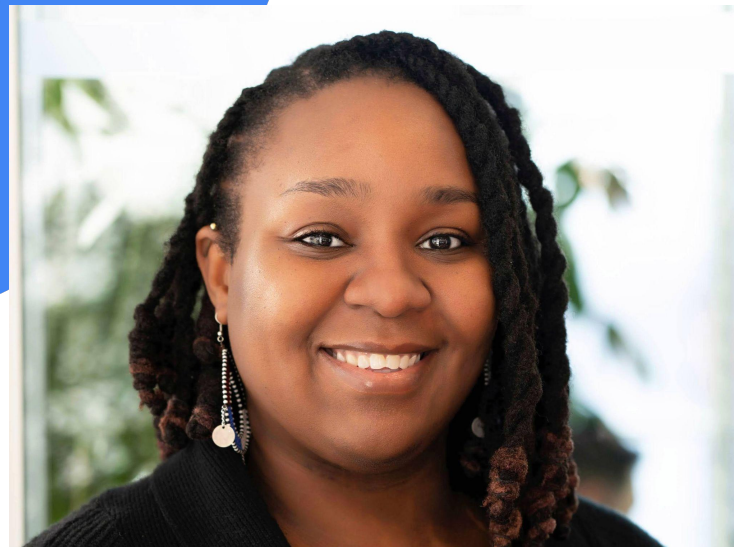
SPTL208 - Out of the Box Thinking: Automated Observability with Datadog and Project44

Google Cloud Next 25



Ajuna Kyaruzi

Manager, Advocacy - SRE & Platform
Datadog

Energy Efficiency

Datadog Platform

Infrastructure

- Infrastructure Monitoring
- Metrics
- Cloud Network Monitoring
- Network Device Monitoring
- Container Monitoring
- Serverless
- Cloud Cost Management
- Cloudcraft

Applications

- Application Performance Monitoring
- Distributed Tracing
- Continuous Profiler
- DB Monitoring
- Universal Service Monitoring
- Data Streams Monitoring
- Data Jobs Monitoring
- LLM Observability
- Error Tracking

Digital Experience

- Synthetics
- Mobile App Testing
- Browser Real User Monitoring
- Mobile Real User Monitoring
- Session Replay
- Error Tracking

Logs

- Log Management
- Observability Pipelines
- Audit Trail
- Log Forwarding
- Error Tracking

Cloud Security

- Cloud Security Management
- Code Security
- Cloud SIEM
- Sensitive Data Scanner
- Workload Protection
- App and API Protection

Software Delivery

- CI Visibility
- Test Optimization
- Continuous Testing

Cloud Service Management

- On-Call
- Incident Management
- Event Management
- Workflow Automation
- App Builder

AI

Natural Language Querying • Root Cause Analysis • Anomaly Detection • Impact Analysis • Proactive Alerts • Autonomous Investigations • Bits AI

Shared Platform Services

Dashboards • CoScreen • Teams • Agent • OpenTelemetry • Notebooks • Software Catalog • IDE Plugins • ChatOps • SLOs • Case Management

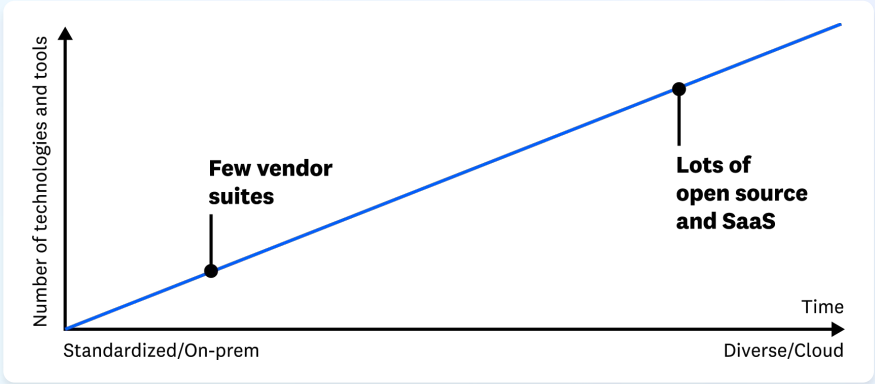
UNIFIED METRICS, LOGS, TRACES, SESSIONS

850+ INTEGRATIONS

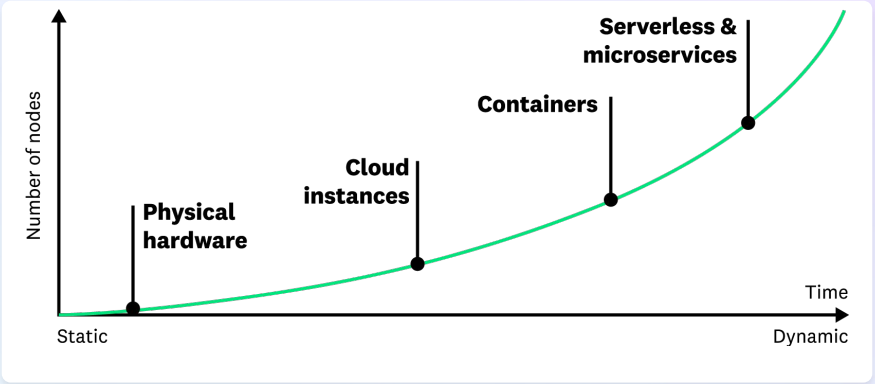
DATADOG

The problem: an explosion of complexity

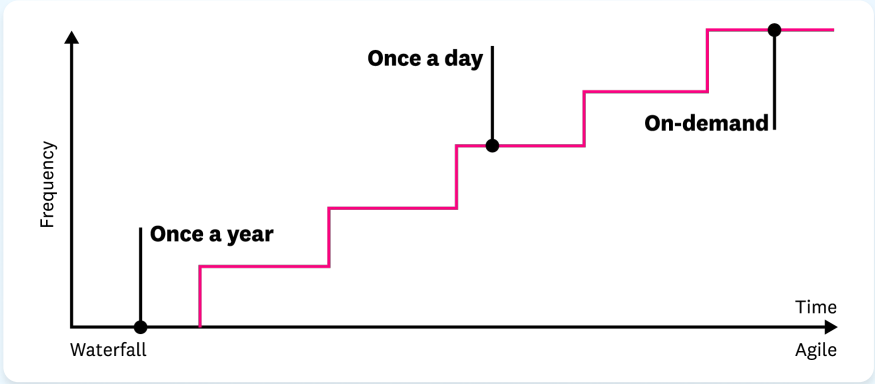
Diversity of technologies in use



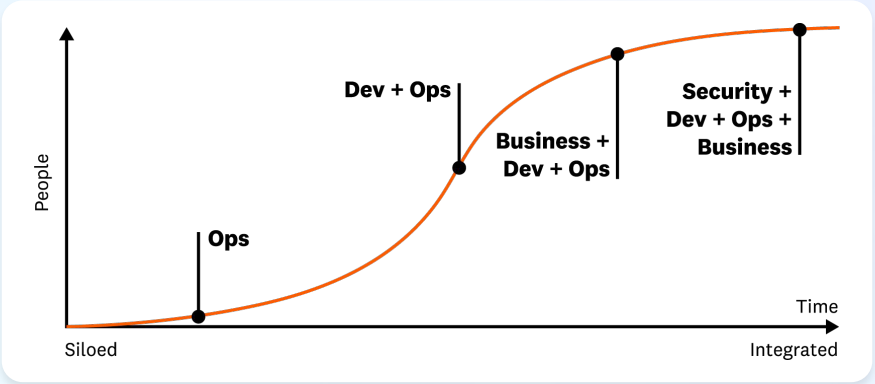
Scale in number of computing units



Frequency of release

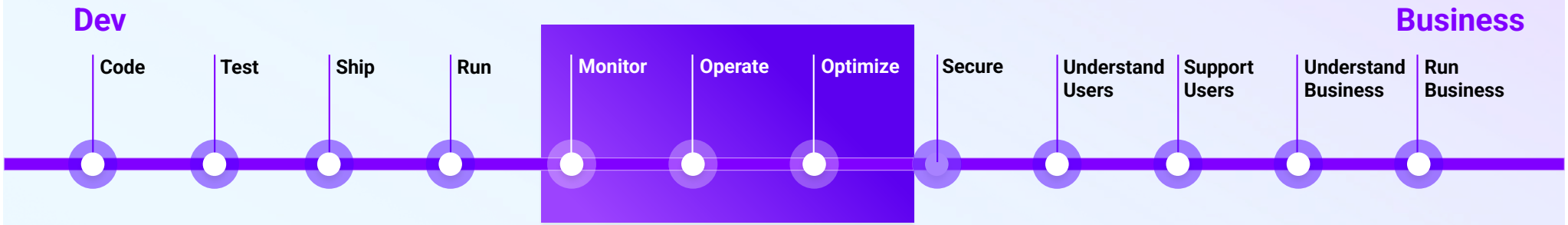


Number of people involved



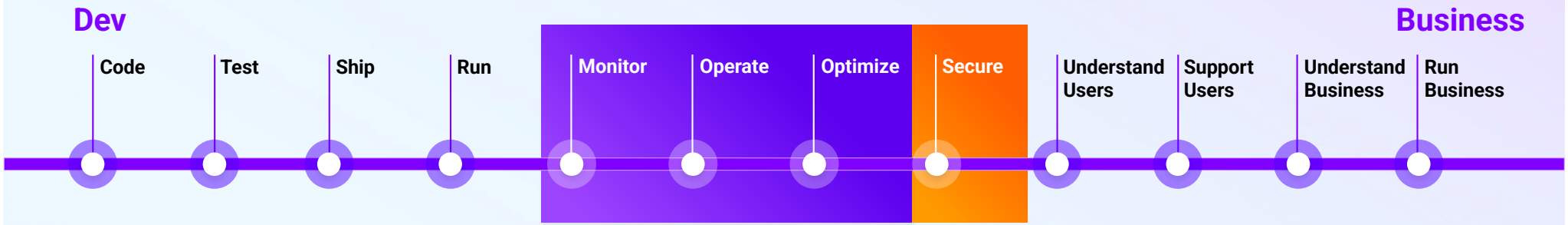
DATADOG

The Datadog Platform



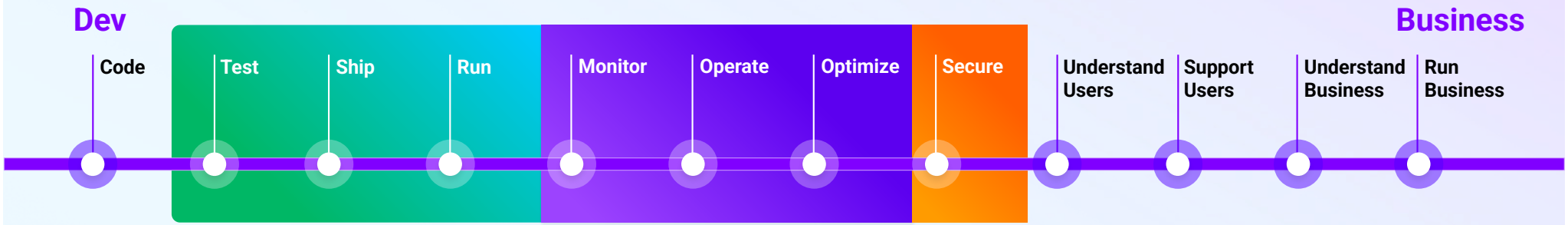
DATADOG

The Datadog Platform



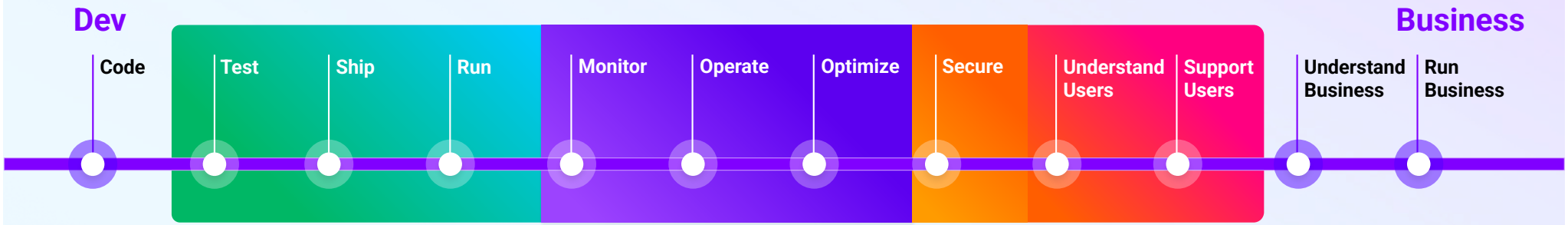
DATADOG

The Datadog Platform



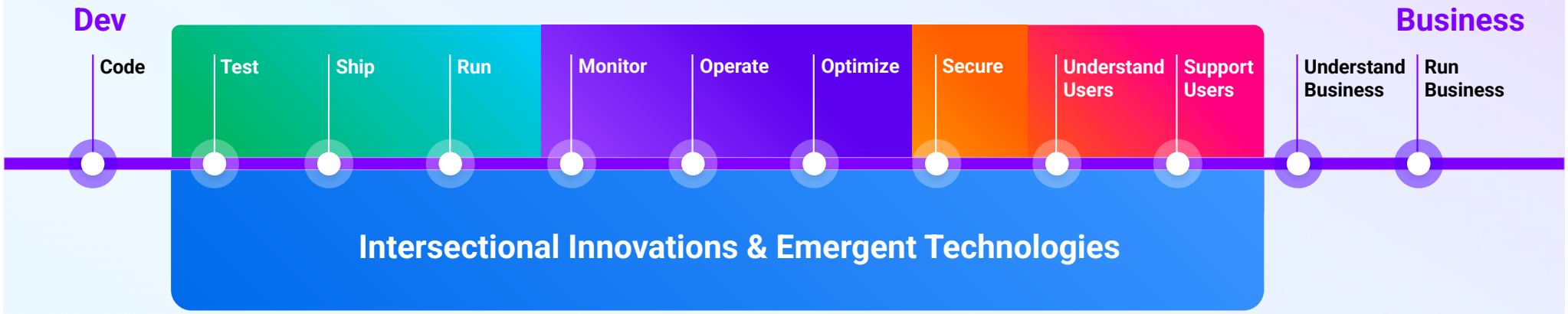
DATADOG

The Datadog Platform



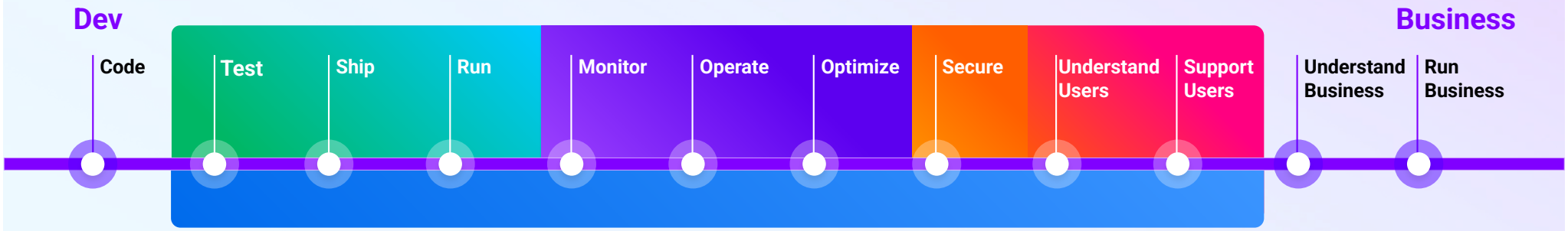
DATADOG

The Datadog Platform



DATADOG

The Datadog Platform



Software Delivery

- CI Visibility
- Test Optimization
- Continuous Testing

Monitor & Operate

- Infra Monitoring
- Network Monitoring
- APM
- Synthetics
- Log Mgmt
- Universal Service Monitoring
- Observability Pipelines
- LLM Observability

