

服务质量目标

服务稳定性治理

服务质量术语概览

服务质量指标SLI

服务质量目标SLO

- SLO，全称为Service Level Objectives，即服务质量目标

- SLO又是基于数据驱动做出事关可靠性决策的关键要素，因此，它是SRE工程实践的核心

- SLO应该以客户为中心，与客户的体验直接相关

- 一个好的SLO，应该是SMART的，应该是具体、有时限且可衡量的

服务质量协议SLA

- SLA全称为Service Level Agreement，即服务质量协议

- 有些服务可能并未存在一个SLA，但确保其较高服务质量却可能事关其形象，进而影响其业务收入

选取SLI和SLO

- 系统稳定性建设的关键就在于选定SLI，并分别为之设定合理的SLO

- 常见的指标

- 关注系统的关键行为

SLI及相关数据的收集

- 常见的服务分类后的重要SLI

- 收集指标

如何选择SLI

- 从众多指标中快速选择SLI的方法：VALET（Volume,Availability,Latency,Error和Ticket）

错误预算

- 如何确保达成SLO？

如何衡量SLO的有效性

落地SLO 要考虑的因素

落地SLO 要考虑的因素（2）

- 设定SLO的原则

- 验证核心链路的SLO的常用方法

服务稳定性治理

SLI/SLO/SLA的制订与落地

故障预防

抑制不可控因素

故障应急演练

业务MTTR

灾备建设

服务质量术语概览

正确运维一个系统的关键前提，是要详细了解服务中各种行为的重要程度，并度量这些行为的正确性程度

■ SRE需要结合主观判断、经验及对服务的理解来定义事关系统关键行为的服务质量指标（SLI）、服务质量目标（SLO）和服务质量协议（SLA）

■ SLI代表服务质量的一个可量化的衡量维度，例如请求延迟、错误率和可用性等

■ SLO代表针对一个SLI所设定的目标值或目标范围，例如可用性指标要大于等于99.9%等

■ SLA负责定义某个SLI没有符合其相关的SLO的定义时要采取的应对计划，例如退款或赔偿损失等

通常，选择出关键的SLI，并为其设定合理的SLO对SRE来说至关重要

SLA的订立则需要业务部门和法务部门进行，SRE仅负责帮助这些人理解SLA的SLO达标的困难程度和代价

服务质量指标SLI

SLI全称为Service Level Indicator，即服务质量指标

用于评估服务的某项服务质量的一个可量化的角度

■性能指标：延迟、吞吐量、处理速率（TPS/QPS）和时效性（Freshness）

■可用性指标：可靠性、故障时间/频率、在线时间等

■质量指标：准确性、正确性、完整性、覆盖率

大部分服务都将请求延迟——处理请求所需要消耗的时长——作为一个关键的SLI

其它常用的SLI包括错误率（请求处理失败的百分比）、系统吞吐量（每秒的请求数量）等

服务可用性（Availability）是另一个关键的SLI，它代表服务可用时间的百分比

■100%的可用性无法实现，但接近100%的可用性指标却是一个可以实现的目标

■运行行业通常用9的数量来描述系统的可用程度

服务质量目标SLO

SLO，全称为Service Level Objectives，即服务质量目标

■用于定义某个SLI的目标值，或者目标范围

◆ $SLI \leq \text{目标值}$ ，例如请求平均延迟低于100ms

◆ $\text{范围下限} \leq SLI \leq \text{范围上限}$

■ 选择一个合理的SLO是非常复杂的过程

◆确立合理的目标值本身很困难，例如QPS由用户请求决定，也就无法为其订立目标值

