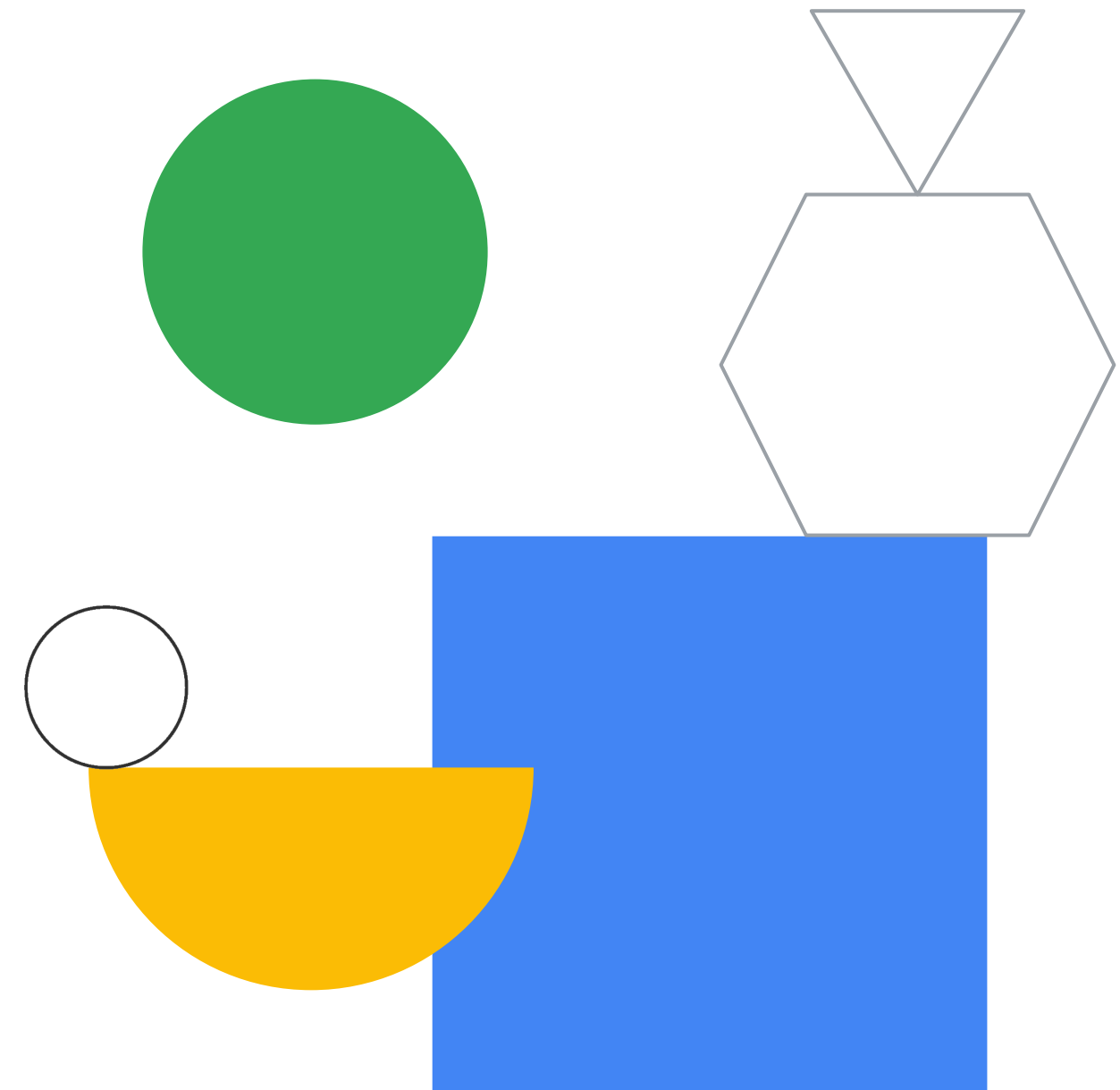
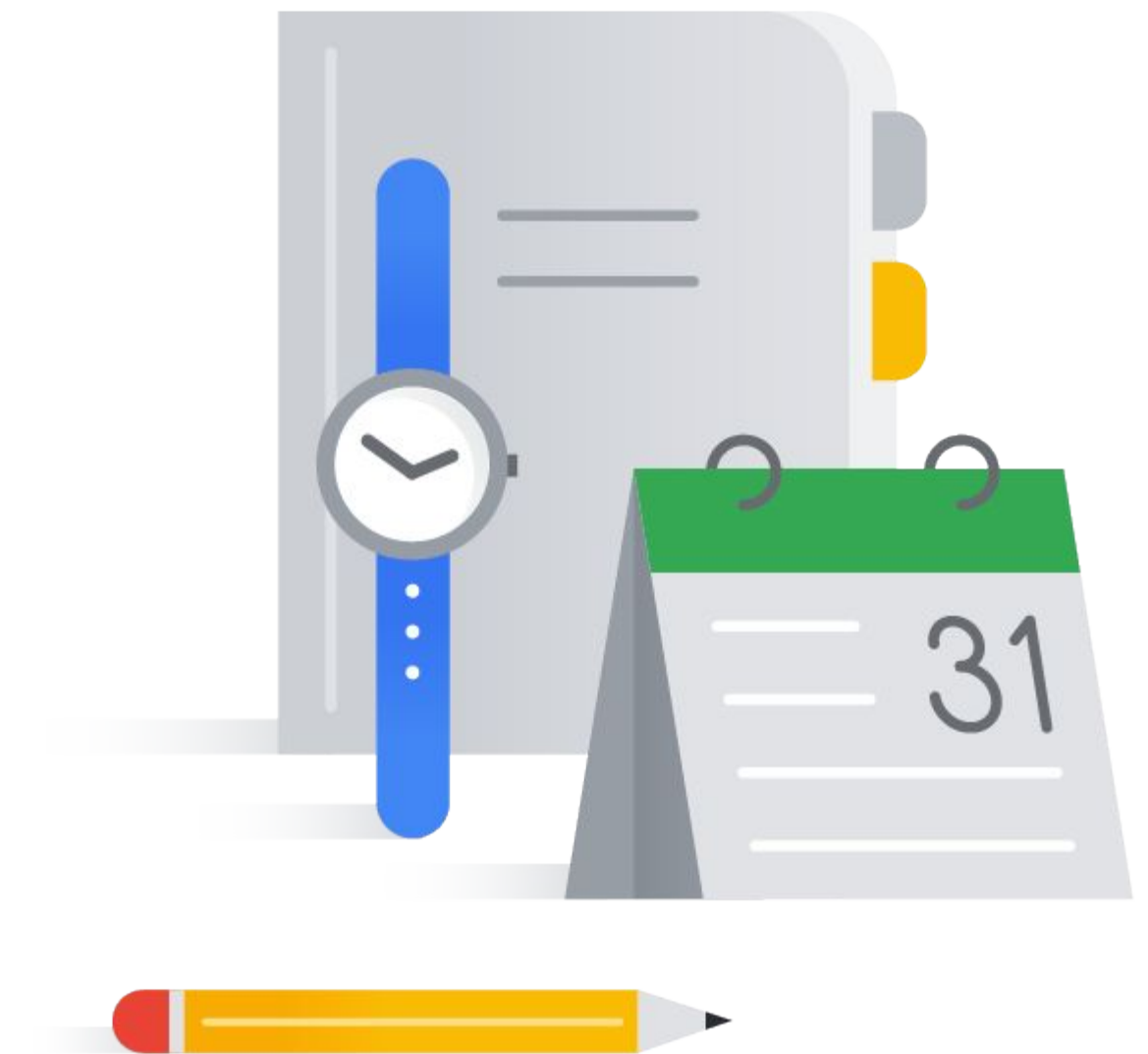


