Site Reliability Engineering

cre.page.link/art-of-slos

Feedback? @GoogleSRE

Google's core practice for balancing Velocity and Reliability.

Terminology

- **MTTD** (Mean Time To Detection) - how long it takes to detect and notify that a risk has occurred.
- **MTTR** (Mean Time To Resolution) - how long it takes to fix the incident once detected.
- **MTBF** (Mean Time Between Failures) - estimated frequency between instances of the risk.

Reliability

The **most important feature** of any system is its **reliability**. A service is **reliable** if it performs as its users expect.

Reliable enough: Acknowledging that a *specific quantity of unreliability is acceptable* provides a budget for failure that can be spent on developing and launching new features.

Improve reliability by reducing: Time to detection | Time to resolution | Impact of outages | Frequency of outages.

Happiness test

Services need SLO targets that capture the performance and availability levels that, if barely met, would keep a typical customer happy.

Service Level Agreement (SLA)

An **external** promise that comes with consequences.

An SLA describes the minimum level of service you promise to provide and what happens otherwise.

Service Level Indicator (SLI)

A **quantifiable** measure of the reliability of your service from your users' perspective.

Good SLIs are a measurable analogy for user happiness.

Our SLI menu provides guidelines for the types of SLIs that may be used when measuring a given CUJ

SLI Menu To track the reliability of a **request response** interaction in a user journey, measure: availability, latency, and quality. For **data processing**: freshness, coverage, correctness and throughput. For **storage**: throughput and latency.

Service Level Objectives (SLO)

Sets the **target** for an SLI over a period of time.

An SLO is a fundamental tool for prioritizing reliability versus other features, and communicating the expectations of a service through objective data.

An SLO is an **internal** promise to meet customer expectations. **Being out of SLO *must* have *consequences*** which redirect engineering effort towards making reliability improvements.

Error budget

An SLO implies an **acceptable level** of unreliability.

This acceptable rate of failure is a ***budget*** that can be actively spent—if it is not consumed by service downtime—on risky development activities activities like releasing new features, making configuration changes, A/B testing, etc.

Setting SLOs and SLIs

SLIs have a consistent format and range from 0-100%.

The SLI Equation

$$\text{SLI} = \left(\frac{\textbf{good}\ \text{events}}{\text{valid events}} \right) \times 100\%$$

*The proportion of **valid events** that were **good**.*

For each **critical user journey** ranked by **business impact:**

1. Choose an **SLI specification** from the menu
2. Specify detailed **SLI implementation**
3. Validate that it doesn't have **coverage gaps**
4. Set **SLOs** based on **past performance** or **business need**

You should choose 3-5 SLIs per user journey.

SLI implementation includes: *event* + success criteria + where/how you record the SLI.
SLO should include: target and a measurement window

Measuring SLIs sources: Log processing, Application Server Metrics, Front-end Infrastructure Metrics, Synthetic Clients (Probers) or Data, Client Instrumentation

Outage Math

Time before 30-day error budget is exhausted						
Error Rate/ Reliability level	**99%**	**99.5%**	**99.9%**	**99.95%**	**99.99%**	**99.999%**
100%	7.2 h	3.6h	43.2m	21.6m	4.32m	25.9s
10%	3d	7.2h	7.2h	3.6h	43.2m	4.32m
1%		15d	3d	36h	7.2h	43.2m
0.1%	All month			15d	3d	7.2h
0.05%					6d	14.4h

V2020.06