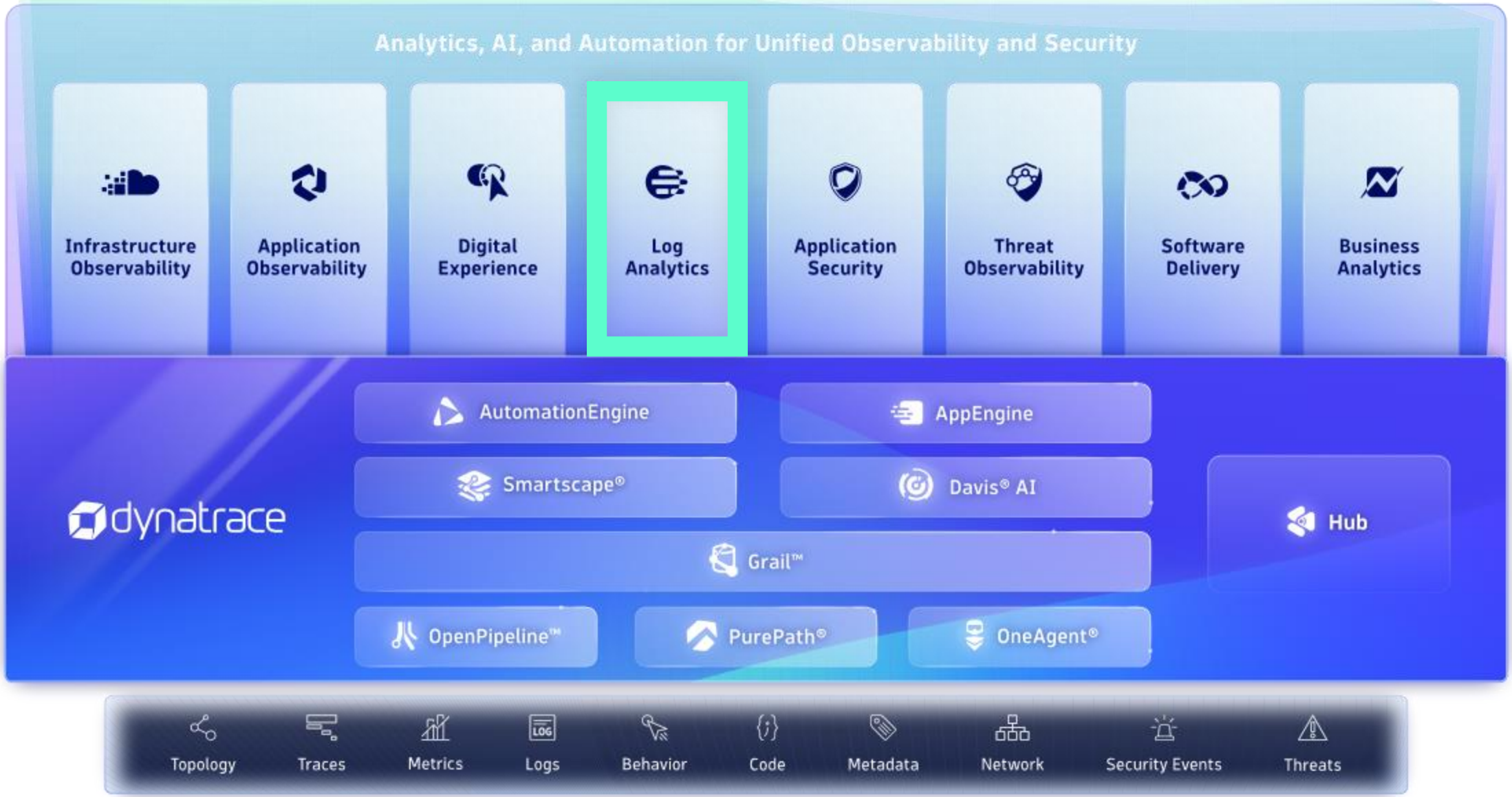
Log Management & Analytics at scale!

Agenda

- Log Management and Analytics in Dynatrace provides instant answers to critical questions across:
  - Business
  - DevOps
  - SRE
  - Operations
  - Security
- This technical session will cover best practices and strategies for Log Management and Analytics in Dynatrace, including scalable implementation.
- Attendees will gain knowledge on managing enterprise logging in Dynatrace and leave with actionable insights for their own implementations.

Agenda Items:

- Log Management and Analytics
- OpenPipeline
- Understanding Query behavior
- Buckets and Segments
- Permissions
- Logs to Metrics / Events
- Dashboarding
- Anomaly Detection
- Live Demo


Analytics, AI, and Automation for Unified Observability and Security
Infrastructure Observability
Application Observability
Digital Experience
Log Analytics
Application Security
Threat Observability
Software Delivery
Business Analytics
dynatrace
AutomationEngine
AppEngine
Smartscape®
Davis® AI
Grail™
Hub
OpenPipeline™
PurePath®
OneAgent®
Topology
Traces
Metrics
Logs
Behavior
Code
Metadata
Network
Security Events
Threats

Discussion Question

What are necessary capabilities for a Log Management and Analytics Solution?

Discussion Question

Ingest logs from many sources

Retain logs for selected timeframe

Permissions and Access Controls

Scalability

at are necessary capabilities for a Log Managem and Analytics Solution?

Filtering

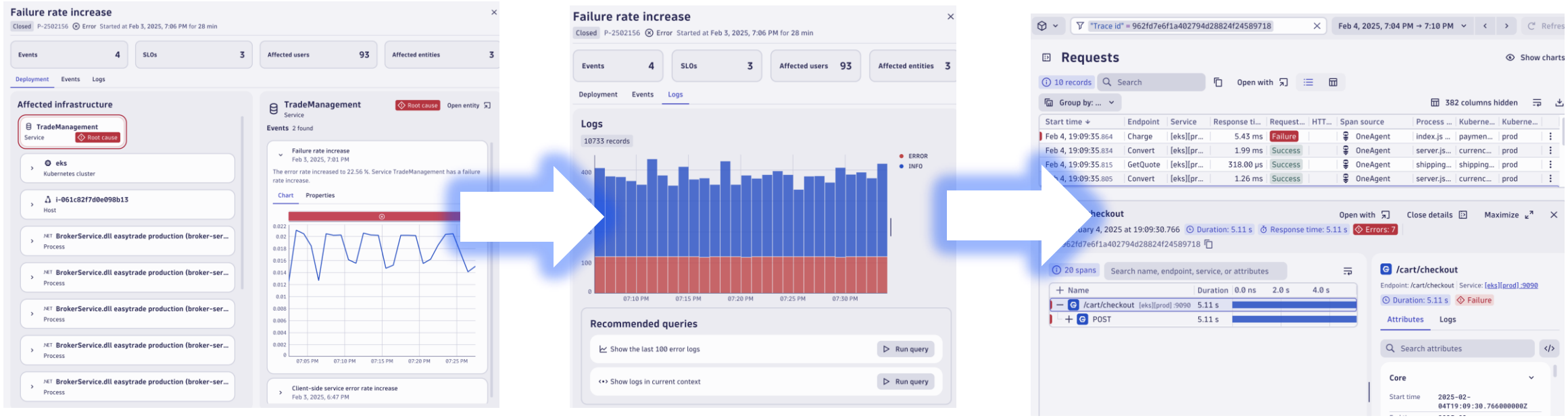
Alerts from log data

Automatic parsing for standard logs

Deep dive analytics

Data security

Logs in context reduces MTTR and provides instant answers



> We used to spend hours manually searching through metrics, logs, and traces to piece together insights about user experience. Now, this takes minutes or seconds.

Discussion Question

How does Dynatrace enrich logs with trace context?

Discussion Question

How does Dynatrace enrich logs with trace context?

Automatically for popular logging frameworks!
Just toggle it on in OneAgent Features

Dynatrace also provides guides for enrichment for other logging methods or for use with OpenTelemetry

The all-in-one unified platform advantage

Log sources

Kubernetes
Technology
Applications

Host
Process
System

aws

Custom API integrations

Log forwarders

OpenTelemetry
Cribl

- Logstash
- FluentD
- Fluent Bit

Log ingestion components

One Agents
Auto-detection

Log ingestion API
JSON, text, OTLP

OpenPipeline

Enrich logs with context

Parse JSON for simplified analytics

Mask sensitive data

Set security context

Extract metrics and business events data

Route to Grail buckets

Causational Data Lakehouse

Grail™

Low cost ingest and retention

No storage tiers, schemas, or indexes to manage

End-to-end context

Davis® AI

Automated
- Entity discovery
- Contextualization
- Baselining
- Anomaly/issue detection

Log insights
- Integrated throughout the platform apps for easy troubleshooting

Simple analysis
- Easy filtering and log exploration
- Narrow by attributes and timeframe

Exploratory analytics
- High speed query processing powered by Data-warping
- Schema-on-read 10X faster parsing than RegEx
- Purpose-built for exabyte scale.

Discussion Question

What is OpenPipeline?

Discussion Question

What is OpenPipeline?

OpenPipeline is the data handling solution (pipeline) for Dynatrace to seamlessly ingest and process data.

Used for configuring processing, transforming, metric or data extraction, permissions and storage.

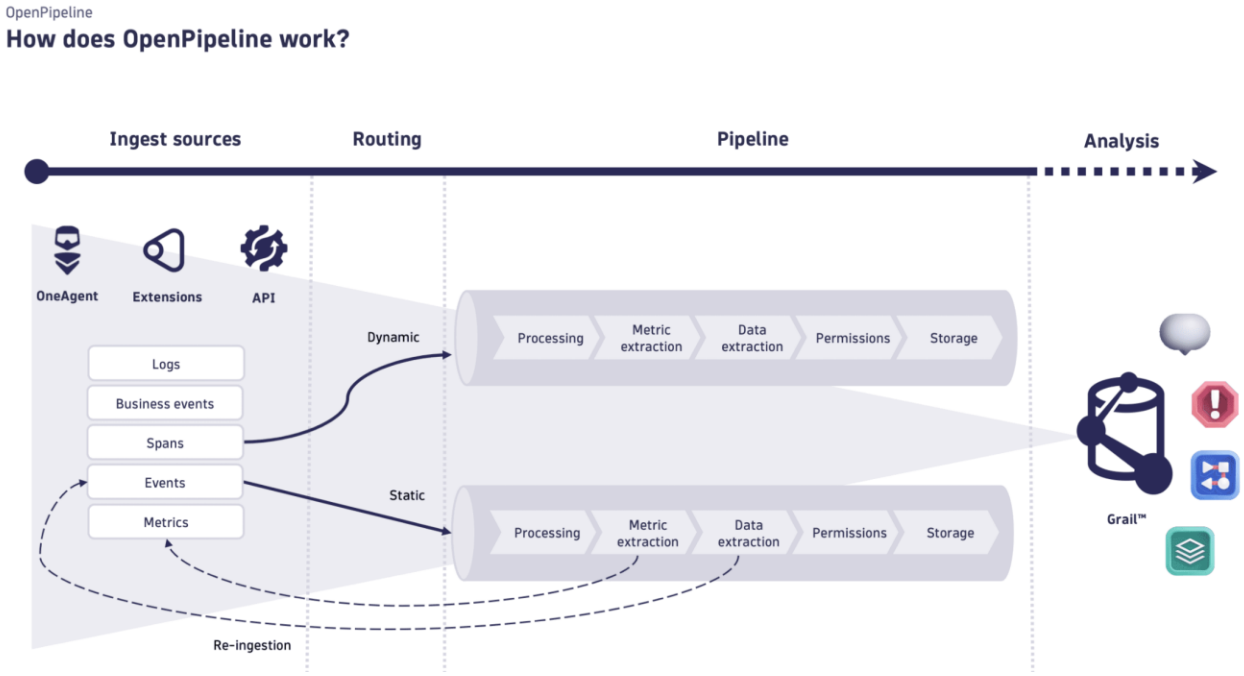
Unified solution for ingestion and processing of different data types, not just logs.