Resource Monitoring



Agenda

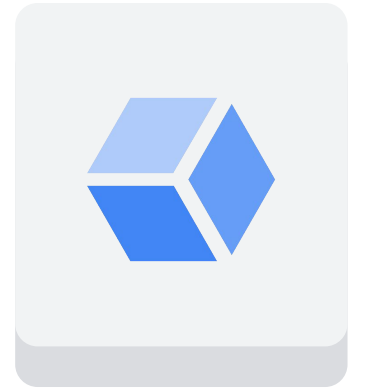
- | | |
|----|----------------------------------------|
| 01 | Google Cloud Observability |
| 02 | Monitoring
Lab: Resource Monitoring |
| 03 | Logging |
| 04 | Error Reporting |
| 05 | Tracing |
| 06 | Profiling |
| 07 | Partner Integrations |





Google Cloud Observability

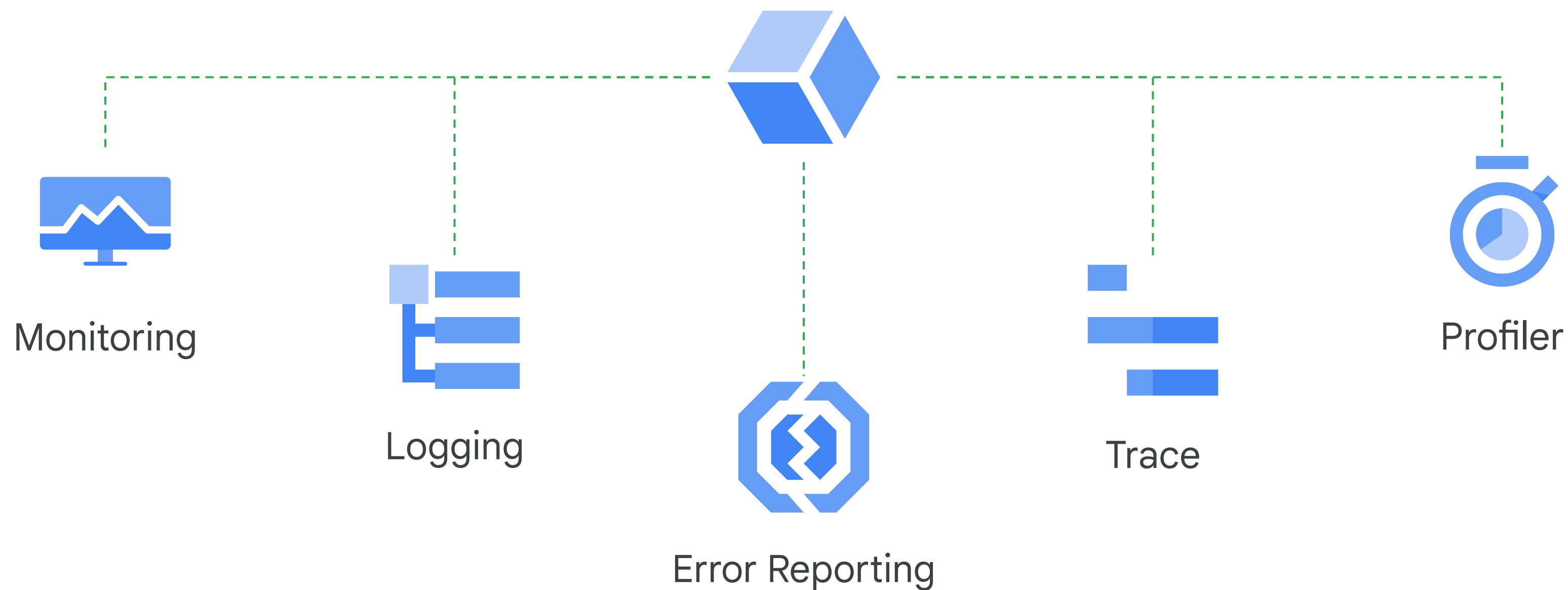
Google Cloud Observability overview



Google Cloud
Observability

- Integrated monitoring, logging, diagnostics
- Manages across platforms
 - Google Cloud and AWS
 - Dynamic discovery of Google Cloud with smart defaults
 - Open-source agents and integrations
- Access to powerful data and analytics tools
- Collaboration with third-party software

