devfest

Site Reliability Engineering natively on the Google Cloud Platform

Google Developer Groups

DevFest Pescara 2023

Jai Campbell

Senior Architect wipro

Founder: London SRE meetup community

Co-Organiser: GDG London

Google Developer Expert: Cloud

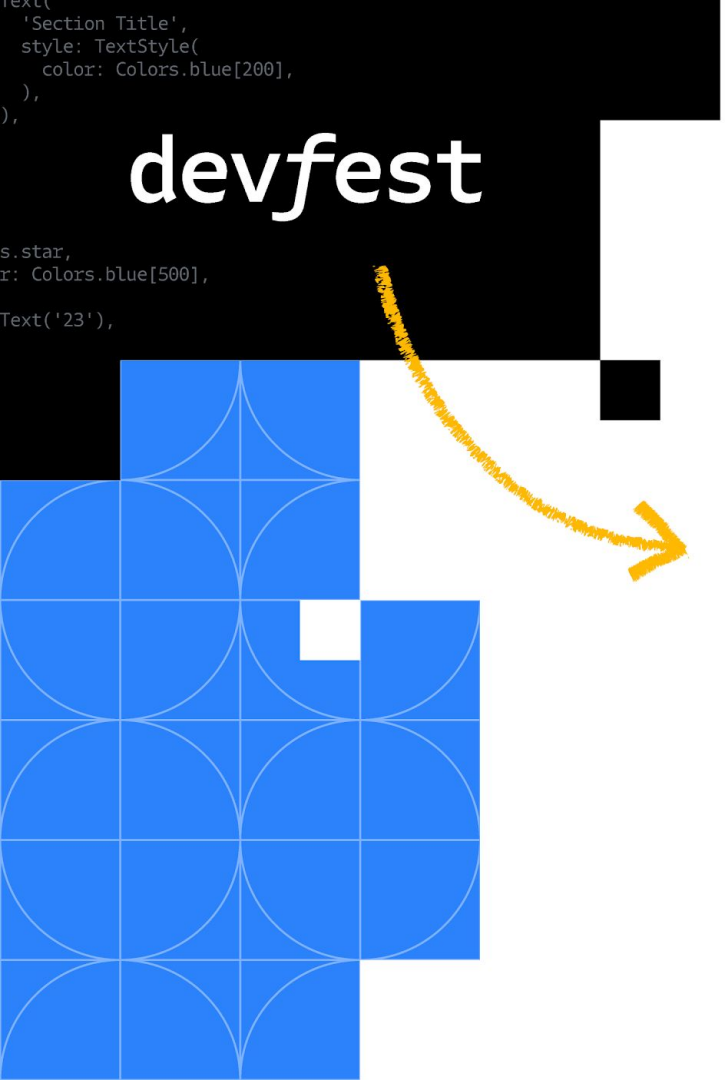
Women Techmaker Ambassador

@sylvanasbeta

https://github.com/sylvanasbeta

https://www.linkedin.com/in/campbelljai/

Google Developer Groups

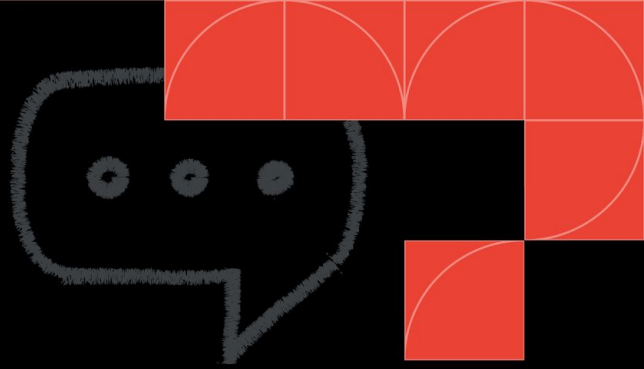


Google Developer Groups

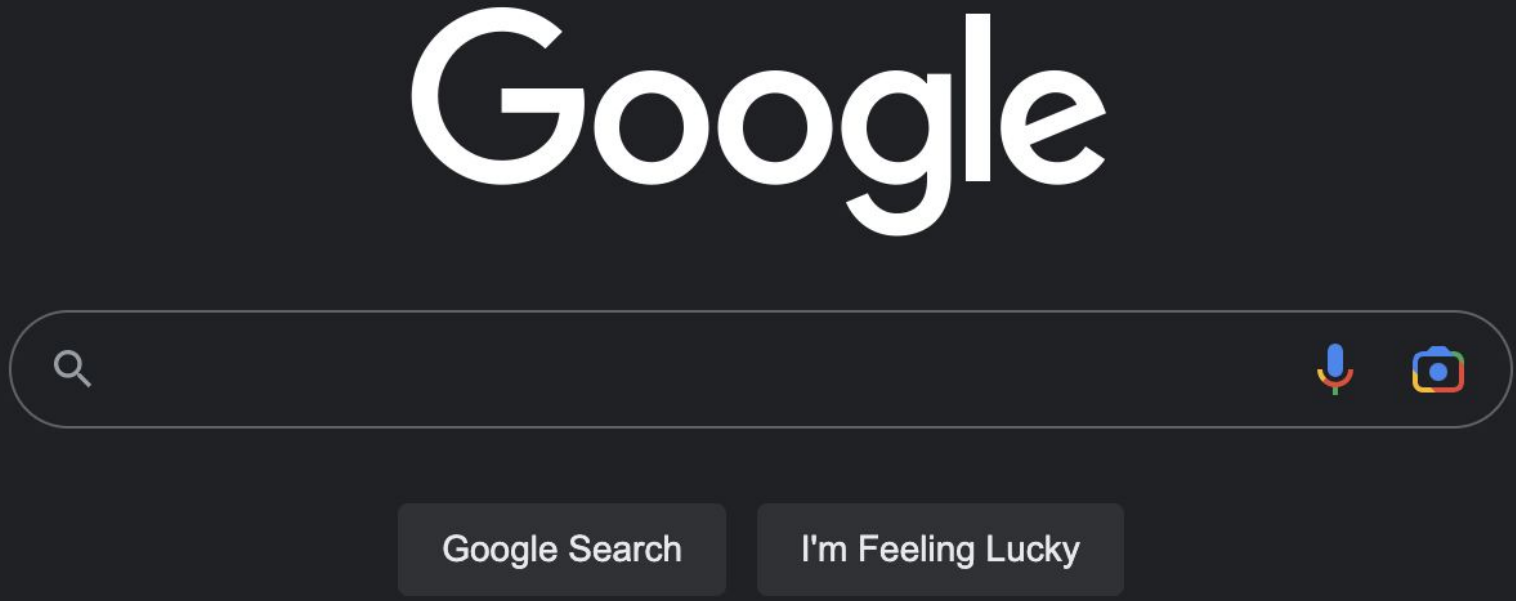
DevFest Pescara 2023

Brief overview of Site Reliability Engineering

devfest

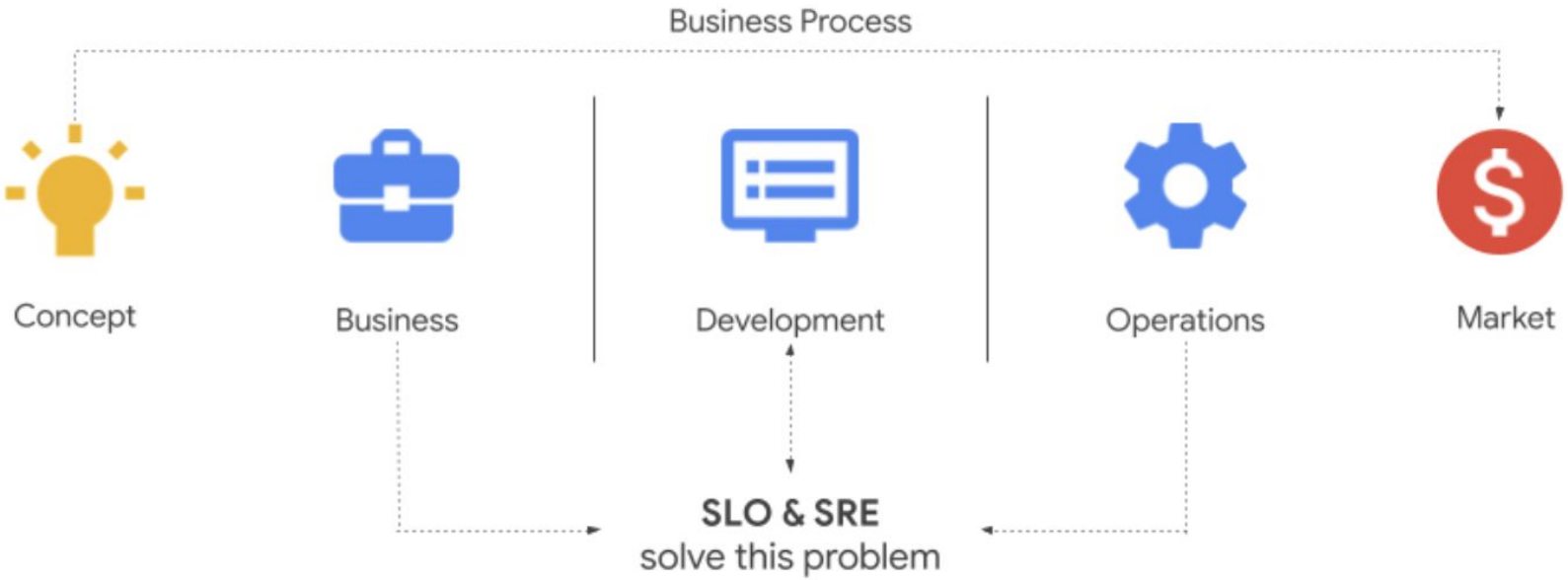


“SRE is what you get when you treat operations as if it’s a software problem, with an ever-watchful eye
on a systems availability, latency, performance, and capacity.”


Google
Google Search
I'm Feeling Lucky

12
9
3
6
=
12
9
3
6
EVERY SECOND
ONE HOUR OF VIDEO
IS UPLOADED TO YOUTUBE

Product lifecycle



Google Developer Groups

https://www.googlecloudcommunity.com/gc/Community-Blogs/Site-Reliability-Engineering-SRE-Fundamentals/ba-p/472590

Google Site Reliability Engineering

Metrics & Monitoring

- SLOs
- Dashboards
- Analytics

Capacity Planning

- Forecasting
- Demand-driven
- Performance

Change Management

- Release process
- Consulting design
- Automation

Emergency Response

- Oncall
- Analysis
- Postmortems

Culture

- Toil management
- Engineering alignment
- Blamelessness

https://www.googlecloudcommunity.com/gc/Community-Blogs/Site-Reliability-Engineering-SRE-Fundamentals/ba-p/472590