◆此时，只能从侧面去为延迟指标订立一个目标值来间接反应QPS

SLO又是基于数据驱动做出事关可靠性决策的关键要素，因此，它是SRE工程实践的核心

SLO应该以客户为中心，与客户的体验直接相关

■SLO的核心目的是用于量化客户对产品和服务可靠性的体验

一个好的SLO，应该是SMART的，应该是**具体、有时限且可衡量的**

- Specific：特有，能明确表达其具体含义

- Measurable：可测量，有具体数值

- Achievable：可达成，不能是无法完成的目标

- Relevant：要反应到用户体验相关

- Timebound：要尽量只覆盖系统负载较重的时间段，以免被平均值稀释

服务质量协议SLA

SLA全称为Service Level Agreement，即服务质量协议

指的是与用户签订的事关服务质量目标的合约，它描述了达成或未达到SLO的后果

- ◆ 一般会具有一定的法律效力，往往涉及到服务付费、质量承诺和违约责任等

- ◆ SLO往往由业务团队自己设立，而SLA则通常是非技术领域的律师或销售人员所设立

- ◆ 但SLA最好能与业务团队的SLO相一致

SRE通常不会参与SLA的书写，这主要由业务部门和法务部门进行，但SRE需要参与帮助避免触发SLA中的惩罚性条款

有些服务可能并未存在一个SLA，但确保其较高服务质量却可能事关其形象，进而影响其业务收入

■例如公共搜索引擎

选取SLI和SLO

系统稳定性建设的关键就在于选定SLI，并分别为之设定合理的SLO

常见的指标

■系统层面：CPU使用率、Load值、Memory使用率、磁盘使用率、磁盘IO和网络IO等

■应用服务器层面：端口存活状态、JVM的GC状况等

■应用运行层面：请求返回的状态码、时延、应用层QPS、TPS及连接数等

■中间件层面：MySQL、Redis、Kafka和分布式文件存储等各组件的类似于应用运行层面的指标

■数据层面：大数量处理平台的批处理或流处理任务，包含吞吐率、及时率和准确率等指标

■业务层面：以电商为例，有在线用户数、新注册用户数、下单数、交易数、支付笔数以及业务层面的成功率等指标

关注系统的关键行为

■实践中，不应该将监控系统中的针对某软件的所有指标都定义为SLI，其应该事关用户最为真实的需求

■过多的SLI会掩盖掉重要的行为，而太少又容易忽略掉关键行为；针对一个特定服务，一般来说四五个具有代表性的指标就足够了

SLI及相关数据的收集

常见的服务分类后的重要SLI

■用户可见的服务系统：可用性、延迟，以及吞吐量

◆可用性：是否能够正常处理请求

◆延迟：每个请求花费的时长

◆吞吐量：有多少请求可以被处理

■存储系统通常强调：延迟、可用性和数据持久性

■大数据系统：吞吐量和端到端的延迟

■所有系统：正确性

◆是否返回了正确的回复，是否读取了正确的数据，或者是否进行了正确的数据操作

◆正确性是系统健康程度的一个重要指标，但它取决于系统内部数据，而非系统本身，因此通常不是SRE负责的

收集指标

■通常，利用监控系统或日志系统在服务器端即可完成收集，必要时，也可加入客户端数据收集

■收集的原始数据通常需要汇总以便快速分析其典型特征，大部分指标都应该以“分布”而非平均值来定义

◆例如Prometheus系统上histogram和summary类型的指标

◆这类指标有助于帮助用户分析数据的分布状态

■对于常见的SLI，定义标准化的模板有助于降低工作量

如何选择SLI

从众多指标中快速选择SLI的方法：VALET (Volume, Availability, Latency, Error和Ticket)