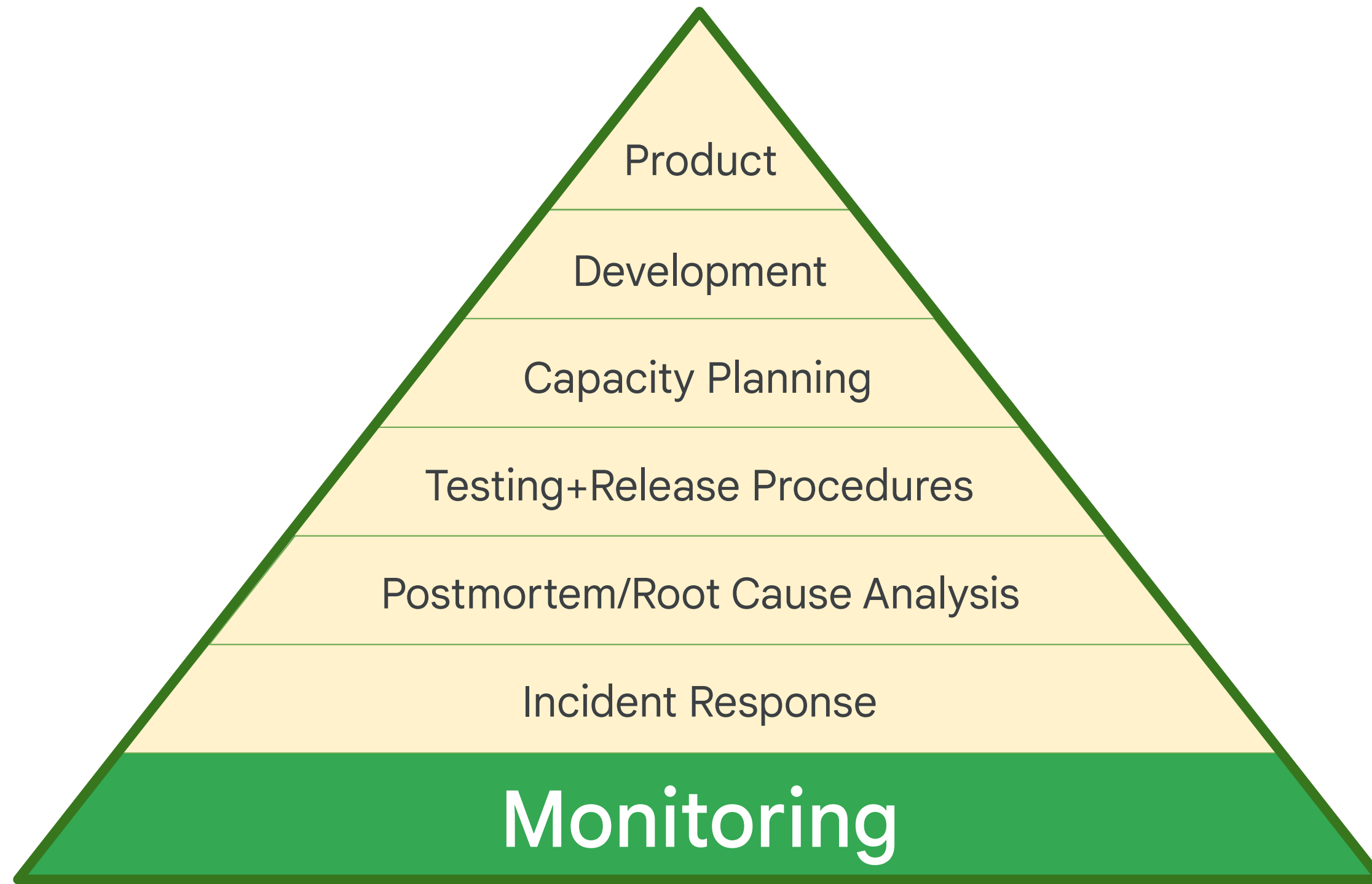
Multiple integrated products





Monitoring

Site reliability engineering



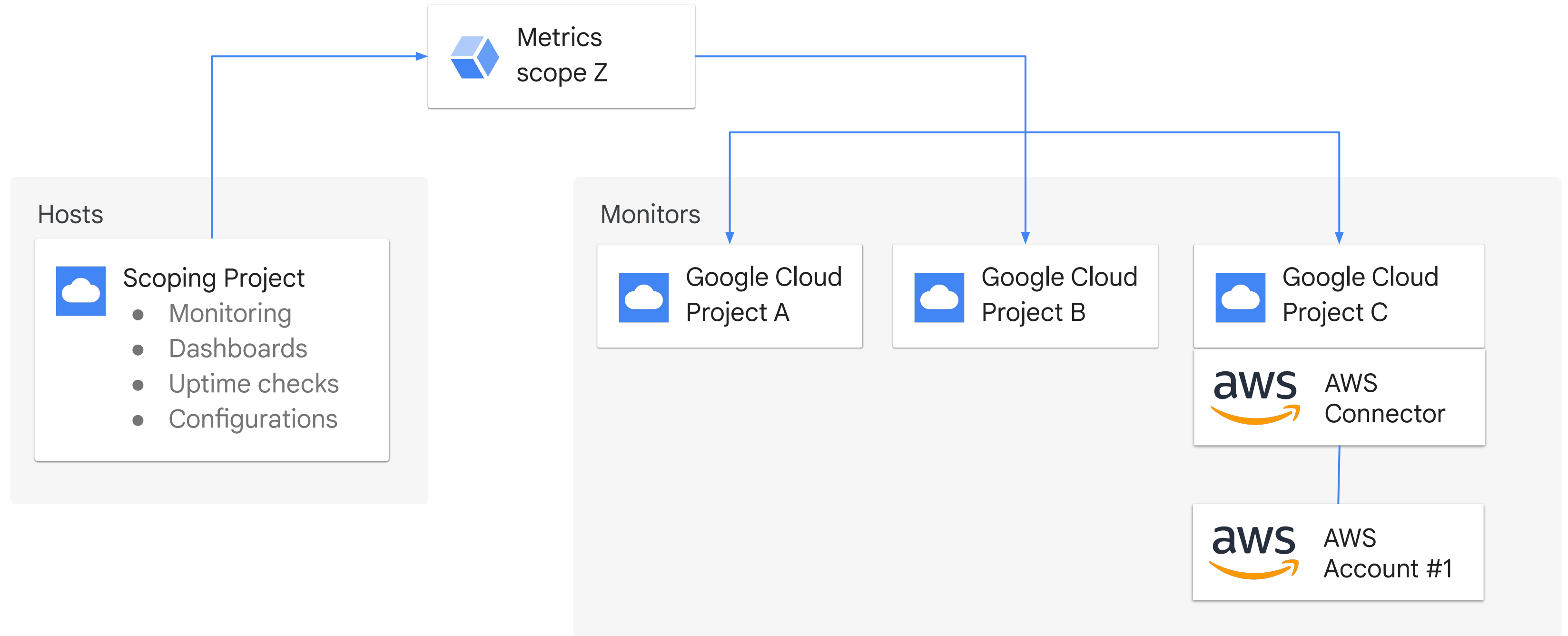
Monitoring



Monitoring

- Dynamic config and intelligent defaults
- Platform, system, and application metrics
 - Ingests data: Metrics, events, metadata
 - Generates insights through dashboards, charts, alerts
- Uptime/health checks
- Dashboards
- Alerts



A metrics scope is the root entity that holds monitoring and configuration information



A metrics scope is a “single pane of glass”

- Determine your monitoring needs up front.
- Consider using separate metrics scopes for data and control isolation.

