

系统设计分析手册

系统分析设计手册V0.1

兼容性

互联网业务场景基本要求系统不间断运行，兼容是为了保障系统在灰度期间、或者上下游可能不能做到同时发布，需要在方案分析设计的时候，对兼容加以考虑。

数据兼容性

1. 在新流程的数据和老流程的数据并存的时候，至少是有3种场景需要考虑到位的：

1. 全是新流程的数据
2. 全是老流程的数据
3. 新流程的数据和老流程的数据并存时

【反例】

1. LFS灰度时，在SLS一个大包中存在走SLS流程和走LFS流程的，验证过程只分别验证了上述的第1点和第2点，而第3点则没有验证，导致线上在灰度过程出现问题；
2. 点线迁移灰度方案只考虑了新流向场景，没有考虑旧流向异常场景，oms请求到了sls，sls已经记录的下单数据，但是超时了，OMS重试，请求灰度到了LFS，又下了相同的单，导致订单重复。

协议兼容性

1. 协议的参数名发生变化，应该知会调用方做充分的评估。

【反例】

LFS给预测模块的数据，相比之前SLS给的数据协议，将SKU的参数名从下划线改成了驼峰式，导致预测结果不准。

1. 服务使用方使用协议的参数值可能超过原有的范围，需和服务提供方确认后再使用。

【反例】

TWS推的轨迹中，存在有location字段，但一直都是赋值成“SG”，之前对这个字段只是展示用，后来用了这个字段来做轨迹描述时，导致问题。

1. 服务切换，比如从coreserver改到spex，需要严格到对比两个不同服务协议规则，以及代码实现前后到对比实现，服务切换不应该有一点差异

【反例】

事故中，需要将原来的coreserver协议，变成spex协议，开发人员无视之前的逻辑，按照自己认为的协议去做了对接，遗漏了几个参数没有传，导致部分场景获取不到地址，最后变成了事故。

消息处理

消息顺序性

1. 异步消息消费时，针对同一个维度，比如同一个订单，要考虑消息的顺序性问题，同一个维度的数据可以使用kafka的同一个partition来实现顺序性，或者对DB加行锁。

【反例】

1. 同一笔运单的轨迹推送有先后两条消息A(旧)、B(新)，LFS先收到A消息，但是处理较慢，后收到B消息，B消息先处理成功了，导致消息B最后被旧的消息A覆盖。

边界分析

边界值

边界值分析是对等价类划分的补充，大量的线上问题经验总结可知，问题往往发生在边界情况上，而不是输入值的中间范围上。所以针对边界情况进行设计分析是防止线上问题出现的必要步骤。

哪些情况需要做边界值分析：

1. 输入条件明确了值的范围
2. 输入条件明确了值的数量
3. 输入条件是一个有序的集合

【反例】

1. 服务范围迁移过程中CB订单校验失败，原因：在location和postcode都没有配置的情况下，lcos服务范围返回了不支持，实际应该按照产品提供的默认逻辑返回支持或不支持。该例子违反了情况1，缺乏对输入值（location和postcode）为空时的分析，导致输出和预期不一致。特别的，在功能迁移的背景下，对输入边界值返回正确性的分析应基于旧返回值和新返回值的对比上（即同一个边界值，新旧逻辑应返回一致）。
2. 运费推送给OMS，如果is_asf_reasonable设置json omitempty，oms默认该类型订单IsAsfReasonable = true，导致业务在运费异常时无法发现问题，忽略了入参为空的场景，缺少对输入值边界考虑；

并发

并发缓存

缓存Value属于全局变量，不应该在函数间传递，函数参数和返回值容易被当做协程的局部变量，进而引发并发问题，应该传递缓存的Key，

【反例】

LFS lane 数据使用本地缓存全局变量，结构数据被上层修改，导致别的场景使用数据错误。

流量评估

分布式缓存

大促活动带来的瞬时流量是平时的几十倍甚至更多，对系统的高可用性带来很大的挑战。因此，在使用缓存时需要正确的评估流量和分析热key，否则在大促时就可能产生严重故障。

评估方式：

1. 分析单次接口调用访问缓存次数
2. 根据PCU或Order信息评估出接口峰值QPS，再乘上单次访问次数得到预估流量
3. 压测验证预估流量的正确性
4. 根据缓存场景分析热key
5. 将热key打散或采用lru内存缓存方式解决分布式缓存流量不均匀问题

【反例】

1. 服务范围迁移过程中为了减少SLS-API对LCOS访问量，在SLS-API利用codis设置了5min的缓存，缓存key包含买卖家地址信息。因为没有根据PCU正确预估峰值流量，也没有提前分析热key，导致4.4大促期间VN市场24分钟PDP和checkout延迟上涨功能不可用