Optimize

- Continuous Profiler
- Cloud Cost Mgmt
- Database Monitoring
- Data Streams Monitoring
- Data Jobs Monitoring

Secure

- Cloud Security Mgmt
- Code Security
- Cloud SIEM
- Sensitive Data Scanner
- Workload Protection
- App and API Protection

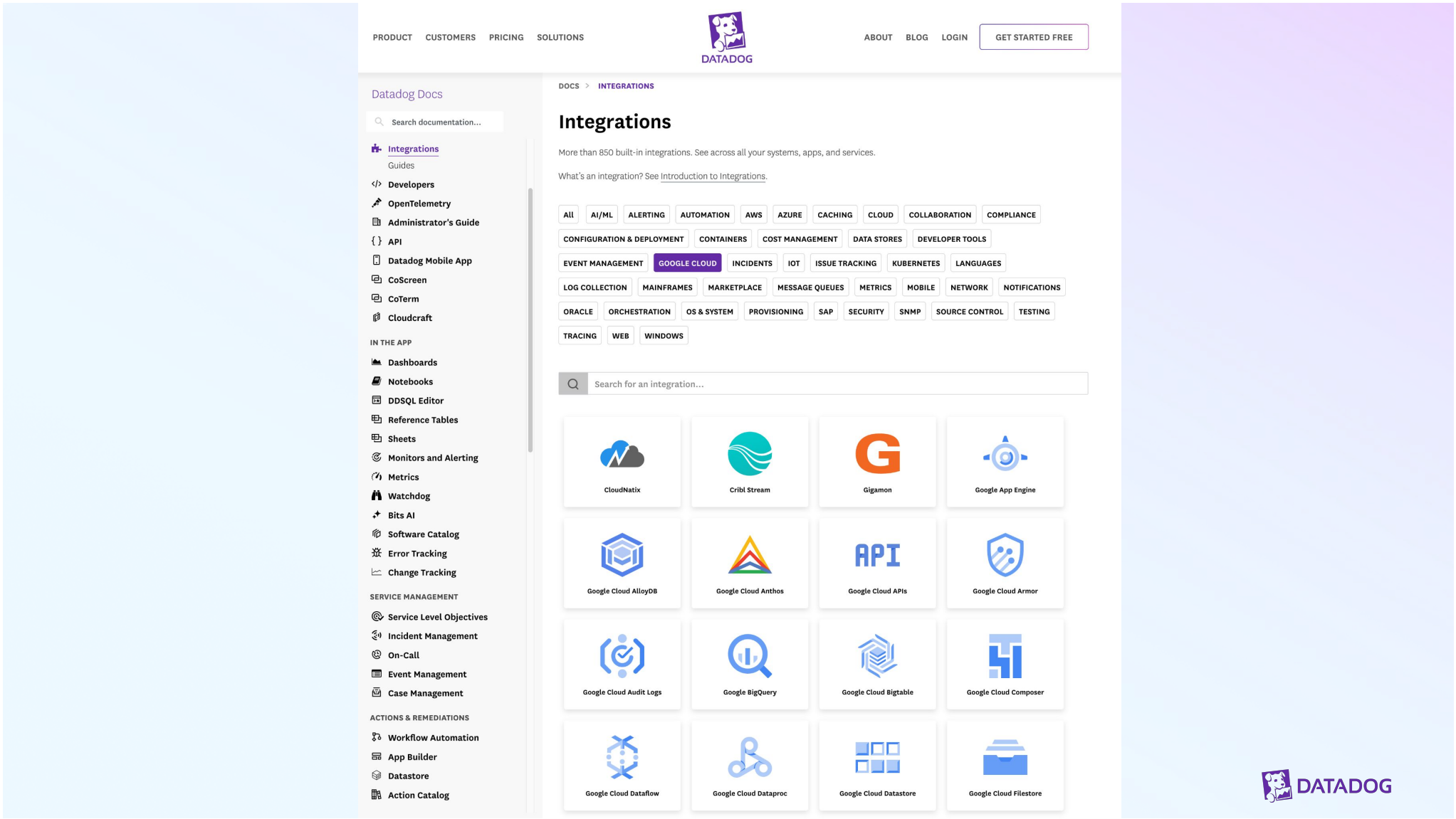
Analyze

- RUM
- Heatmap/ Clickmap/ Scrollmap
- Mobile App Testing
- Session Replay

Cloud Service Management

- Incident Management
- On-Call
- Workflow Automation
- App Builder
- Software Catalog
- Resource Catalog
- Case Management
- Event Management

DATADOG


PRODUCT
CUSTOMERS
PRICING
SOLUTIONS
DATADOG
ABOUT
BLOG
LOGIN
GET STARTED FREE
Datadog Docs
Search documentation...
Integrations
Guides
Developers
OpenTelemetry
Administrator's Guide
API
Datadog Mobile App
CoScreen
CoTerm
Cloudcraft
IN THE APP
Dashboards
Notebooks
DDSQL Editor
Reference Tables
Sheets
Monitors and Alerting
Metrics
Watchdog
Bits AI
Software Catalog
Error Tracking
Change Tracking
SERVICE MANAGEMENT
Service Level Objectives
Incident Management
On-Call
Event Management
Case Management
ACTIONS & REMEDIATIONS
Workflow Automation
App Builder
Datastore
Action Catalog
DOCS > INTEGRATIONS
Integrations
More than 850 built-in integrations. See across all your systems, apps, and services.
What's an integration? See Introduction to Integrations.
All
AI/ML
ALERTING
AUTOMATION
AWS
AZURE
CACHING
CLOUD
COLLABORATION
COMPLIANCE
CONFIGURATION & DEPLOYMENT
CONTAINERS
COST MANAGEMENT
DATA STORES
DEVELOPER TOOLS
EVENT MANAGEMENT
GOOGLE CLOUD
INCIDENTS
IOT
ISSUE TRACKING
KUBERNETES
LANGUAGES
LOG COLLECTION
MAINFRAMES
MARKETPLACE
MESSAGE QUEUES
METRICS
MOBILE
NETWORK
NOTIFICATIONS
ORACLE
ORCHESTRATION
OS & SYSTEM
PROVISIONING
SAP
SECURITY
SNMP
SOURCE CONTROL
TESTING
TRACING
WEB
WINDOWS
Search for an integration...
CloudNatix
Cribl Stream
Gigamon
Google App Engine
Google Cloud AlloyDB
Google Cloud Anthos
API
Google Cloud APIs
Google Cloud Armor
Google Cloud Audit Logs
Google BigQuery
Google Cloud Bigtable
Google Cloud Composer
Google Cloud Dataflow
Google Cloud Dataproc
Google Cloud Datastore
Google Cloud Filestore
DATADOG

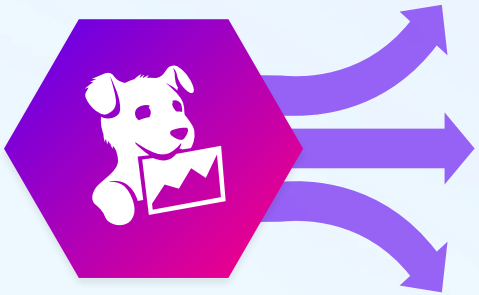
Millions of hosts

Tens of millions of containers

Billions of serverless functions

Trillions of metrics

Trillions of logs



PLATFORM SERVICES

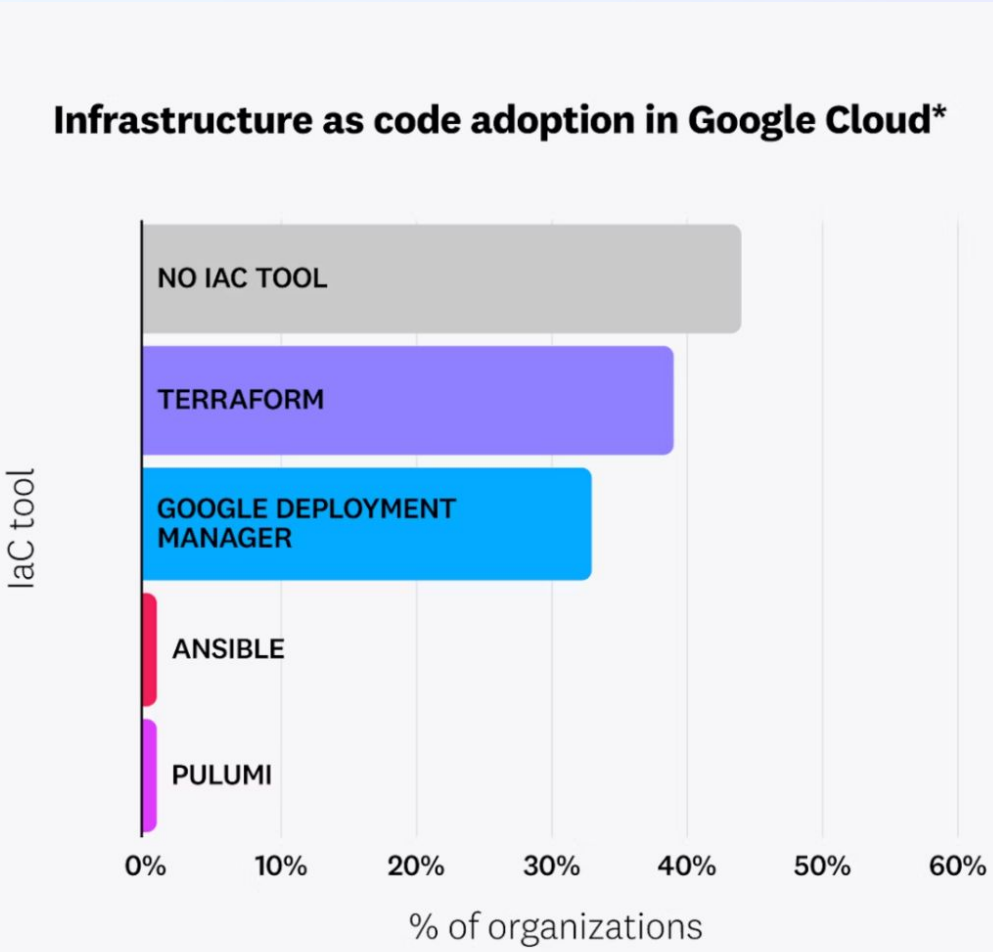
- Dashboards
- Agents
- Collaboration
- Mobile
- Workflows
- Watchdog AI
- Open Telemetry

PRODUCTS / USE CASES

- Infrastructure
- APM
- DBM
- Log Management
- Cloud SIEM
- CI Visibility
- Continuous Profiler
- RUM
- Network
- Synthetics
- Cloud Security Management
- App Security Management
- Observability Pipelines
- Cloud Cost Management
- ... and more