Metrics monitored by this project

 **Filter** Filter projects 

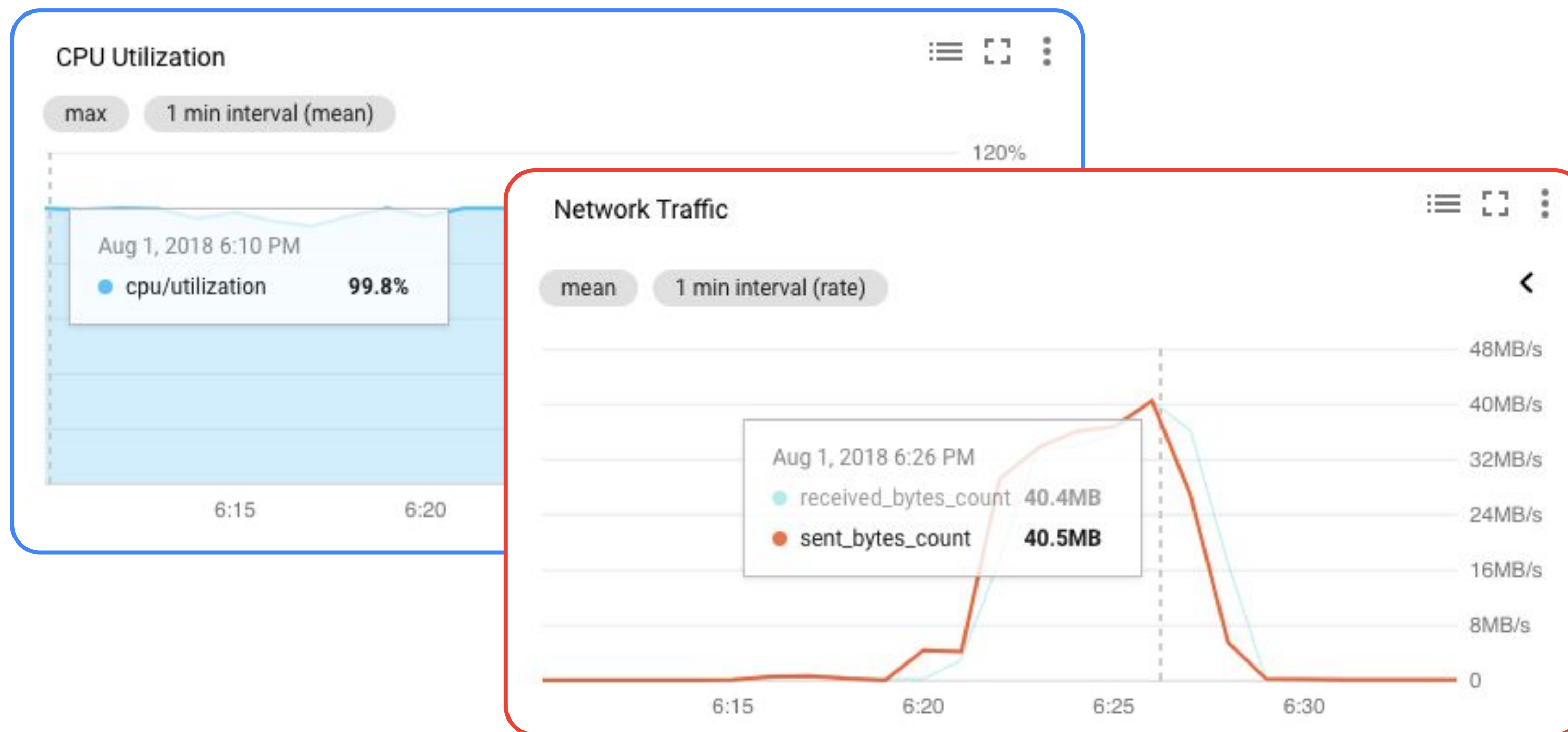
Project name	Project ID	Project role ↓
AllEnvironments	allenvironments	Scoping project
Production	production-315019	Monitored project
Staging	staging-315019	Monitored project

[Add Cloud projects to metrics scope](#)

The projects listed below can view this project's metrics

This project's metrics are visible only in this project

Dashboards visualize utilization and network traffic



Alerting policies can notify you of certain conditions



Creating an alerting policy

Create new alerting policy

1 Conditions

Basic Conditions

HTTP check on instance summer01
Violates when: Uptime Check Health on Instance (GCE) summer01 fails
[Edit](#) [Delete](#)

+ Add Another Condition

2 Notifications (optional)

When alerting policy violations occur, you will be notified via these channels. [Learn more](#)

Email demo@example.com X

+ Add Another Notification

3 Documentation (optional)

When email notifications are sent, they'll include any text entered here. This can convey useful information about the problem and ways to approach fixing it.

[Edit](#) [Preview](#) [Markdown Formatting Help](#)

 Main Server health check failed
+ Server named summer01 failed a Stackdriver uptime check
+ IP Address of the server is: 104.197.58.79

4 Name this policy

A policy's name is used in identifying which policies were triggered, as well as managing configurations of different policies.

