9 Things you should Do When starting to use SLOs

Sal Furino

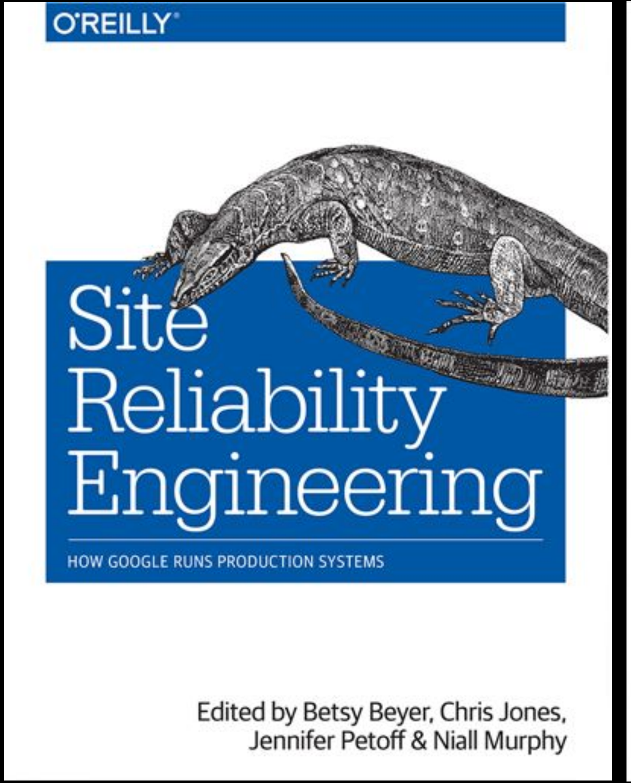
Customer Reliability Engineer

@sfurino

He/Him

/in/salvatore-furino/

Who has read these?

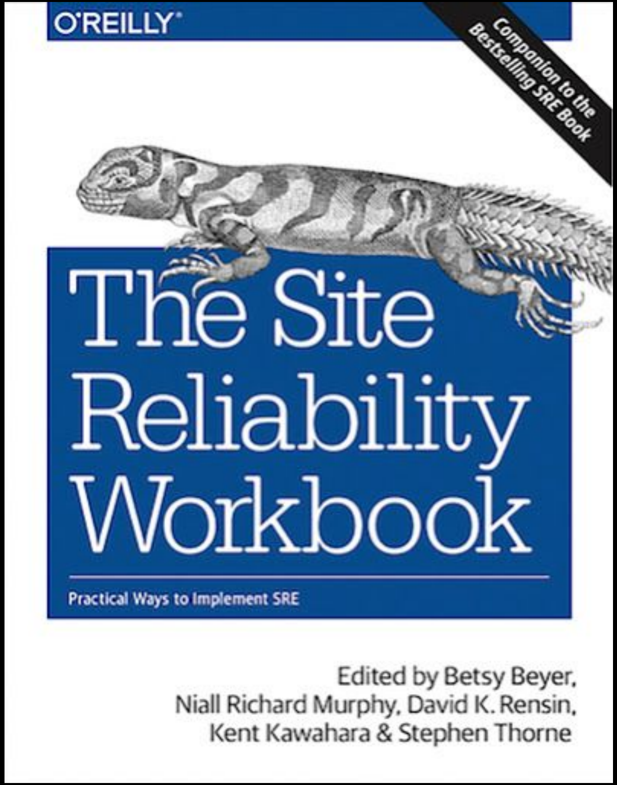


O'REILLY

Site Reliability Engineering

HOW GOOGLE RUNS PRODUCTION SYSTEMS

Edited by Betsy Beyer, Chris Jones,
Jennifer Petoff & Niall Murphy



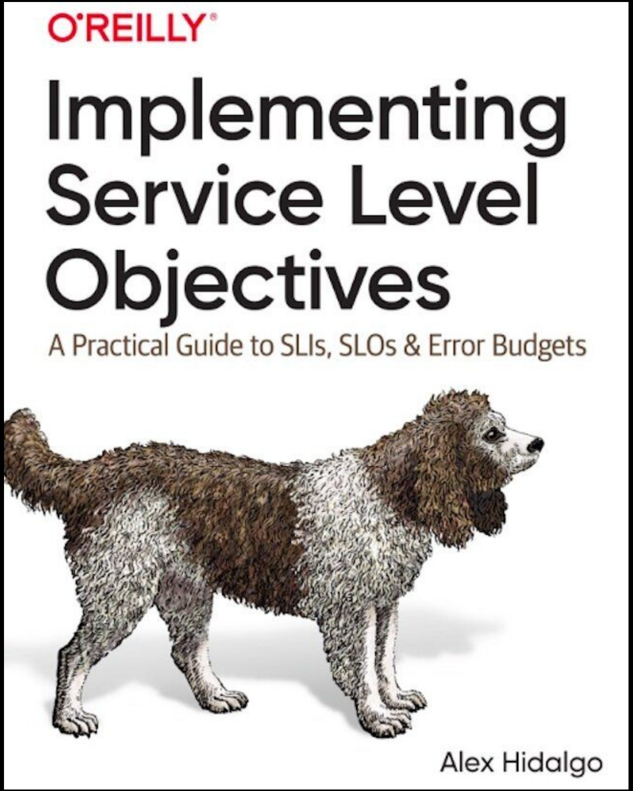
O'REILLY

Companion to the Bestselling SRE Book

The Site Reliability Workbook

Practical Ways to Implement SRE

Edited by Betsy Beyer,
Niall Richard Murphy, David K. Rensin,
Kent Kawahara & Stephen Thorne