DATADOG

DATADOG



IN GOOGLE CLOUD, NEARLY HALF OF ORGANIZATIONS DON'T USE ANY INFRASTRUCTURE AS CODE TOOL

*IaC tools are not mutually exclusive

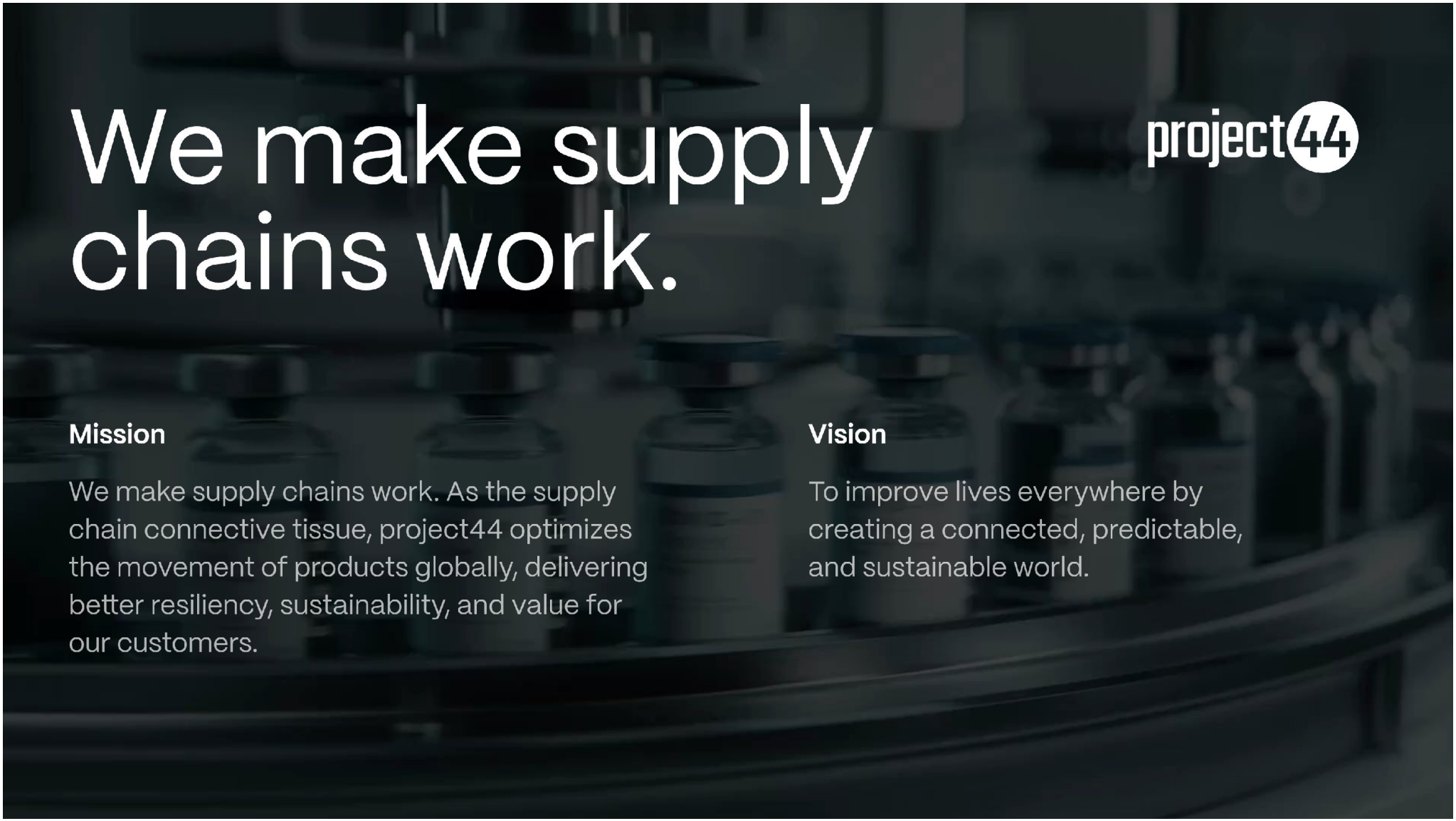
Source: Datadog

DATADOG

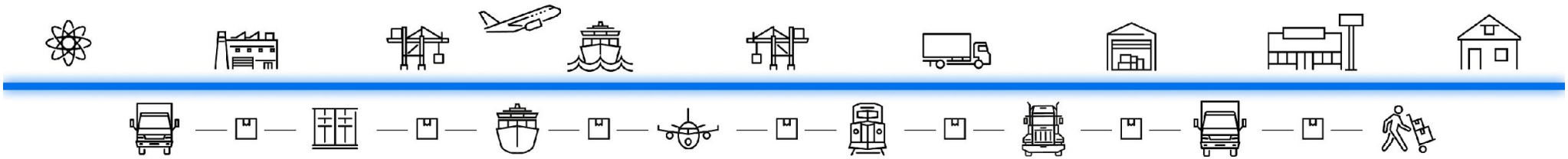
Infrastructure as Code

Manan Kothari

Staff Site Reliability Engineer
Project44


We make supply chains work.
project44
Mission
We make supply chains work. As the supply chain connective tissue, project44 optimizes the movement of products globally, delivering better resiliency, sustainability, and value for our customers.
Vision
To improve lives everywhere by creating a connected, predictable, and sustainable world.

We built the connective tissue to optimize supply chains



1. All modes and geos

E2E shipment visibility for inbound and outbound shipments, across all modes, from door to door, anywhere in the world

2. Inventory Visibility

Visibility of inventory in motion by order—whether it's a PO, STO, or SO—and by SKU

3. Transfer point efficiency

Capabilities to optimize transfer points including ports, terminals, and yards, along with solutions for shipment initiation and eCommerce delivery

4. AI-powered supply chain intelligence

Enhanced data quality, advanced ETAs, exception management and collaboration, supply chain insights and analytics

Observability on it's own can be overwhelming.

We want to reduce the overhead for our developers.

1

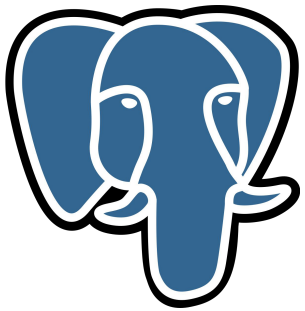
Building Out A Self-Service Platform

What is Self-Service Infrastructure

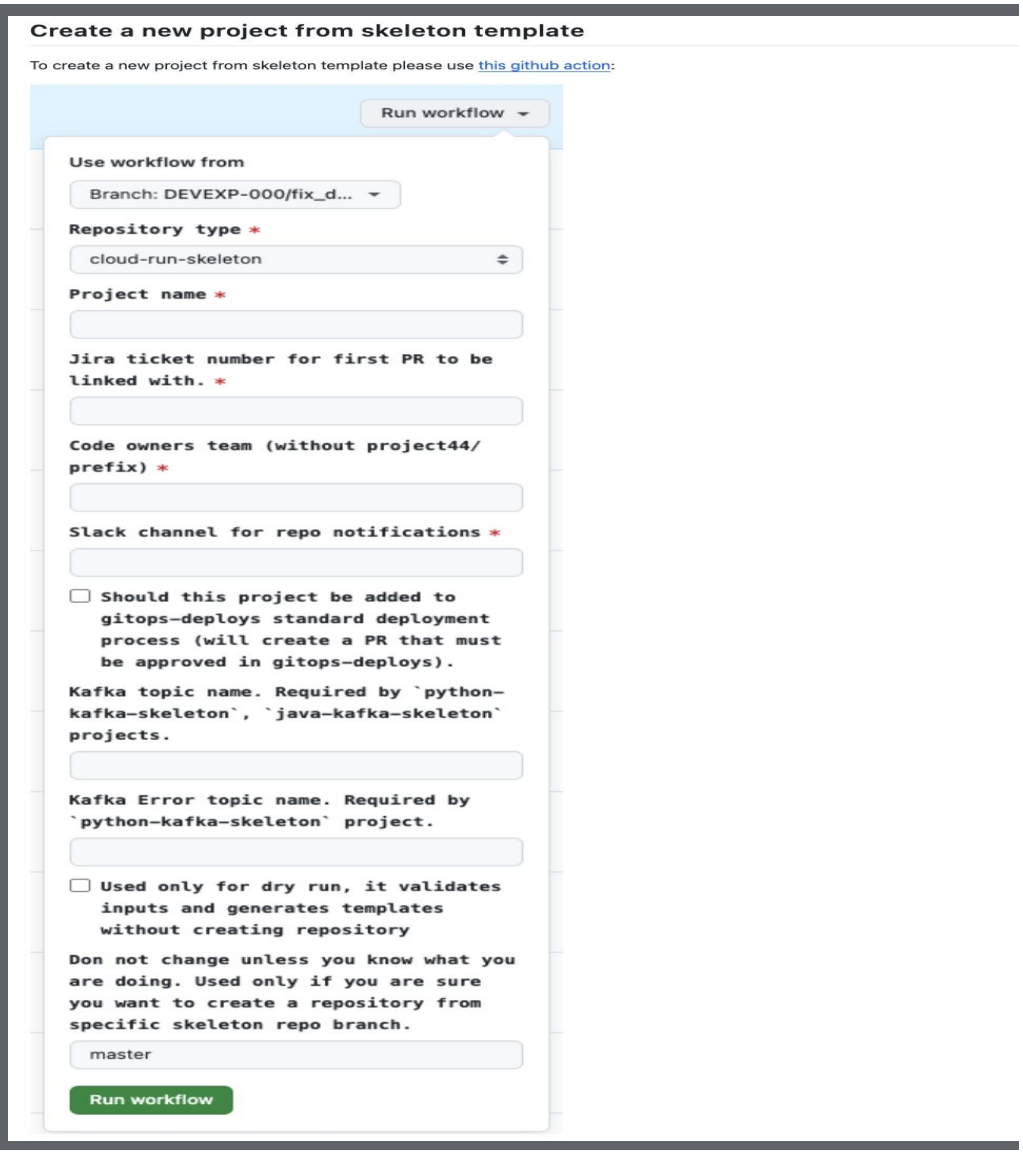
- We've built "skeleton" repos for java, python, nodeJS allowing our developers to spin up a new microservice and deploy it to GKE in minutes

- We've built "skeleton" repos for java, python, nodeJS allowing our developers to spin up a new microservice and deploy it to GKE in minutes

- These include optional basic building blocks like getting started with postgres, kafka or other technologies with the appropriate infrastructure and tooling like liquibase



Setting Up a New Microservice



CV-2188 Initial PR #1

Merged **dmorton2297** merged 6 commits into `master` from `CV-2188-initial-setup` on Aug 15, 2024

Jira Resolved **CV-2188 - Scaffolding for Task Management Core Service** Reported by **Matt Galloway** and assigned to M

Conversation 2 | Commits 6 | Checks 3 | Files changed 45

opsteamp44 commented on Aug 15, 2024

This PR was opened by https://github.com/project44/repo-skeletons with config:

```
{
  "codeOwners": "collaborative-visibility",
  "deployType": "",
  "projectName": "task-service",
  "projectType": "java-microservice-skeleton",
  "gitopsDeploy": true,
  "jiraTicketNumber": "CV-2188",
  "slackChannel": "cv-team-deploys",
  "skeletonBranch": "master"
}
```

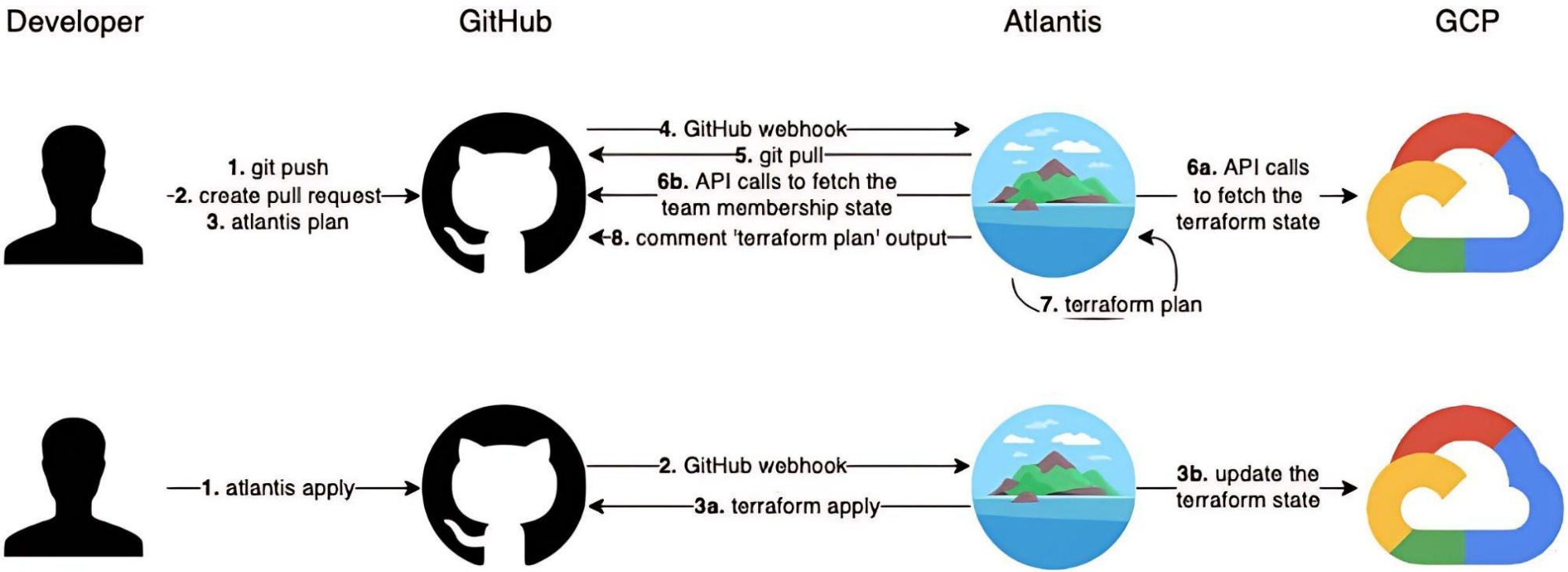
opsteamp44 added 6 commits 7 months ago

`CV-2188 Initial PR [batch 1]` `af9ddd5`

Automating Terraform in CI

Setting up Atlantis on GKE and Github.

We run one atlantis deployment on a shared management cluster used by all of our various terraform repos (for creating infrastructure in GCP or Elastic Cloud or Mongo Atlas, etc, alerts in datadog and more.