SLIs SLOs & SLAs

An SLI is a **service level indicator**—a carefully defined quantitative measure of some aspect of the level of service that is provided.

An SLO is a **service level objective**: a target value or range of values for a service level that is measured by an SLI.

SLAs are **service level agreements**: an explicit or implicit contract with your users that includes consequences for missing the SLOs they contain

Google Developer Groups

https://cloud.google.com/blog/products/management-tools/learn-how-to-set-slos-for-an-sre-or-cre-practice

Class SRE Implements DevOps

We have to look at traditional IT teams, these teams were made of Developers and Operators.

Developers are responsible for writing the code for the systems and the operators who are responsible for ensuring that these systems operate reliably.

Developers want to push more features and then ‘throw them over the fence’ for the operators to look after.

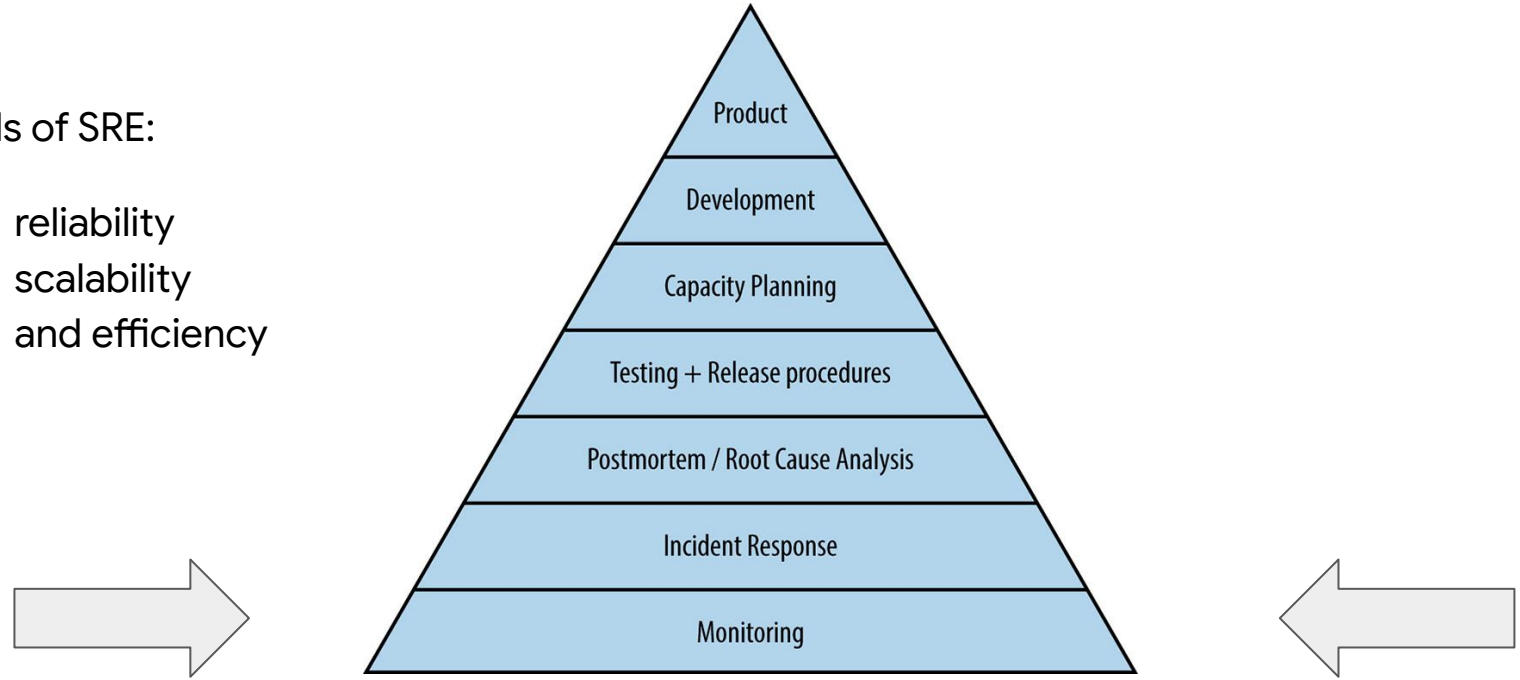
Developers traditionally want to work fast so the can push more features to compete with other organizations, the operators want to work slower to ensure these new features work as expected and are reliable.

This doesn’t scale well, especially when systems grow in complexity and scale, to combat this a new culture emerged and with best practices known as DevOps.

Service Reliability Hierarchy

Goals of SRE:

- reliability
- scalability
- and efficiency



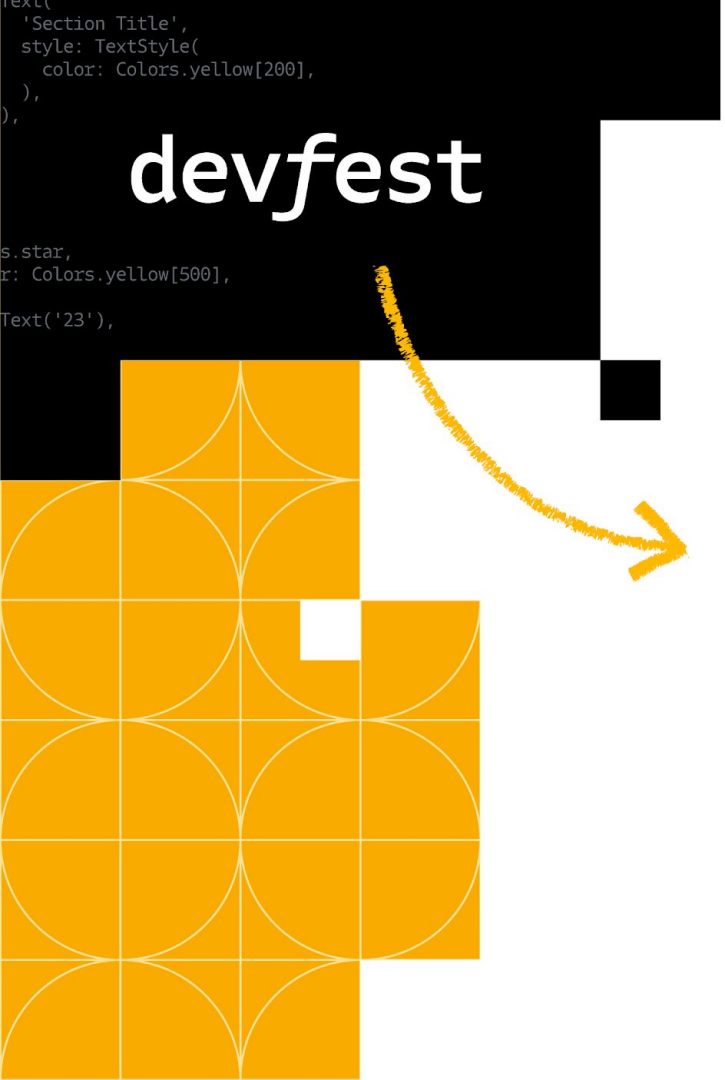
'Pessimistically optimistic'

"The glass is half empty... But hey, at least it's not empty."