■ Volume（容量）

◆ 服务承诺的最大容量，例如一个应用集群的QPS、TPS、会话数及连接数等等，这些就是容量相关的SLI

◆ 对这些指标设定一个日常目标，就是日常的SLO，对大促期间设定一个目标，就是大促的SLO

■ Availability（可用性）

◆ 服务是否能够正常响应客户端请求

◆ 请求调用的非5xx类响应码的占比，就是衡量可用性的常用指标

■ Latency（延迟）

◆ 用于评估是否能够足够快地响应请求，通常用于评估每个请求是否能在规定的时间内完成

◆ 该指标与用户访问体验相关

◆ 时长通常符合正态分布，因而不应该使用平均值进行衡量，而要使用Histogram或Summary类型的指标

●同时也应考虑极端情况，例如404的响应时间短到会影响整体分布，或者个别延迟较大的请求会长到影响整体分布

■Errors（错误率）

- ◆常规错误，例如5xx，以及高频度的影响到用户体验的4xx响应
- ◆自定义状态码，包括对业务有损的状态码，例如热门商品的高缺货率、例如验证码的高错误率等

■Tickets（人工介入）

- ◆人工介入通常意味着低效
- ◆可以为某服务设定一个Tickets总的数量指标

错误预算

如何确保达成SLO？

■将容许的犯错空间转换为具体可行的指标（犯错的次数或标准），即错误预算（Error Budget）

■以具有冲击力的方式提示剩余的犯错机会

如何应用Error Budget？

■稳定性燃尽图

■故障定级

◆可以按错误预算在单次故障中消耗的比例进行定级

◆效用：借助于错误预算将故障定级量化

■稳定性共识机制

◆剩余错误预算充足或未尽之前，对问题要有充分的容忍度

◆剩余错误预算消耗过快或即将耗尽之前，SRE有权终止和拒绝任何线上变更

●避免“带病工作”，并等待下一个预算周期

●确保运维、产品和开发就此达成一致

■基于错误预算的告警

◆制订好告警收敛策略

◆基于错误预算进行告警

Error Budget-错误预算	单次消耗比例	故障等级
25, 000	比例 \leq 5%	P4
	5% < 比例 \leq 20%	P3
	20% < 比例 \leq 30%	P2
	30% < 比例 \leq 50%	P1
	50% < 比例	P0

如何衡量SLO的有效性

根据实际运行结果判定有效性，通常存在三个关键纬度

■ SLO达成情况：可用“达成（Met）”和“未达成（Missed）”标识

■ 人工介入程度（Toil）：泛指大量人工投入、重复、繁琐的低价值事务，可用投入程度高（High）和低（Low）标识

用户满意度（Customer Satisfaction）：可通过客服投诉、客户访谈或舆情监控等真实渠道获取，可用满意度高（High）和低（Low）来标识

总体处置策略有三种：
1、收紧SLO
2、放宽SLO
3、保持现状

SLOs	Toil	Customer satisfaction	Action
Met	Low	High	Choose to (a) relax release and deployment processes and increase velocity, or (b) step back from the engagement and focus engineering time on services that need more reliability.
Met	Low	Low	Tighten SLO.
Met	High	High	If alerting is generating false positives, reduce sensitivity. Otherwise, temporarily loosen the SLOs (or offload toil) and fix product and/or improve automated fault mitigation.
Met	High	Low	Tighten SLO.
Missed	Low	High	Loosen SLO.
Missed	Low	Low	Increase alerting sensitivity.
Missed	High	High	Loosen SLO.
Missed	High	Low	Offload toil and fix product and/or improve automated fault mitigation.

落地SLO 要考虑的因素

确定核心链路

■一般来说，分布式在线系统的各服务通常可划分为核心应用和非核心应用两大类

■核心应用及其强依赖项组成的调用关系即为核心链路

◆以电商系统为例，核心应用通常是指电商交易关键路径上的应用，例如首页、详情页、购物车、价格、优惠券、SKU和支付等

找出核心应用之后，还要确认这些应用的依赖项，核心应用彼此间的依赖构成“强依赖”，对核心应用的依赖即为弱依赖



落地SLO 要考虑的因素（2）

设定SLO的原则

- 核心应用的SLO要严格，而非核心应用的可以放宽
- 强依赖之间的核心应用，SLO要一致
- 弱依赖项中，核心应用对非核心的依赖，要有降级、熔断和限流等服务治理手段
- 错误预算策略中，任意核心应用的错误预算影响范围都是整个核心链路
 - ◆ 若某核心应用错误预算耗尽，原则上，整个链路都要暂停变更操作，直至问题完全解决

验证核心链路的SLO的常用方法

■全链路压测

- ◆判定容量目标能否达成，主要评估QPS和TPS，并确认扩容水位
- ◆判定在极端容量场景下，预设的限流、降级、熔断策略是否能正常触发

■混沌工程

- ◆主动模拟故障场景，测试线上应急机制，提前发现隐患
 - 模拟机房断电，测试双活机房或备用机制的流量切换
 - 模拟网络丢包或流量满载、磁盘写满、CPU满载、服务器重启、接口延迟、返回异常、线程池满载等
- ◆会对线上业务造成影响
 - 必须事先在隔离环境中反复演练
 - 在模拟的故障超出预估时要能快速进行隔离，快速恢复业务
- ◆是SRE稳定性体系建设的高级阶段，要在服务治理、全链路压测、链路跟踪、监控告警和运维自动化建设完善后考虑