2

Automating Observability

Using the admission controller + helm to instrument tracing

- We use a central helm template that all of our microservices pull from and can extend

Using the admission controller + helm to instrument tracing

- We use a central helm template that all of our microservices pull from and can extend
- We setup defaults for injecting the tracing library via the admission controller

Using the admission controller + helm to instrument tracing

- We use a central helm template that all of our microservices pull from and can extend
- We setup defaults for injecting the tracing library via the admission controller
- Enabling the right env variables to enable various observability features

Using the admission controller + helm to instrument tracing

- We use a central helm template that all of our microservices pull from and can extend
- We setup defaults for injecting the tracing library via the admission controller
- Enabling the right env variables to enable various observability features
- Integrating with Datadog's postgres, mongodb, elasticsearch, redis, and other integrations

```yaml
datadog:
  environment: ""
  customMetricsEnabled: false
  customMetricsList: []
  maxMetricsReturned: 10000
  dataStreamsEnabled: true
  dbmPropagationMode: full
  dynamicInstrumentationEnabled: true
  elasticsearchBodyEnabled: true
  elasticsearchEnabled: true
  elasticsearchParamsEnabled: true
  exceptionDebuggingEnabled: false
  gitCommitSha: ""
  gitRepoUrl: ""
  headerTags: 'x-request-id:p44.request_id,x-real-ip:p44.request_ip,x-tenant-id:p44.user_context.tenant_id,x-tenant-uuid:p44.user_context.tenant_uuid,x-user-id:p44.user_context.user_id,x-p44-vendor-code:p44.vendor_code,x-p
  httpServerQueryString: true
  httpClientQueryString: true
  hystrixTagsEnabled: true
  # see releases availabe here https://github.com/DataDog/dd-trace-java/releases
  javaLibVersion: v1.41.1
  jdbcEnabled: true
  libraryInjection: true
  logInjection: true
  mongodbEnabled: true
  # set to true to enable otel trace collection - https://docs.datadoghq.com/opentelemetry/otlp_ingest_in_the_agent/?tab=kubernetes#sending-opentelemetry-traces-and-metrics-to-datadog-agent
  otelEnabled: true
  postgresEnabled: true
  profiling: false
  profilingAllocationEnabled: false
  profilingHeapEnabled: false
  profilingHeapHistogramEnabled: false
  profilingDirectAllocationEnabled: false
  profilingLiveHeapEnabled: false
  redisEnabled: true
  removeIntegrationServiceNamesEnabled: true
  # set to true if you want to inject headers, default behavior is just to extract
  tracePropagationExtractAndInject: false
  tracePropagationStyle: b3multi,b3single,Datadog
```

```
# see releases availabe here https://github.com/DataDog/dd-trace-java/releases
javaLibVersion: v1.41.1
jdbcEnabled: true
libraryInjection: true
logInjection: true
mongodbEnabled: true
# set to true to enable otel trace collection - https://docs.datadoghq.com/opentelemetry/otlp_ingest_in_the_agent/?tab=kubernetes#sending-opentelemetry-traces-and-metrics-to-datadog-agent
otelEnabled: true
postgresEnabled: true
```

3

Setting up Monitors

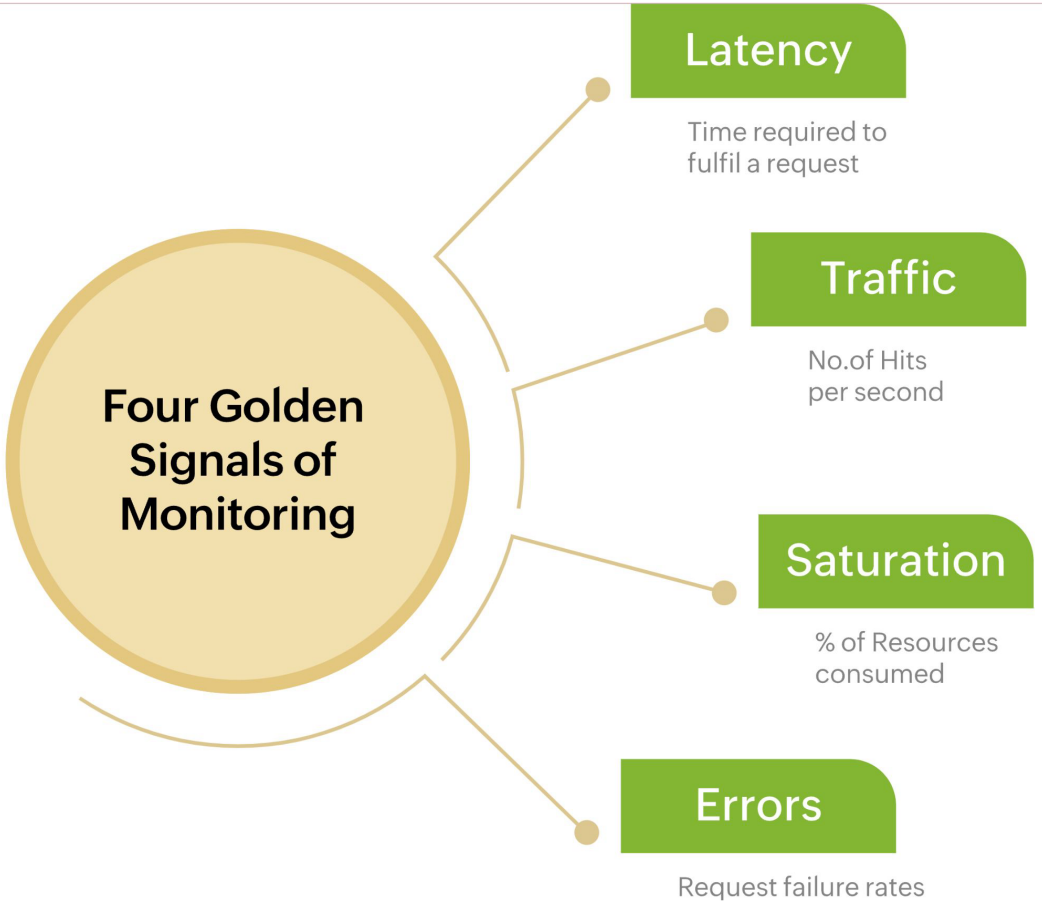
Repo Design

How we use GitHub + Atlantis + Terraform to manage observability

Building Alerting Modules

Instead of making developers build alerts on their own, we've built standardized modules based on golden signals metrics as well as other business specific monitors, allowing developers to just provide minimal details

Golden Signals

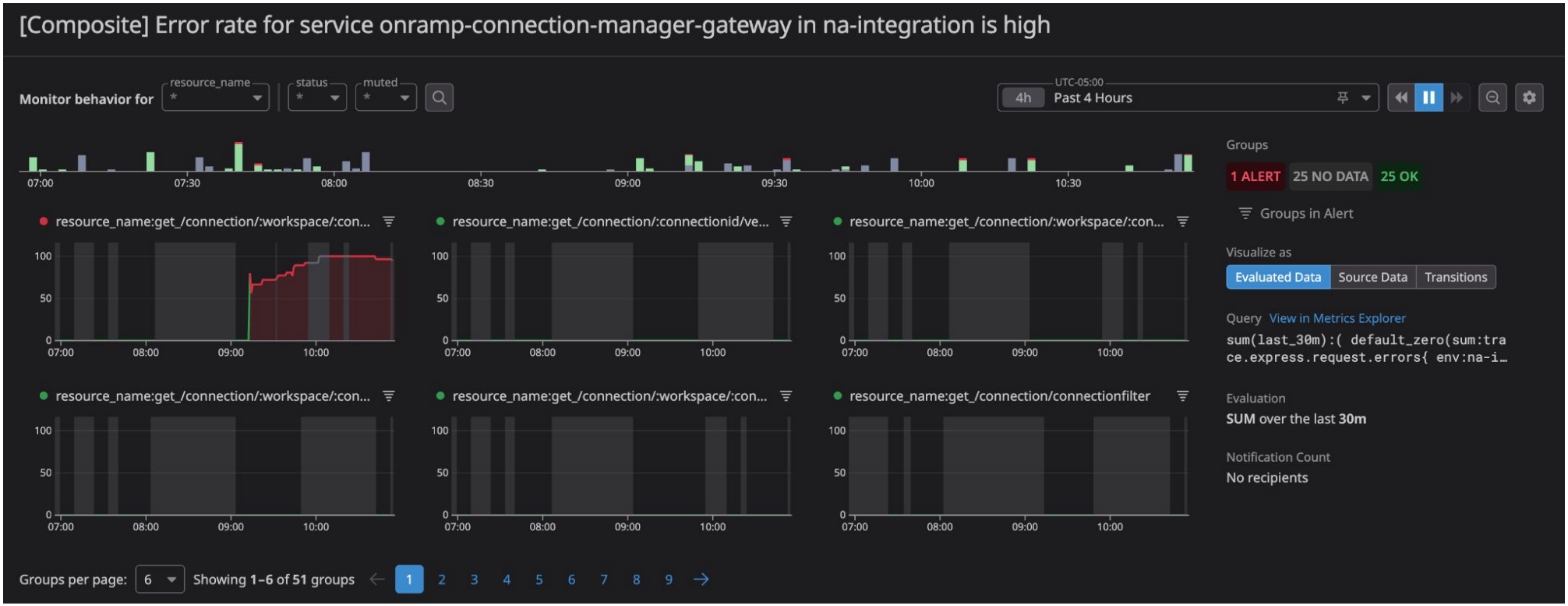


IAM-team	IAM-5069: add zero pod monitor module, rewrite NX k8s ...	2 weeks ago
actionable-insights-team	ACTINSTS-1: Fix burn rate monitors (#1262)	last week
air-team	[TRC-1208] Adding P90 Latency Monitors for air (#1140)	2 months ago
ami-team	SRE-2841 env label fixes (#1012)	4 months ago
api-enablement-team	CENGINE-3355-separate-out-atlas-search-update-monit...	2 weeks ago
china-datadog	SRE-9999 fixing link for latency golden signals (#1254)	2 weeks ago
collaborative-visibility-team	CV-2650 raise restart threshold (#1263)	last week
command-center	SRE-2859 adding webhook configuration for command c...	3 months ago
data-quality-operations	IHEALTH-1662: Update the team name. (#1174)	2 months ago
datadog-integrations	IHEALTH-1662: Configure the golden signals for p44-con...	2 months ago
datadog-internal-tooling	CENGINE-3331: Add integrations.p-44.com ssl monitor (...	last month
dataengineering-team	SRE-2887 (#1137)	3 months ago
devexp-team	DEVEXP-000 Update github-workflows-monitors.tf (#1132)	3 months ago

Error Rate Module

```
resource "datadog_monitor" "error_rate_monitor" {
  name               = "[Composite] Error rate for service ${var.service} in ${var.environment} is high"
  evaluation_delay   = "600"
  include_tags       = "true"
  type               = "query alert"
  query              = <<EOT
sum(last_${var.timePeriod}):(
  default_zero(sum:trace.${join(".", compact([var.operation, var.subOperation]))}.errors{
    env:${var.environment},
    service:${var.service},
    resource_name:${var.resourceName},
    ${join(",", [for r in var.resourceExclusion : format("!resource_name:%s", r)])},
    ${join(",", [for code in var.excludeErrorCodes : format("!http.status_code:%s", code)])}
  } by {resource_name}.as_count())
  /
  sum:trace.${join(".", compact([var.operation, var.subOperation]))}.hits{
    env:${var.environment},
    service:${var.service},
    resource_name:${var.resourceName},
    ${join(",", [for r in var.resourceExclusion : format("!resource_name:%s", r)])}
  } by {resource_name}.as_count()
) * 100 > ${var.errorCriticalThreshold}
EOT
  new_group_delay     = "60"
  notify_audit        = "false"
  notify_no_data      = "true"
  priority            = "2"
  require_full_window = "false"
  timeout_h           = "0"
  message             = ""
  Erik Bates, 13 months ago | 2 authors (Erik Bates and one other)
  monitor_thresholds {
    critical = var.errorCriticalThreshold
    warning  = var.errorWarningThreshold
  }      Manan Kothari, 2 years ago • SRE-70 migrating from terraform-datadog to new …

  tags = ["team:${var.team}", "env:${var.environment}", "service:${var.service}", "terraform:true", "composite_monitor"]

}
```

Error Rate Monitor



Latency Module

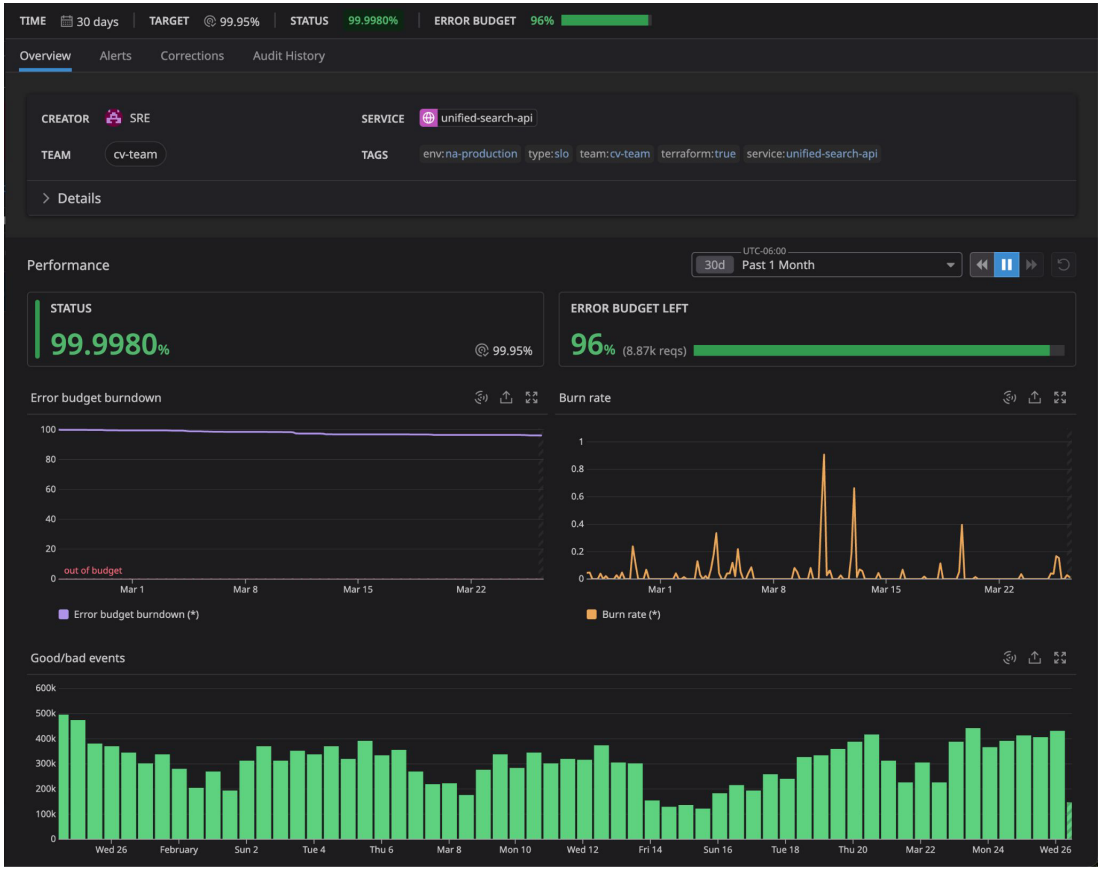
```
locals {
  endpoints = {
    bulk-loads-na = {
      environment              = "na-production"
      latencyWarningThreshold  = "6"
      latencyCriticalThreshold = "10"
      resourceName             = "post_/api/v1/tenants/bulk-loads"
    }

    bulk-loads-eu = {
      environment              = "eu-production"
      latencyWarningThreshold  = "3"
      latencyCriticalThreshold = "5"
      resourceName             = "post_/api/v1/tenants/bulk-loads"
    }

    another-endpoint-na = {
      environment              = "na-production"
      latencyWarningThreshold  = "15"
      latencyCriticalThreshold = "20"
      resourceName             = "post_/api/v1/tenants/abc"
    }
  }
}

module "tenant-service-resource-post_api_v1_tenants_bulk-loads-na" {
  for_each                 = local.endpoints
  source                   = "../modules/golden-signals-v2/latency-monitor"
  service                  = "tenant-service"
  environment              = each.value.environment
  pagerdutyService         = "@pagerduty-gnet"
  slackChannel             = "@slack-monitoring-gnet"
  timePeriod               = "15m"
  team                     = "network-growth-team"
  resourceName             = each.value.resourceName
  latencyWarningThreshold  = each.value.latencyWarningThreshold
  latencyCriticalThreshold = each.value.latencyCriticalThreshold
  operation                = "servlet"
  anyAdditionalInfo        = "Go to https://project44.atlassian.net/wiki/spaces/PD/pages/35755
  latencyPercentile        = "p99"
}
```