“SREs are focused on the whole production stack, making systems that a really wide range of internal and external customers are using.”

Ciara, a software engineer in Site Reliability Engineering @ Google

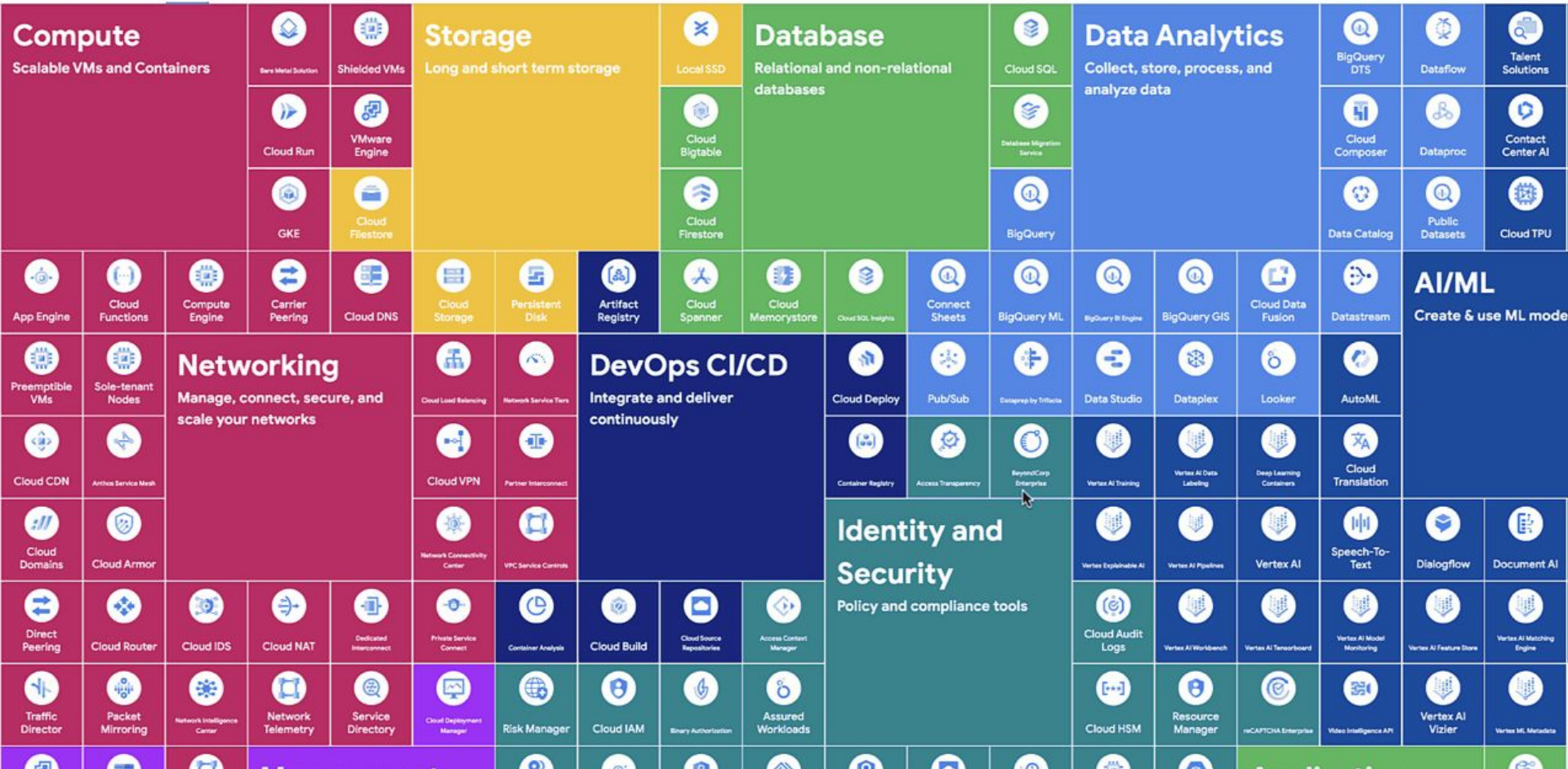
https://careers.google.com/stories/site-reliability-engineering-profile-google/



Google Developer Groups

DevFest Pescara 2023

Native APIs on the Google Cloud Platform


Compute
Scalable VMs and Containers
Bare Metal Solution
Shielded VMs
Cloud Run
VMware Engine
GKE
Cloud Filestore
Storage
Long and short term storage
Local SSD
Cloud Bigtable
Cloud Firestore
Database
Relational and non-relational databases
Cloud SQL
Database Migration Service
BigQuery
Data Analytics
Collect, store, process, and analyze data
BigQuery DTS
Dataflow
Talent Solutions
Cloud Composer
Dataproc
Contact Center AI
Data Catalog
Public Datasets
Cloud TPU
App Engine
Cloud Functions
Compute Engine
Carrier Peering
Cloud DNS
Cloud Storage
Persistent Disk
Artifact Registry
Cloud Spanner
Cloud Memorystore
Cloud SQL Insights
Connect Sheets
BigQuery ML
BigQuery BI Engine
BigQuery GIS
Cloud Data Fusion
Datastream
AI/ML
Create & use ML models
Preemptible VMs
Sole-tenant Nodes
Networking
Manage, connect, secure, and scale your networks
Cloud Load Balancing
Network Service Tiers
Cloud CDN
Anthos Service Mesh
Cloud VPN
Partner Interconnect
Cloud Domains
Cloud Armor
Network Connectivity Center
VPC Service Controls
DevOps CI/CD
Integrate and deliver continuously
Cloud Deploy
Pub/Sub
Dataprep by Trifacta
Container Registry
Access Transparency
BeyondCorp Enterprise
Data Studio
Dataplex
Looker
AutoML
Vertex AI Training
Vertex AI Data Labeling
Deep Learning Containers
Cloud Translation
Identity and Security
Policy and compliance tools
Vertex Explainable AI
Vertex AI Pipelines
Vertex AI
Speech-To-Text
Dialogflow
Document AI
Direct Peering
Cloud Router
Cloud IDS
Cloud NAT
Dedicated Interconnect
Private Service Connect
Container Analysis
Cloud Build
Cloud Source Repositories
Access Context Manager
Cloud Audit Logs
Vertex AI Workbench
Vertex AI Tensorboard
Vertex AI Model Monitoring
Vertex AI Feature Store
Vertex AI Matching Engine
Traffic Director
Packet Mirroring
Network Intelligence Center
Network Telemetry
Service Directory
Cloud Deployment Manager
Risk Manager
Cloud IAM
Binary Authorization
Assured Workloads
Cloud HSM
Resource Manager
reCAPTCHA Enterprise
Video Intelligence API
Vertex AI Vizier
Vertex ML Metadata

Stackdriver Monitoring API

Google Enterprise API

The Stackdriver Monitoring API gives you access to over 900 Stackdriver Monitoring metrics from Google Cloud Platform and Amazon Web Services. You can create your own custom metrics and can organize your cloud resources using groups.

Error Reporting API

Google

Groups and counts similar errors from cloud services and applications, reports new errors, and provides access to error groups and their associated errors.