Uptime Check Policy

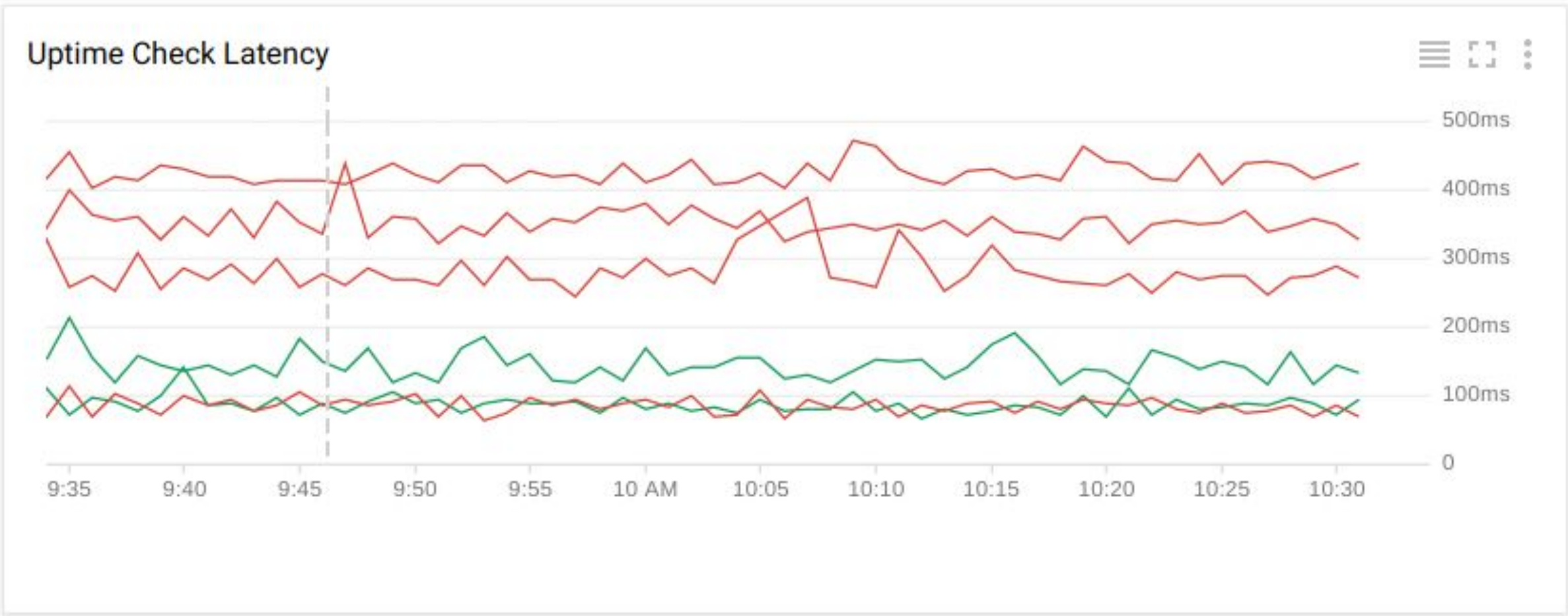
Save Policy

Cancel

Uptime checks test the availability of your public services

CHECKS	VIRGINIA	OREGON	IOWA	BELGIUM	SINGAPORE	SAO PAULO	POLICIES
Instance 1							
Instance 2							
Instance 3							

Uptime check example



Uptime [?]
100.000%

Outages [?]
0 minutes

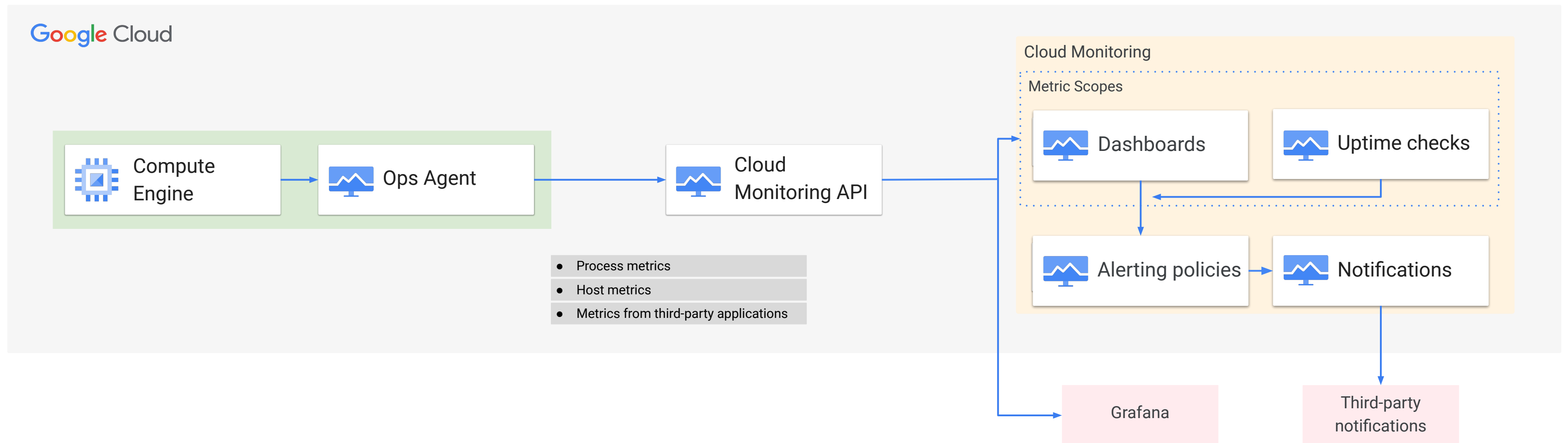
Location Results **All locations passed**

Check config

Check Type	HTTP
Resource	summer01
Path	/
Check Every	1 minute
Port	80
Locations	Global
Timeout	10 seconds

What is Ops Agent?

Ops Agent gathers system and application metrics from **VM instances** and sends them to Monitoring.