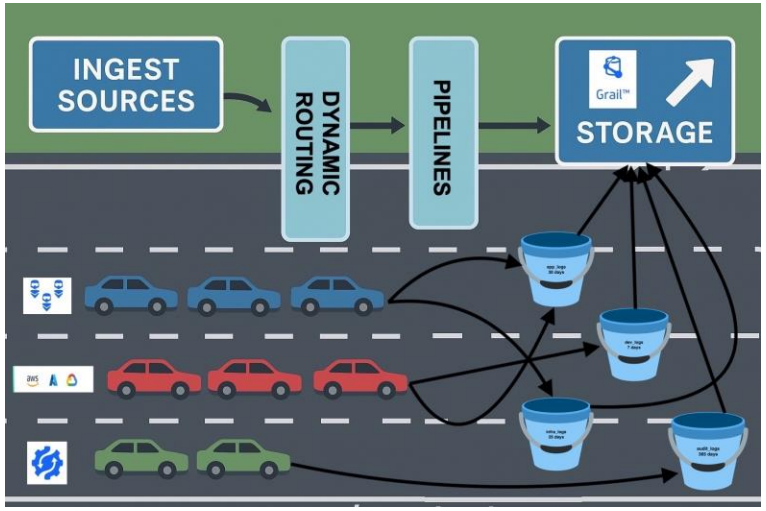
OpenPipeline

- OpenPipeline is used to normalize log records with processing rules, create metrics, create events, assign security context, and define storage to buckets

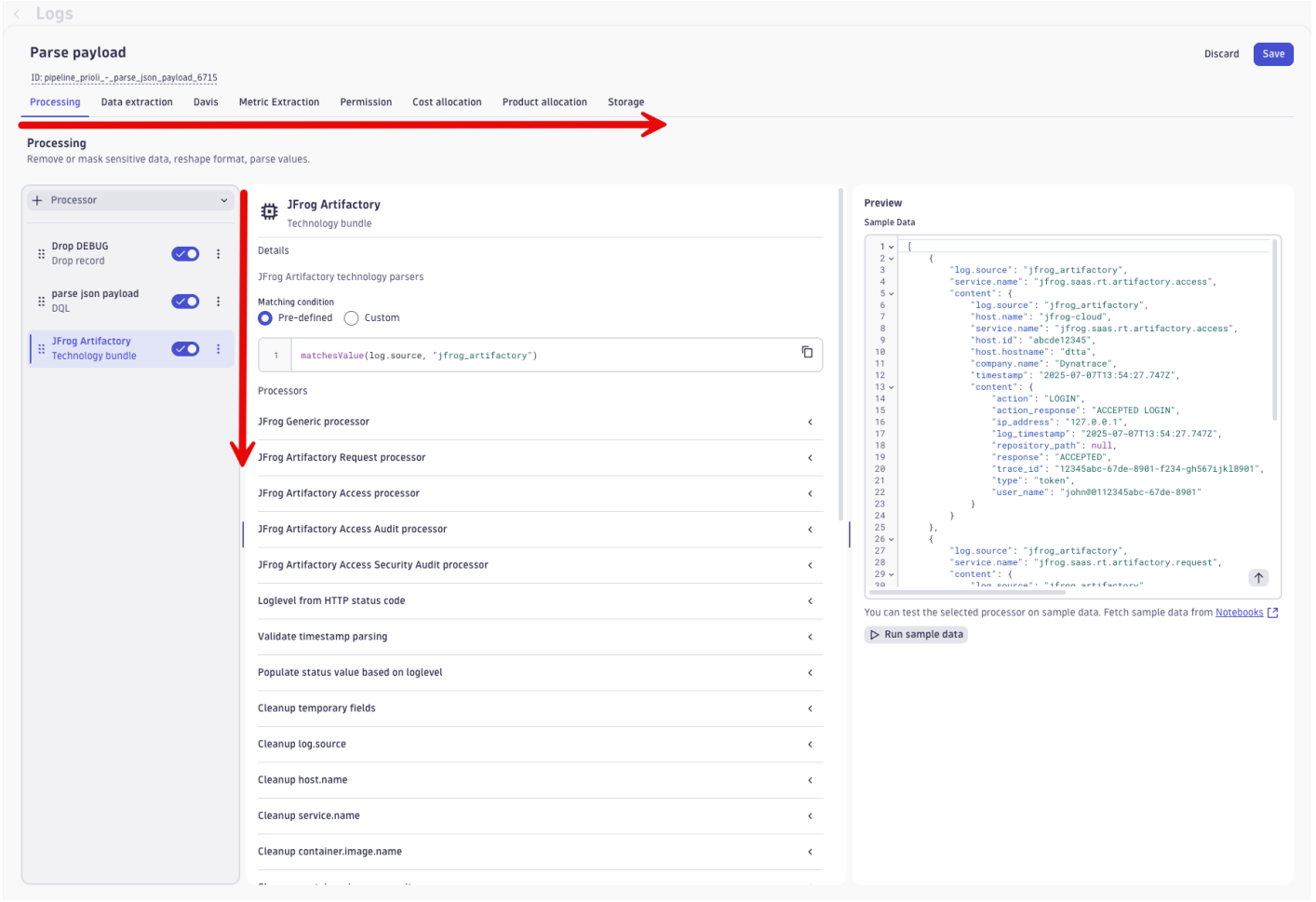
- Mapping incoming log records to specific buckets is done via OpenPipeline.

- OpenPipeline should be strategized in a similar manner to buckets.
  - 1:1 pipeline to bucket
  - 1:many pipeline to buckets





OpenPipeline

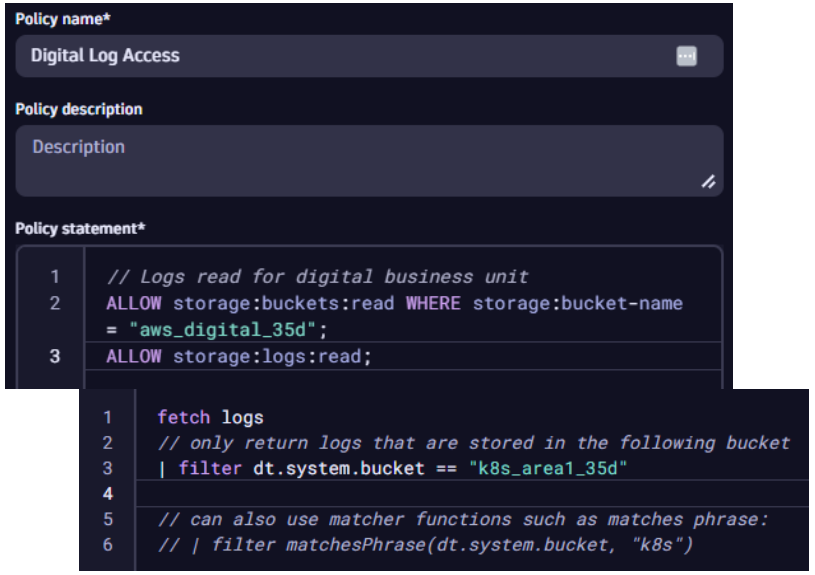


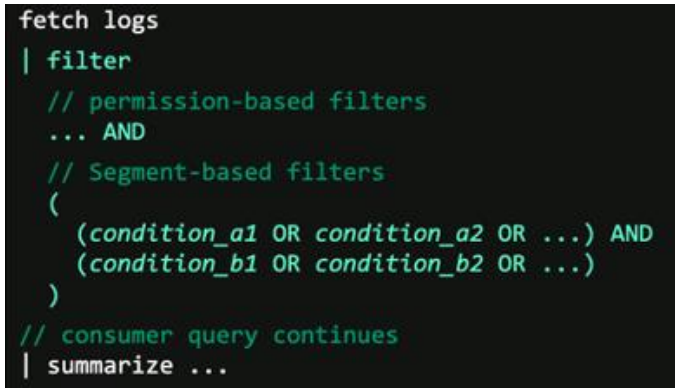
- OpenPipeline works in sequential mode, first vertical and then horizontal for each pipeline.
- Processing allows you to parse, add/remove/rename fields and drop record.
- Metric extraction allows you to create counter or value metric.

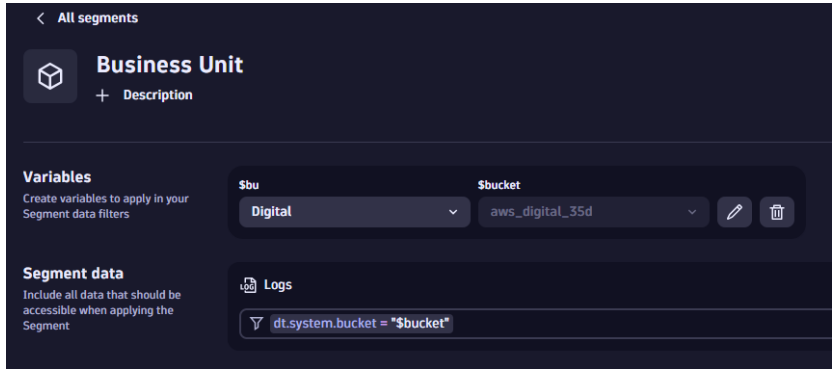
Scaling Log Analytics!

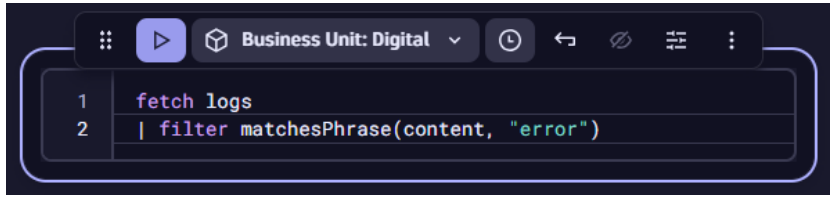
Understanding Query Behavior

- DQL and in app queries **will scan all buckets** a user has permission to read by default.
- Dynatrace **automatically optimizes queries** based on filters
- To realize performance and cost optimizations from a bucket strategy, queries must be filtered to relevant buckets.
  - This can be accomplished in three ways:
    - Only give users **access to relevant buckets or log records***
    - Apply **dt.system.bucket filter** into DQL query
    - Utilize **segments** for targeting appropriate buckets*