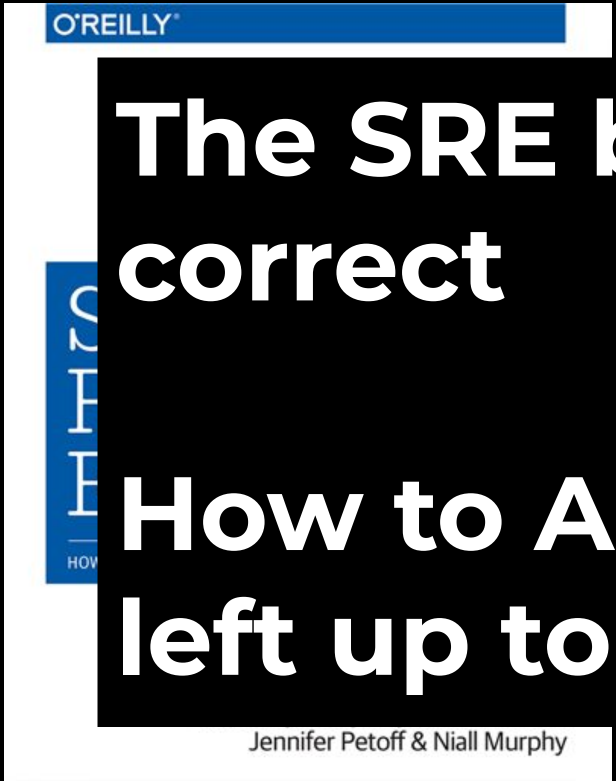


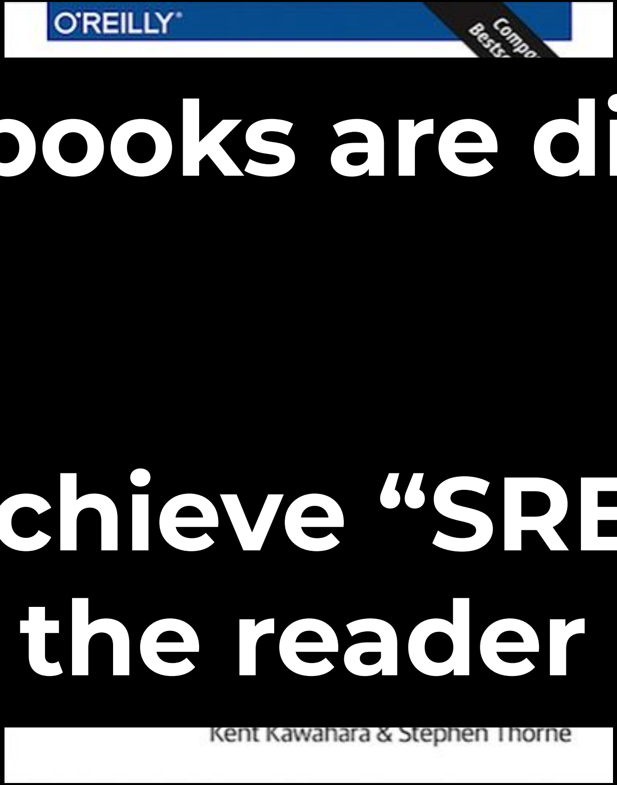
O'REILLY

Implementing Service Level Objectives

A Practical Guide to SLIs, SLOs & Error Budgets

Alex Hidalgo





The SRE books are directionly correct

How to Achieve “SRE” still left up to the reader

SLOs are Hard to get Right

SLOs are Hard to get Right

Common questions

- Which metrics best instrument the service?
- What is a reasonable objective for this service?
- How should the error budget be used?
- Who else needs to understand these signals?

What are the 9 things?

Understanding the systems

- Measure meaningful things
- Success rates > Error rates
- Raw data for SLOs

Understanding expectations of system be

- Examine historical performance
- Define and take actions
- Different time windows for different folks

Recognizing when a system doesn’t match those expectations

- Enrich your dashboards with context
- Document your SLOs
- Adopt open standards

Understanding Systems

Measure Meaningful Things

Shine
Light
On…

Don’t Measure

CPU Utilization

Disk IO

What Matters to Users

Customer Journeys

Interactions

Measure Meaningful Things "Fortress Bank"

Industry:

Finance - Card Processor

Motivation:

Contractual and regulatory impacts

Goal:

Insure reliability for quicker person to person payments

Luca Colapinto on Unsplash

Measure Meaningful Things “Fortress Bank”

Journey:

As a user, I want to send money to quickly to a 3rd party

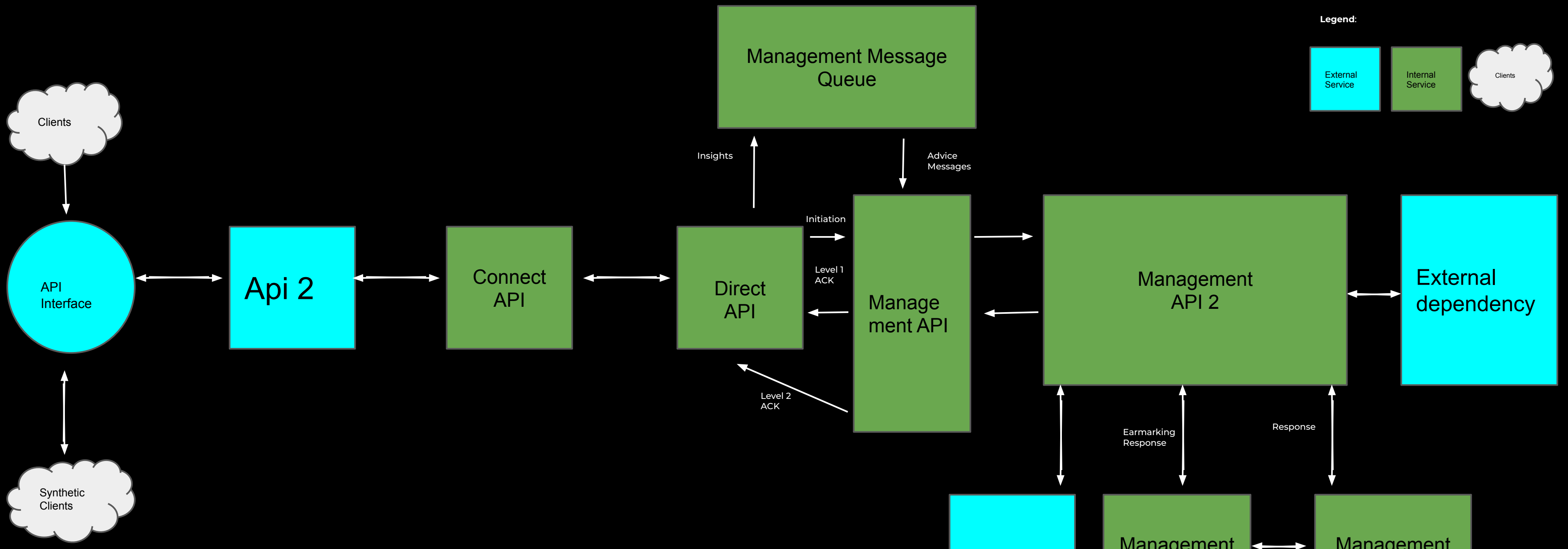
Expectation:

I should receive notification that the payment is being processed

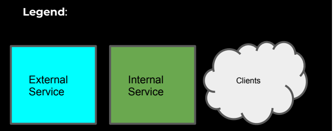
Dependencies:

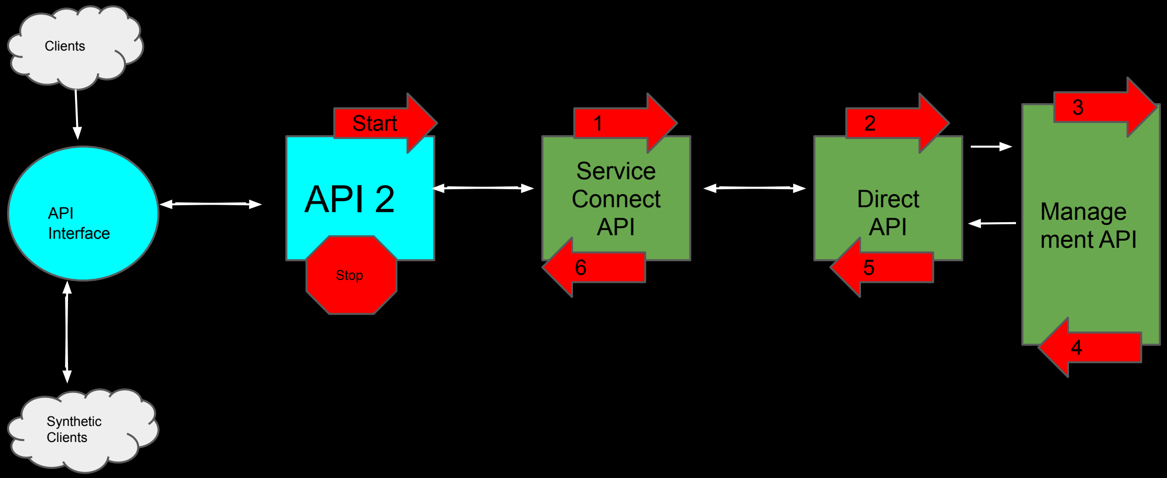
3rd party application gateway, clearing house, screening

Measure Meaningful Things "Fortress Bank"