Cloud Trace API

Google Enterprise API

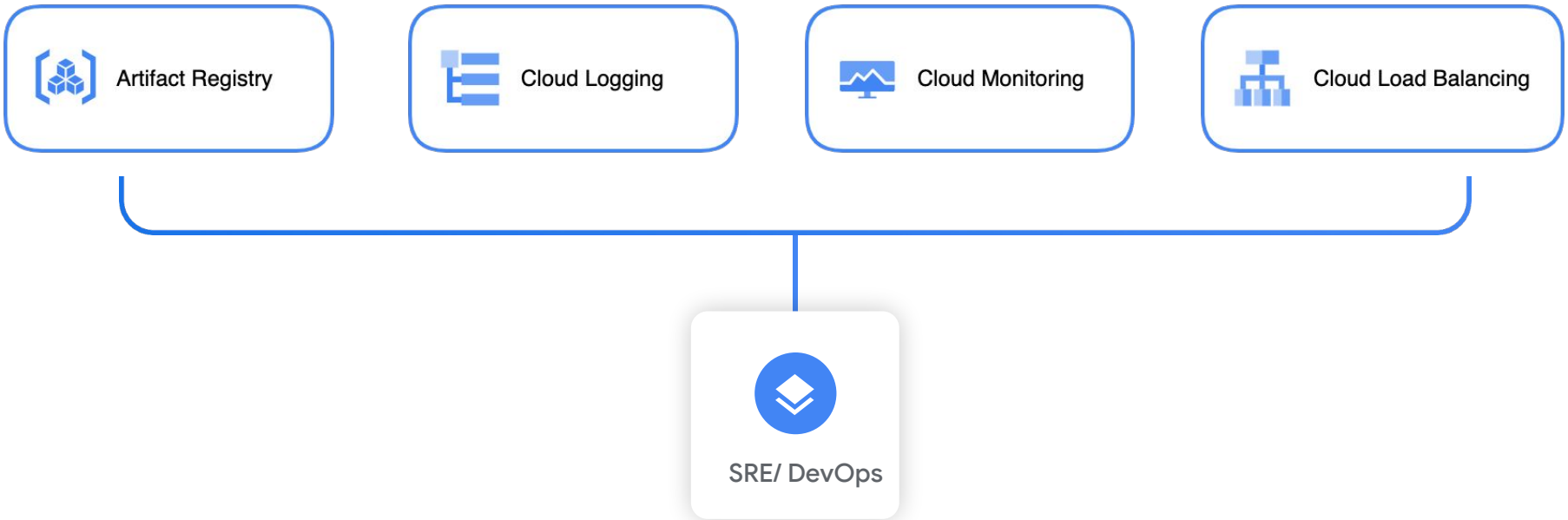
Sends application trace data to Cloud Trace for viewing. Trace data is collected for all App Engine applications by default. Trace data from other applications can be provided using this API.

Config Monitoring for Ops API

Google

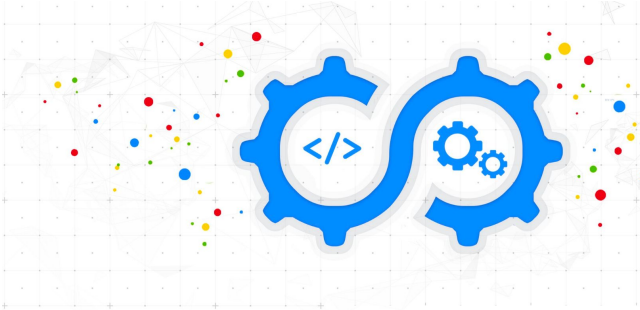
Config Monitoring for Ops API collects, aggregates and indexes resource configuration and operational metadata of resources and services in GCP, including the status and relations, and makes it actionable to enable rich, out of the box monitoring for Cloud.

Snapshot of available native APIs

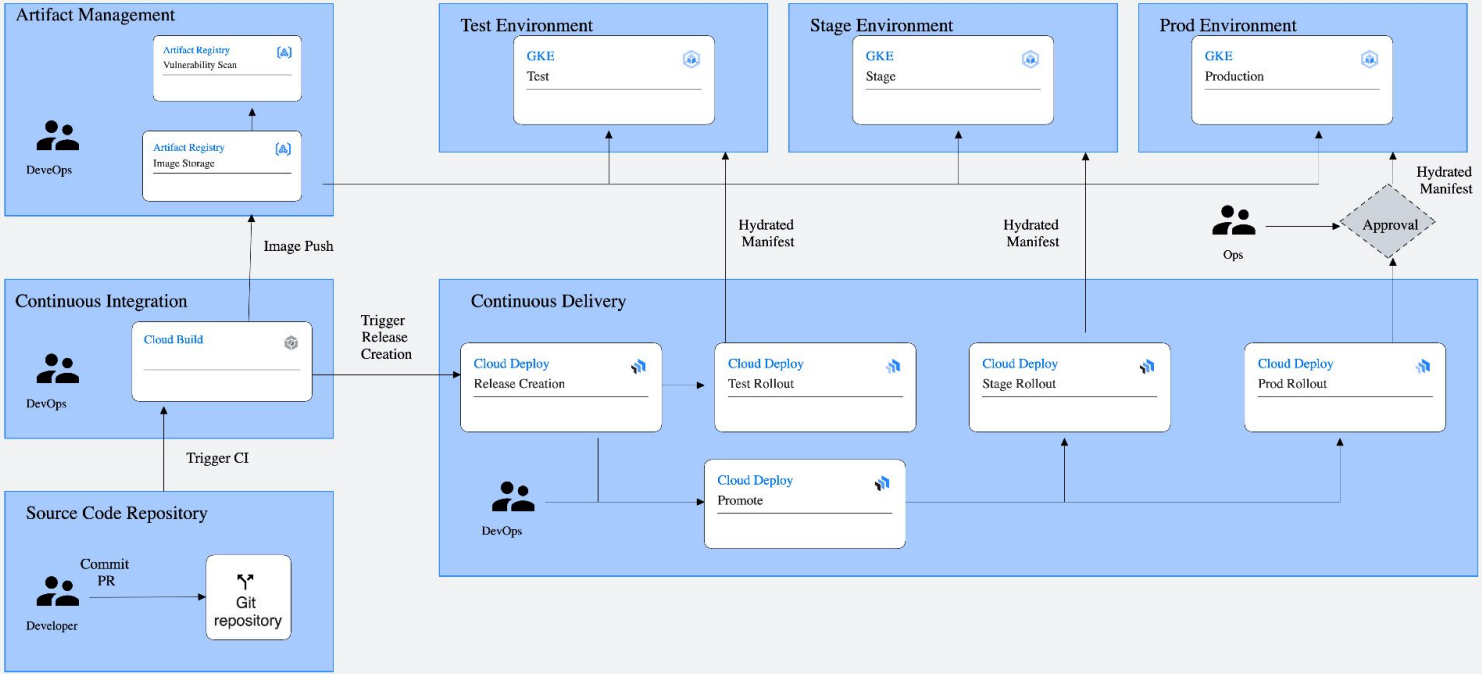


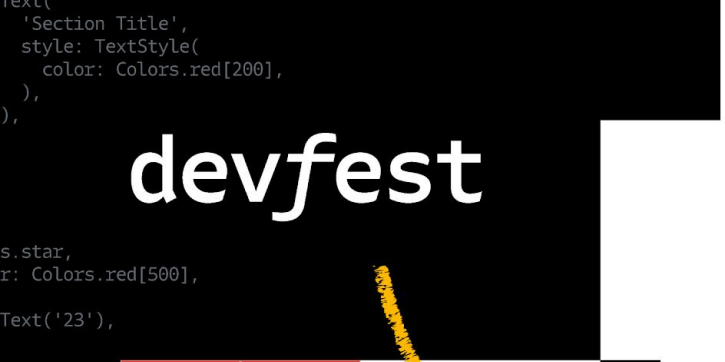
Why use native APIs for SRE tasks ?

- **Out of the box integrations** Leverage native integrations between services to benefit from security, ease of use, known implementations.
- **One integrated view for faster resolution** A unified view across logs, metris, events and SLOs across cloud services and deployed applications.
- **Seamless integration with GCP services** Google is known for SRE and those learnings are built into Google Cloud Platform and are easily accessible.
- **Save costs by not using third party software** There's a vast amount of options available to meet your SRE requirements without having to use software outside the platform.