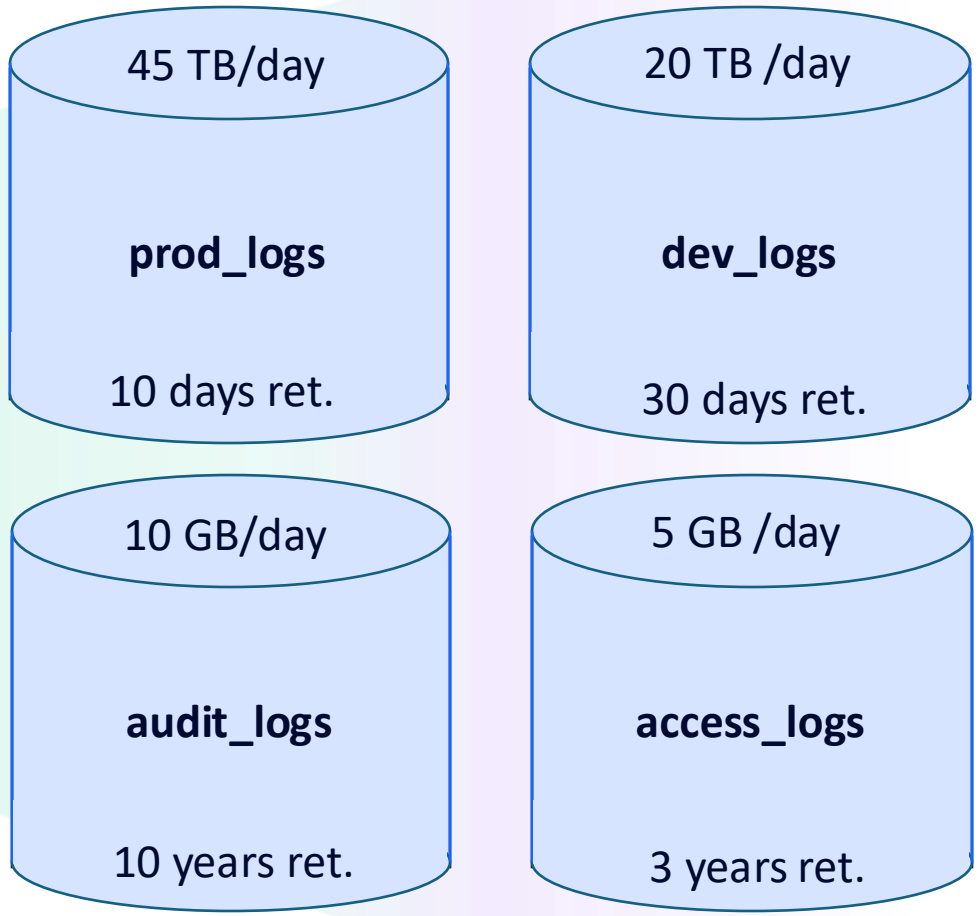


Bucket Query Strategy - Unoptimized

DQL Query:
fetch logs

IAM Policy:
ALLOW storage:logs:read

Default Scan Limit:
500 GB

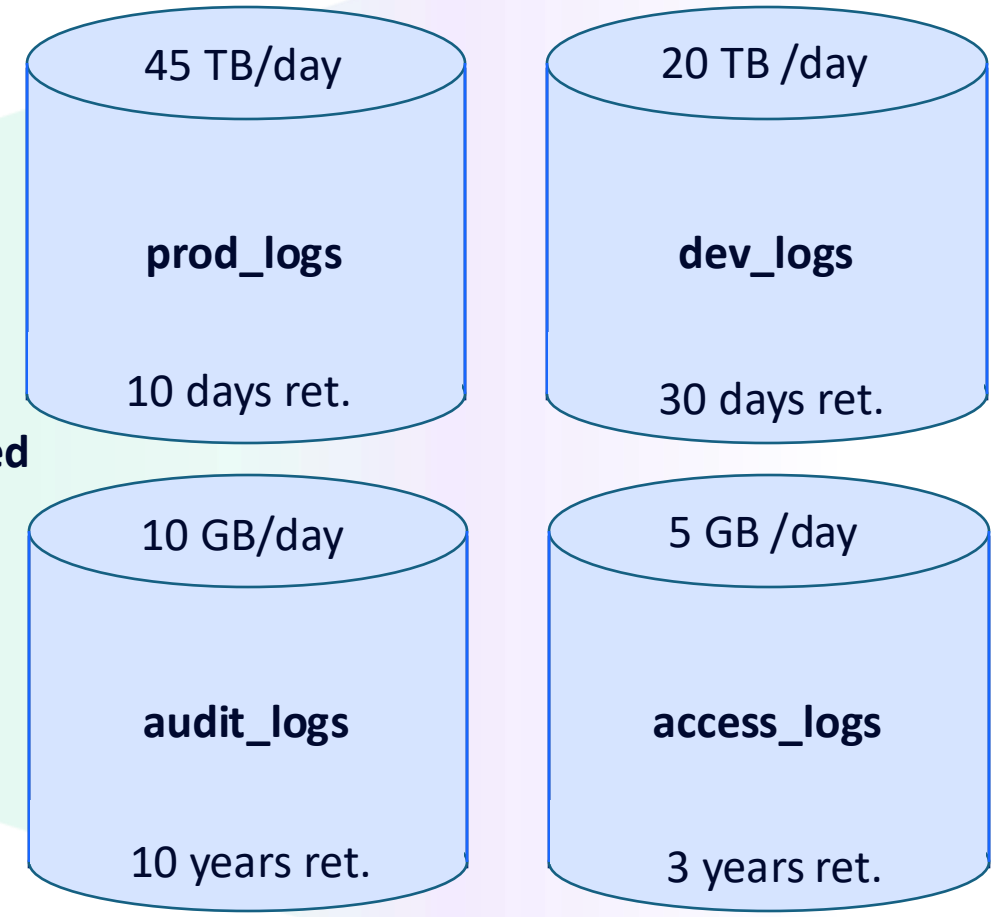


Bucket Query Strategy - Unoptimized

DQL Query:
fetch logs,
scanlimitGBytes=-1

IAM Policy:
ALLOW storage:logs:read

Scan Limit uncapped result:
~80 TB

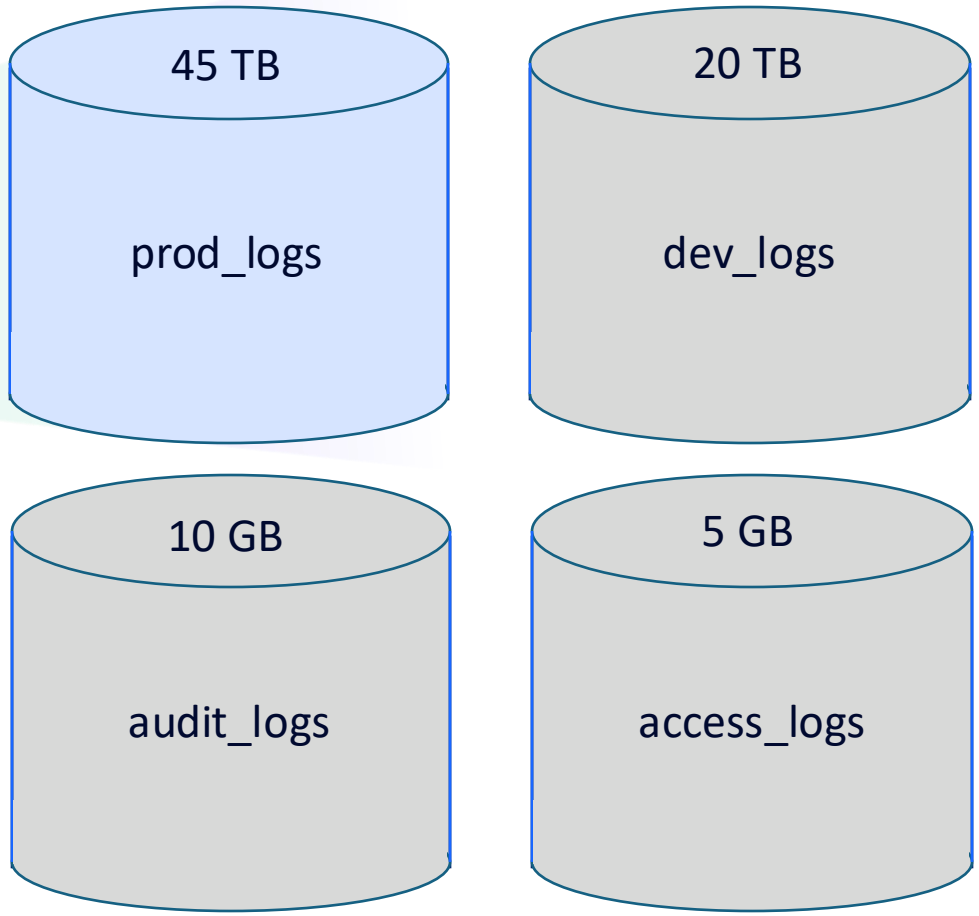


Bucket Query Strategy – Filtered, but Unoptimized Bucket

DQL Query:
fetch logs, bucket:
“**prod_logs**"

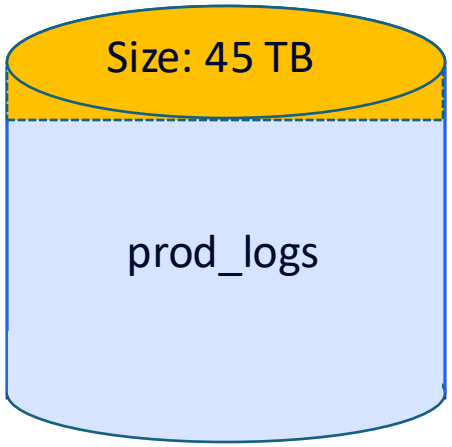
IAM Policy:
ALLOW storage:logs:read

Default Scan Limit:
500 GB



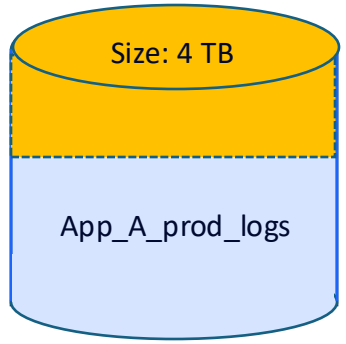
Bucket Strategy – Why you should have one?

Default Scan Limit: 500 GB (0.5 TB)
1.11% of 45 TB per Day = 0.5 TB



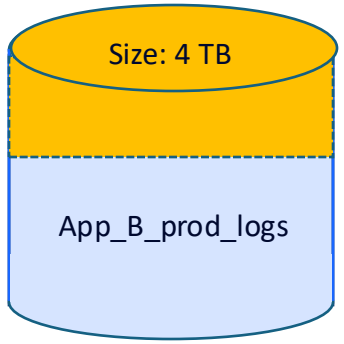
0.5 TB scanned logs
16 minutes (1.11%)

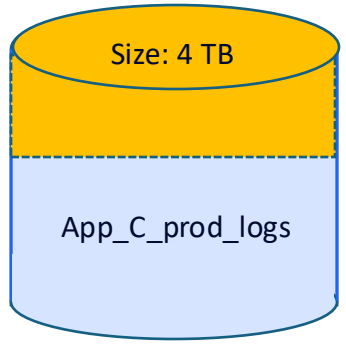
24h of retained logs



24h of retained logs

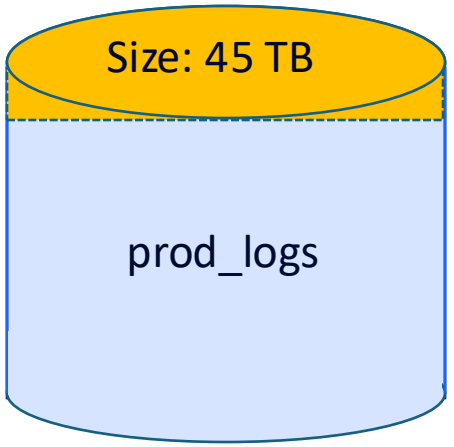
0.5 TB scanned logs
3 hours (12.5%)





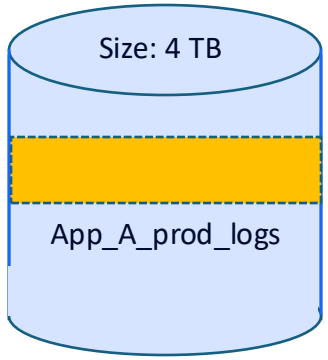
Bucket Query Strategy – Optimize Your Query Window

Default Scan Limit: 500 GB (0.5 TB)
fetch logs, bucket: “App_x_prod_logs“,
from: -6h, to:-4h



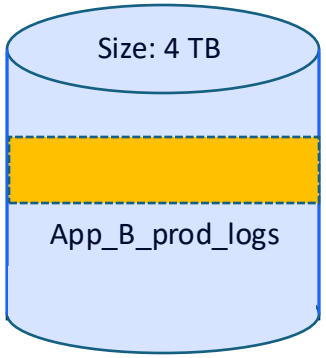
0.5 TB scanned logs
16 minutes (1.11%)

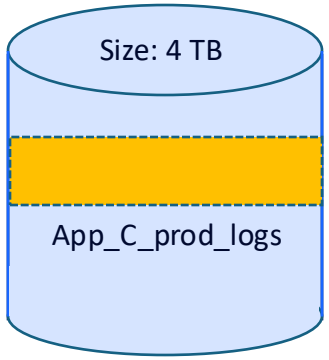
24h of retained logs



24h of retained logs

0.33 TB scanned logs
2 hours (8.3%)





Bucket Query Strategy – Filter using Segments

DQL Query:
fetch logs

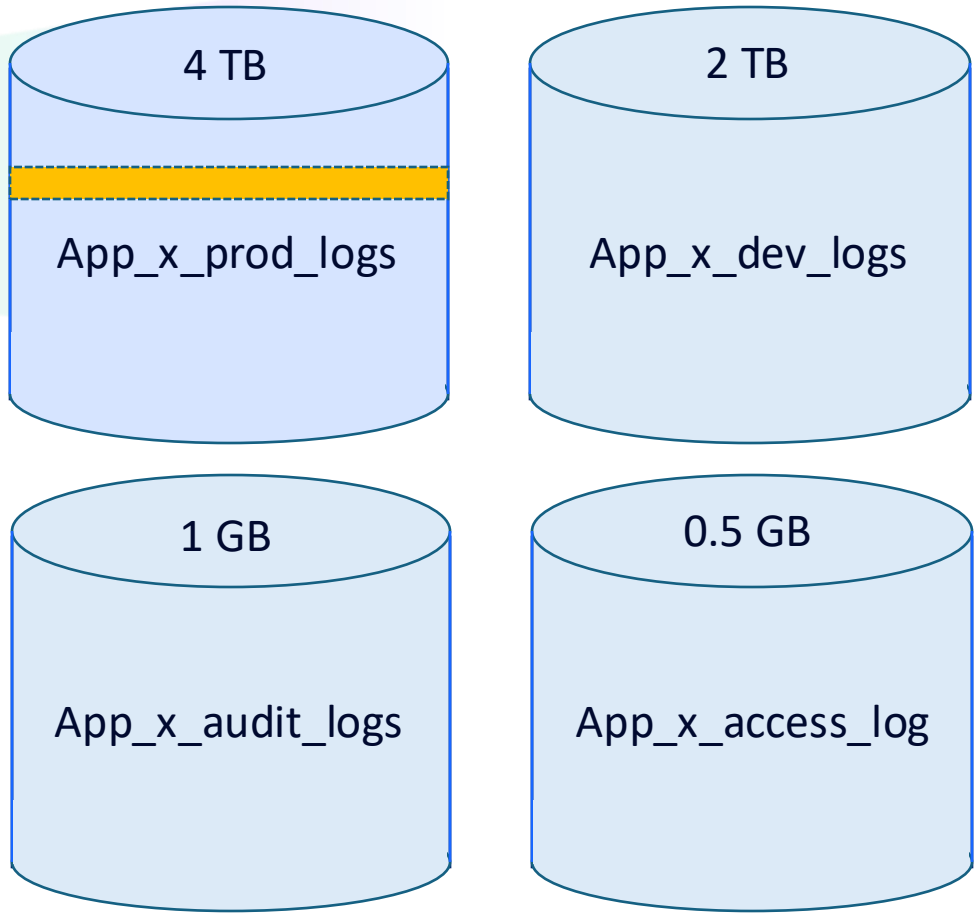
IAM Policy:
ALLOW storage:logs:read

Segment:
Bucket: "App_x_prod_logs"
k8s.namespace="App_X"