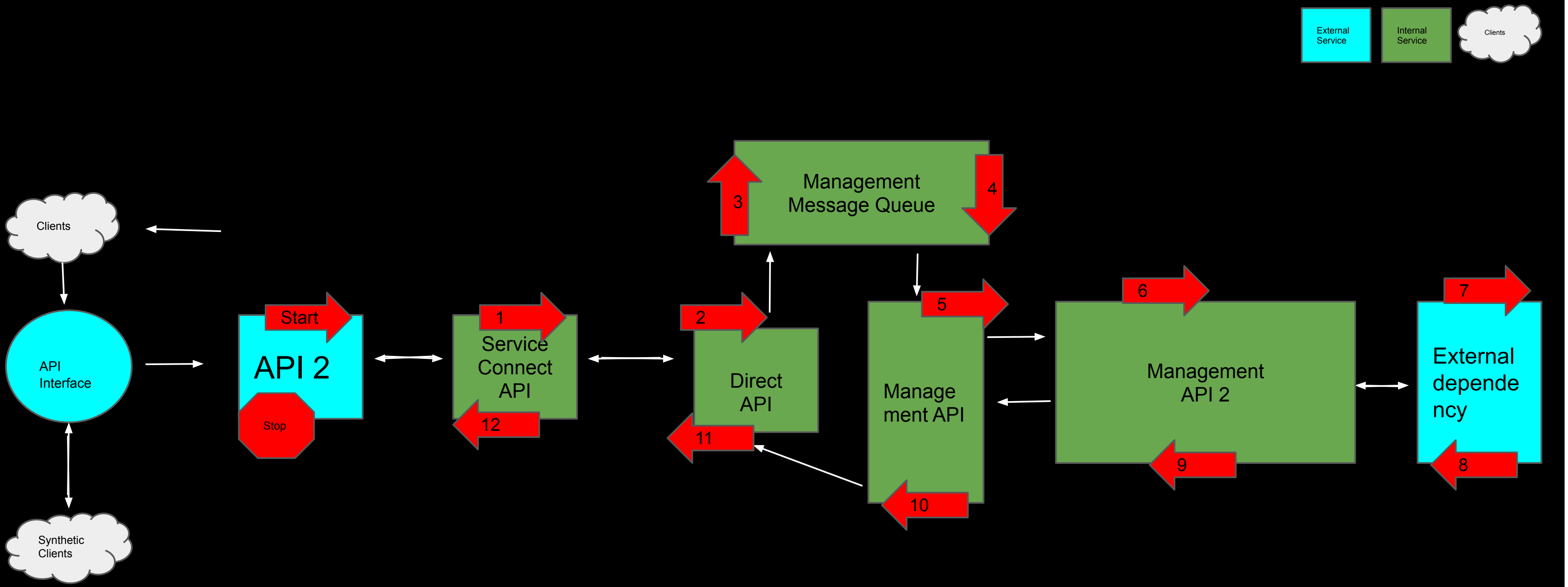


Measure Meaningful Things
"Fortress Bank" - Level 1 Ack





Measure Meaningful Things
“Fortress Bank” - Level 2 ack



Measure Success Rate

Don’t Define

Bad Behavior

Maybe Easier

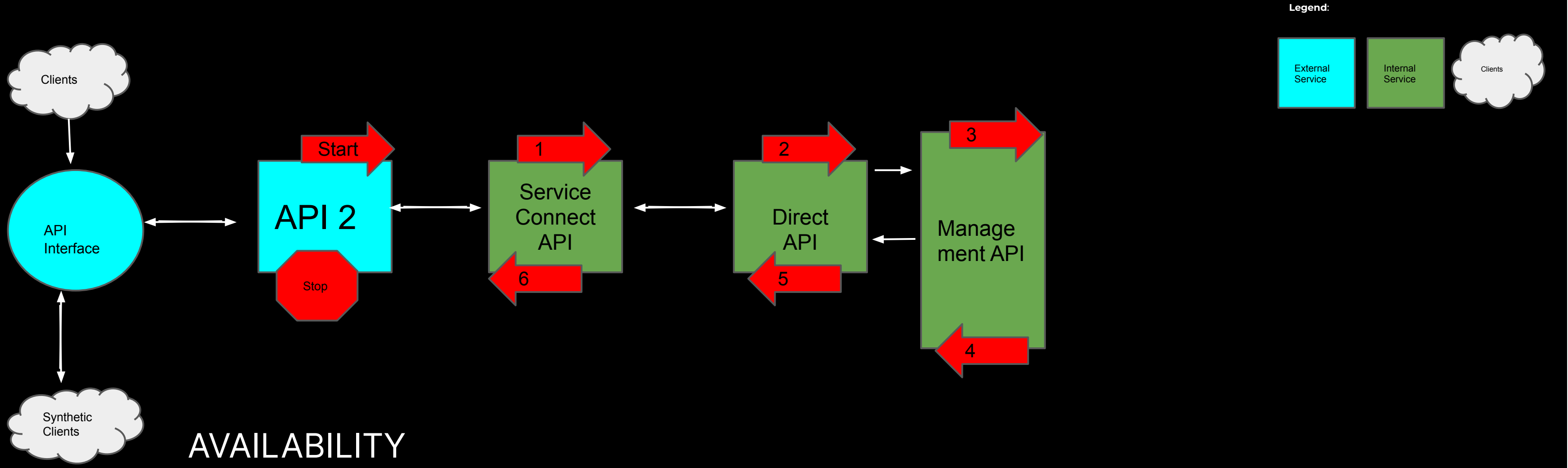
Has a limited scope

Known undesired problems

Define Success

Everything else is uncertain

Measure Success Rate
“Fortress Bank” - Level 1 Ack



AVAILABILITY

The proportion of requests to Service Connect API that have *2XX*, *3XX* or *4XX* status measured at the *Load Balancer*

Provide Raw Data for SLOs

Don’t Down Sample

avg(query)

percentile(95, query)

sum(query)

Simple Queries

Provide High Fidelity Information to Your SLOs

Provide Raw Data for SLOs
Streams “R” Us

Industry:

Entertainment - Streaming Video

Motivation:

Contractual requirement

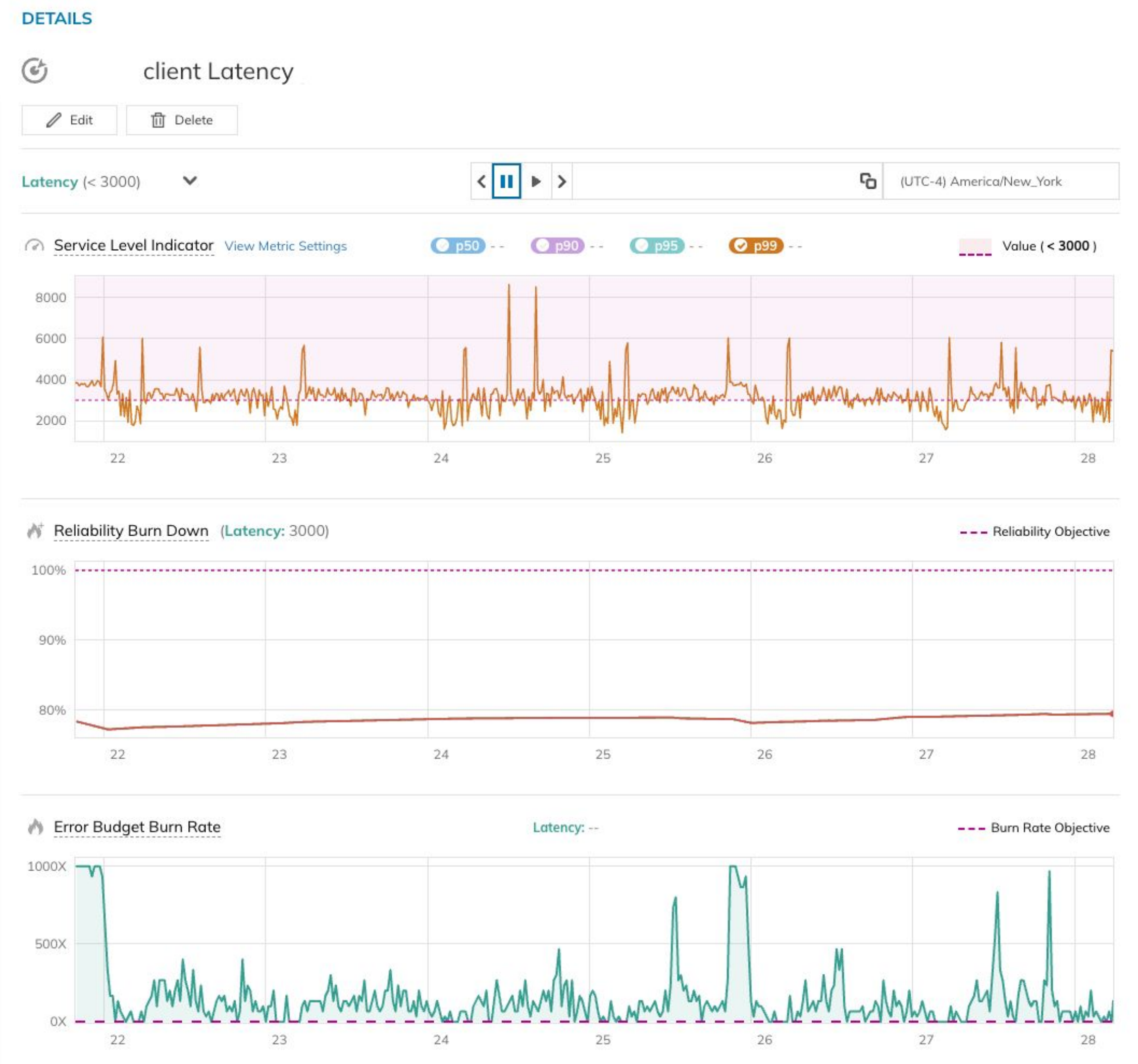
Competitive requirements

Goal:

Insure low latency streaming video to consumers

pascal Stöckmann on Unsplash

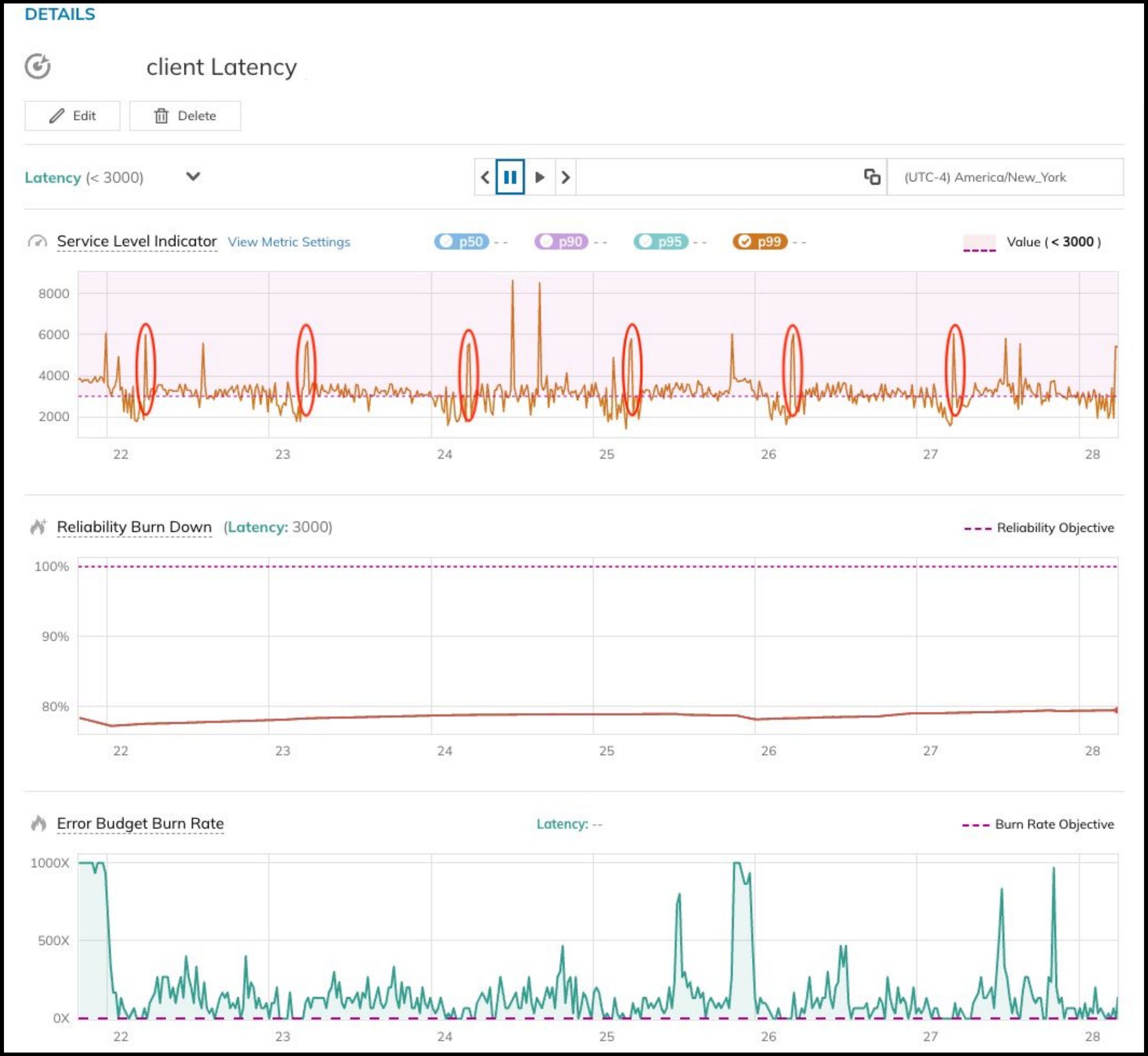
Streams “R” Us



DETAILS

client Latency

Streams “R” Us



DETAILS

client Latency

Latency (< 3000)

(UTC-4) America/New_York

Service Level Indicator

Reliability Burn Down (Latency: 3000)

Error Budget Burn Rate

Latency: --

Understanding Expectations of Systems Behavior

Look Back to Look Forward

Don’t say

“All Our Services Need

To Have 99.999%

Or Better Reliability”

Look Back to Look Forward

Ask your teams:

- How reliable does the service need to be to be competitive?
- How much un-reliability can we accept with the current design?
- How much are we willing to spend to achieve those goals?

Look Back to Look Forward

Don’t say

“All Our Services Need

To Have 99.999%

Or Better Reliability”

Do

Benchmark Achievable

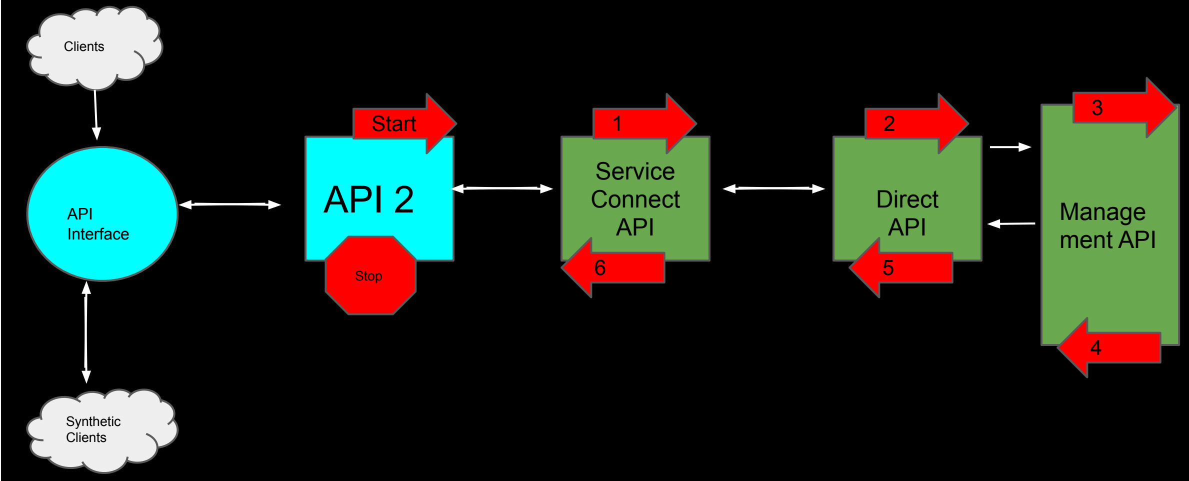
Reliability targets

Capture SLI Activity and

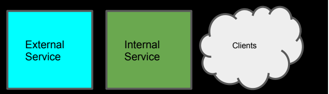
Calculate Error Budget

After a recent incident

Look Back to Look Forward
“Fortress Bank” - Level 1 Ack



Legend:

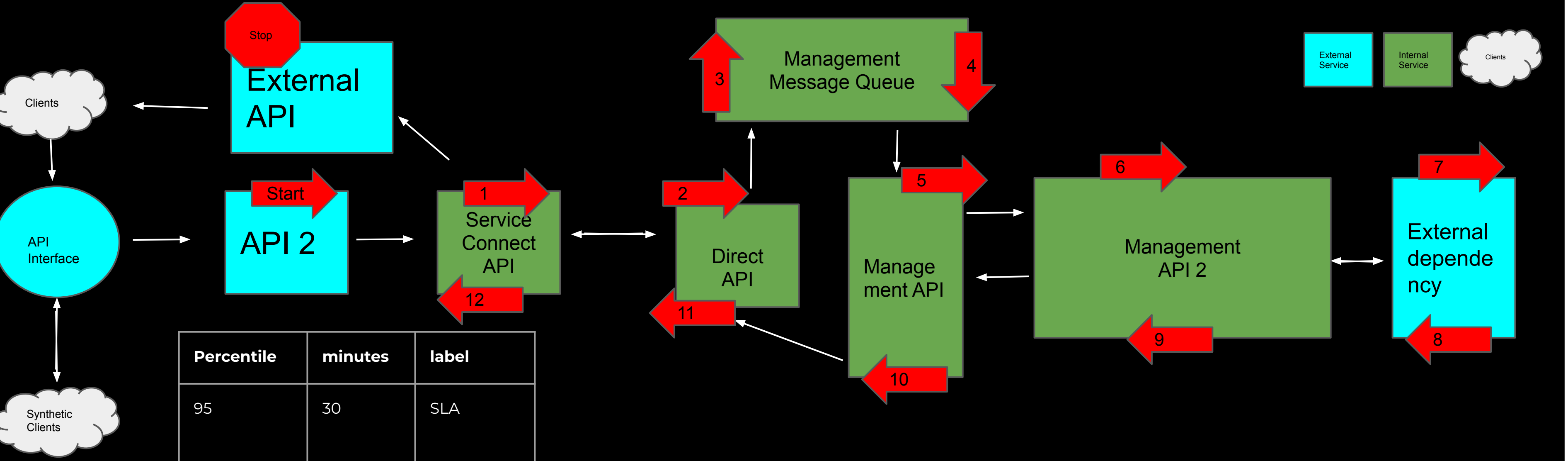


AVAILABILITY

The proportion of requests to
 Service Connect API
that have *2XX*, *3XX* or *4XX* status
measured at the *Load Balancer*

Percentile	ms	label
99.9	2000	SLA
99	1700	Something is wrong
95	1500	Starting to lag
90	1000	Okay
50	100	lightning

Look Back to Look Forward “Fortress Bank” - Level 2 ack



Percentile	minutes	label
95	30	SLA
92	25	Something is wrong
90	15	Okay
50	5	lightning

Observibility Without Action is Just Storage

Don't

Set and forget

Dashboards for the sake of

Dashboards

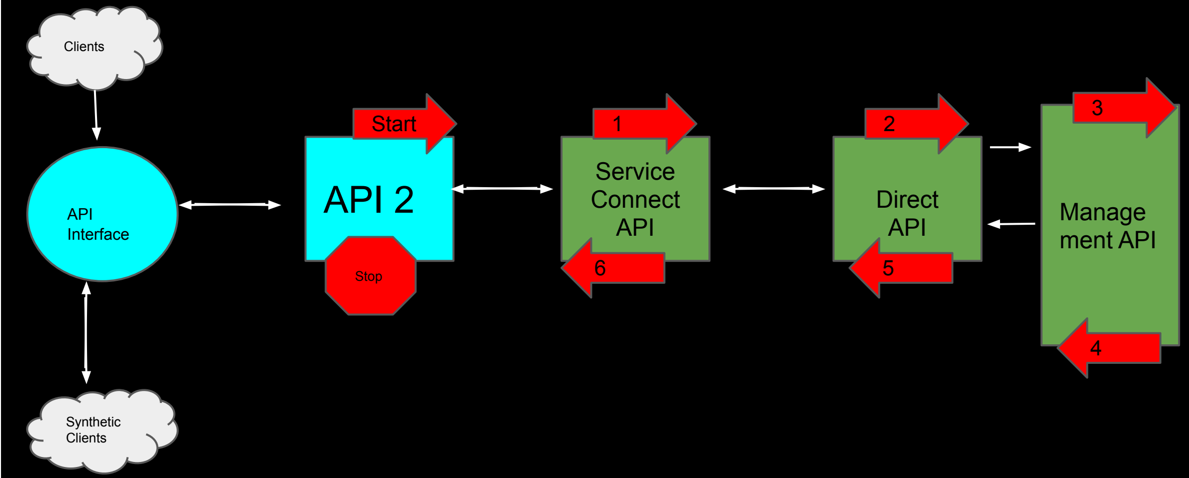
Do

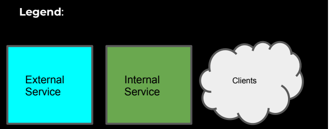
Codify Your Reliability Targets

Set Automated Action

Alert when necessary

Look Back to Look Forward “Fortress Bank” - Level 1 Ack





AVAILABILITY

The proportion of requests to Service Connect API that have *2XX*, *3XX* or *4XX* status measured at the *Load Balancer*

Percentile	**ms**	**label**	**Action**
99.9	2000	SLA	Full Outage Crisis Mode
99	1700	Something is wrong	Page on Call
95	1500	Starting to lag	Service Now Ticket
90	1000	Okay	Warn / Send Slack message
50	100	lightning	Steady State N/A

Different Time Windows for Different Folks

Don’t say

“SLOs Are Only For Operations and Site Reliability Engineers”

Do

Different Groups Take Different Actions to Reliability

Same SLIs But Different Time Windows

Different Time Windows for Different Folks

Don’t say

“SLOs Are Only For Operations and Site Reliability Engineers”

