



# Site Reliability Engineering

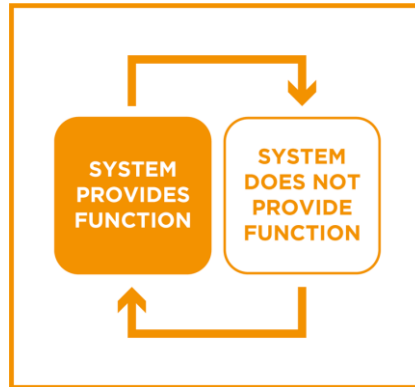
SWISSED20

John Lunney & Sebastian Klages  
September, 2020



## Definition:

**Systems reliability is the probability, that a system will provides its function uninterrupted over a defined time span.**



**Why is systems  
reliability so  
important?**

**The worth of a system is  
determined by its functions.**

**The system is worthless, if  
the functions are  
unavailable.**

- Service Level Agreement
- Continuous Deployment

Google

# Site Reliability Engineering

SWISSED20

John Lunney & Sebastian Klages  
September, 2020



# The SRE approach to reliability

Use data to guide decision-making.

Treat operations like a software engineering problem:

- Hire people motivated and capable to write automation.
- Use software to accomplish tasks normally done by sysadmins.
- Design more reliable and operable service architectures from the start.



# What do SRE teams do?

- ▶ Site Reliability Engineers develop solutions to design, build, and run large-scale systems **scalably, reliably, and efficiently.**
- ▶ We **guide system architecture** by operating at the intersection of software development and systems engineering.

- ▶ SRE is a job function, a mindset, and a set of **engineering approaches** to running better production systems.
- ▶ We approach our work with a spirit of constructive pessimism: we **hope for the best, but plan for the worst.**

“

100% is the wrong reliability target for basically everything.”

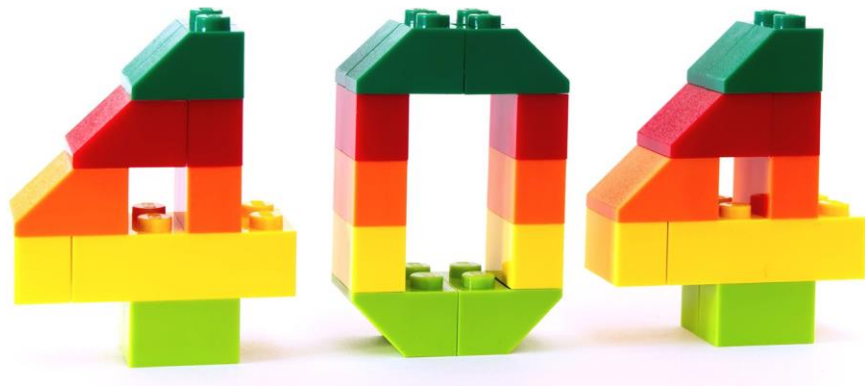
Benjamin Treynor Sloss, Vice President of 24x7 Engineering,  
Google





# Error budgets

- Product management & SRE define an **availability target**.
  - "how many nines?"
  - 99, 99.9, 99.99, 99.999
- 100% minus availability target is a “budget of unreliability” (or the **error budget**).
- Monitoring measures **actual uptime**.
- Control loop for utilizing budget!



[Public Domain Image](#)

# Benefits of error budgets

- ▶ **Common incentive for devs and SREs**

Find the right balance between innovation and reliability

- ▶ **Dev team can manage the risk themselves**

They decide how to spend their error budget

- ▶ **Unrealistic reliability goals become unattractive**

Such goals dampen the velocity of innovation

- ▶ **Dev team becomes self-policing**

The error budget is a valuable resource for them

- ▶ **Shared responsibility for system uptime**

Infrastructure failures eat into the devs' error budget

- Availability in %
- From “Plan, Analyze, Implement” to continuous deployment



# Site Reliability Engineering



SWISSED20

John Lunney & Sebastian Klages  
September, 2020



**OPERATIONAL ANALYSIS - OPERATIONAL FAILURE MODES**  
 - Operational Failure Modes

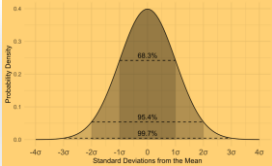
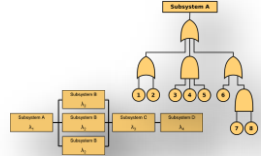
**QUALITATIVE RELIABILITY ANALYSIS - FUNCTIONAL FMECA**  
 - Functional FMECA  
 - Severity  
 - Probability