Filter: K8s App_X

* User or admin defined

Filtered on-demand by segment



Bucket Query Strategy – Filter using Segments

DQL Query:
fetch logs

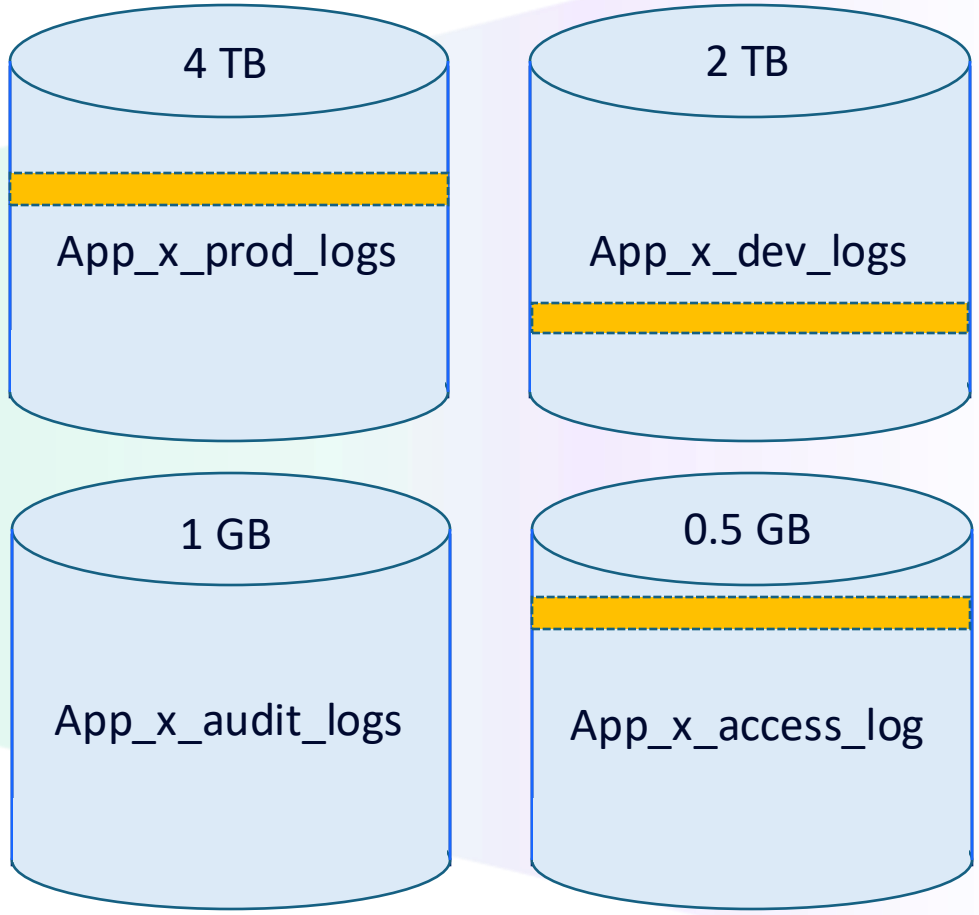
IAM Policy:
ALLOW storage:logs:read

Segment:
k8s.namespace="App_X"

Filter: K8s App_X

* User or admin defined

Filtered on-demand by segment



Bucket Query Strategy – Limit using Security context

DQL Query:
fetch logs

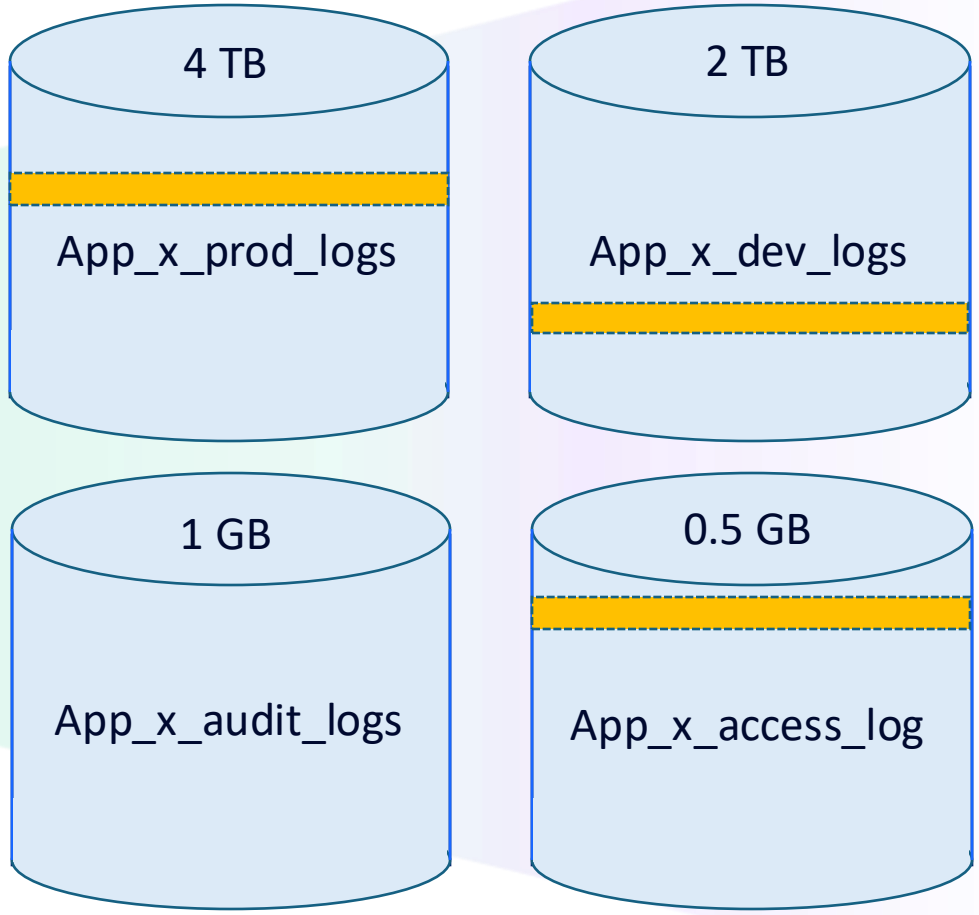
IAM Policy:
ALLOW storage:logs:read
WHERE
dt.security.context="App_X";

Security context:
dt.security.context is App_X

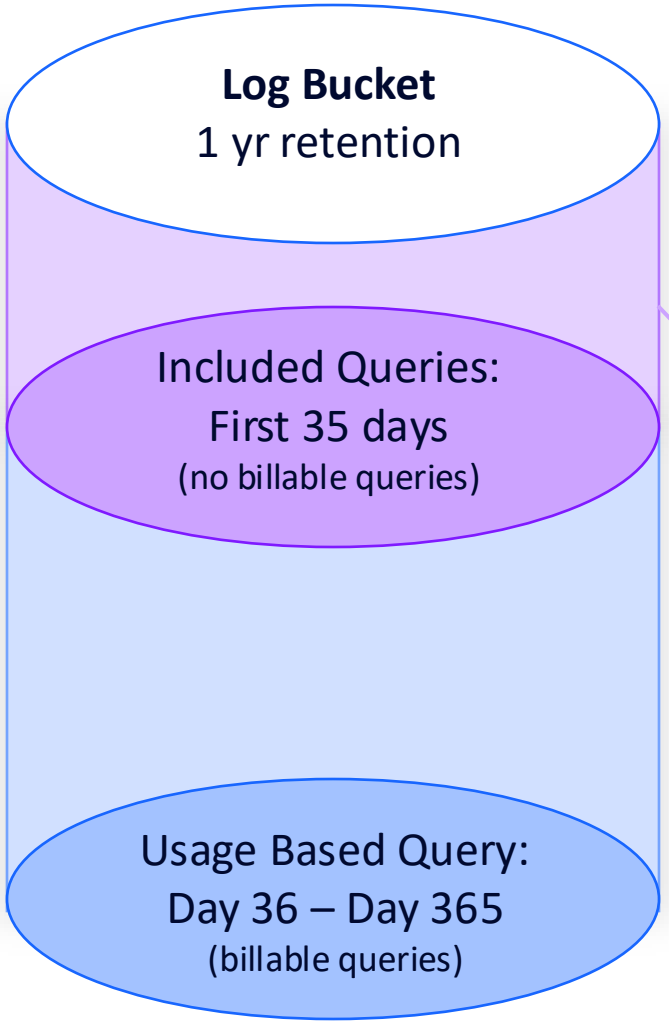
Filter: K8s App_X

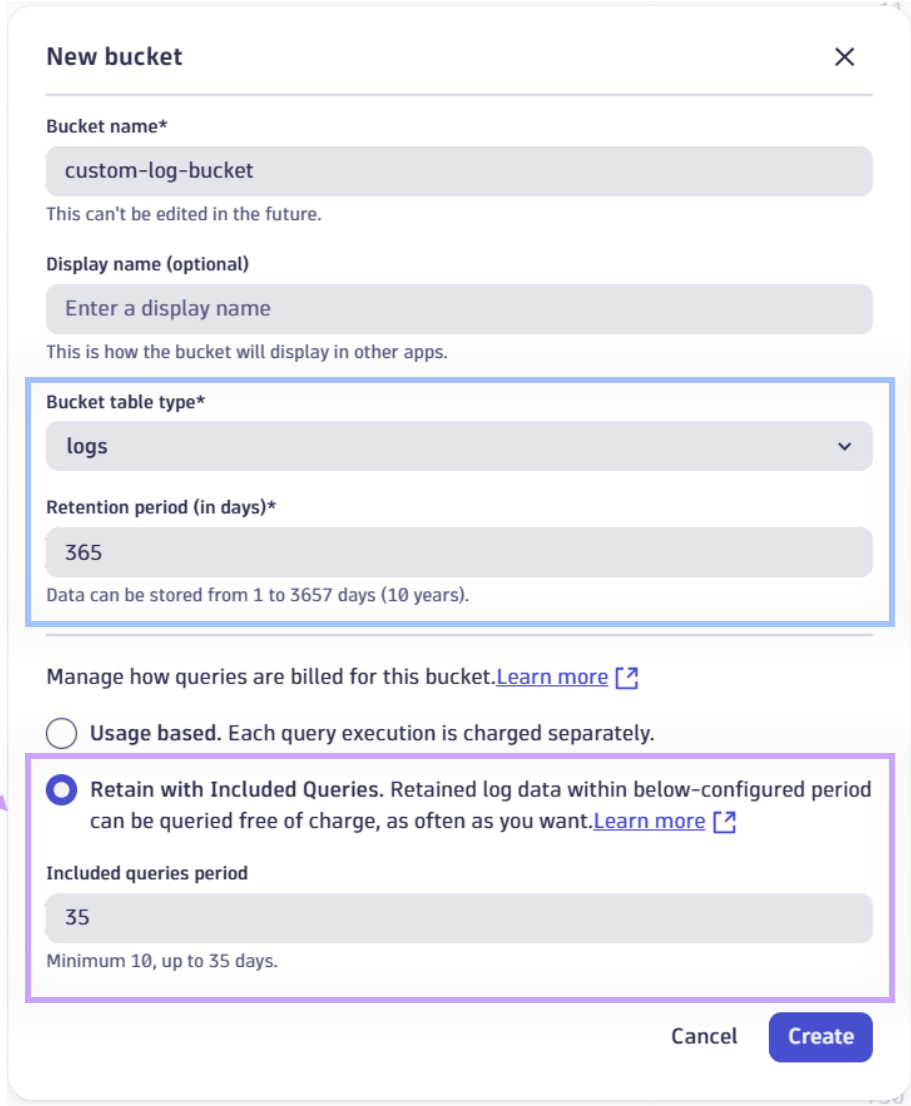
* Admin defined

Limit access based on Security context



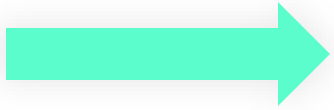
Retain with Included Queries – Configuration Details





Define your bucket scaling strategy

- Developing a tailored data partitioning (bucket) strategy should gather the following information:
  - Ingest sources
  - Estimated Ingest volume
  - Retention desires
  - Compliance Requirements (audit)
  - Sensitive data & masking
  - Permission requirements

