口与服务的调用来源、依赖关系。当遇到请求耗时较长的情况，通过调用链分析
调用瓶颈，快速定位异常。

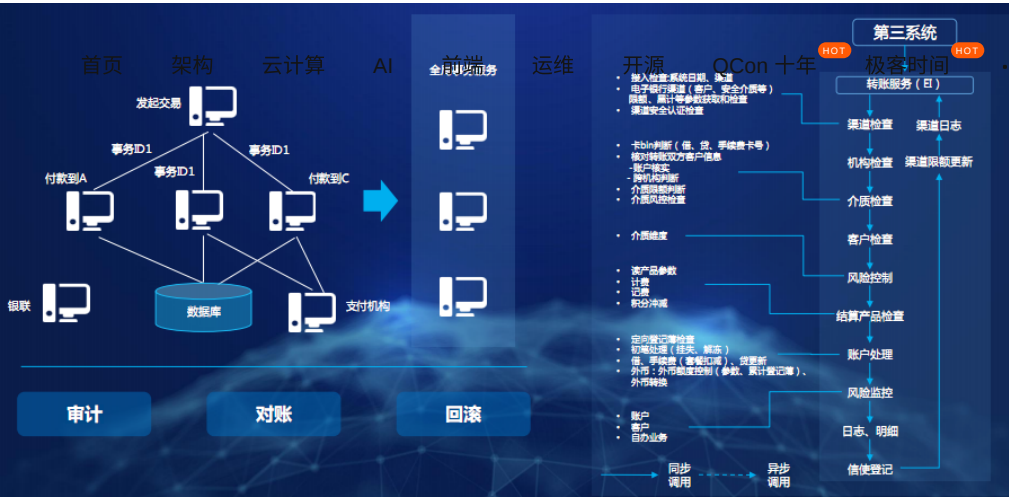


监控方面包括 **IaaS** 基础监控和应用监控。IaaS 基础监控的指标有实例 CPU、内存、网络和磁盘等，应用监控的指标包括应用的 QPS, 请求时间和请求出错率等。

分布式调用链分析包括调用链查询和调用链详情。调用链查询用于查看系统中的调用链路状态，尤其是慢业务和出错业务，避免后端雪崩；调用链详情是为了定位在分布式链路调用过程中每个环节的耗时和异常，用户可以根据时间范围和服务名等条件来查询一组调用链，有助于提前发现架构的瓶颈点。

• 自研 **TDTS** 分布式事务服务

金融场景下，对数据一致性有着严苛要求，通常一笔交易将会跨多个业务集群。TSF 采用自研分布式事务服务中间件 TDTS，基于 TCC 模式，提供无业务侵入性、无改造数据库改造成本的最终一致性中间件，保证每一笔交易可靠达成。



TCC 模式通过应用层的两阶段提交，可以解决数据库的两阶段并发性差、数据库对 XA 协议的支持可能不完善、以及部分情况下主业务系统智能间接操作数据库等问题。其实现原理是先执行 try, 再执行 confirm；若 try 失败，则执行 cancel。具体事务执行流程如下：

1. 依次调用从业务系统的 Try 接口；
2. 若 Try 阶段成功，则依次调用从业务系统的 Confirm 接口；
3. 若在 Try 阶段出现失败，则调用从业务系统的 Cancel 接口，对 Try 接口的操作进行恢复；
4. 若在 Confirm 阶段出现失败，则重试，设置重试次数。若重试仍然失败，则需要事后处理；
5. 若调用 Cancel 接口失败，也进行重试操作；

衡、熔断等能力。

TSF 在传统企业的微服务改造实践

腾讯云的 TSF 分布式微服务框架雏形可追溯至 2013 年，其初期以 Devops 服务系统的形式，支撑了包括财付通、腾讯游戏、手机 QQ 等内部核心业务的架构升级。2017 年，腾讯云整合内部微服务体系，提炼出了具备金融级容灾能力，高性能、高可靠的微服务解决方案，帮助传统企业快速解决在微服务改造过程中遇到的问题。此处以某房地产为例，简单讲述 TSF 在传统企业的微服务改造实践。

传统房地产企业面临的问题

传统房地产企业在互联网转型升级过程中，IT 系统架构极为复杂。因为单体 IT 架构的固有属性，其 IT 子系统各自为战，标准不一，无法相互通信，如会员信息、订单信息、消费者行为记录等多套系统链接分散，一旦涉及到业务变更情况，经常数百人加班参与，对后续的系统迭代带来了巨大挑战。

国家新型城镇化规划出台后，“城市配套服务商”的战略一时引发热潮，围绕系列新业务，包括：装修、社区教育、酒店、商办（写字楼）、长租公寓、养老地产、物流地产等系统协同发展，此时，传统的 IT 架构显然已经无法满足新的业务发展需求。

腾讯云 TSF 微服务改造方案

腾讯云在原有的 IT 系统上，梳理出关键模块和关键资源，同时，在云中间件的平台上，搭建了系列抽象出来的中间件系统，包括负载均衡、API 网关服务、分布式配置、分布式事务、消息队列服务、分布式数据库以及腾讯云的 IaaS 能力，将独立的 IT 系统通过微服务的方式进行管理，对于关键资源通过合并调用减少调用册数，对于非关键资源同步异化处理，减少实施强依赖。



通过微服务改造，将房地产的业务拆分成一手房销售系统、土地拍卖系统、营销包装系统与认筹系统，每个系统底下都有自己独立的运行进程，各系统之间通过轻量级的方式进行通信，每个系统独立而又相互连接。

此时，可以在公有云或者其他小程序上，根据自身业务需求，开发出新的应用或者系统如管家微信小程序、在线家政预约服务等，因为每个系统的运行相互独立，新的应用即便呈现爆发式增长，也不会影响其他模块的运行。

据悉，通过搭建分布式微服务中间件 TSF，该房地产的 IT 系统在 80 天内完成微服务架构改造上线，版本迭代平均周期从一个月变为两周，当有新应用时，可以快速上线互联网应用。目前，系统每天能够稳定处理超



当然，这个项目的整体以渐进性开发，循序渐进，前期主要进行三核心系统迁移，当完成主体架构搭建后，后期再根据需求进行适当的迭代和更新。

因此，对于传统企业的微服务架构引入策略上，建议在开始时可以考虑引入部分合适的微服务架构原则，对已有系统进行改造或新建微服务应用，然后逐步探索及积累微服务架构经验，而非全盘实施微服务架构。

文章版权归极客邦科技 InfoQ 所有，未经许可不得转载。

架构 语言 & 开发 腾讯云 微服务



0 人喜欢



收藏



评论



微信



微博



写下你的想法，一起交流

0



喜欢



收藏



评论

发表评论

注册/登录 InfoQ 发表评论

注册/登录



促进软件开发领域知识与创新的传播

特别专题



百度技术沙龙 华为云特惠 云+未来 Intel
华为云 MeetUp 百度 AI AWS
云+社区开发者大会 迅雷链技术专区
工业大数据创新竞赛

关于我们

关于我们
合作伙伴
关注我们
我要投稿
加入我们

联系我们

内容投稿: editors@geekbang.com
业务合作: hezuo@geekbang.org
反馈投诉: feedback@geekbang.org

InfoQ 近期会议

软件开发大会 2019年5月6-8日
架构师峰会 2019年7月12-13日

全球 InfoQ

InfoQ En
InfoQ 日本
InfoQ Fr
InfoQ Br