**QUANTITATIVE RELIABILITY ANALYSIS**  
 - RBD  
 - FTA

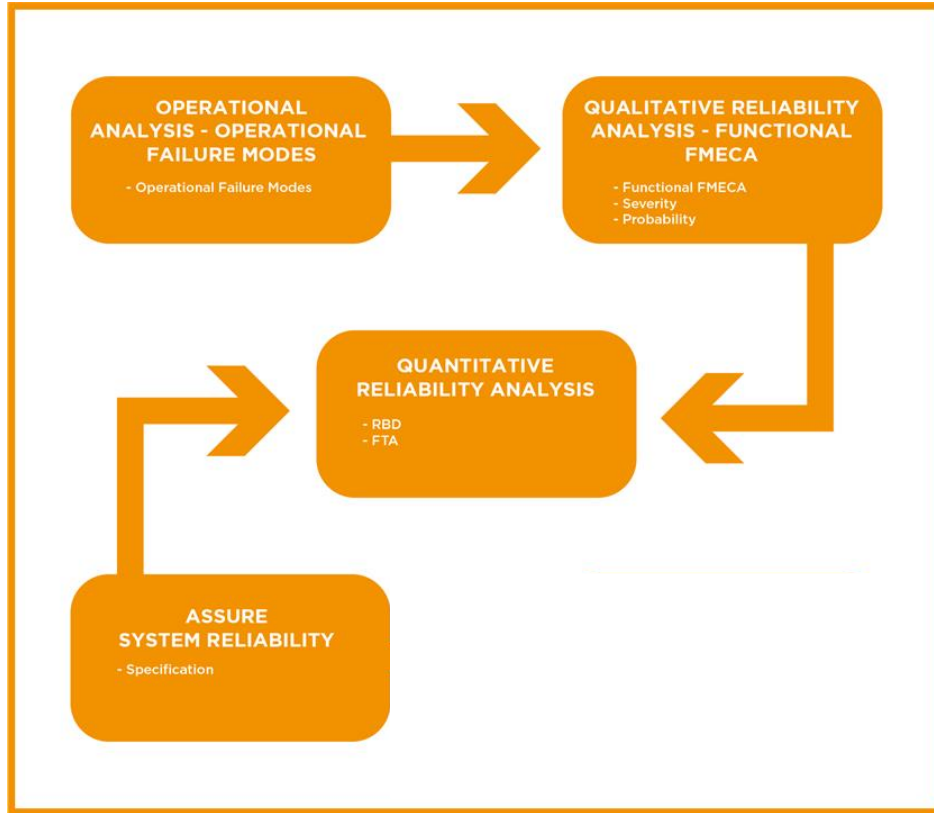
**ASSURE SYSTEM RELIABILITY**  
 - Specification

**MONITOR SYSTEMS RELIABILITY**

Process/FCA	Reference	History
Process/FCA		
FMEA Name	Intermittent - BI	Intermittent - BI
Project Name	Intermittent - BI	Intermittent - BI
Start Date/End Date	Intermittent - BI	Intermittent - BI
Failure Mode and Effect Analysis		
Product	Intermittent - BI	Intermittent - BI
Process	Intermittent - BI	Intermittent - BI
FMEA Type	Intermittent - BI	Intermittent - BI
Version	Intermittent - BI	Intermittent - BI
Created By	Intermittent - BI	Intermittent - BI
Reviewed By	Intermittent - BI	Intermittent - BI
Approved By	Intermittent - BI	Intermittent - BI
Created Date	Intermittent - BI	Intermittent - BI
Reviewed Date	Intermittent - BI	Intermittent - BI
Approved Date	Intermittent - BI	Intermittent - BI



# RAM Analysis for a System Archetype



Outlook for industrial applications:

- Increasing frequency of system updates (obsolescence, SW patches, etc.)
- Functionality is increasingly determined by IT services (the IT service lifecycle increasingly influences the system life cycle).