End-to-End Google Cloud DevOps Flow





Google Developer Groups

Benefits of Using Native APIs For SRE tasks

Benefits of using native APIs for SRE tasks

- **Seamless integration with GCP services** Googles emphasis on reliability ensures those learnings and capabilities are built into Google Cloud Platform and are easily accessible across various services.
- **Consistent and reliable performance** With a wide user base and consistent testing you have ensured reliability when developing on the platform.
- **Easy automation, integration and scripting** Development tasks are easier with a rich set of tools that tie observability into your solutions
- **Comprehensive documentation and support** With a large user base providing constant feedback there is a large amount of information readily available and accessible.

Google Cloud integration

Discover and monitor all Google Cloud resources and services, with no additional configuration, integrated right into the Google Cloud console.

Start your Monitoring journey by simply running workloads on any of the supported Google Cloud products. Learn more

EXPLORE FEATURED PRODUCTS

CONFIGURE METRICS SCOPE

Compute Engine

Cloud Storage

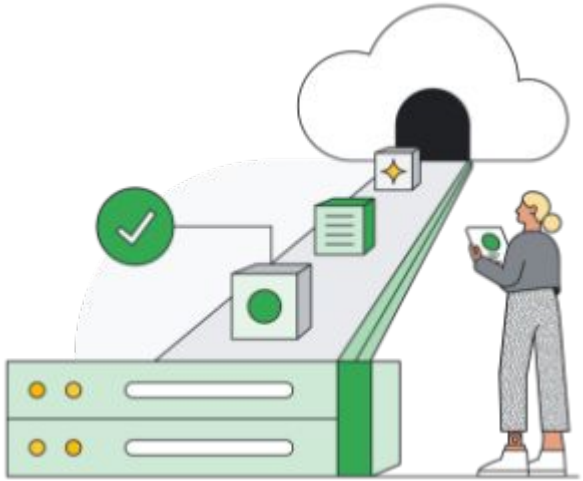
SQL

Kubernetes Engine

Dataflow

Cloud Run

Cloud Functions



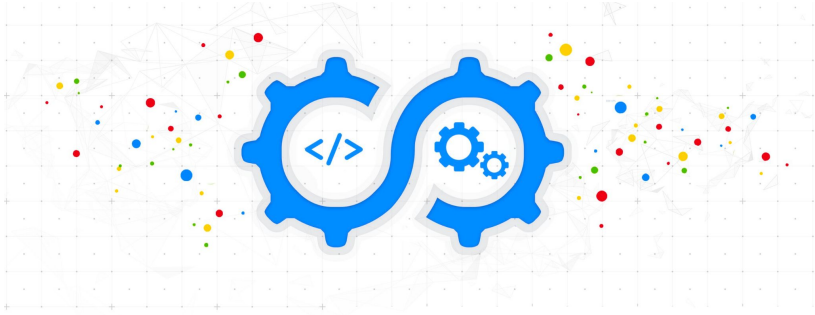
Google Developer Groups

DevFest Pescara 2023

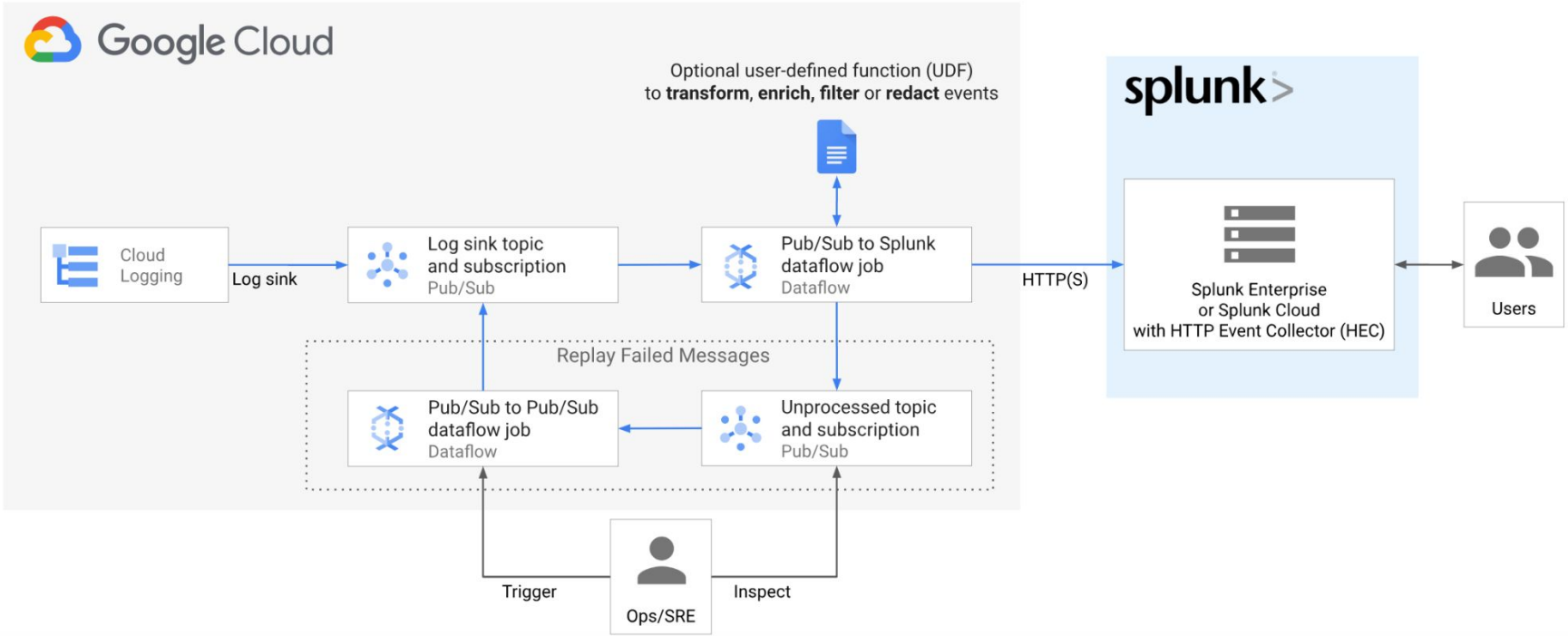
SRE Tasks Using Native APIs

SRE tasks using native APIs

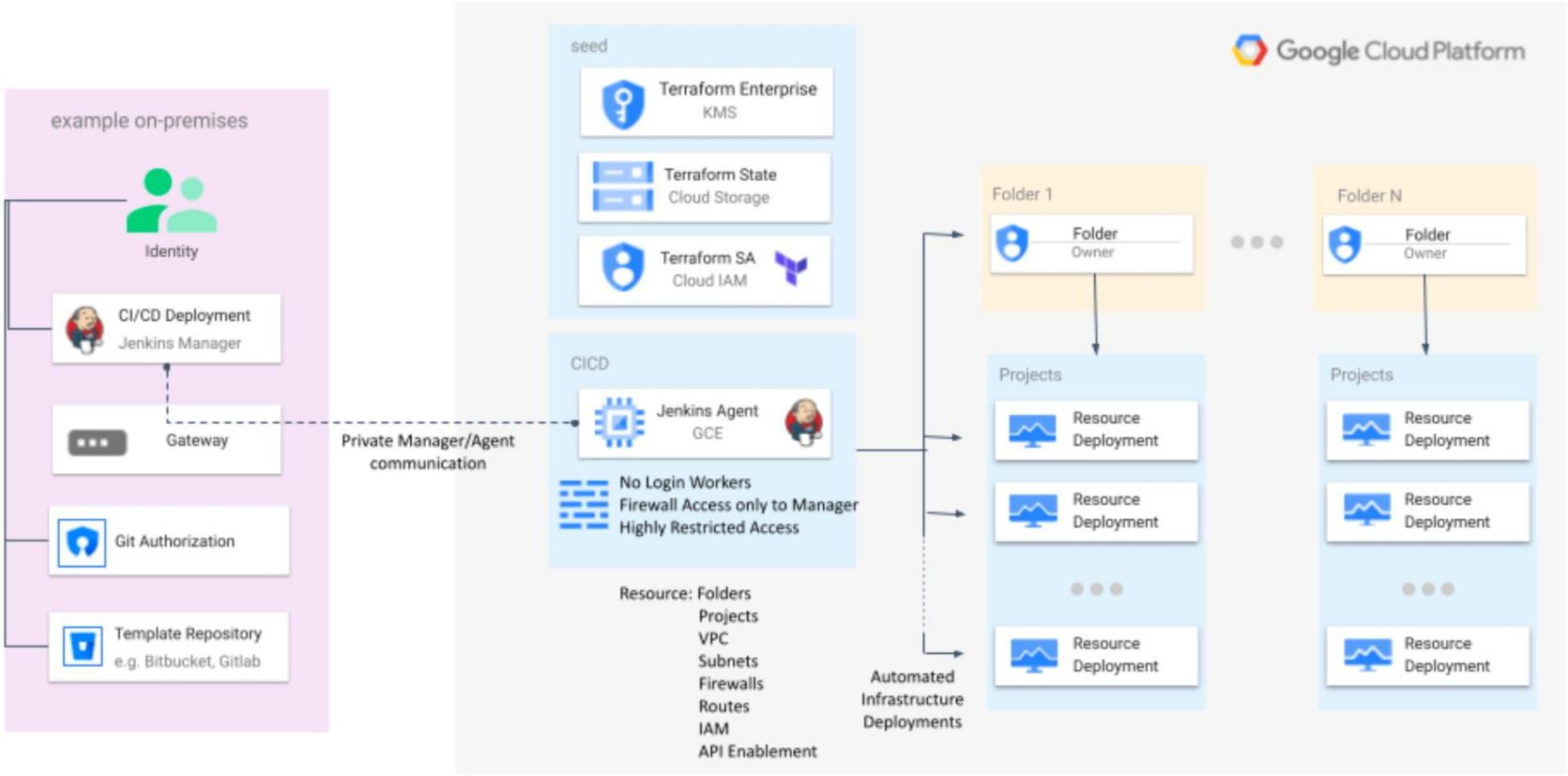
- **Monitoring and alerting with Cloud Monitoring API** Out of the box access to over 1,500 cloud monitoring metrics
- **Autoscaling with Compute Engine API** Inbuilt capabilities to automatically add or remove VM instances from managed instance groups.
- **Log analysis and management with Cloud Logging API** Easy access to manage, monitor and retrieve insights from log data in real time
- **Source Code Management** The ability to privately host, track and manage changes to large code bases. (GitOps)

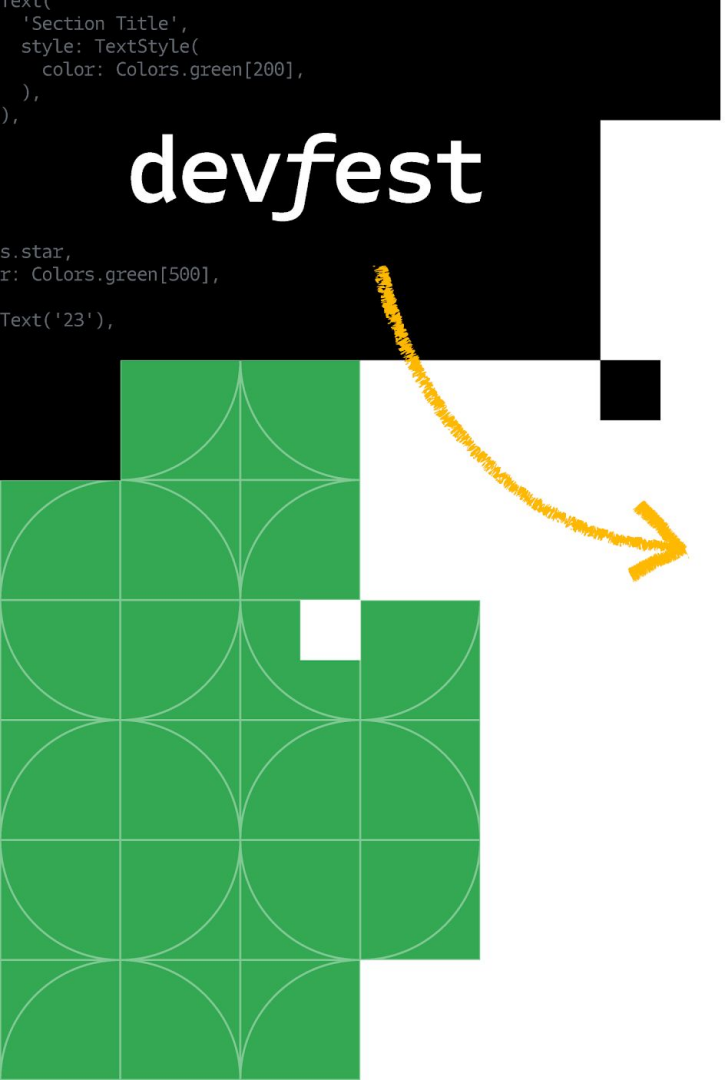