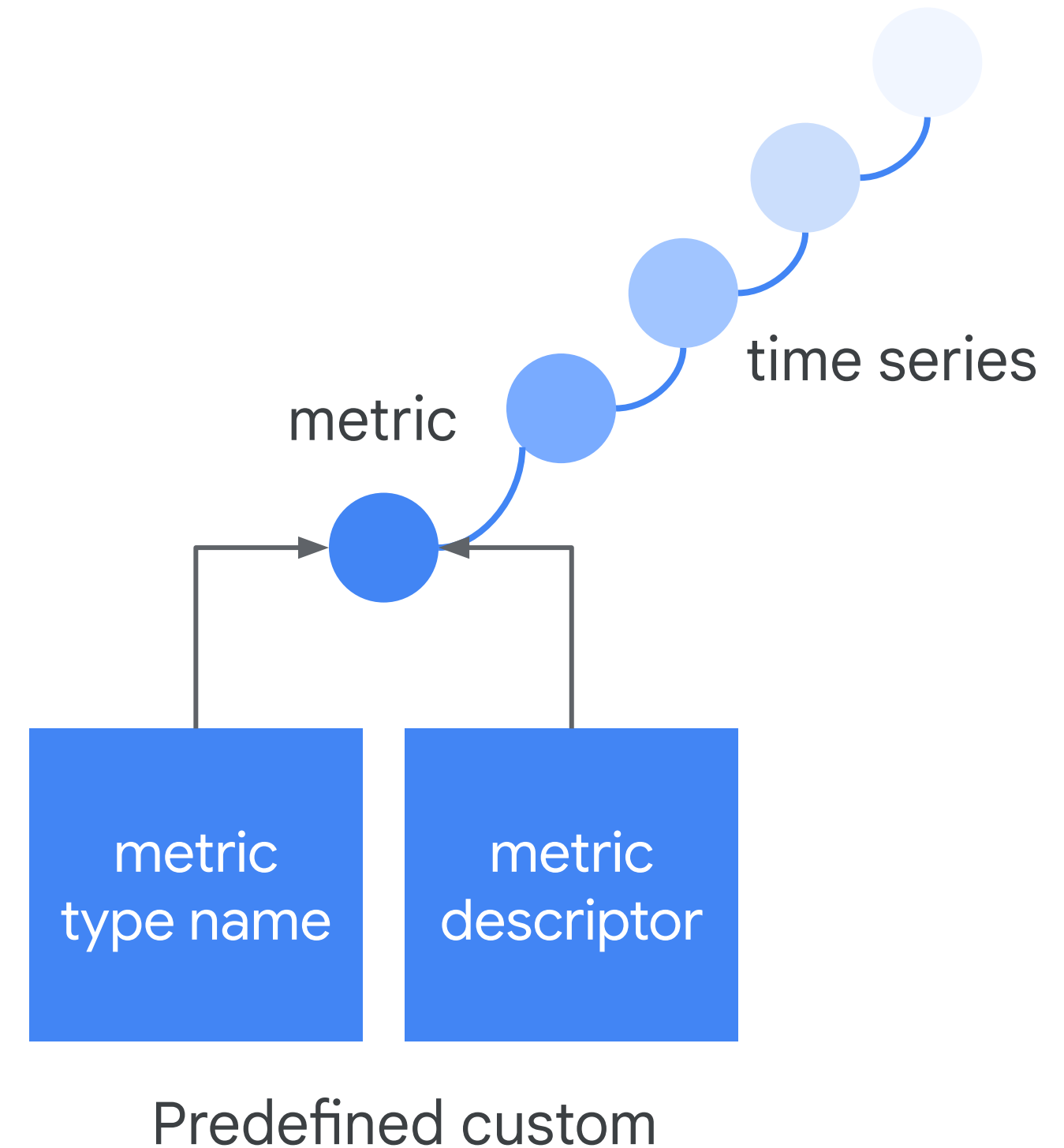


Custom metrics

Custom metric example in Python:

```
client = monitoring.Client()
descriptor = client.metric_descriptor(
    'custom.googleapis.com/my_metric',

metric_kind=monitoring.MetricKind.GAUGE,
    value_type=monitoring.ValueType.DOUBLE,
    description='This is a simple example
of a custom metric.')
descriptor.create()
```



Autoscale to maintain a metric at a target value

To maintain a metric at a target value, specify a utilization target.

If the metric comes from each VM in your MIG,



the average metric value across all VMs is compared with the utilization target.

If the metric applies to the whole MIG and does not come from the VMs in your MIG,



the metric value is compared with the utilization target.

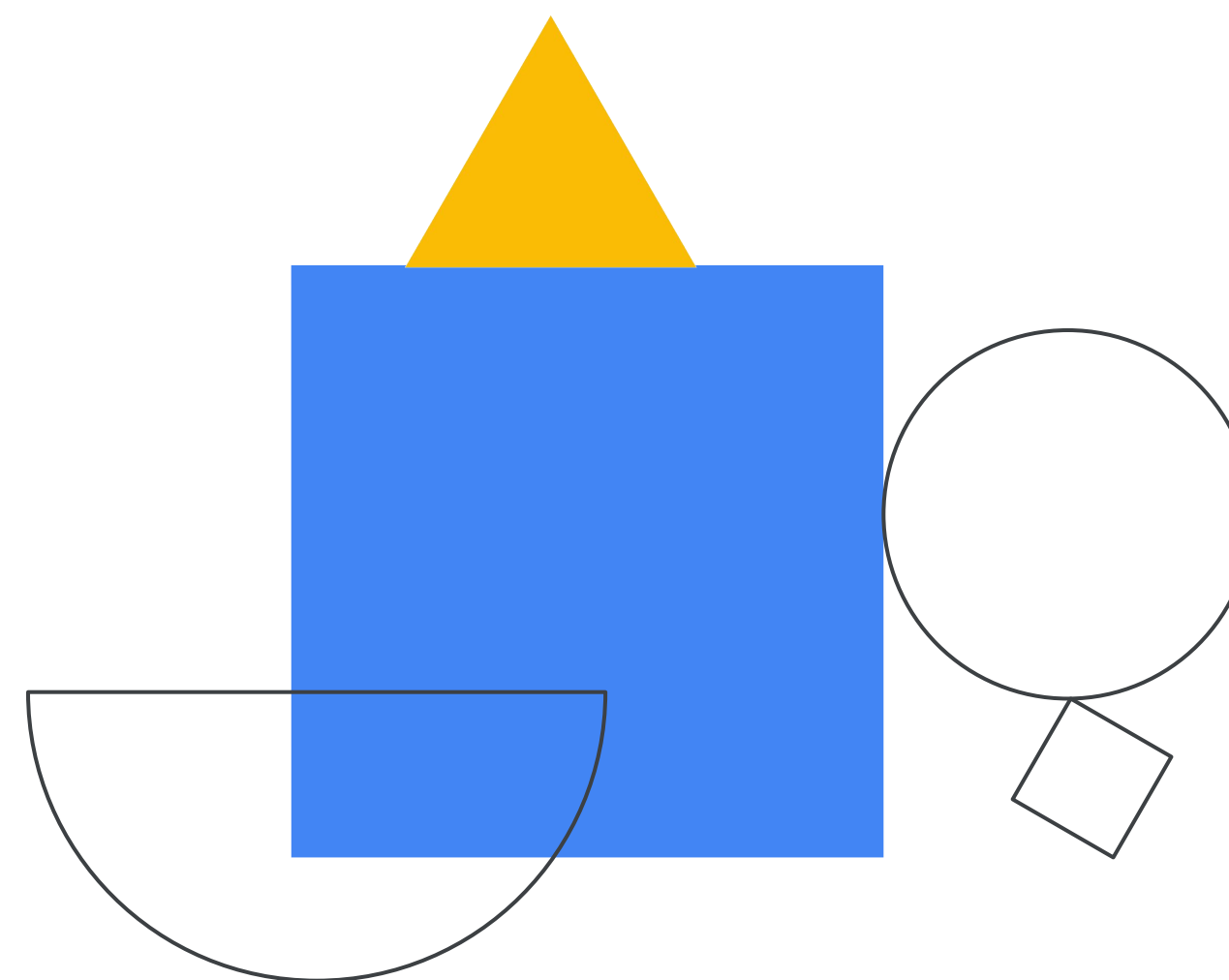
If your metric has multiple values,



apply a filter to autoscale using an individual value from the metric.

Lab Intro

Resource Monitoring



Lab objectives

01

Enable Cloud Monitoring

02

Add charts to dashboards

03

Create alerts with multiple conditions

04

Create resource groups

05

Create uptime checks



03



Logging

Logging



Logging

- Platform, systems, and application logs
 - API to write to logs
 - 30-day retention
- Log search/view/filter
- Log-based metrics
- Monitoring alerts can be set on log events
- Data can be exported to Cloud Storage, BigQuery, and Pub/Sub

Analyze logs in BigQuery and visualize in Looker Studio

RUN QUERY

Save Query

Save View

Format Query

Show Options

ResultsDetails

Download as CSV

Row	vpc_name	bytes	subnetwork_name	dest_ip	src_ip	dest_port	protocol	
1	vpc-demo	23529368	vpc-demo-web	74.125.28.95	10.1.1.2	443.0	6.0	
2	vpc-demo	15237089	vpc-demo-web	74.125.197.95	10.1.1.2	443.0	6.0	
3	vpc-demo	4390076	vpc-demo-web	74.125.135.95	10.1.1.2	443.0	6.0	
4	vpc-demo	1606002	vpc-demo-web	74.125.199.95	10.1.1.2	443.0	6.0	
5	vpc-demo	1479280	vpc-demo-web	108.177.98.95	10.1.1.2	443.0	6.0	
6	vpc-demo	828169	vpc-demo-web	173.194.202.95	10.1.1.2	443.0	6.0	
7	null	150991	null	10.1.1.2	151.101.52.204	48668.0	6.0	
8	null	18024	null	10.1.1.2	74.125.199.95	37910.0	6.0	
9	null	17573	null	10.1.1.2	74.125.199.139	58010.0	6.0	
10	null	16687	null	10.1.1.2	74.125.28.95	46118.0	6.0	