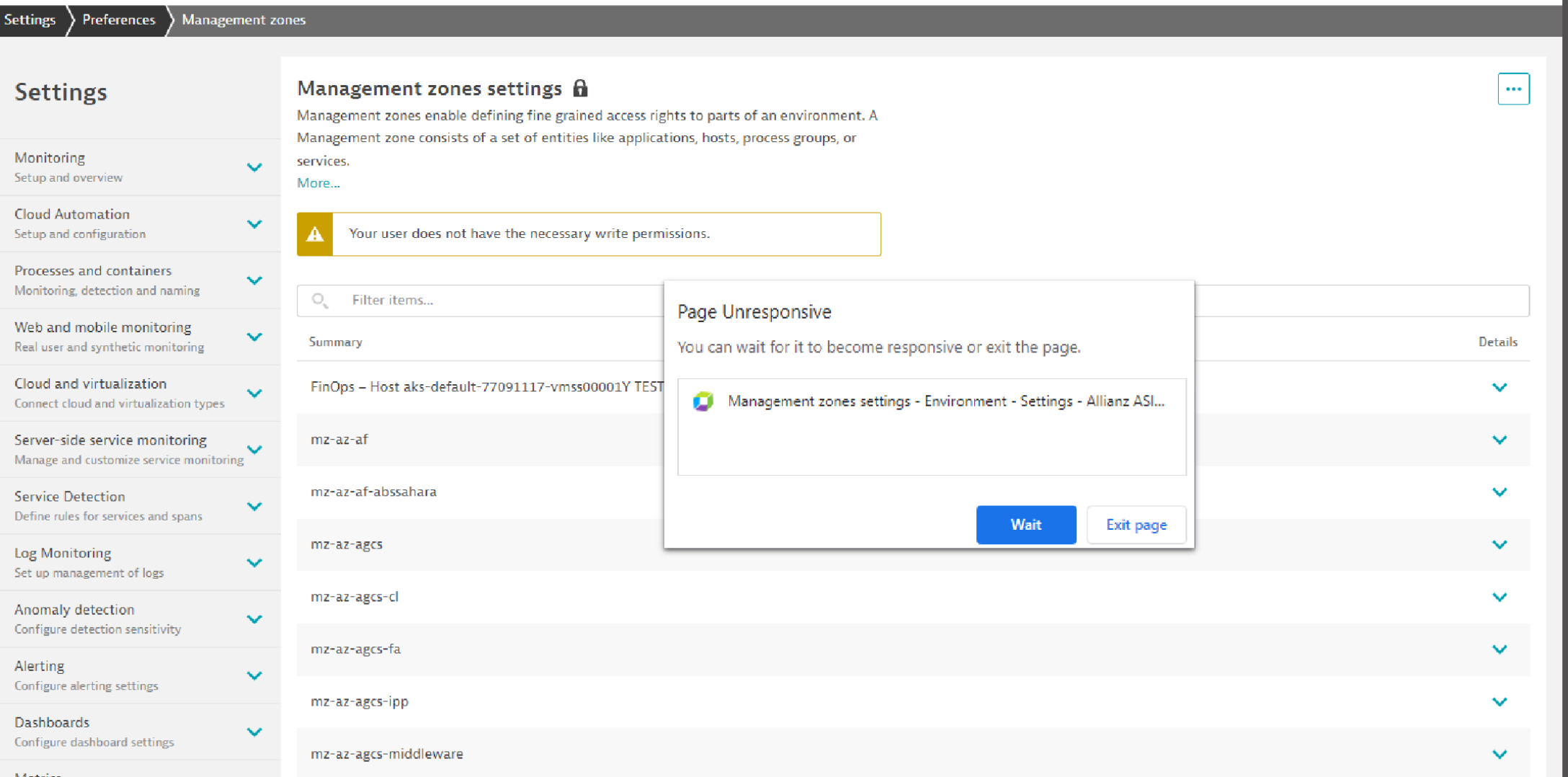


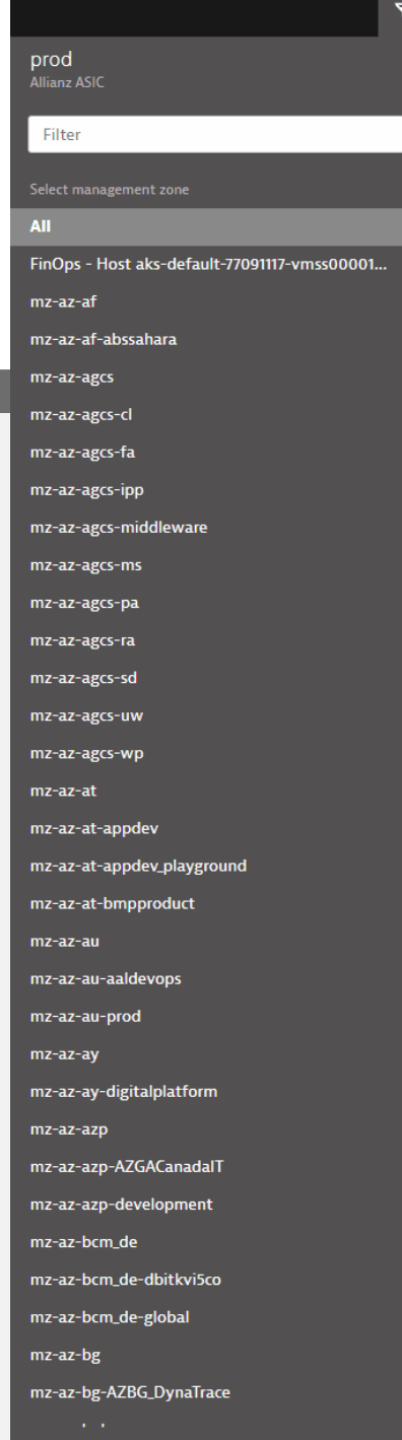
<u>Bucket strategy best practice:</u>

- Define a strict naming standard
  - <provider>_<type>_<retention>
  - <provider>_<bu>_retention.
- Consider Business units or App Groupings
- Ingest of 1-2 TB per day is ideal.
- 80 bucket limit per tenant by default
  - Can be significantly increased based on ingest volume – contact support
  - More buckets = more setup / admin management – strike a balance
- Strategy can be replicated for other high volume data types such as BizEvents or spans.

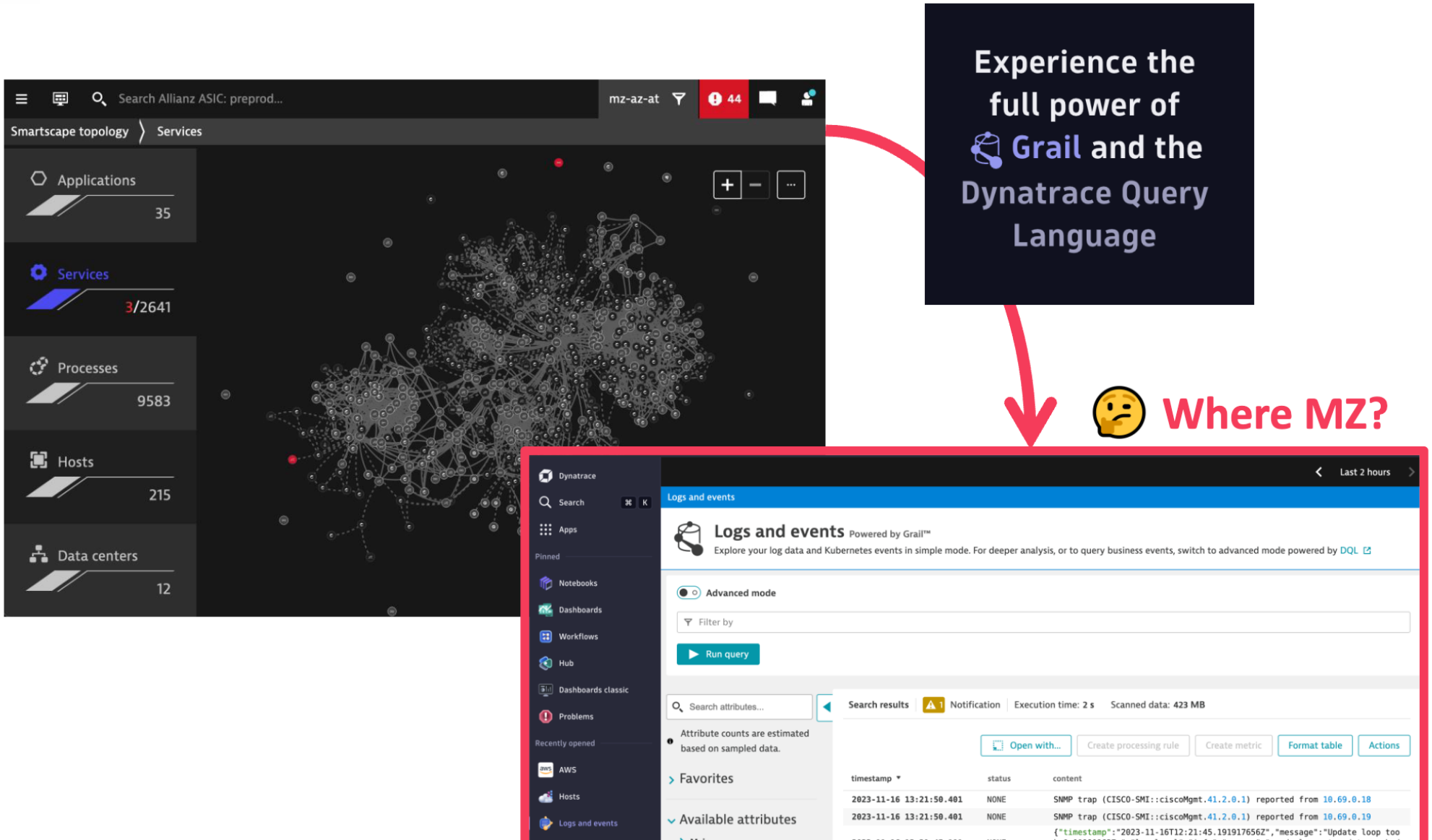
Segments

Why Segments?

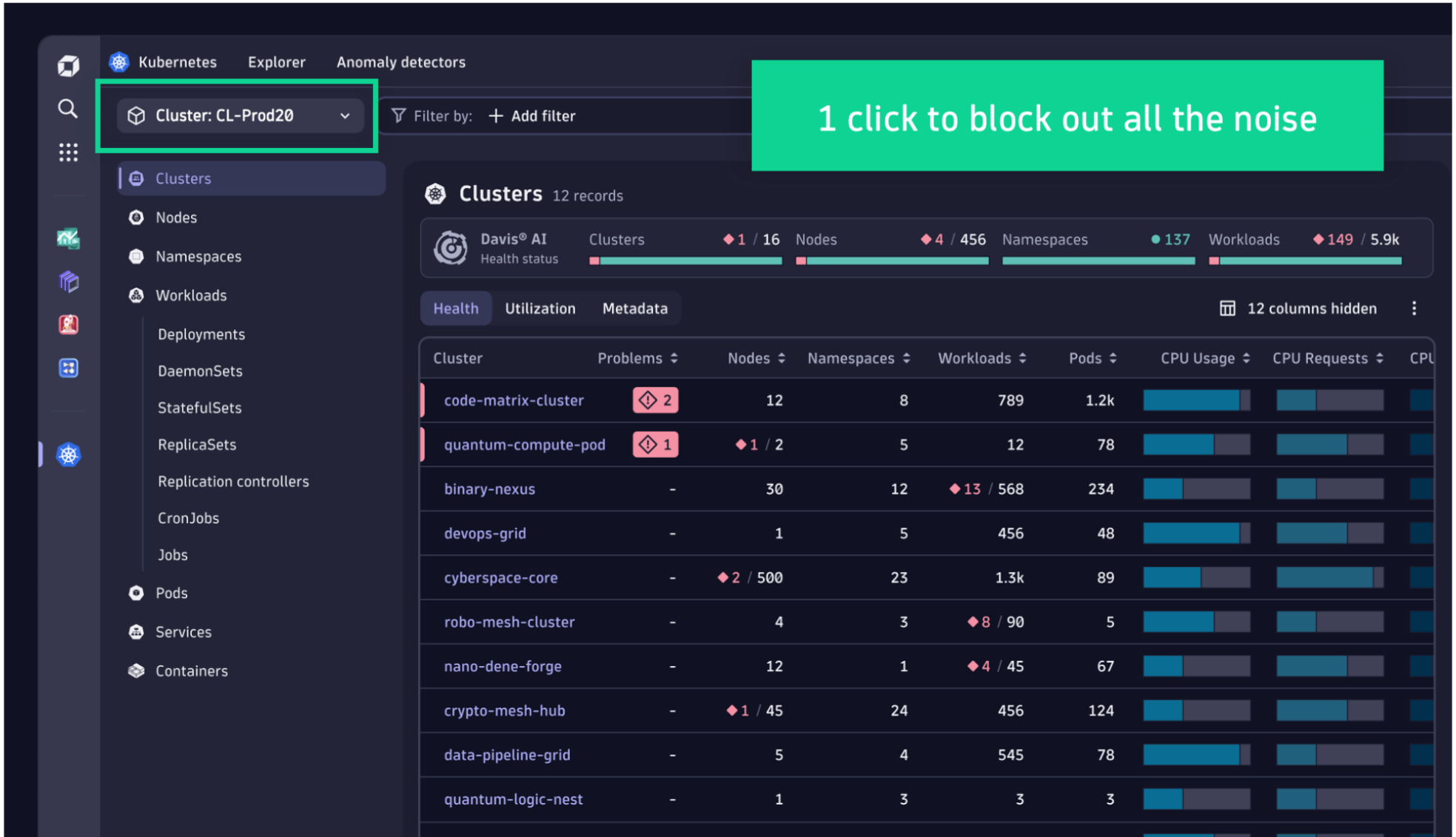




Why Segments?