Stream logs from Google Cloud to Splunk



https://cloud.google.com/architecture/stream-logs-from-google-cloud-to-splunk


example on-premises
Identity
CI/CD Deployment
Jenkins Manager
Gateway
Git Authorization
Template Repository
e.g. Bitbucket, Gitlab
Private Manager/Agent communication
seed
Terraform Enterprise
KMS
Terraform State
Cloud Storage
Terraform SA
Cloud IAM
CICD
Jenkins Agent
GCE
No Login Workers
Firewall Access only to Manager
Highly Restricted Access
Resource: Folders
Projects
VPC
Subnets
Firewalls
Routes
IAM
API Enablement
Automated Infrastructure Deployments
Google Cloud Platform
Folder 1
Folder
Owner
Projects
Resource Deployment
Resource Deployment
Resource Deployment
Folder N
Folder
Owner
Projects
Resource Deployment
Resource Deployment
Resource Deployment



Google Developer Groups

DevFest Pescara 2023

Case Study: Implementing SRE with Native APIs

Dejima Concept and Transformation Initiative:

- JCB initiated a project three years ago to develop high-value services with agility.
- The "Dejima" concept was introduced, focusing on starting small from scratch, independent of existing systems.
- Inspired by the Edo periods Dejima, JCB aimed to create an organization independent from traditional practices for effective digital transformation.

Digital Enablement Platform and Google Cloud Integration:

- JCB successfully transformed with the help of Google Cloud, implementing services like GKE, Cloud Spanner, and Anthos Service Mesh.

Introduction of Site Reliability Engineering (SRE) Practices:

- SRE practices were introduced to break down barriers between business, development, and operations teams.
- Categories addressed included defining organizational culture and practices, ensuring practical policies, and measuring appropriate reliability.

https://cloud.google.com/blog/topics/financial-services/how-jcb-transforms-its-infrastructure-using-a-dejima-concept

JCB's SRE team comprises two sub-teams:

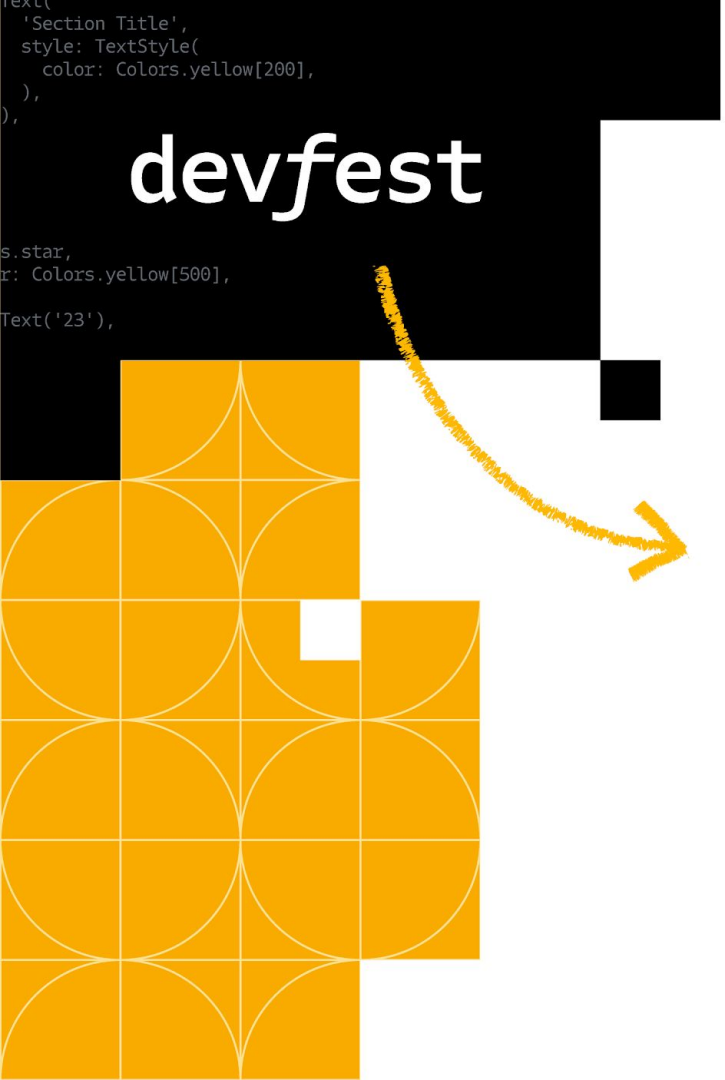
Sheriff, functioning as the platform SRE, delivering infrastructure services to the application team;