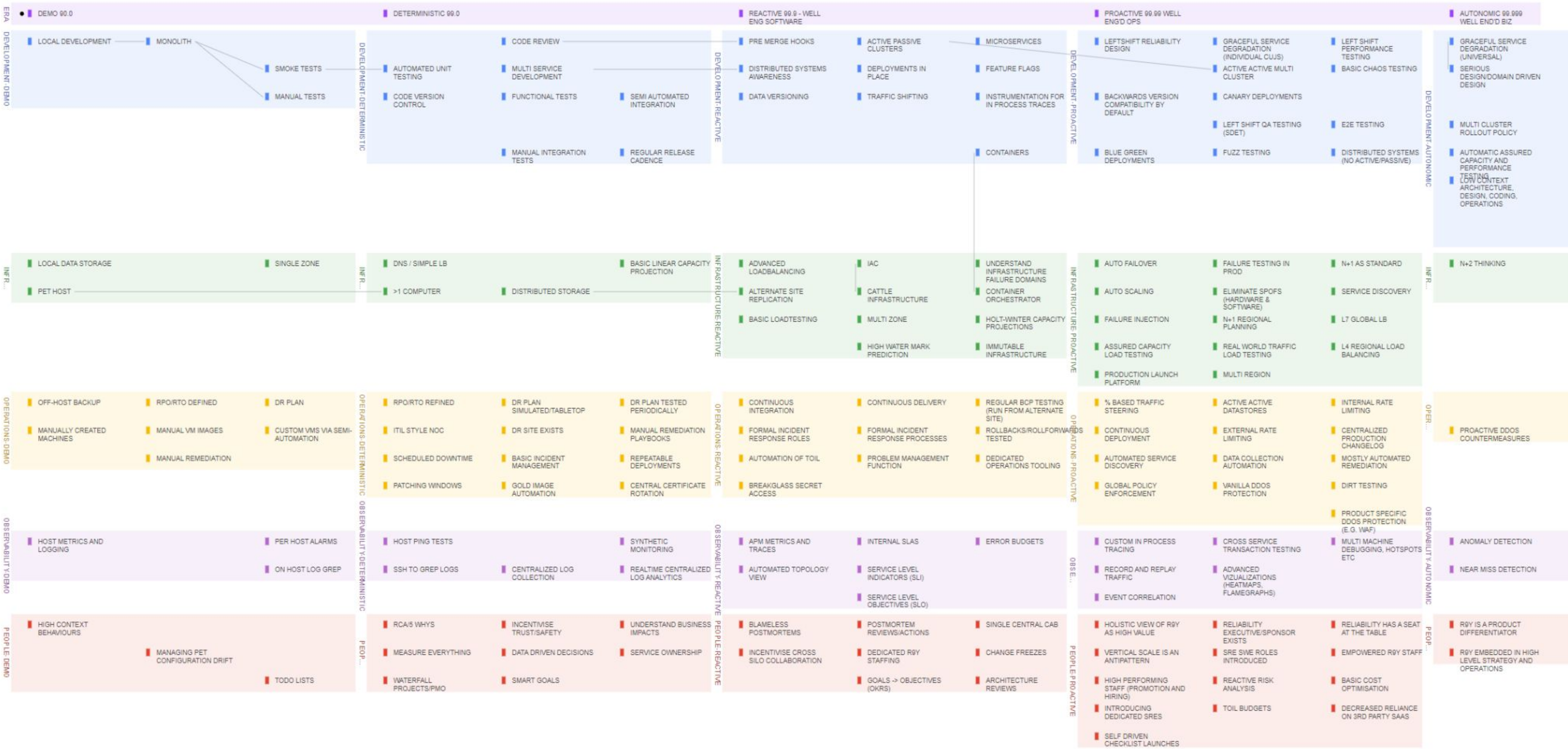
Do

24h/48h –SRE/OPs

14D/2W – Dev teams

Monthly / Quarterly – Arch / Leadership

R9Y.dev

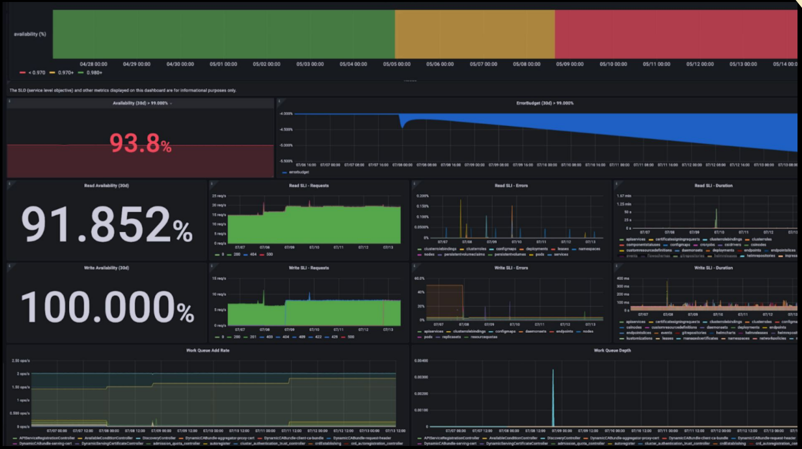


ERA	DEMO 90.0	DETERMINISTIC 99.0	REACTIVE 99.9 - WELL ENG SOFTWARE	PROACTIVE 99.99 WELL ENG'D OPS	AUTONOMIC 99.999 WELL END'D BIZ
DEVELOPMENT-DEMO	LOCAL DEVELOPMENT MONOLITH SMOKE TESTS MANUAL TESTS	DEVELOPMENT-DETERMINISTIC AUTOMATED UNIT TESTING CODE VERSION CONTROL CODE REVIEW MULTI SERVICE DEVELOPMENT FUNCTIONAL TESTS MANUAL INTEGRATION TESTS SEMI AUTOMATED INTEGRATION REGULAR RELEASE CADENCE	DEVELOPMENT-REACTIVE PRE MERGE HOOKS DISTRIBUTED SYSTEMS AWARENESS DATA VERSIONING ACTIVE PASSIVE CLUSTERS DEPLOYMENTS IN PLACE TRAFFIC SHIFTING MICROSERVICES FEATURE FLAGS INSTRUMENTATION FOR IN PROCESS TRACES CONTAINERS	DEVELOPMENT-PROACTIVE LEFTSHIFT RELIABILITY DESIGN BACKWARDS VERSION COMPATIBILITY BY DEFAULT BLUE GREEN DEPLOYMENTS GRACEFUL SERVICE DEGRADATION (INDIVIDUAL CUJS) ACTIVE ACTIVE MULTI CLUSTER CANARY DEPLOYMENTS LEFT SHIFT QA TESTING (SDET) FUZZ TESTING LEFT SHIFT PERFORMANCE TESTING BASIC CHAOS TESTING E2E TESTING DISTRIBUTED SYSTEMS (NO ACTIVE/PASSIVE)	DEVELOPMENT-AUTONOMIC GRACEFUL SERVICE DEGRADATION (UNIVERSAL) SERIOUS DESIGN/DOMAIN DRIVEN DESIGN MULTI CLUSTER ROLLOUT POLICY AUTOMATIC ASSURED CAPACITY AND PERFORMANCE TESTING LOW CONTEXT ARCHITECTURE, DESIGN, CODING, OPERATIONS
INFR...	LOCAL DATA STORAGE PET HOST SINGLE ZONE	INFR... DNS / SIMPLE LB >1 COMPUTER DISTRIBUTED STORAGE BASIC LINEAR CAPACITY PROJECTION	INFRASTRUCTURE-REACTIVE ADVANCED LOADBALANCING ALTERNATE SITE REPLICATION BASIC LOADTESTING IAC CATTLE INFRASTRUCTURE MULTI ZONE HIGH WATER MARK PREDICTION UNDERSTAND INFRASTRUCTURE FAILURE DOMAINS CONTAINER ORCHESTRATOR HOLT-WINTER CAPACITY PROJECTIONS IMMUTABLE INFRASTRUCTURE	INFRASTRUCTURE-PROACTIVE AUTO FAILOVER AUTO SCALING FAILURE INJECTION ASSURED CAPACITY LOAD TESTING PRODUCTION LAUNCH PLATFORM FAILURE TESTING IN PROD ELIMINATE SPOFS (HARDWARE & SOFTWARE) N+1 REGIONAL PLANNING REAL WORLD TRAFFIC LOAD TESTING MULTI REGION N+1 AS STANDARD SERVICE DISCOVERY L7 GLOBAL LB L4 REGIONAL LOAD BALANCING	INFR... N+2 THINKING
OPERATIONS-DEMO	OFF-HOST BACKUP MANUALLY CREATED MACHINES RPO/RTO DEFINED MANUAL VM IMAGES MANUAL REMEDIATION DR PLAN CUSTOM VMS VIA SEMI-AUTOMATION	OPERATIONS-DETERMINISTIC RPO/RTO REFINED ITIL STYLE NOC SCHEDULED DOWNTIME PATCHING WINDOWS DR PLAN SIMULATED/TABLETOP DR SITE EXISTS BASIC INCIDENT MANAGEMENT GOLD IMAGE AUTOMATION DR PLAN TESTED PERIODICALLY MANUAL REMEDIATION PLAYBOOKS REPEATABLE DEPLOYMENTS CENTRAL CERTIFICATE ROTATION	OPERATIONS-REACTIVE CONTINUOUS INTEGRATION FORMAL INCIDENT RESPONSE ROLES AUTOMATION OF TOIL BREAKGLASS SECRET ACCESS CONTINUOUS DELIVERY FORMAL INCIDENT RESPONSE PROCESSES PROBLEM MANAGEMENT FUNCTION REGULAR BCP TESTING (RUN FROM ALTERNATE SITE) ROLLBACKS/ROLLFORWARDS TESTED DEDICATED OPERATIONS TOOLING	OPERATIONS-PROACTIVE % BASED TRAFFIC STEERING CONTINUOUS DEPLOYMENT AUTOMATED SERVICE DISCOVERY GLOBAL POLICY ENFORCEMENT ACTIVE ACTIVE DATASTORES EXTERNAL RATE LIMITING DATA COLLECTION AUTOMATION VANILLA DDOS PROTECTION INTERNAL RATE LIMITING CENTRALIZED PRODUCTION CHANGELOG MOSTLY AUTOMATED REMEDIATION DIRT TESTING PRODUCT SPECIFIC DDOS PROTECTION (E.G. WAF)	OPER... PROACTIVE DDOS COUNTERMEASURES
OBSERVABILITY-DEMO	HOST METRICS AND LOGGING PER HOST ALARMS ON HOST LOG GREP	OBSERVABILITY-DETERMINISTIC HOST PING TESTS SSH TO GREP LOGS CENTRALIZED LOG COLLECTION SYNTHETIC MONITORING REALTIME CENTRALIZED LOG ANALYTICS	OBSERVABILITY-REACTIVE APM METRICS AND TRACES AUTOMATED TOPOLOGY VIEW INTERNAL SLAS SERVICE LEVEL INDICATORS (SLI) SERVICE LEVEL OBJECTIVES (SLO) ERROR BUDGETS	OBSE... CUSTOM IN PROCESS TRACING RECORD AND REPLAY TRAFFIC EVENT CORRELATION CROSS SERVICE TRANSACTION TESTING ADVANCED VIZUALIZATIONS (HEATMAPS, FLAMEGRAPHS) MULTI MACHINE DEBUGGING, HOTSPOTS ETC	OBSERVABILITY-AUTONOMIC ANOMALY DETECTION NEAR MISS DETECTION
PEOPLE-DEMO	HIGH CONTEXT BEHAVIOURS MANAGING PET CONFIGURATION DRIFT TODO LISTS	PEOP... RCA/5 WHYS MEASURE EVERYTHING WATERFALL PROJECTS/PMO INCENTIVISE TRUST/SAFETY DATA DRIVEN DECISIONS SMART GOALS UNDERSTAND BUSINESS IMPACTS SERVICE OWNERSHIP	PEOPLE-REACTIVE BLAMELESS POSTMORTEMS INCENTIVISE CROSS SILO COLLABORATION POSTMORTEM REVIEWS/ACTIONS DEDICATED R9Y STAFFING GOALS -> OBJECTIVES (OKRS) SINGLE CENTRAL CAB CHANGE FREEZES ARCHITECTURE REVIEWS	PEOPLE-PROACTIVE HOLISTIC VIEW OF R9Y AS HIGH VALUE VERTICAL SCALE IS AN ANTIPATTERN HIGH PERFORMING STAFF (PROMOTION AND HIRING) INTRODUCING DEDICATED SRES SELF DRIVEN CHECKLIST LAUNCHES RELIABILITY EXECUTIVE/SPONSOR EXISTS SRE SWE ROLES INTRODUCED REACTIVE RISK ANALYSIS TOIL BUDGETS RELIABILITY HAS A SEAT AT THE TABLE EMPOWERED R9Y STAFF BASIC COST OPTIMISATION DECREASED RELIANCE ON 3RD PARTY SAAS	PEOP... R9Y IS A PRODUCT DIFFERENTIATOR R9Y EMBEDDED IN HIGH LEVEL STRATEGY AND OPERATIONS