Burn Rate Module

```
resource "datadog_service_level_objective" "availability_slo" {      Manan Kothari, 2 years ago • SRE-70 migrating from terraform-datadog to new …
  name = "${var.environment} - ${var.service} - ${var.resourceName} Availability SLO"
  type        = "metric"
  description = "Monitors the overall availability of the ${var.service} service in the ${var.environment} environment, ensuring it meets the defined performance standards and reliability expecta
  Erik Bates, 2 months ago | 2 authors (Erik Bates and one other)
  query {
    numerator   = "sum:trace.${join(".", compact([var.operation, var.subOperation]))}.hits{env:${var.environment},service:${var.service},resource_name:${var.resourceName}}.as_count() - sum:trace.
    denominator = "sum:trace.${join(".", compact([var.operation, var.subOperation]))}.hits{env:${var.environment},service:${var.service},resource_name:${var.resourceName}}.as_count()"
  }

  Manan Kothari, 16 months ago | 1 author (Manan Kothari)
  thresholds {
    timeframe = var.sloTimeFrame
    target    = var.sloThreshold
    warning   = var.sloWarningThreshold
  }

  tags = ["team:${var.team}", "service:${var.service}", "terraform:true", "env:${var.environment}", "type:slo"]
}


Erik Bates, 7 days ago | 1 author (Erik Bates)
resource "datadog_monitor" "slo_burn_rate_alert" {
  name = "${var.environment} - ${var.service} - ${var.resourceName} SLO Burn Rate Alert"
  type = "slo alert"

  query = <<EOT
    burn_rate("${datadog_service_level_objective.availability_slo.id}").over("${var.sloTimeFrame}").long_window("${var.longWindow}").short_window("${var.shortWindow}") > ${var.burnRate}
    EOT

  depends_on = [datadog_service_level_objective.availability_slo]
  message = <<EOT
      🚨 SLO breach detected in ${var.service} 🚨
      [SLO Details](https://p44.datadoghq.com/slo?slo_id=${datadog_service_level_objective.availability_slo.id}) for this specific SLO.

      Recommendation: If you find that your burn rate alert is consistently too flaky, this is an indication that you should
      make your short window slightly larger. However, note that the larger you make your short window, the slower the monitor
      will be in recovering after an issue has ended. For more information see [Datadog's documentation](https://docs.datadoghq.com/service_management/service_level_objectives/burn_rate/).
      ${var.slackChannel}
      ${var.pagerdutyService}

      ${var.additionalMessage}
      EOT

  Erik Bates, 2 months ago | 1 author (Erik Bates)
  monitor_thresholds {
    critical = var.burnRate
  }
  tags = [
    "team:${var.team}", "service:${var.service}", "terraform:true", "env:${var.environment}",
    "monitor:slo-burn-rate-alert", "type:slo"
  ]
}
```

Burn Rate Monitor

```
module "unified-search-api-production-eu" {
  source               = "../modules/slos"
  slackChannel         = local.slack
  service              = "unified-search-api"
  environment          = "eu-production"
  operation            = "servlet"
  sloThreshold         = 99.95
  sloWarningThreshold = 99.97
  team                 = local.team
}
```

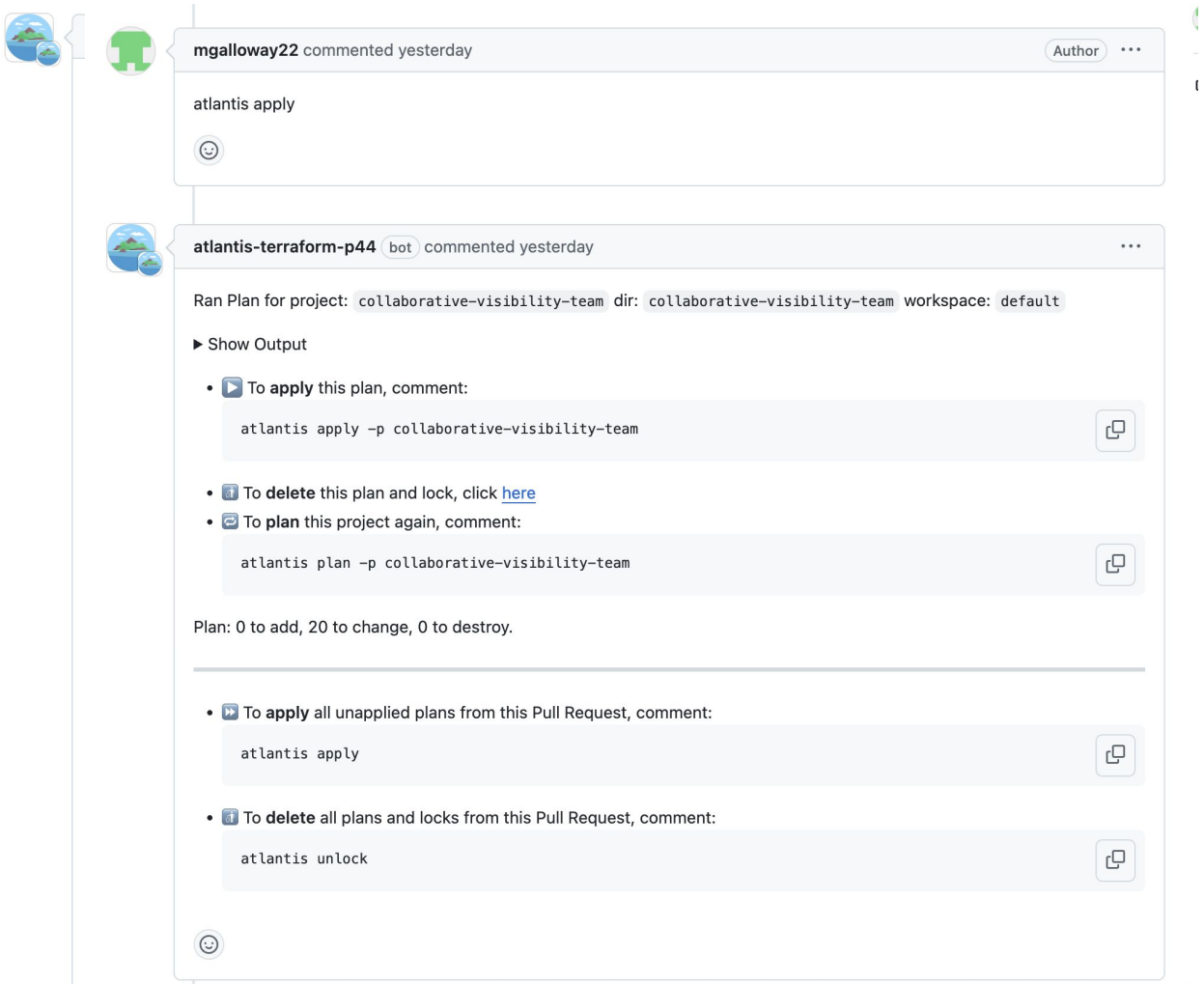


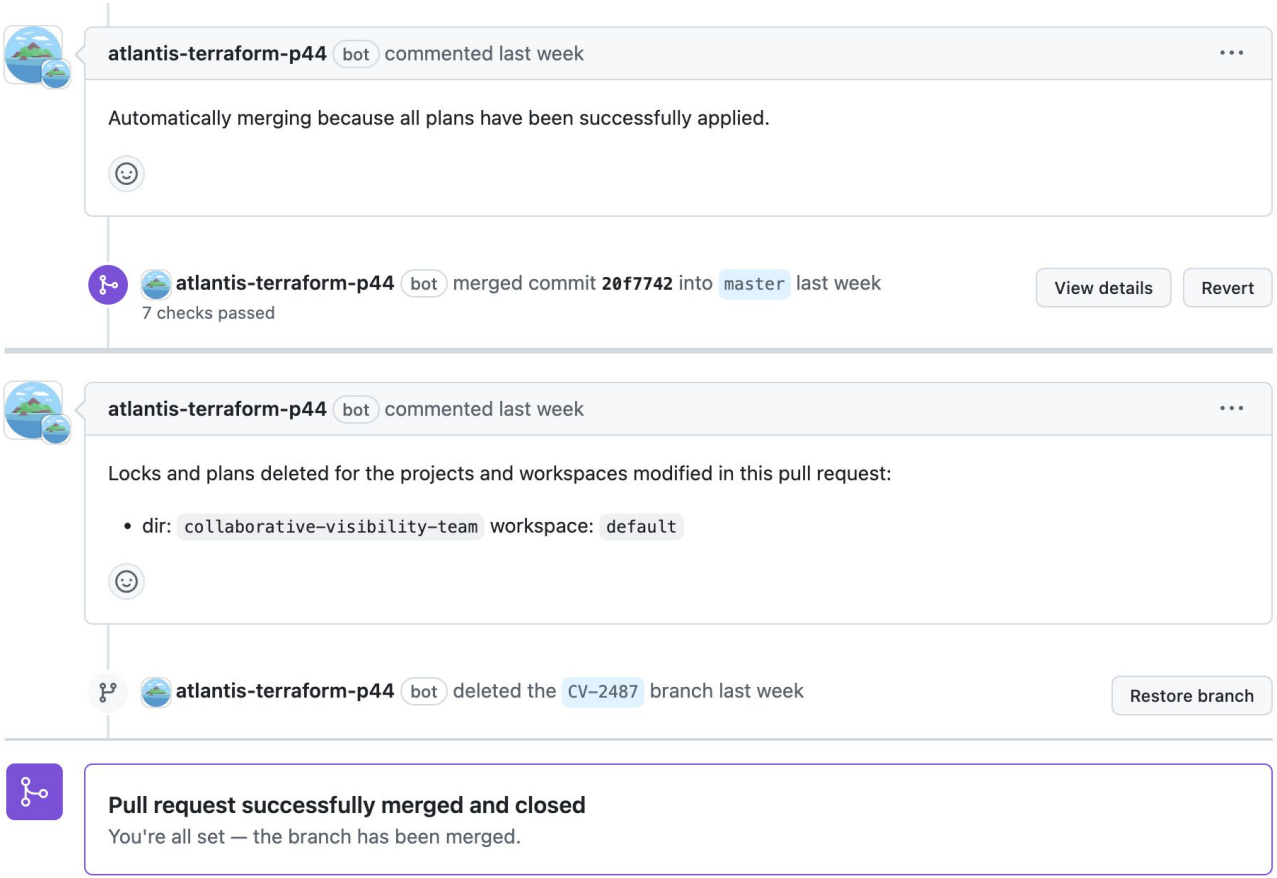
atlantis-terraform-p44 bot commented last week

Ran Plan for project: `collaborative-visibility-team` dir: `collaborative-visibility-team` workspace: `default`

▼ Show Output

```
Terraform used the selected providers to generate the following execution
plan. Resource actions are indicated with the following symbols:
! update in-place

Terraform will perform the following actions:

  # module.task-service-http-p95["add-comment-eu"].datadog_monitor.latency-monitor will be updated in-place
! resource "datadog_monitor" "latency-monitor" {
        id                  = "1247306"
        name                = "Latency for resource {{resource_name.name}} from task-service in eu-production
!       query               = "avg(last_1h):p95:trace.servlet.request{env:eu-production,resource_name:post_/i
        tags                = [
            "env:eu-production",
            "service:task-service",
            "team:cv-team",
            "terraform:true",
            "type:latency-golden-signal",
        ]
        # (16 unchanged attributes hidden)

!       monitor_thresholds {
!           critical = "0.3" -> "0.8"
!           warning  = "0.1" -> "0.4"
        }
    }

  # module.task-service-http-p95["add-comment-na"].datadog_monitor.latency-monitor will be updated in-place
! resource "datadog_monitor" "latency-monitor" {
        id                  = "1247255"
        name                = "Latency for resource {{resource_name.name}} from task-service in na-production
!       query               = "avg(last_1h):p95:trace.servlet.request{env:na-production,resource_name:post_/i
        tags                = [
            "env:na-production",
            "service:task-service",
            "team:cv-team",
            "terraform:true",
            "type:latency-golden-signal",
        ]
        # (16 unchanged attributes hidden)
```