Segments: Preconfigured Dynamic Filtering



Simple Segment

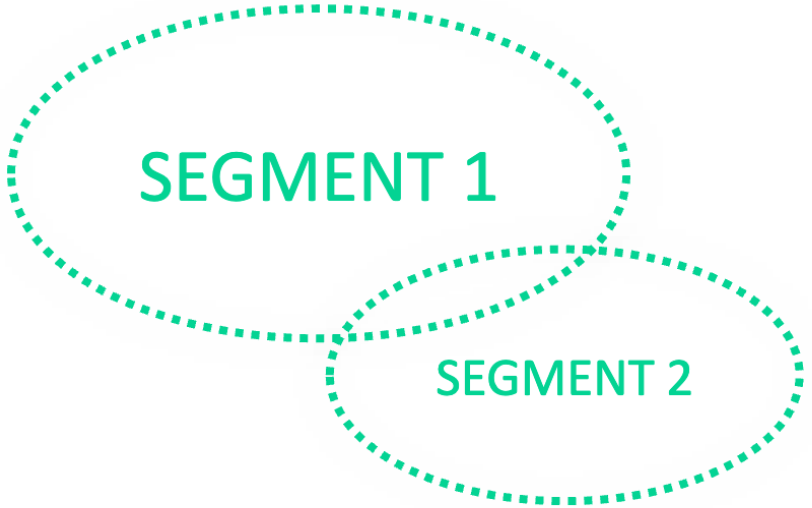
Show me entities, logs, events, etc. of my application

- Equivalent to single MZ

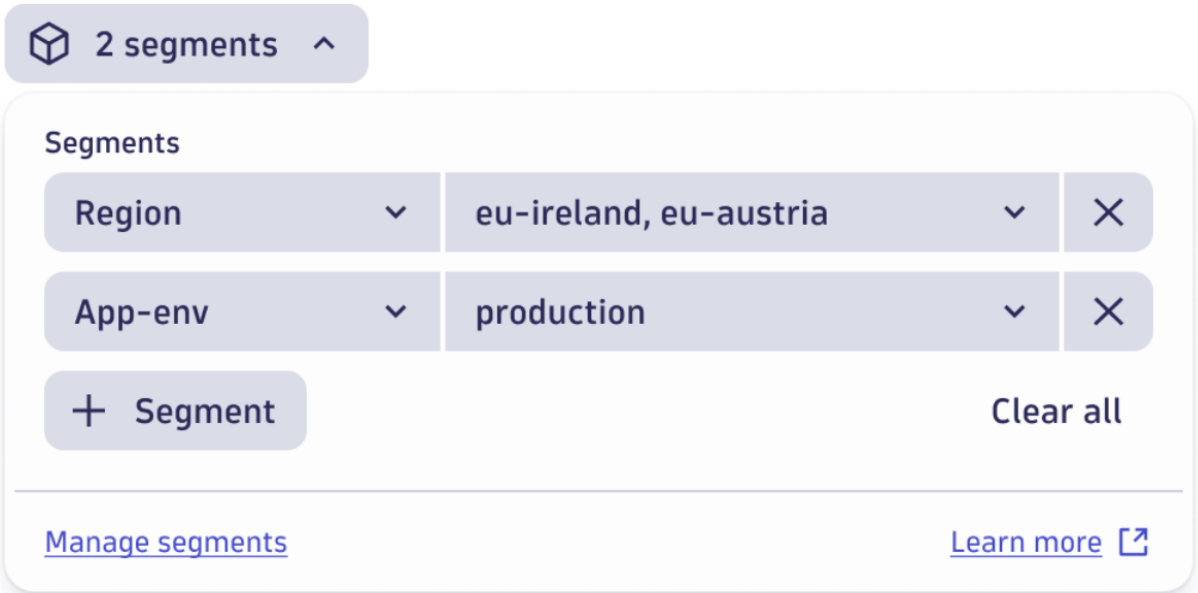
Multiple Segments

Show me data of my application from a specific infrastructure

- Intersection of multiple segments (eu-ireland OR eu-austria) AND production



How it all works



1) Segment IDs and values are passed to query API

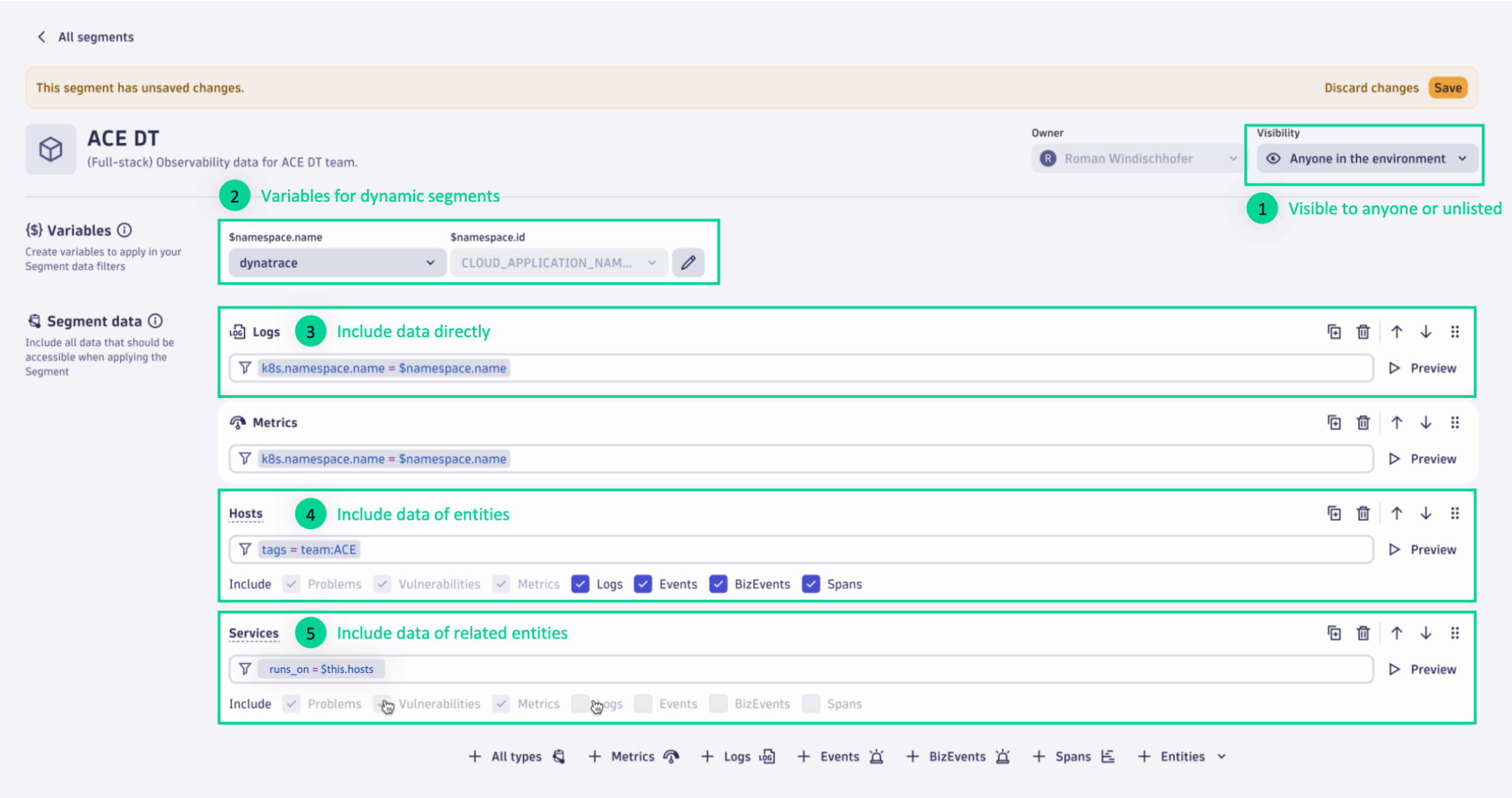
FiRmGMvWzPY(eu-ireland, eu-austria), HCkccp8aAIU(production)

2) Grail filters consumer queries with conditions defined in segments

```
fetch logs
| filter
  // permission-based filters
  ... AND
  // Segment-based filters
  (
    (condition_a1 OR condition_a2 OR ...) AND
    (condition_b1 OR condition_b2 OR ...)
  )
// consumer query continues
| summarize ...
```

