

Bigdata-微服务契约治理

Bigdata 契约 规范 协作

微服务既是一种技术手段，也是一种治理方式，通过分而治之方式，达到组织团队和业务开发规模化 and 快速迭代的目标。异地团队协同开发，采用微服务架构，可以考虑采用契约优先（**contract first**）开发模式，例如：**grpc**或者**spring/swagger**支持契约驱动开发。开发前，两边团队定义好契约，根据契约开展开发和测试，可以提升一致性和研发效率，当然团队之间两盒的沟通和写作也非常重要。

一、为什么要治理服务契约？

在日常工作中发现契约导致的沟通问题很多。另外由于契约定义不严谨，导致线上Bug的出现也有发生。比如忽略字段长度，导致数据落库失败。由点到面梳理问题如下表：

问题	开发阶段	测试阶段	运行阶段	后期迭代
契约不严谨	1.增加了开发人员的理解成本 2.增加了契约相关方的沟通成本 3.这种情况下，再缺少沟通的话，各相关方按照自己的理解开发导致后期联调成本增加。	1.增加了测试人员的理解成本 2.增加了测试与开发的沟通成本	服务不健壮	1.增加了产品的理解成本 2.增加了产品与开发的沟通成本
服务契约与文档契约不一致	1.增加了联调成本 2.导致联调时某一方的返工	1.增加测试与开发的沟通成本 2.导致测试用例的返工	服务不健壮	1.增加了产品的理解成本 2.增加了产品与开发的沟通成本
契约改动通知不到位	1.增加了联调成本 2.导致联调时某一方的返工	1.增加测试与开发的沟通成本 2.导致测试用例的返工	线上故障	
契约信任	1.契约验证开发成本 2.契约验证维护成本	契约相关的测试成本		

另外，还存在职责问题，例如：

服务提供者的问题，被客户端发现。而客户端对这种非自身问题关注是不够的，导致问题不被重视。

二、怎么治理服务契约呢？

三、治理什么？

一个健壮契约的元数据：

- 协议
- 结构
 - 基元类型
 - 文本、数字、布尔、时间
 - 约束
 - 复杂类型
 - 类、数组

四、支撑该设计的原则

1. 契约先行
2. 问题尽早发现
3. 谁的问题谁发现