Sheriff (CoP)	Diplomat (APL Team alignment)
Formulation and introduction of release method	Monitoring Dashboard construction
Planning and execution of disability training	SLI / SLO Manage . update
Workload vulnerability detection	CI / CD Design • Build
Toil response (temporary request from APL Team, security response)	
On-call correspondence	
Daily Monitoring	

Diplomat, serving as the embedded SRE, actively engaging in the application team to oversee productionisation.

Additionally, there's an independent Architecture team tasked with advising the SRE on system design and conducting architecture reviews.

“We saw change within the organization. The business team started evangelizing SRE to other members in the business department, and the development and operations teams started collaborating autonomously”.



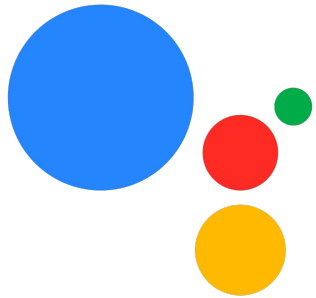
Google Developer Groups

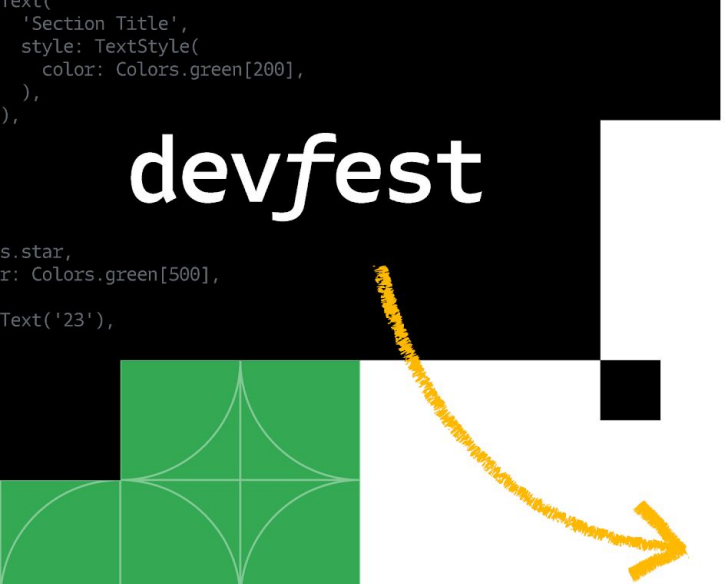
DevFest Pescara 2023

Best Practices for SRE with Native APIs

Best Practices for SRE with native APIs

- **Regularly review and update API usage** Leverage usage and review any changes that may need to be implemented
- **Implement proper authentication and authorisation** Ensure security you implement best practices and ensure least privilege across all API usage
- **Leverage GCP's IAM (Identity and Access Management for security)** Strengthen your security posture by utilising GCP IAM
- **Monitor API usage and performance metrics** Regularly review API usage and quota usage to ensure you are making the most of your cloud presence and ensure APIs are not redundant
- **Stay up to date on improvements and capabilities** Stay informed for updates on APIs that you have enabled and additional capabilities that may have been introduced





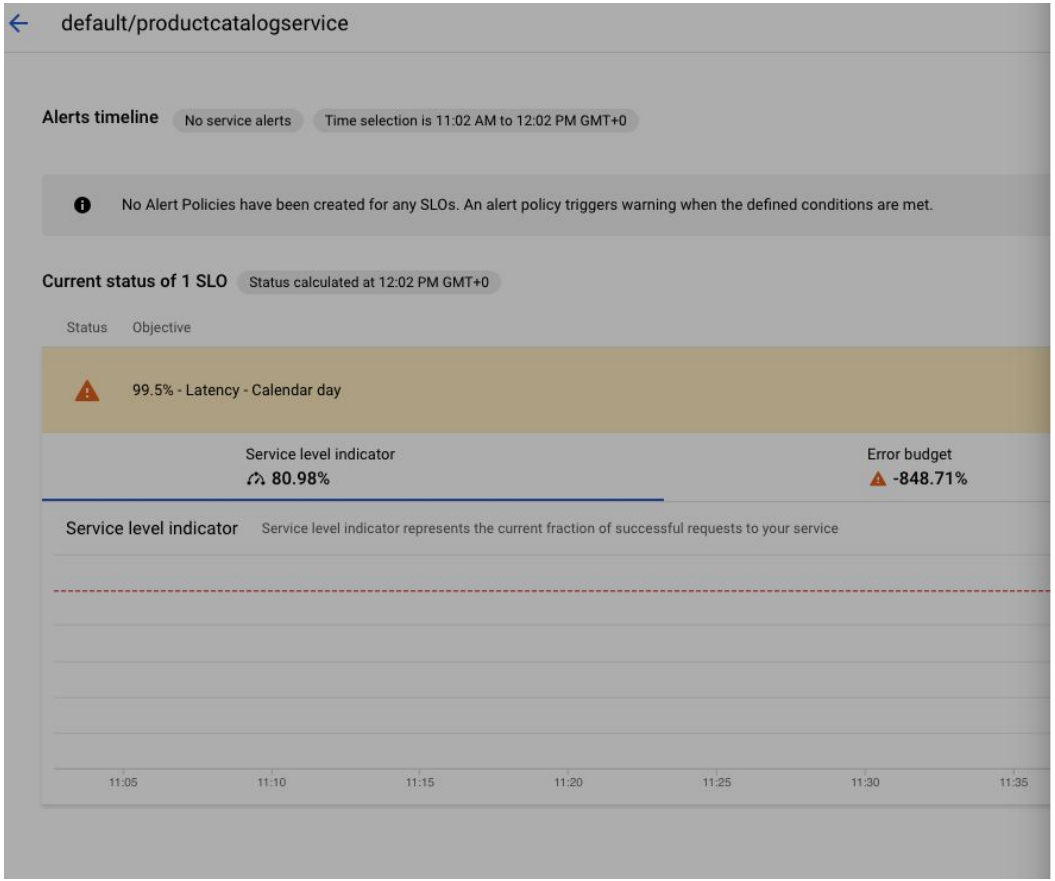
Google Developer Groups

Editable Location

Demo

Ecommerce website running on Kubernetes GKE

1. Cluster deployed on (GKE)
2. Configured Anthos service mesh
3. Installed the online boutique application on the GKE cluster
4. Deployed a new version of the product catalogue and split traffic across the two versions.
5. Explore the increased latency on the product catalogue service
6. Create a Latency SLO for the product catalogue service



1 Set SLO alert conditions

Creating an alert condition on your service-level objectives (SLOs) will let you know whether you are in danger of violating an SLO.

Target: 99.5% - Latency - Calendar day

Select a burn rate threshold value that constitutes a violation, and a lookback duration period for which the violation is permitted. If the burn rate threshold is exceeded for more than the allowable period, an incident is created. Learn more

Display name *
Burn rate on 99.5% - Latency - Calendar day

Lookback duration * 60 minute(s)

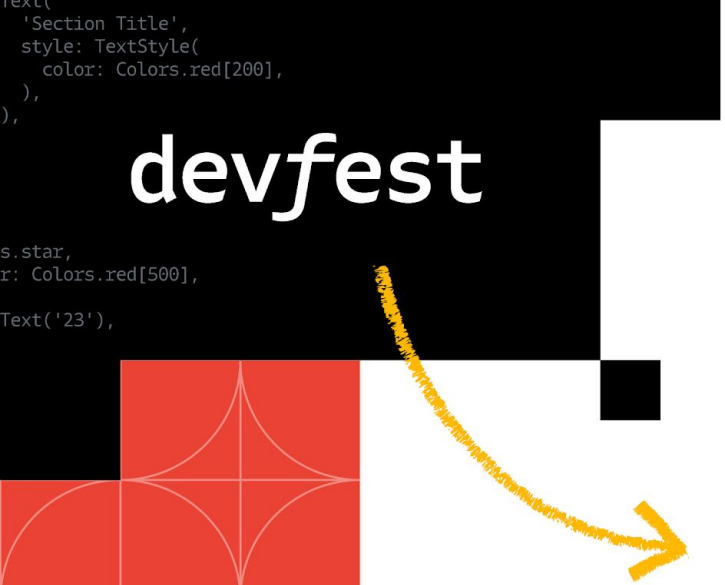
Burn rate threshold * 10

NEXT

2 Who should be notified? (optional)

When alerting policy violations occur, you will be notified via these channels.