Table

JSON



BigQuery



Looker Studio



Error Reporting

Error Reporting

Aggregate and display errors for running cloud services

- Error notifications
- Error dashboard
- App Engine, Apps Script, Compute Engine, Cloud Run, Cloud Run functions, GKE, Amazon EC2
- Go, Java, .NET, Node.js, PHP, Python, and Ruby

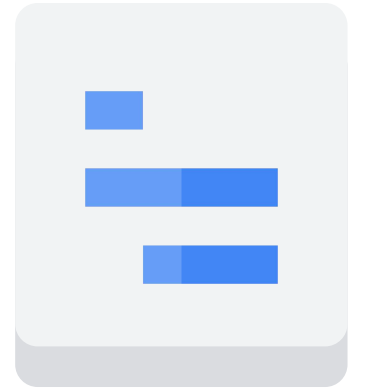


Error Reporting



Tracing

Tracing



Trace

Tracing system

- Displays data in near real-time
- Latency reporting
- Per-URL latency sampling

Collects latency data

- App Engine
- Global external Application Load Balancers
- Applications instrumented with the Cloud Trace SDKs



Profiling

Profiling

- Continuously analyze the performance of CPU or memory-intensive functions executed across an application.
- Uses statistical techniques and extremely low-impact instrumentation.
- Runs across all production instances.
- Java, Go, Node.js, and Python



Profiler



Partner Integrations

bluemedora[®]

 bmc

(x) matters[®]

 sumologic[®]