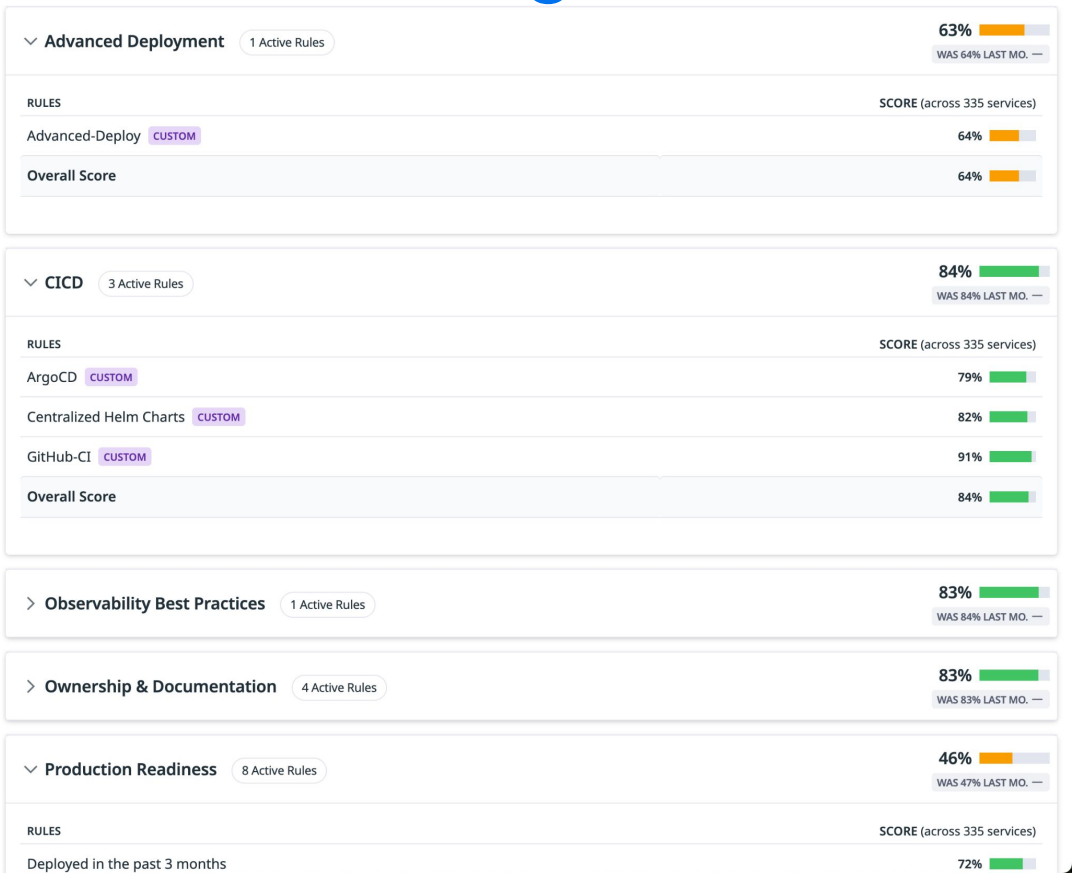

mgalloway22 commented yesterday
Author
atlantis apply
atlantis-terraform-p44 bot commented yesterday
Ran Plan for project: collaborative-visibility-team dir: collaborative-visibility-team workspace: default
▶ Show Output
To apply this plan, comment:
atlantis apply -p collaborative-visibility-team
To delete this plan and lock, click here
To plan this project again, comment:
atlantis plan -p collaborative-visibility-team
Plan: 0 to add, 20 to change, 0 to destroy.
To apply all unapplied plans from this Pull Request, comment:
atlantis apply
To delete all plans and locks from this Pull Request, comment:
atlantis unlock


atlantis-terraform-p44 bot commented last week
Automatically merging because all plans have been successfully applied.
atlantis-terraform-p44 bot merged commit 20f7742 into master last week
7 checks passed
View details
Revert
atlantis-terraform-p44 bot commented last week
Locks and plans deleted for the projects and workspaces modified in this pull request:
dir: collaborative-visibility-team workspace: default
atlantis-terraform-p44 bot deleted the CV-2487 branch last week
Restore branch
Pull request successfully merged and closed
You're all set — the branch has been merged.

4

Conclusion

Measuring Success



Time Savings

- We've reduced the time it takes to spin up a new microservice by 5-7 days (including infrastructure like a memorystore or cloudsql instance)

Time Savings

- We've reduced the time it takes to spin up a new microservice by 5-7 days (including infrastructure like a memorystore or cloudsql instance)

- Our Engineers have the right monitors and observability built within minutes for a new service

Time Savings

- We've reduced the time it takes to spin up a new microservice by 5-7 days (including infrastructure like a memorystore or cloudsql instance)

- Our Engineers have the right monitors and observability built within minutes for a new service

- Reduce overhead for our engineers on what is actually important to monitor & page on

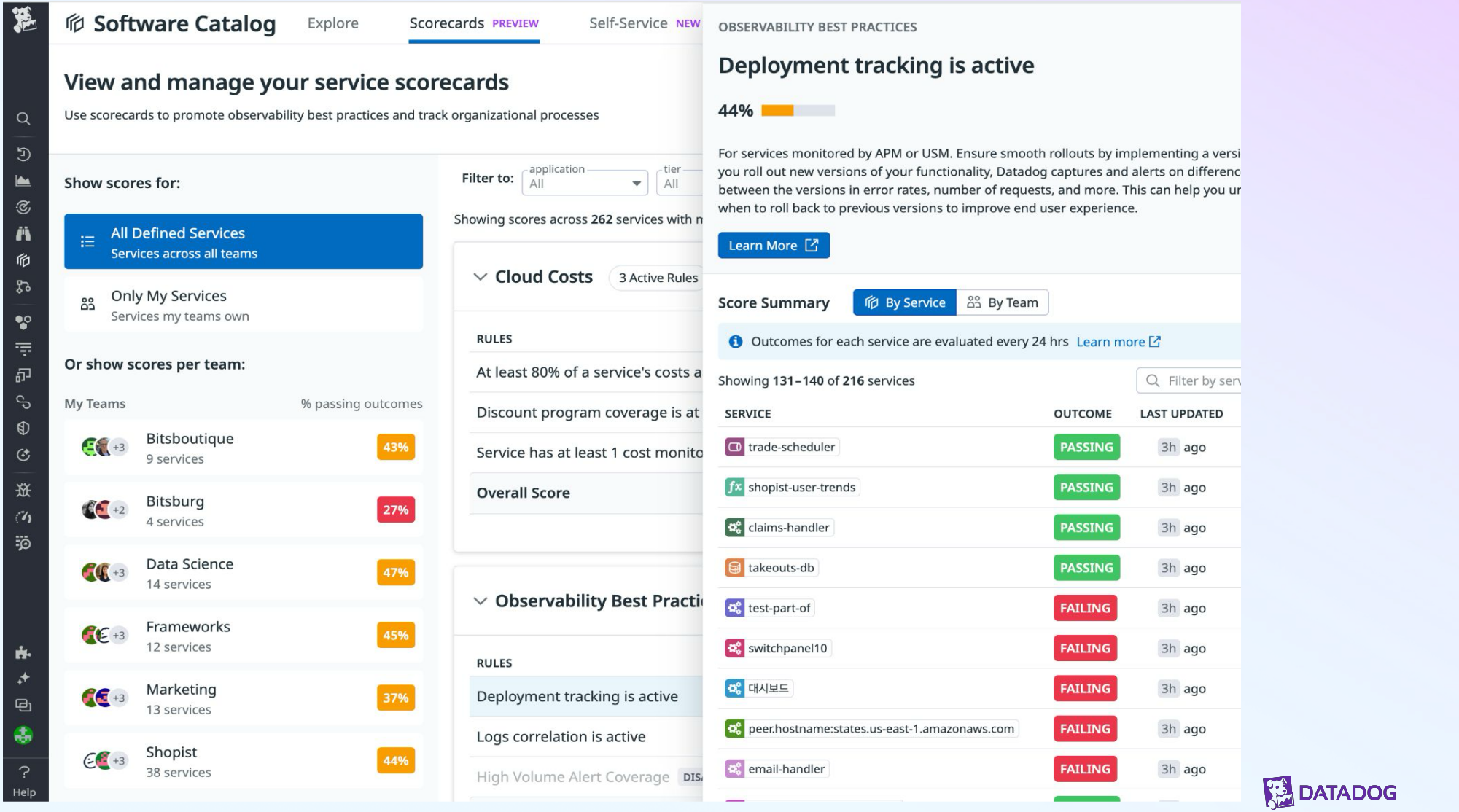
Time Savings

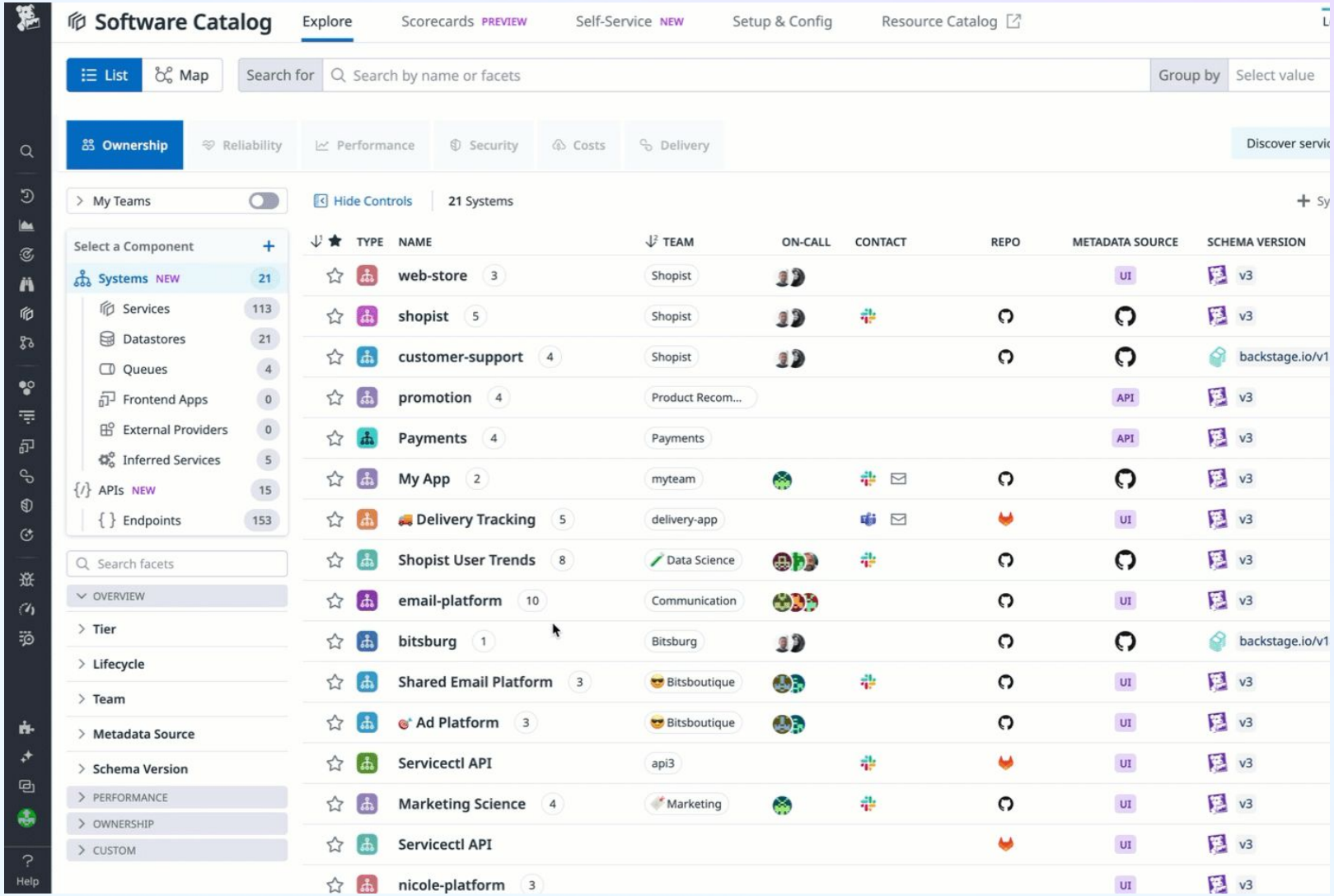
- We've reduced the time it takes to spin up a new microservice by 5-7 days (including infrastructure like a memorystore or cloudsql instance)

- Our Engineers have the right monitors and observability built within minutes for a new service

- Reduce overhead for our engineers on what is actually important to monitor & page on