Transparent filter conditions

Building a Segment

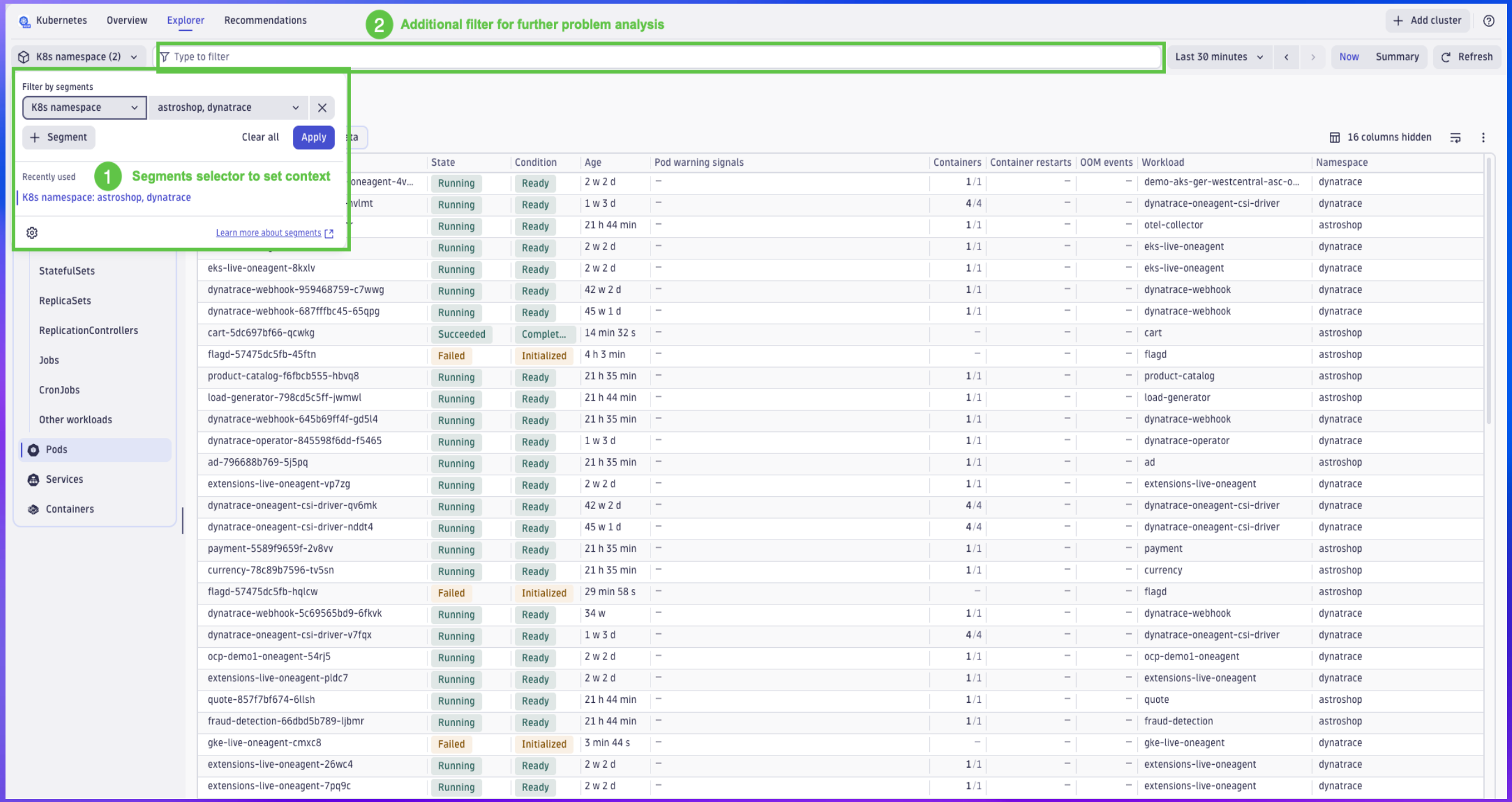


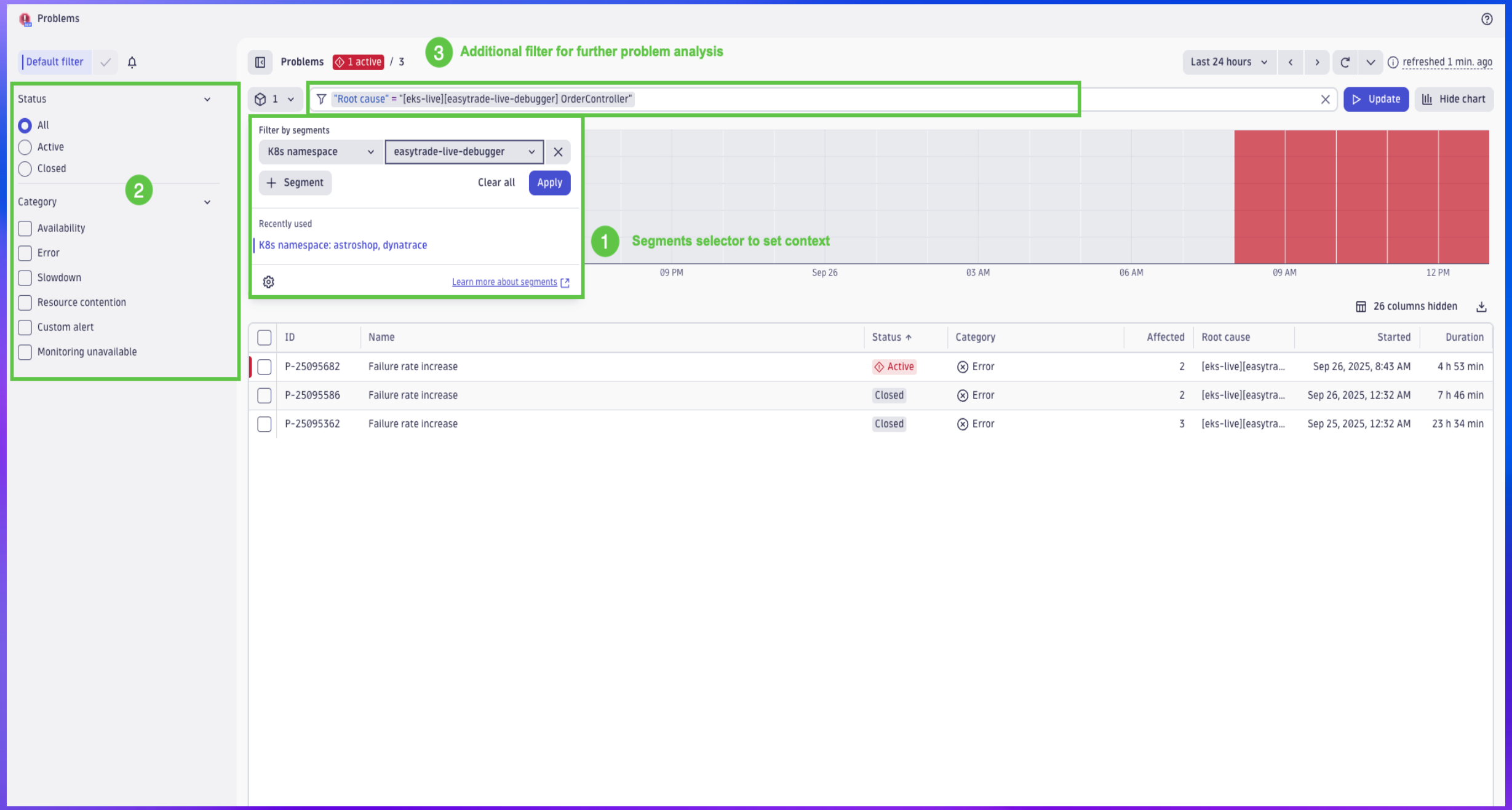
20 include blocks per segment

1 filter for data of entities (logs, events, spans)

No limit for problems, vulnerabilities, metrics

1 topology traversal step only


Kubernetes
Overview
Explorer
Recommendations
2 Additional filter for further problem analysis
Add cluster
K8s namespace (2)
Type to filter
Last 30 minutes
Now
Summary
Refresh
Filter by segments
K8s namespace
astroshop, dynatrace
Segment
Clear all
Apply
Recently used
1 Segments selector to set context
K8s namespace: astroshop, dynatrace
Learn more about segments
16 columns hidden
StatefulSets
ReplicaSets
ReplicationControllers
Jobs
CronJobs
Other workloads
Pods
Services
Containers
State
Condition
Age
Pod warning signals
Containers
Container restarts
OOM events
Workload
Namespace
eks-live-oneagent-8kxlv Running Ready 2 w 2 d 1/1 eks-live-oneagent dynatrace
dynatrace-webhook-959468759-c7wwg Running Ready 42 w 2 d 1/1 dynatrace-webhook dynatrace
dynatrace-webhook-687fffbc45-65qpg Running Ready 45 w 1 d 1/1 dynatrace-webhook dynatrace
cart-5dc697bf66-qcwkg Succeeded Complet... 14 min 32 s cart astroshop
flagd-57475dc5fb-45ftn Failed Initialized 4 h 3 min flagd astroshop
product-catalog-f6fbcb555-hbvq8 Running Ready 21 h 35 min 1/1 product-catalog astroshop
load-generator-798cd5c5ff-jwmwl Running Ready 21 h 44 min 1/1 load-generator astroshop
dynatrace-webhook-645b69ff4f-gd5l4 Running Ready 21 h 35 min 1/1 dynatrace-webhook dynatrace
dynatrace-operator-845598f6dd-f5465 Running Ready 1 w 3 d 1/1 dynatrace-operator dynatrace
ad-796688b769-5j5pq Running Ready 21 h 35 min 1/1 ad astroshop
extensions-live-oneagent-vp7zg Running Ready 2 w 2 d 1/1 extensions-live-oneagent dynatrace
dynatrace-oneagent-csi-driver-qv6mk Running Ready 42 w 2 d 4/4 dynatrace-oneagent-csi-driver dynatrace
dynatrace-oneagent-csi-driver-nddt4 Running Ready 45 w 1 d 4/4 dynatrace-oneagent-csi-driver dynatrace
payment-5589f9659f-2v8vv Running Ready 21 h 35 min 1/1 payment astroshop
currency-78c89b7596-tv5sn Running Ready 21 h 35 min 1/1 currency astroshop
flagd-57475dc5fb-hqlcw Failed Initialized 29 min 58 s flagd astroshop
dynatrace-webhook-5c69565bd9-6fkvk Running Ready 34 w 1/1 dynatrace-webhook dynatrace
dynatrace-oneagent-csi-driver-v7fqx Running Ready 1 w 3 d 4/4 dynatrace-oneagent-csi-driver dynatrace
ocp-demo1-oneagent-54rj5 Running Ready 2 w 2 d 1/1 ocp-demo1-oneagent dynatrace
extensions-live-oneagent-pldc7 Running Ready 2 w 2 d 1/1 extensions-live-oneagent dynatrace
quote-857f7bf674-6llsh Running Ready 21 h 44 min 1/1 quote astroshop
fraud-detection-66dbd5b789-ljbmr Running Ready 21 h 44 min 1/1 fraud-detection astroshop
gke-live-oneagent-cmxc8 Failed Initialized 3 min 44 s gke-live-oneagent dynatrace
extensions-live-oneagent-26wc4 Running Ready 2 w 2 d 1/1 extensions-live-oneagent dynatrace
extensions-live-oneagent-7pq9c Running Ready 2 w 2 d 1/1 extensions-live-oneagent dynatrace

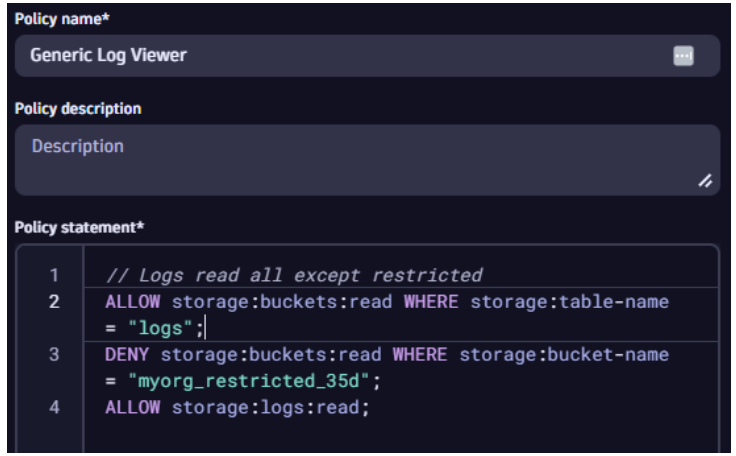

Problems
Default filter
Problems 1 active / 3
3 Additional filter for further problem analysis
Last 24 hours
refreshed 1 min. ago
Status
All
Active
Closed
2
Category
Availability
Error
Slowdown
Resource contention
Custom alert
Monitoring unavailable
1
"Root cause" = "[eks-live][easytrade-live-debugger] OrderController"
Update
Hide chart
Filter by segments
K8s namespace
easytrade-live-debugger
Segment
Clear all
Apply
Recently used
K8s namespace: astroshop, dynatrace
Learn more about segments
1 Segments selector to set context
09 PM
Sep 26
03 AM
06 AM
09 AM
12 PM
26 columns hidden
ID
Name
Status
Category
Affected
Root cause
Started
Duration
P-25095682
Failure rate increase
Active
Error
2
[eks-live][easytra...
Sep 26, 2025, 8:43 AM
4 h 53 min
P-25095586
Failure rate increase
Closed
Error
2
[eks-live][easytra...
Sep 26, 2025, 12:32 AM
7 h 46 min
P-25095362
Failure rate increase
Closed
Error
3
[eks-live][easytra...
Sep 25, 2025, 12:32 AM
23 h 34 min

Permissions

Permissions

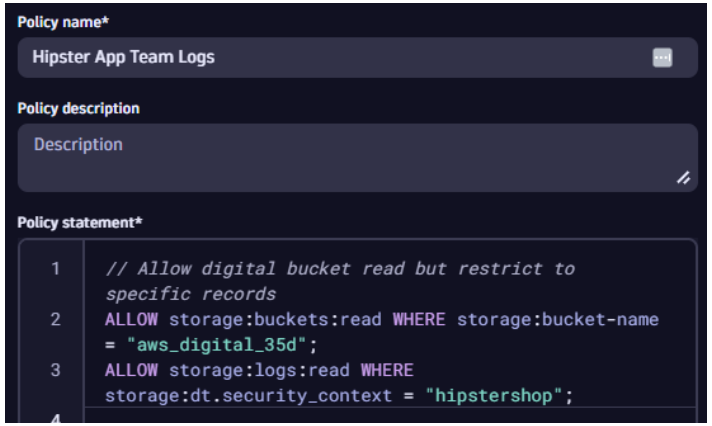
1. Bucket and table access

Basic functionality
- Organize records in buckets and table
- Control access per table and per bucket
- Optimized for query performance



2. Record-level security

Record filtering
- Filter out records based on known fields
- Optimized for query performance



3. Field-level security

Data masking
- Filter out fields that contain sensitive data by default
- Admin can access sensitive fields
- Optimized for query performance

Record/Field-level security brings fine-grained access controls

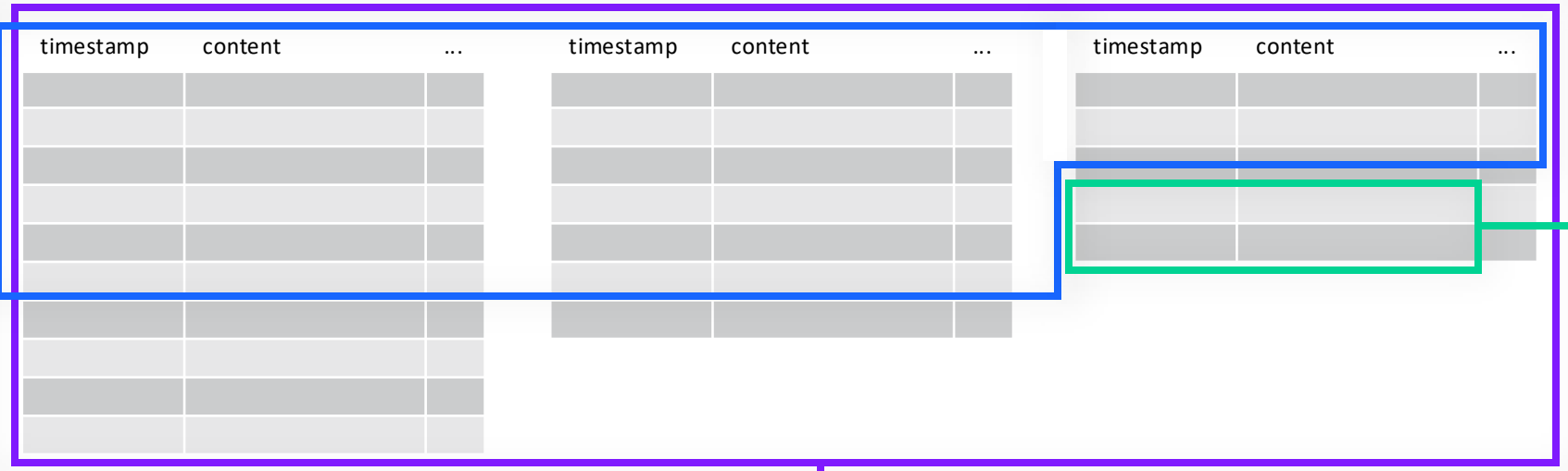
- Allows customer to create access policies on **record** or **field** level in Grail

Bucket: audit
Retention: 7y
"All audit logs"

Bucket: infra
Retention: 3mo
"Business unit X infra logs"

Bucket: debug
Retention: 10d
"All debug logs"

I needs access to my business unit's infra logs, security events and debug data



When developing or troubleshooting services, I need to monitor performance and regression

Makes sure the Dynatrace monitoring components are deployed, updated and working fine.