 **tenable**[®]
network security

 OpsGenie

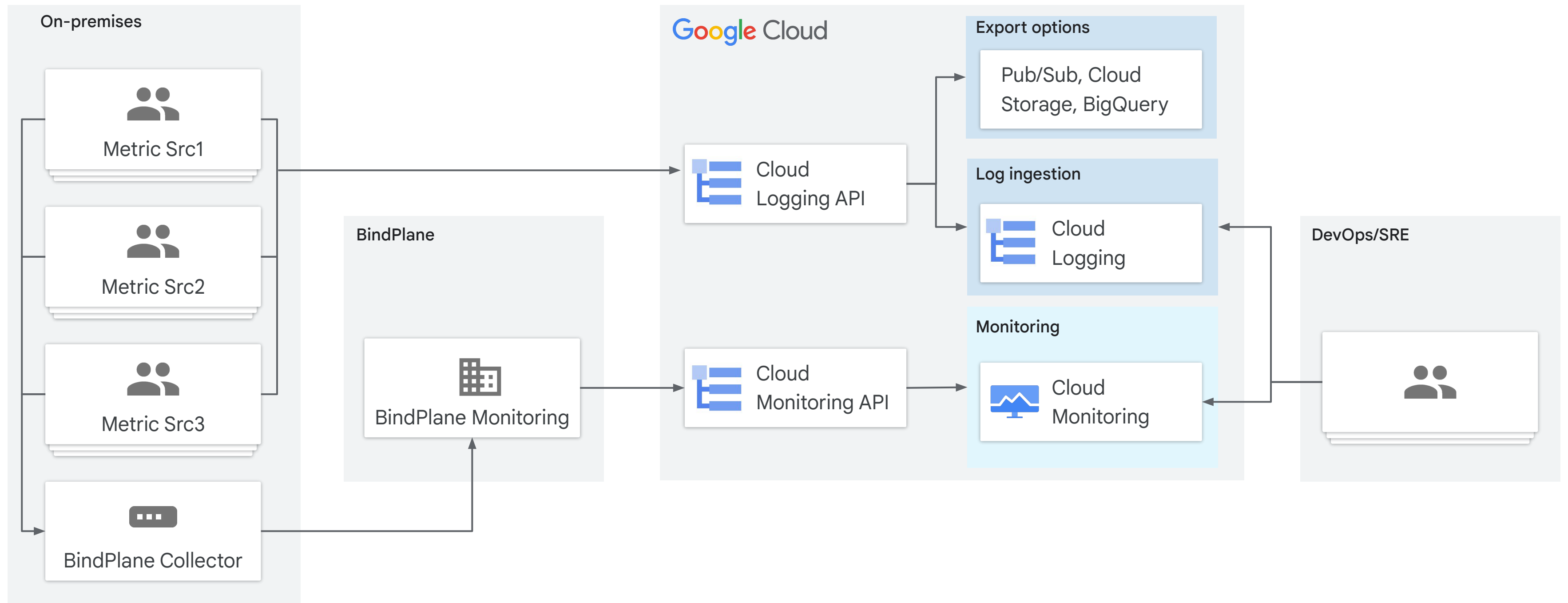
splunk>enterprise

 netskope

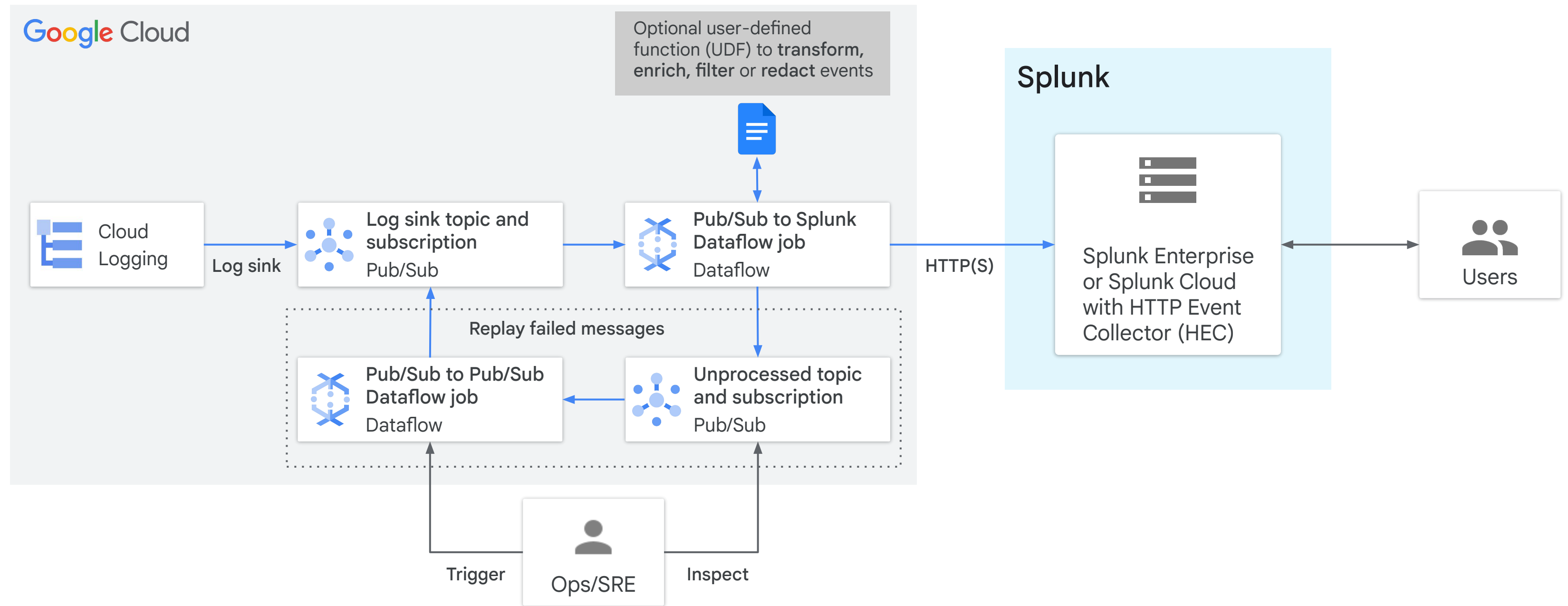
 insightfinder

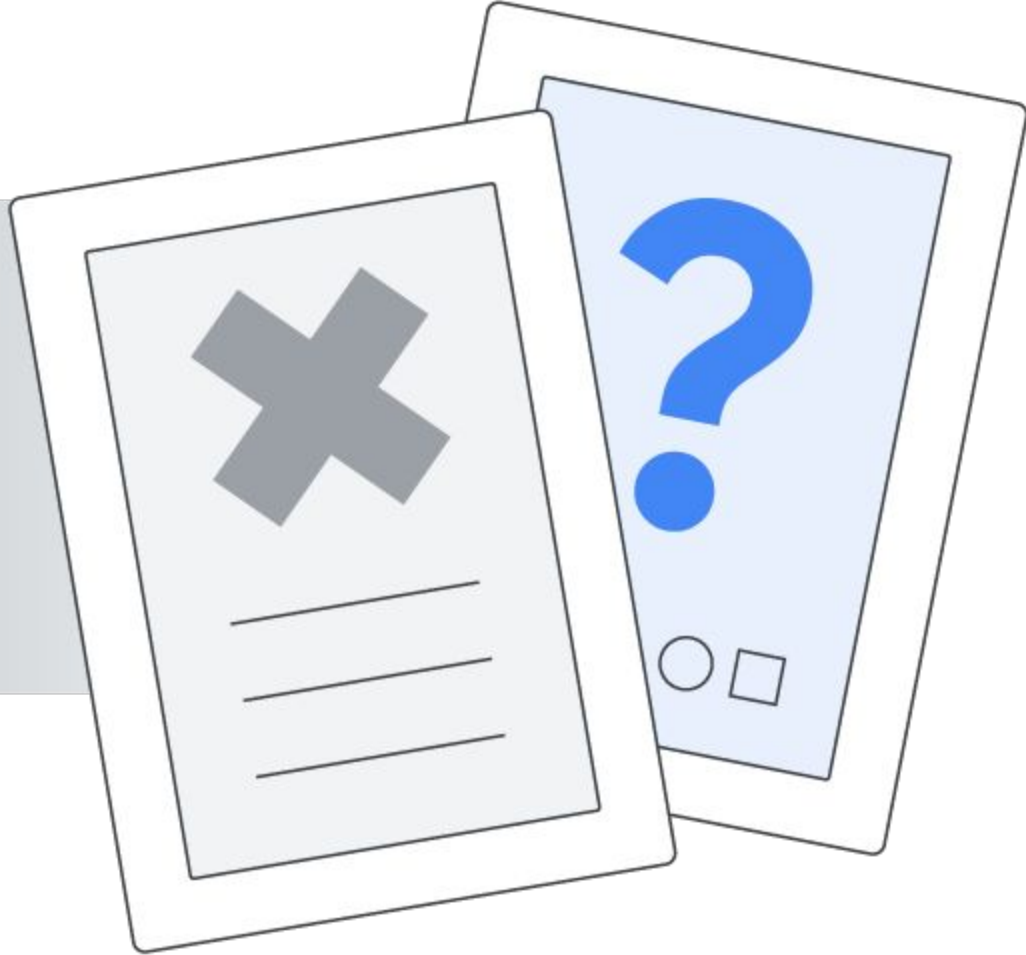
pagerduty

Reference architecture for logging and monitoring on-premises resources with BindPlane



Reference architecture for log data flows from Google Cloud to Splunk





Quiz



Question #1

Question

What is the foundational process at the base of Google's Site Reliability Engineering (SRE)?

- A. Capacity planning
- B. Testing and release procedures
- C. Monitoring
- D. Root cause analysis

Question #1

Answer

What is the foundational process at the base of Google's Site Reliability Engineering (SRE)?

- A. Capacity planning
- B. Testing and release procedures
- C. Monitoring
- D. Root cause analysis



Question #2

Question

What is the purpose of the Cloud Trace service?

- A. Reporting on latency as part of managing performance
- B. Reporting on Google Cloud system errors
- C. Reporting on application errors
- D. Reporting on Google Cloud resource consumption as part of managing performance

Question #2

Answer

What is the purpose of the Cloud Trace service?

- A. Reporting on latency as part of managing performance
- B. Reporting on Google Cloud system errors
- C. Reporting on application errors
- D. Reporting on Google Cloud resource consumption as part of managing performance



Question #3

Question

Google Cloud Observability integrates several technologies, including monitoring, logging and error reporting, that are commonly implemented in other environments as separate solutions using separate products. What are key benefits of integration of these services?

- A. Reduces over head, reduces noise, streamlines use, and fixes problems faster
- B. Ability to replace one tool with another from a different vendor
- C. Detailed control over the connections between the technologies
- D. Better for Google Cloud only so long as you don't need to monitor other applications or clouds

Question #3

Answer

Google Cloud Observability integrates several technologies, including monitoring, logging and error reporting, that are commonly implemented in other environments as separate solutions using separate products. What are key benefits of integration of these services?

- A. Reduces over head, reduces noise, streamlines use, and fixes problems faster
- B. Ability to replace one tool with another from a different vendor
- C. Detailed control over the connections between the technologies
- D. Better for Google Cloud only so long as you don't need to monitor other applications or clouds



Review: Resource Monitoring