- We've reduced the number of incidents caught by our customers first

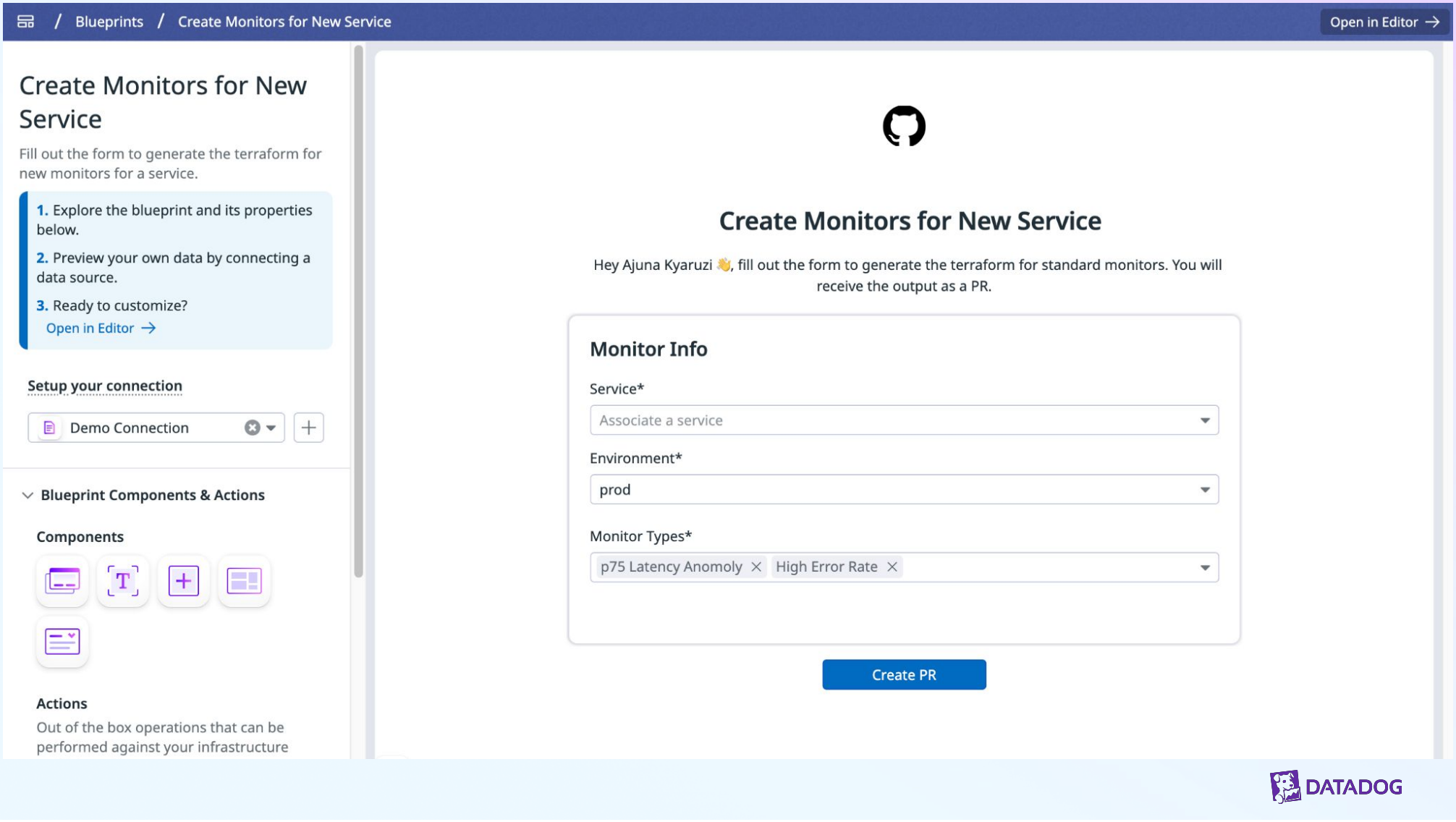

Software Catalog
Explore
Scorecards PREVIEW
Self-Service NEW
View and manage your service scorecards
Use scorecards to promote observability best practices and track organizational processes
Show scores for:
All Defined Services
Services across all teams
Only My Services
Services my teams own
Or show scores per team:
My Teams
% passing outcomes
Bitsboutique
9 services
43%
Bitsburg
4 services
27%
Data Science
14 services
47%
Frameworks
12 services
45%
Marketing
13 services
37%
Shopist
38 services
44%
Help
Filter to:
application
All
tier
All
Showing scores across 262 services with n
Cloud Costs
3 Active Rules
RULES
At least 80% of a service's costs a
Discount program coverage is at
Service has at least 1 cost monito
Overall Score
Observability Best Practi
RULES
Deployment tracking is active
Logs correlation is active
High Volume Alert Coverage
OBSERVABILITY BEST PRACTICES
Deployment tracking is active
44%
For services monitored by APM or USM. Ensure smooth rollouts by implementing a versi
you roll out new versions of your functionality, Datadog captures and alerts on differenc
between the versions in error rates, number of requests, and more. This can help you ur
when to roll back to previous versions to improve end user experience.
Learn More
Score Summary
By Service
By Team
Outcomes for each service are evaluated every 24 hrs Learn more
Showing 131–140 of 216 services
Filter by serv
SERVICE
OUTCOME
LAST UPDATED
trade-scheduler
PASSING
3h ago
shopist-user-trends
PASSING
3h ago
claims-handler
PASSING
3h ago
takeouts-db
PASSING
3h ago
test-part-of
FAILING
3h ago
switchpanel10
FAILING
3h ago
대시보드
FAILING
3h ago
peer.hostname:states.us-east-1.amazonaws.com
FAILING
3h ago
email-handler
FAILING
3h ago


Software Catalog
Explore
Scorecards PREVIEW
Self-Service NEW
Setup & Config
Resource Catalog
List
Map
Search for
Search by name or facets
Group by
Select value
Ownership
Reliability
Performance
Security
Costs
Delivery
Discover servic
My Teams
Hide Controls
21 Systems
Select a Component
Systems NEW 21
Services 113
Datastores 21
Queues 4
Frontend Apps 0
External Providers 0
Inferred Services 5
APIs NEW 15
Endpoints 153
Search facets
OVERVIEW
Tier
Lifecycle
Team
Metadata Source
Schema Version
PERFORMANCE
OWNERSHIP
CUSTOM
Help
TYPE
NAME
TEAM
ON-CALL
CONTACT
REPO
METADATA SOURCE
SCHEMA VERSION
web-store 3 Shopist UI v3
shopist 5 Shopist v3
customer-support 4 Shopist backstage.io/v1
promotion 4 Product Recom... API v3
Payments 4 Payments API v3
My App 2 myteam v3
Delivery Tracking 5 delivery-app UI v3
Shopist User Trends 8 Data Science v3
email-platform 10 Communication UI v3
bitsburg 1 Bitsburg backstage.io/v1
Shared Email Platform 3 Bitsboutique UI v3
Ad Platform 3 Bitsboutique UI v3
Servicectl API api3 UI v3
Marketing Science 4 Marketing UI v3
Servicectl API UI v3
nicole-platform 3 UI v3