Recognizing when a system doesn’t match those expectations

Enrich Dashboards with Contextual Information

Don’t



The SLO (service level objective) and other metrics displayed on this dashboard are for informational purposes only.

Do

Annotations

Badges

Dependency Mapping

Document your SLOs

Don’t say

“What is This Service?”

“Who’s Responsible For it?”

“Where is it Deployed?”

“How Do We Fix it?”

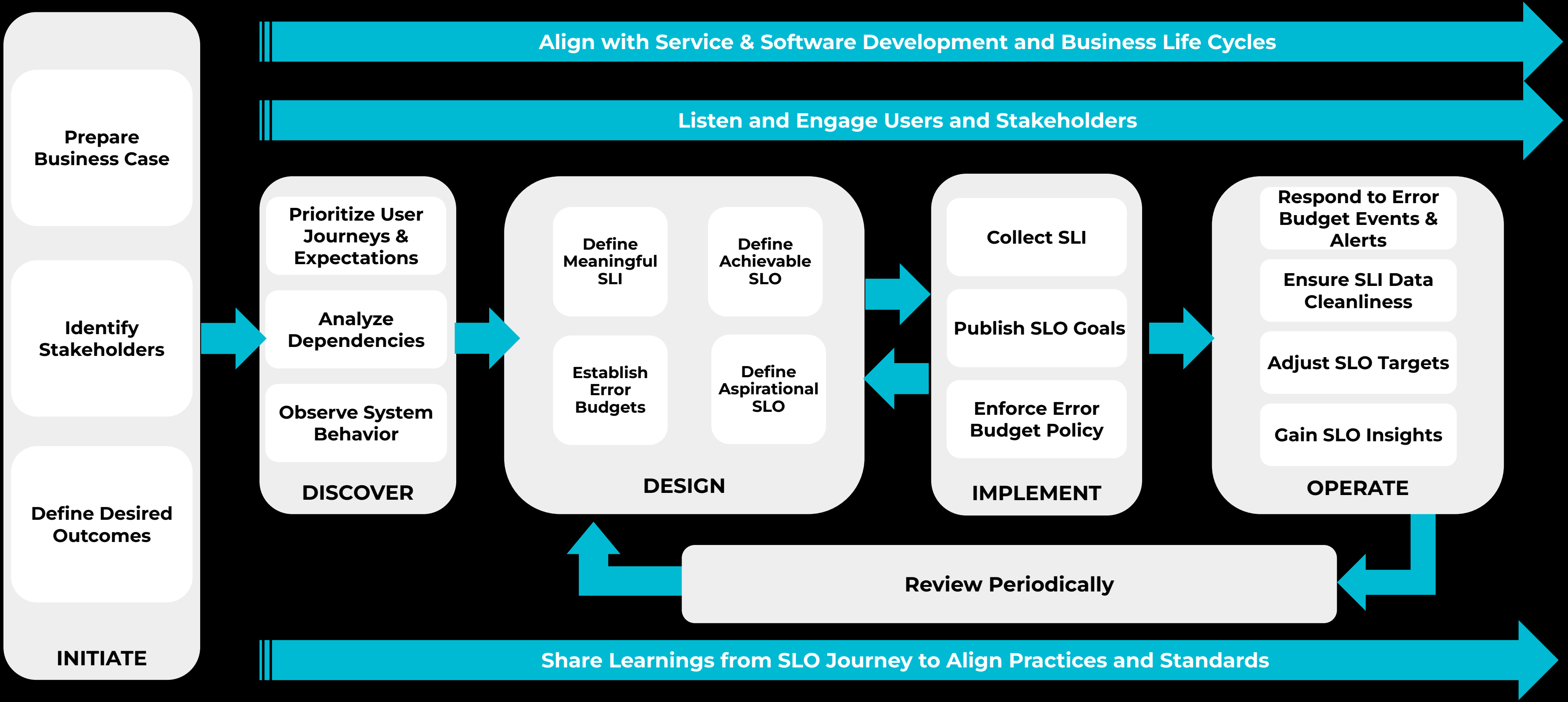
“Why is This Metric Important?”