3 What are the steps to fix the issue? (optional)

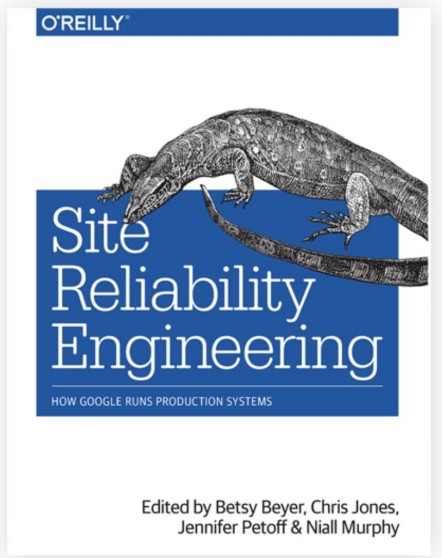


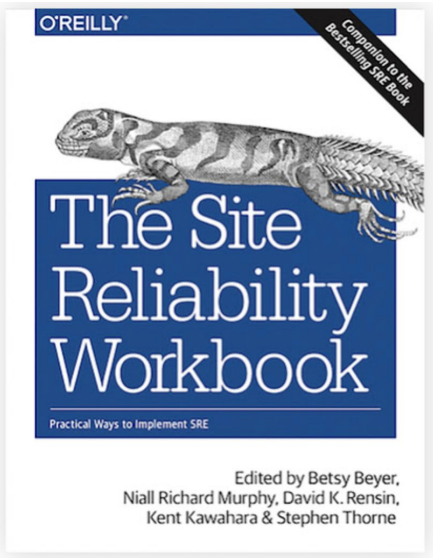
Google Developer Groups

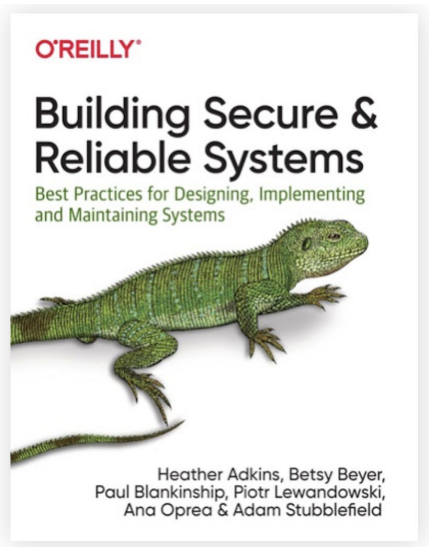
DevFest Pescara 2023

Tips and Tricks

Google Site Reliability Engineering







DevOps & SRE

News, updates, and best practices for Cloud Operations, DevOps and SRE at Google

DevOps & SRE

Synthetic Monitoring in Cloud Monitoring is now Generally Available

By Amol Devgan • 3-minute read

Application Development

Artifact Registry remote and virtual repositories are now generally available

By Rishi Mukhopadhyay • 2-minute read

DevOps & SRE

Success through culture: why embracing failure encourages better software delivery

By James Pashutinski • 6-minute read

Cloud Operations Sandbox Alpha

Cloud Operations Sandbox is an open source tool* that helps practitioners to learn Cloud Operations on GCP. It offers:

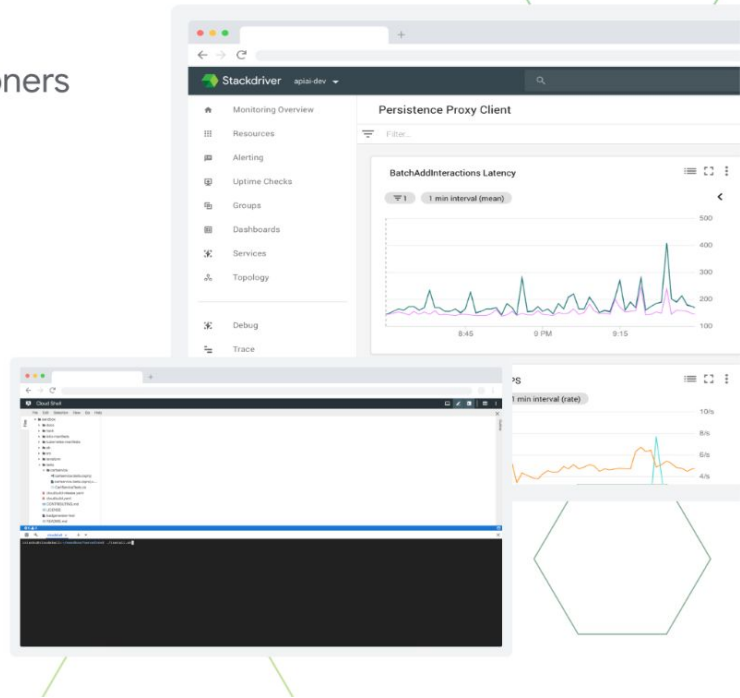
- **Demo Service** - an application built using microservices architecture on modern, cloud native stack.
- **One-click deployment script** of the service to Google Cloud Platform
- **Load Generator** - a component that produces synthetic traffic on a demo service
- **(Soon) SRE Runbook** - for operating deployed sample service that follows best SRE practices

View source code

Get started now

Open in Google Cloud Shell

Warning! If you have a Cloud Shell session running, clicking the button will restart it.



Google Developer Groups

One stop shop for ‘all’ your SRE needs!

Awesome Site Reliability Engineering

awesome

A curated list of awesome Site Reliability and Production Engineering resources.

What is Site Reliability Engineering?

> "Fundamentally, it's what happens when you ask a software engineer to design an operations function." - Ben Treynor Sloss, VP Google Engineering, founder of Google SRE

Contributing

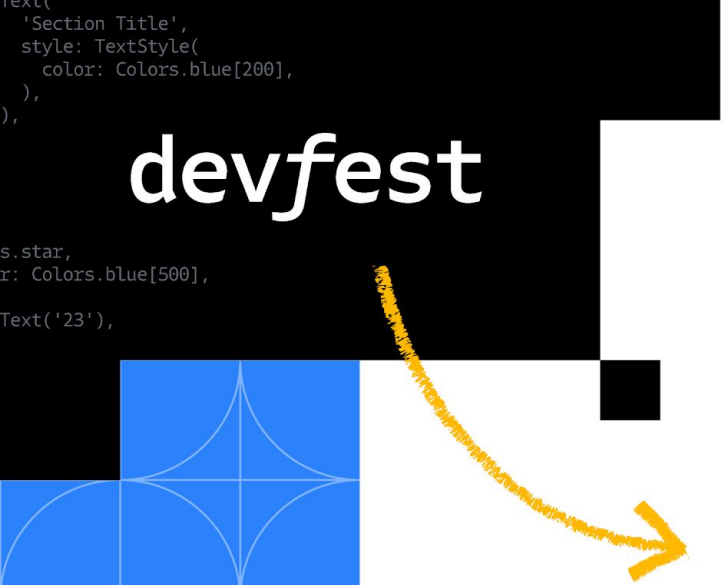
Please take a look at the contribution guidelines first. Contributions are always welcome!

Contents

- Culture
- Education
- Books
- Hiring
- Reliability
- Monitoring & Observability & Alerting
- On-Call
- Post-Mortem
- Capacity Planning

Google Developer Groups

https://github.com/dastergon/awesome-sre



Google Developer Groups

DevFest Pescara 2023

Conclusion

Thank you

Jai Campbell
@sylvanasbeta

Google Cloud