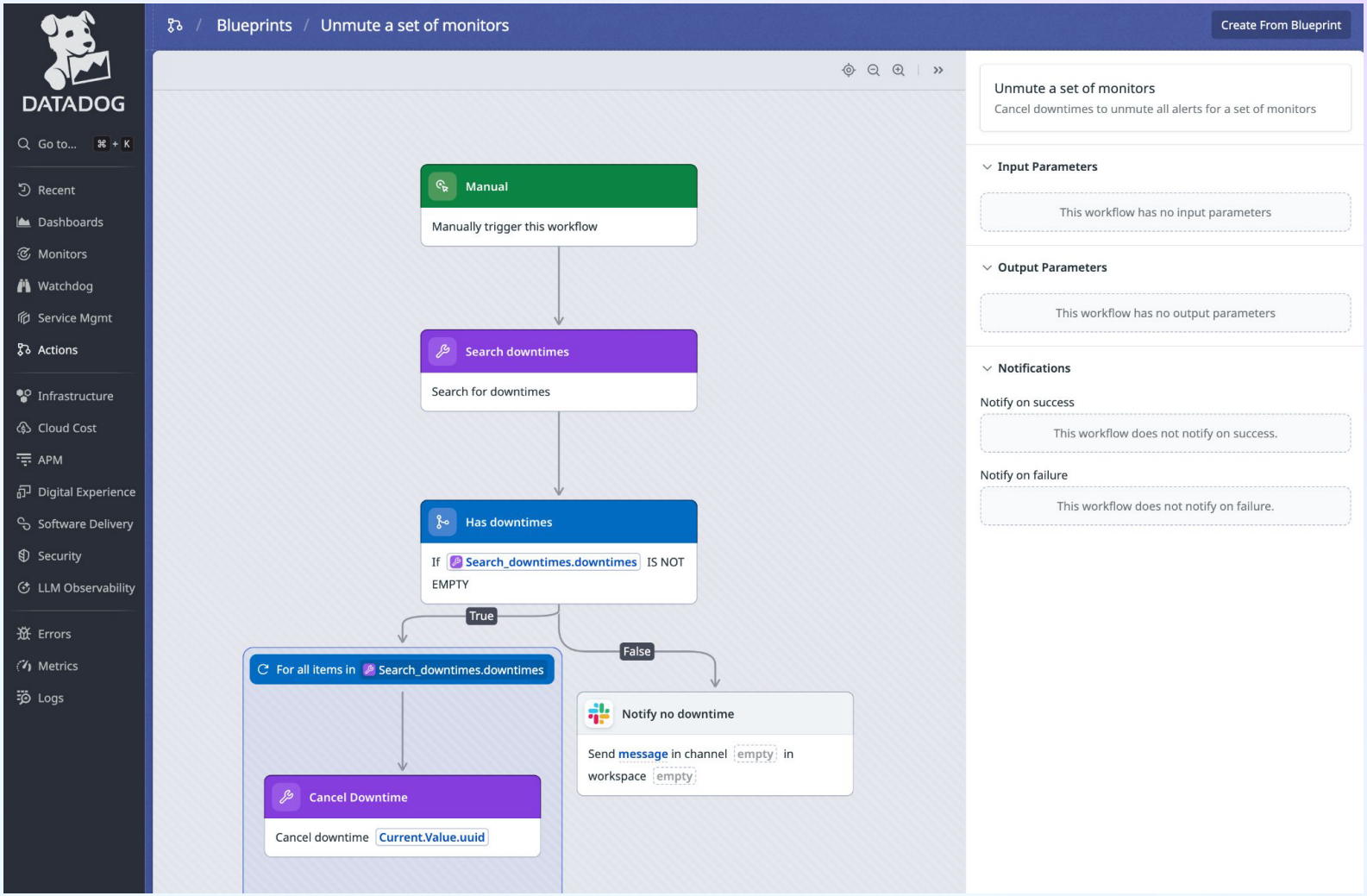
DATADOG

Workflow Automation

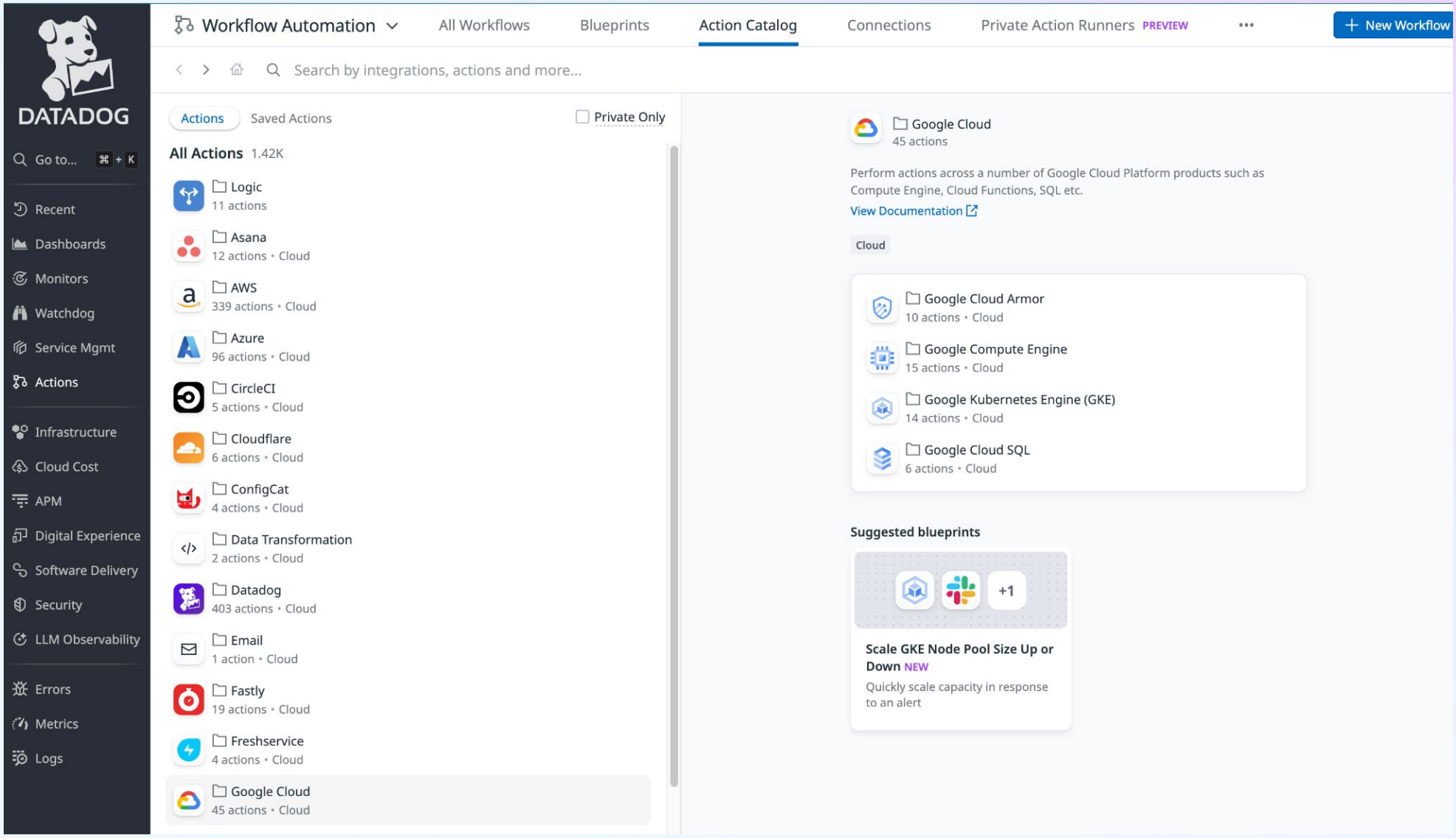
DATADOG

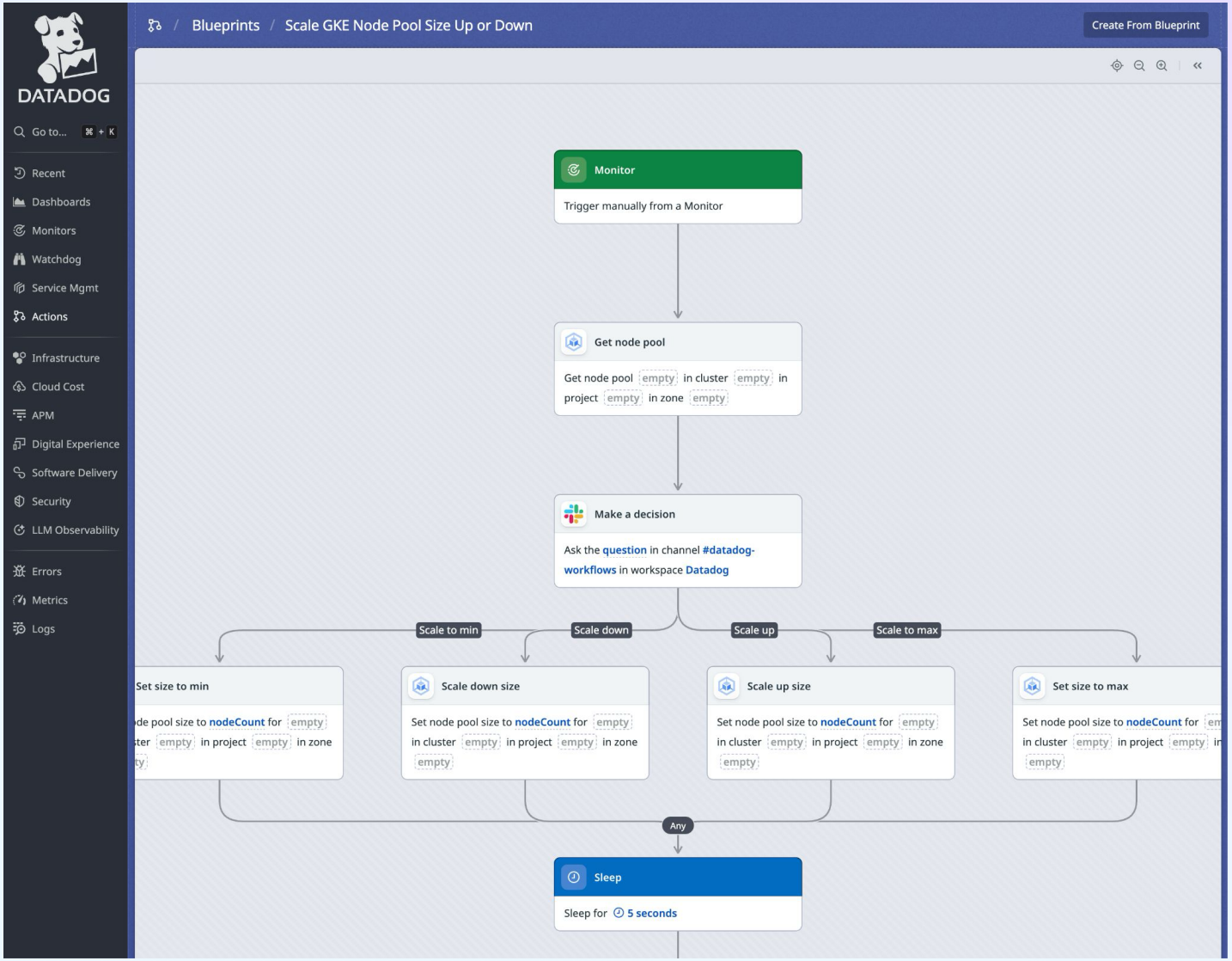

/ Blueprints / Create Monitors for New Service
Open in Editor
Create Monitors for New Service
Fill out the form to generate the terraform for new monitors for a service.
1. Explore the blueprint and its properties below.
2. Preview your own data by connecting a data source.
3. Ready to customize?
Open in Editor
Setup your connection
Demo Connection
Blueprint Components & Actions
Components
Actions
Out of the box operations that can be performed against your infrastructure
Create Monitors for New Service
Hey Ajuna Kyaruzi 👋, fill out the form to generate the terraform for standard monitors. You will receive the output as a PR.
Monitor Info
Service*
Associate a service
Environment*
prod
Monitor Types*
p75 Latency Anomoly
High Error Rate
Create PR

DATADOG

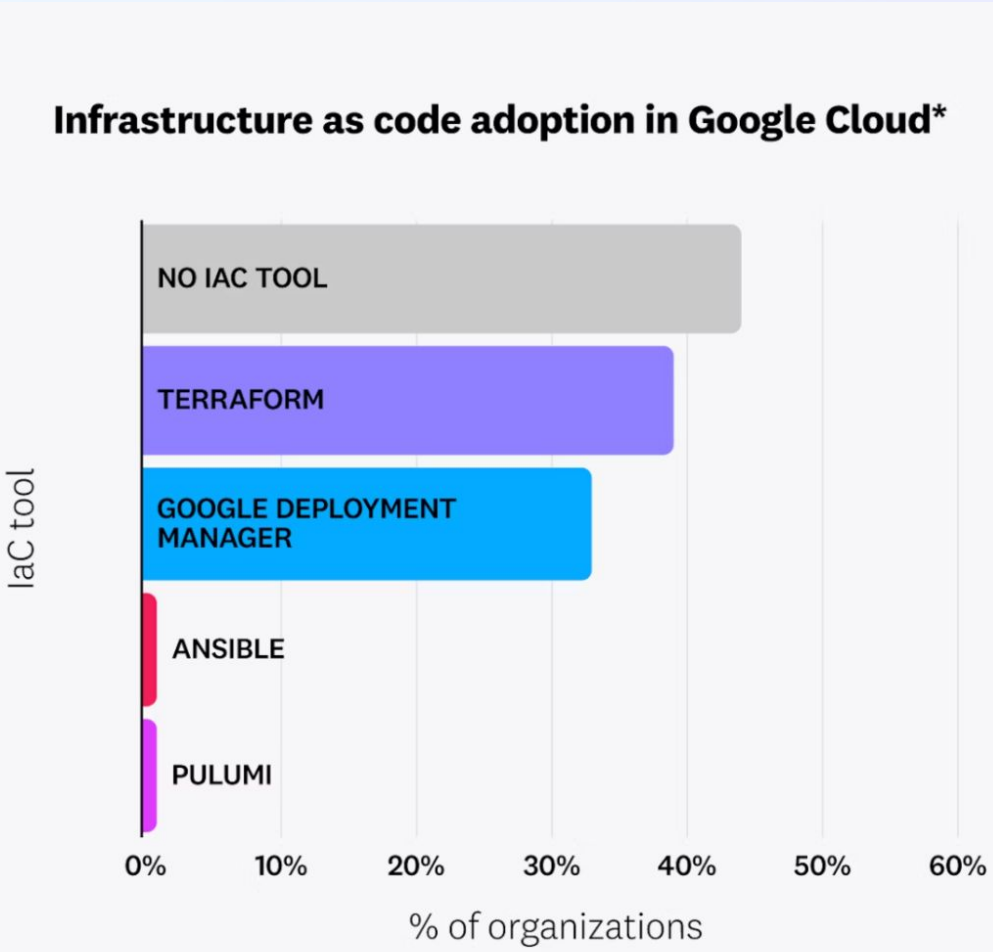

DATADOG
Go to... ⌘ + K
Recent
Dashboards
Monitors
Watchdog
Service Mgmt
Actions
Infrastructure
Cloud Cost
APM
Digital Experience
Software Delivery
Security
LLM Observability
Errors
Metrics
Logs
Blueprints / Unmute a set of monitors
Create From Blueprint
Manual
Manually trigger this workflow
Search downtimes
Search for downtimes
Has downtimes
If Search_downtimes.downtimes IS NOT EMPTY
True
False
For all items in Search_downtimes.downtimes
Cancel Downtime
Cancel downtime Current.Value.uuid
Notify no downtime
Send message in channel empty in workspace empty
Unmute a set of monitors
Cancel downtimes to unmute all alerts for a set of monitors
Input Parameters
This workflow has no input parameters
Output Parameters
This workflow has no output parameters
Notifications
Notify on success
This workflow does not notify on success.
Notify on failure
This workflow does not notify on failure.

DATADOG


DATADOG
Go to... ⌘ + K
Recent
Dashboards
Monitors
Watchdog
Service Mgmt
Actions
Infrastructure
Cloud Cost
APM
Digital Experience
Software Delivery
Security
LLM Observability
Errors
Metrics
Logs
Workflow Automation
All Workflows
Blueprints
Action Catalog
Connections
Private Action Runners PREVIEW
New Workflow
Search by integrations, actions and more...
Actions
Saved Actions
Private Only
All Actions 1.42K
Logic
11 actions
Asana
12 actions • Cloud
AWS
339 actions • Cloud
Azure
96 actions • Cloud
CircleCI
5 actions • Cloud
Cloudflare
6 actions • Cloud
ConfigCat
4 actions • Cloud
Data Transformation
2 actions • Cloud
Datadog
403 actions • Cloud
Email
1 action • Cloud
Fastly
19 actions • Cloud
Freshservice
4 actions • Cloud
Google Cloud
45 actions • Cloud
Google Cloud
45 actions
Perform actions across a number of Google Cloud Platform products such as Compute Engine, Cloud Functions, SQL etc.
View Documentation
Cloud
Google Cloud Armor
10 actions • Cloud
Google Compute Engine
15 actions • Cloud
Google Kubernetes Engine (GKE)
14 actions • Cloud
Google Cloud SQL
6 actions • Cloud
Suggested blueprints
+1
Scale GKE Node Pool Size Up or Down NEW
Quickly scale capacity in response to an alert


DATADOG
Go to...
Recent
Dashboards
Monitors
Watchdog
Service Mgmt
Actions
Infrastructure
Cloud Cost
APM
Digital Experience
Software Delivery
Security
LLM Observability
Errors
Metrics
Logs
Blueprints / Scale GKE Node Pool Size Up or Down
Create From Blueprint
Monitor
Trigger manually from a Monitor
Get node pool
Get node pool empty in cluster empty in project empty in zone empty
Make a decision
Ask the question in channel #datadog-workflows in workspace Datadog
Scale to min
Scale down
Scale up
Scale to max
Set size to min
Scale down size
Set node pool size to nodeCount for empty in cluster empty in project empty in zone empty
Scale up size
Set node pool size to nodeCount for empty in cluster empty in project empty in zone empty
Set size to max
Set node pool size to nodeCount for empty in cluster empty in project empty
Any
Sleep
Sleep for 5 seconds

DATADOG



IN GOOGLE CLOUD, NEARLY HALF OF ORGANIZATIONS DON'T USE ANY INFRASTRUCTURE AS CODE TOOL

*IaC tools are not mutually exclusive

Source: Datadog

DATADOG

Continue your Datadog learning journey!

Booth #2140

BRK1-122 Accelerate your Google Cloud migration journey

Wed 4/9 02:45 - 03:30pm

BRK2-224 Exploring and securing Google Cloud default service accounts: Deep dive and real-world adoption trends

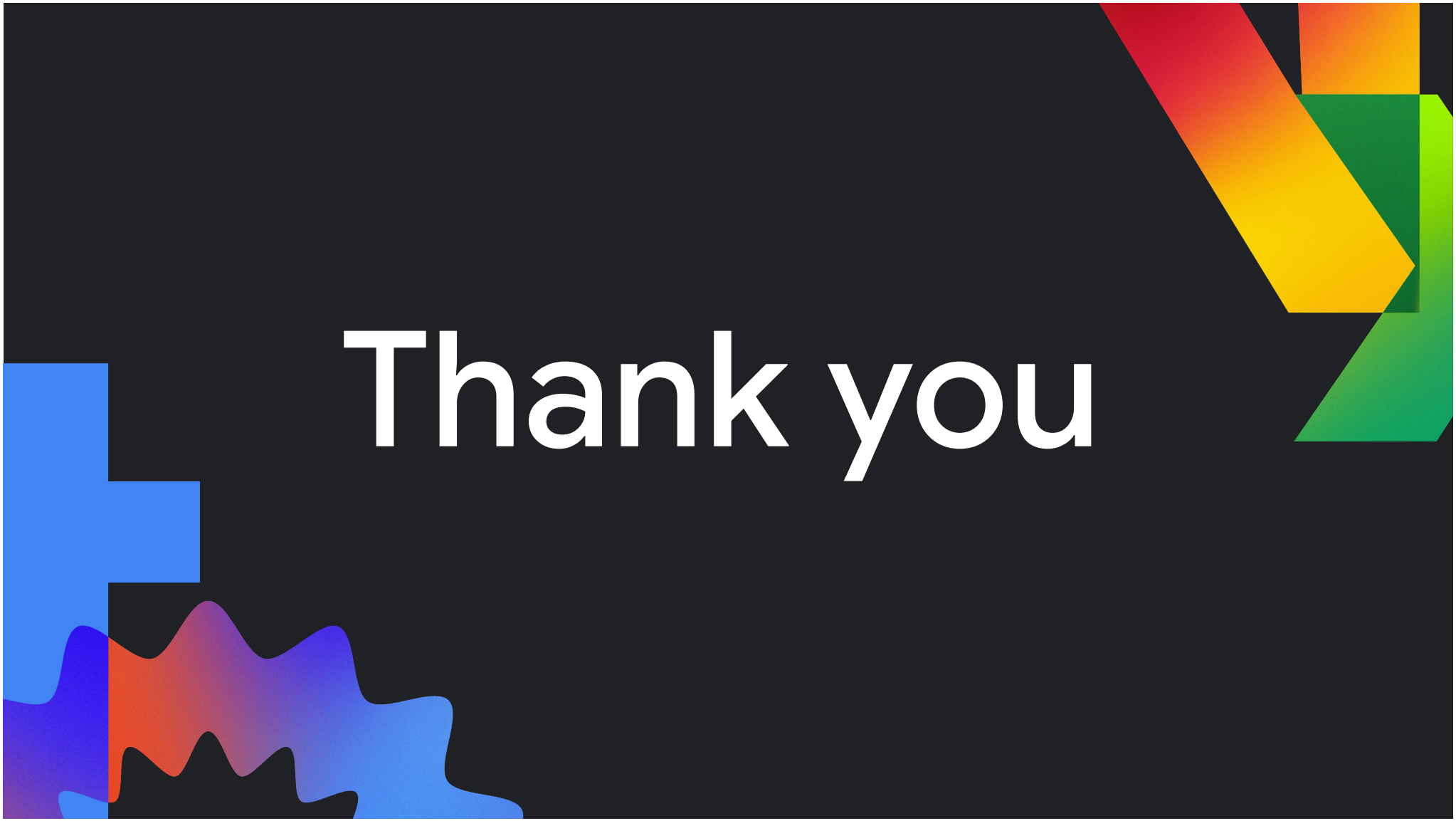
Thu 4/10 10:30 - 11:15 am

BRK2-163 PANEL - AI code assist tools: Boost productivity while enhancing app performance

Fri 4/11 09:45 - 10:30 am

CT1-12 So you're in the cloud. Now what?

Fri 4/11 12:00 - 12:20 PM


Thank you