Do

Document the WWWWHW
For the SLOs Existence

Track and Version With
Your SLO Definitions

SLODLC



Align with Service & Software Development and Business Life Cycles

Listen and Engage Users and Stakeholders

Prepare Business Case

Identify Stakeholders

Define Desired Outcomes

INITIATE

Prioritize User Journeys & Expectations

Analyze Dependencies

Observe System Behavior

DISCOVER

Define Meaningful SLI

Define Achievable SLO

Establish Error Budgets

Define Aspirational SLO

DESIGN

Collect SLI

Publish SLO Goals

Enforce Error Budget Policy

IMPLEMENT

Respond to Error Budget Events & Alerts

Ensure SLI Data Cleanliness

Adjust SLO Targets

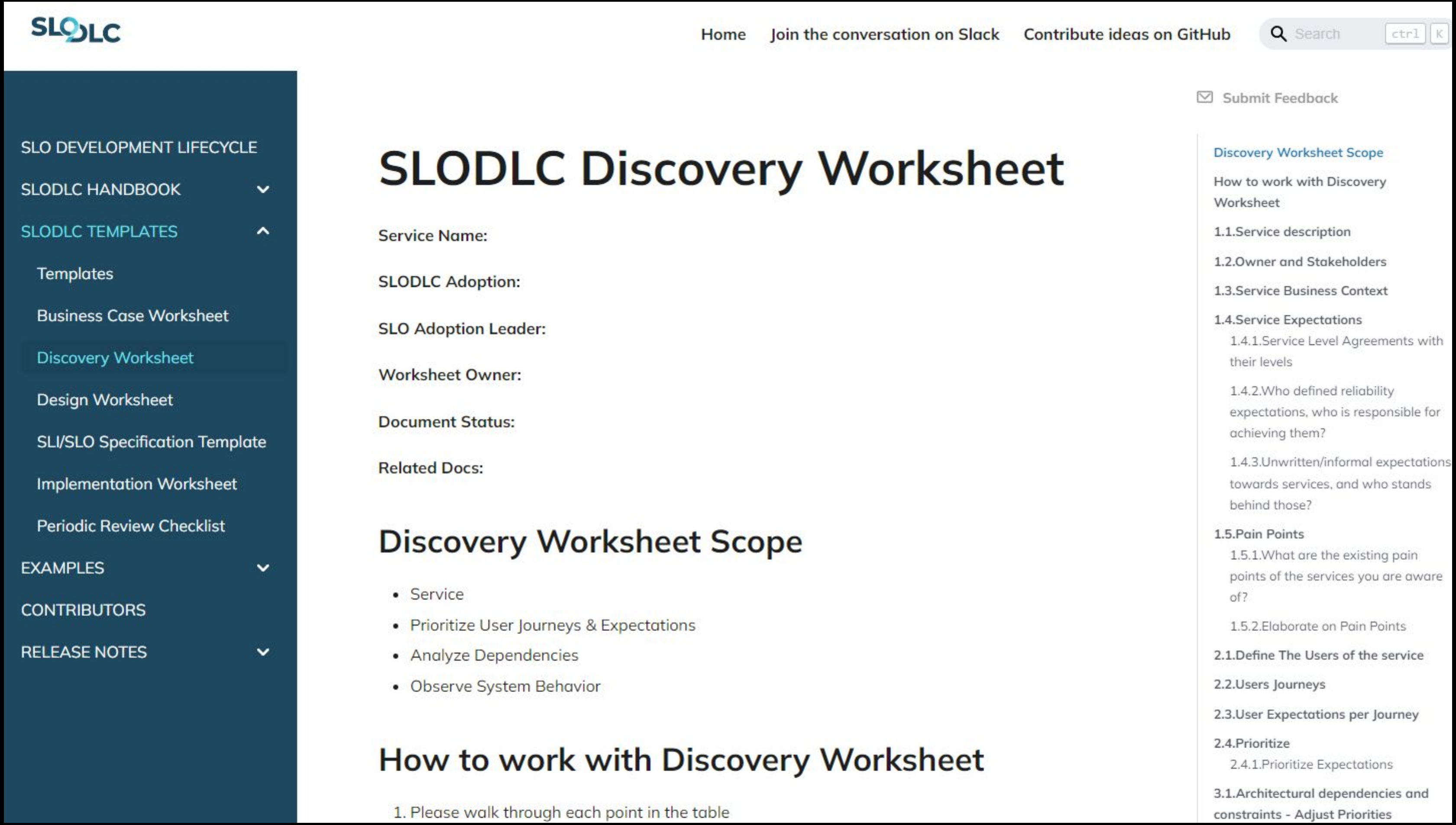
Gain SLO Insights

OPERATE

Review Periodically

Share Learnings from SLO Journey to Align Practices and Standards

SLODLC



SLODLC Discovery Worksheet

Service Name:

SLODLC Adoption:

SLO Adoption Leader:

Worksheet Owner:

Document Status:

Related Docs:

Discovery Worksheet Scope

- Service
- Prioritize User Journeys & Expectations
- Analyze Dependencies
- Observe System Behavior

How to work with Discovery Worksheet

1. Please walk through each point in the table

Your Problems Aren’t Unique

Don’t

Operate in a Silo

Do

Embrace open standards

Your Problems Aren't Unique

Standards: Open Telemetry & Open SLO

Frameworks: DORA, R9Y.dev, SLODLC

Communities: LFI, Art of SLOs

Slacks: DevOpsDays, SRECon, HangOps, Rands Leadership, Reliability.org

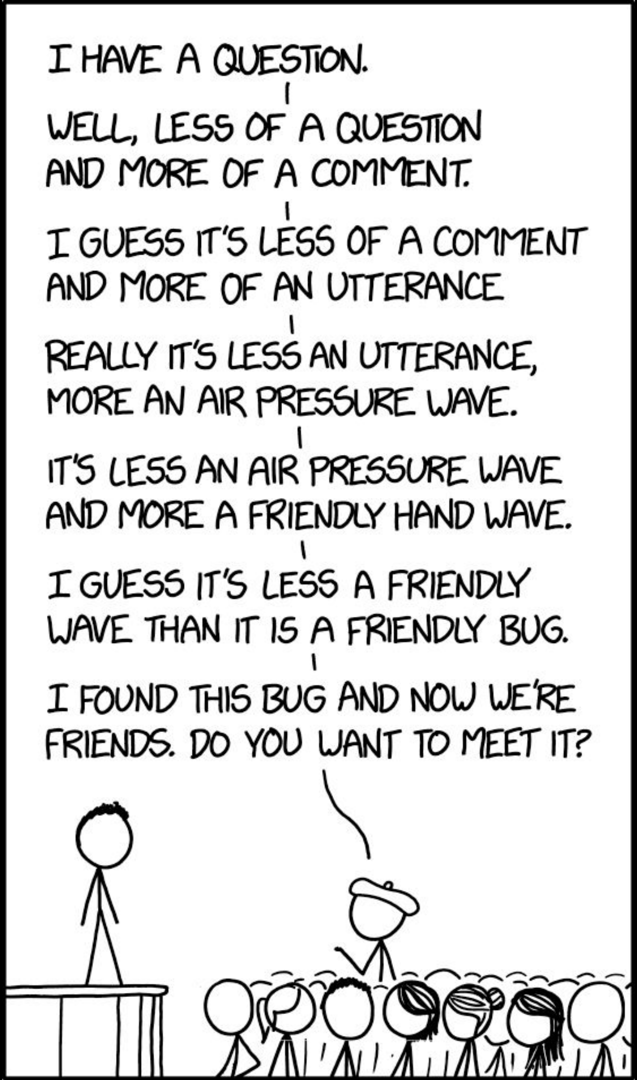
Socials: /r/sre, /r/devops, Tech Twitter?, Mastodon, bluesky?, etc

Thank you!

Salvatore Furino

CRE - Customer Reliability Engineer

Q & A


I HAVE A QUESTION.
WELL, LESS OF A QUESTION AND MORE OF A COMMENT.
I GUESS IT'S LESS OF A COMMENT AND MORE OF AN UTTERANCE
REALLY IT'S LESS AN UTTERANCE, MORE AN AIR PRESSURE WAVE.
IT'S LESS AN AIR PRESSURE WAVE AND MORE A FRIENDLY HAND WAVE.
I GUESS IT'S LESS A FRIENDLY WAVE THAN IT IS A FRIENDLY BUG.
I FOUND THIS BUG AND NOW WE'RE FRIENDS. DO YOU WANT TO MEET IT?

Source: xkcd

Your Problems Aren’t Unique