How do we set record-level permissions?

Field name	IAM condition	Supported IAM tables
`event.kind`	`storage:event.kind`	`events`, `bizevents`, `system`
`event.type`	`storage:event.type`	`events`, `bizevents`, `system`
`event.provider`	`storage:event.provider`	`events`, `bizevents`, `system`
`k8s.namespace.name`	`storage:k8s.namespace.name`	`events`, `bizevents`, `logs`, `metrics`, `spans`
`k8s.cluster.name`	`storage:k8s.cluster.name`	`events`, `bizevents`, `logs`, `metrics`, `spans`
`host.name`	`storage:host.name`	`events`, `bizevents`, `logs`, `metrics`, `spans`
`dt.host_group.id`	`storage:dt.host_group.id`	`events`, `bizevents`, `logs`, `metrics`, `spans`
`metric.key`	`storage:metric.key`	`metrics`
`log.source`	`storage:log.source`	`logs`
`dt.security_context`	`storage:dt.security_context`	`events`, `bizevents`, `system`, `logs`, `metrics`, `spans`, `entities`
`gcp.project.id`	`storage:gcp.project.id`	`events`, `bizevents`, `logs`, `metrics`
`aws.account.id`	`storage:aws.account.id`	`events`, `bizevents`, `logs`, `metrics`
`azure.subscription`	`storage:azure.subscription`	`events`, `bizevents`, `logs`, `metrics`
`azure.resource.group`	`storage:azure.resource.group`	`events`, `bizevents`, `logs`, `metrics`

storage:logs:read

Grants permission to read records from the logs-table

conditions:

- `storage:bucket-name` - This condition reduces the effect of the record-level permission to a defined list of buckets.
  operators: `=`, `!=`, `IN`, `NOT IN`, `startsWith`, `NOT startsWith`
- `storage:k8s.namespace.name` - The name of the namespace that the pod is running in.
  operators: `=`, `IN`, `startsWith`
- `storage:k8s.cluster.name` - The name of the cluster that the pod is running in.
  operators: `=`, `IN`, `startsWith`
- `storage:host.name` - Name of the host.
  operators: `=`, `IN`, `startsWith`
- `storage:dt.host_group.id` - Id of the host group.
  operators: `=`, `IN`, `startsWith`
- `storage:log.source` - The location where the log comes from.
  operators: `=`, `IN`, `startsWith`
- `storage:dt.security_context` - Custom field for security context.
  operators: `=`, `IN`, `startsWith`
- `storage:gcp.project.id` - Google Cloud Platform Project ID.
  operators: `=`, `IN`, `startsWith`
- `storage:aws.account.id` - Amazon Web Services Account ID.
  operators: `=`, `IN`, `startsWith`
- `storage:azure.subscription` - Azure subscription.
  operators: `=`, `IN`, `startsWith`
- `storage:azure.resource.group` - Azure resource group.
  operators: `=`, `IN`, `startsWith`

How do we set field-level permissions?

Field permissions

1

Request URL

```
POST https://myapps.mydomain.com/platform/storage/fieldsets/v1/fieldsets
```

Request body

```
{
  "name": "sensitive-fields-retail",
  "description": "Sensitive fields retail",
  "enabled": true,
  "scope": "BUCKET",
  "fields": [
    "credit_card",
    "DOB"
  ],
  "buckets": [
    "logs_retail"
  ]
}
```

2

To unmask the `credit_card` and `DOB` fields, you need the following permission:

```
ALLOW storage:fieldsets:read WHERE storage:fieldset-name="sensitive-fields-retail"
```

Discussion Question

Why would you want to create a metric from a log?

Discussion Question

Why would you want to create a metric from a log?

Query Performance

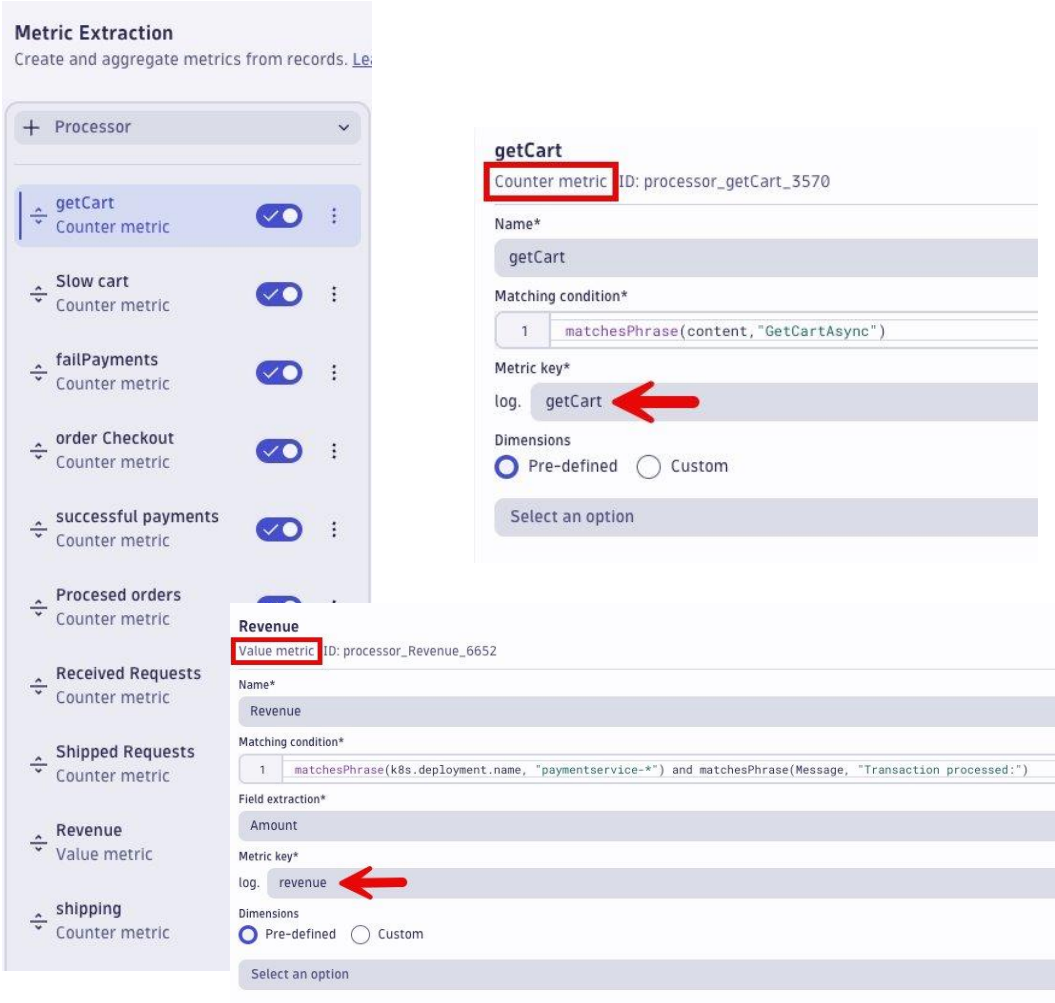
Cost efficient storage and query

Simplified alert creation

Why Logs to Metrics?

- **Enhanced Performance:**
  - Efficient storage
  - Faster query performance
  - Better scalability for handling of large volumes of data
- **Simplified Alerting:**
  - Easier to setup and manage Metrics-based alerts
  - Easier to generate dynamic thresholds
  - Easy to aggregate and summarize data

- A metric can be either a Counter or Value type.
- Counter metric - # of HTTP requests
- Value metric - Product quantity, Revenue



Log to Events

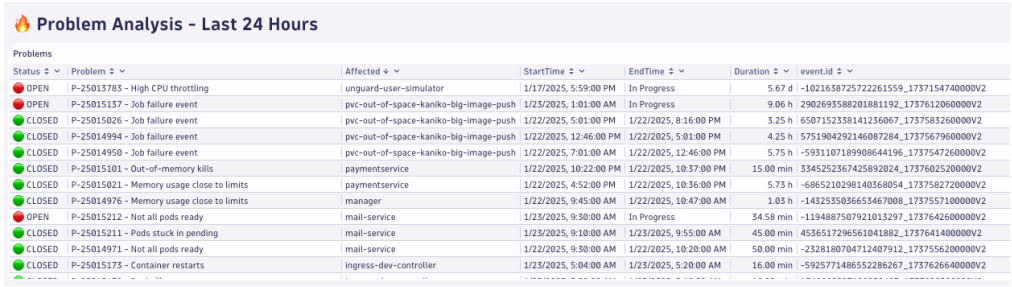
An event can be either Davis or Business event:

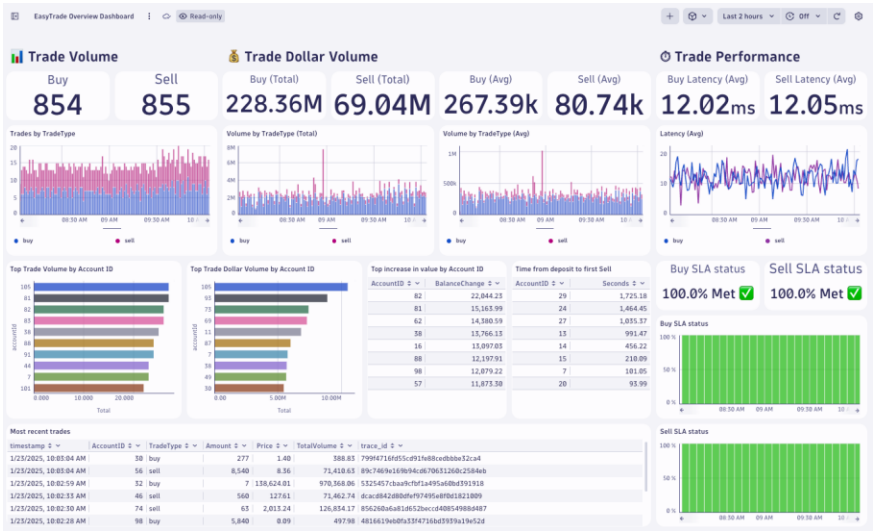
Davis Event

- Does this anomaly need to be reported?
- Is it an INFO Event that needs to be generated?

Business Event

- Is this data for reporting business data?
- Is this data for analytics for your Ops team?





Dashboarding

- Spot the differences between these two dashboards?

Use logs when:

Low frequency refresh

Involve a complex DQL

Joining different types of data

Analyzing short timeframes

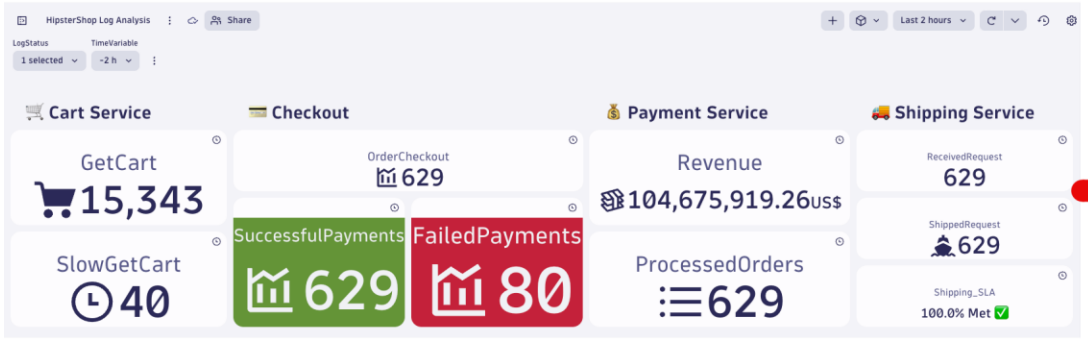
Use metrics when:

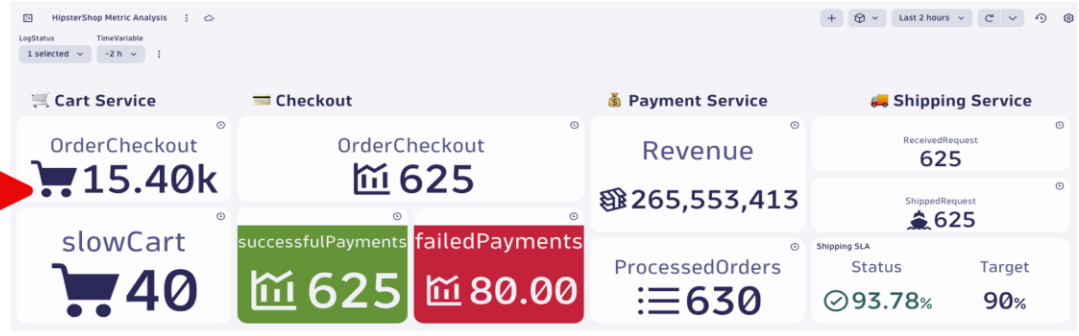
High frequency refresh

Trending over long timeframes (timeseries)

Optimize query cost

To create SLOs alerts

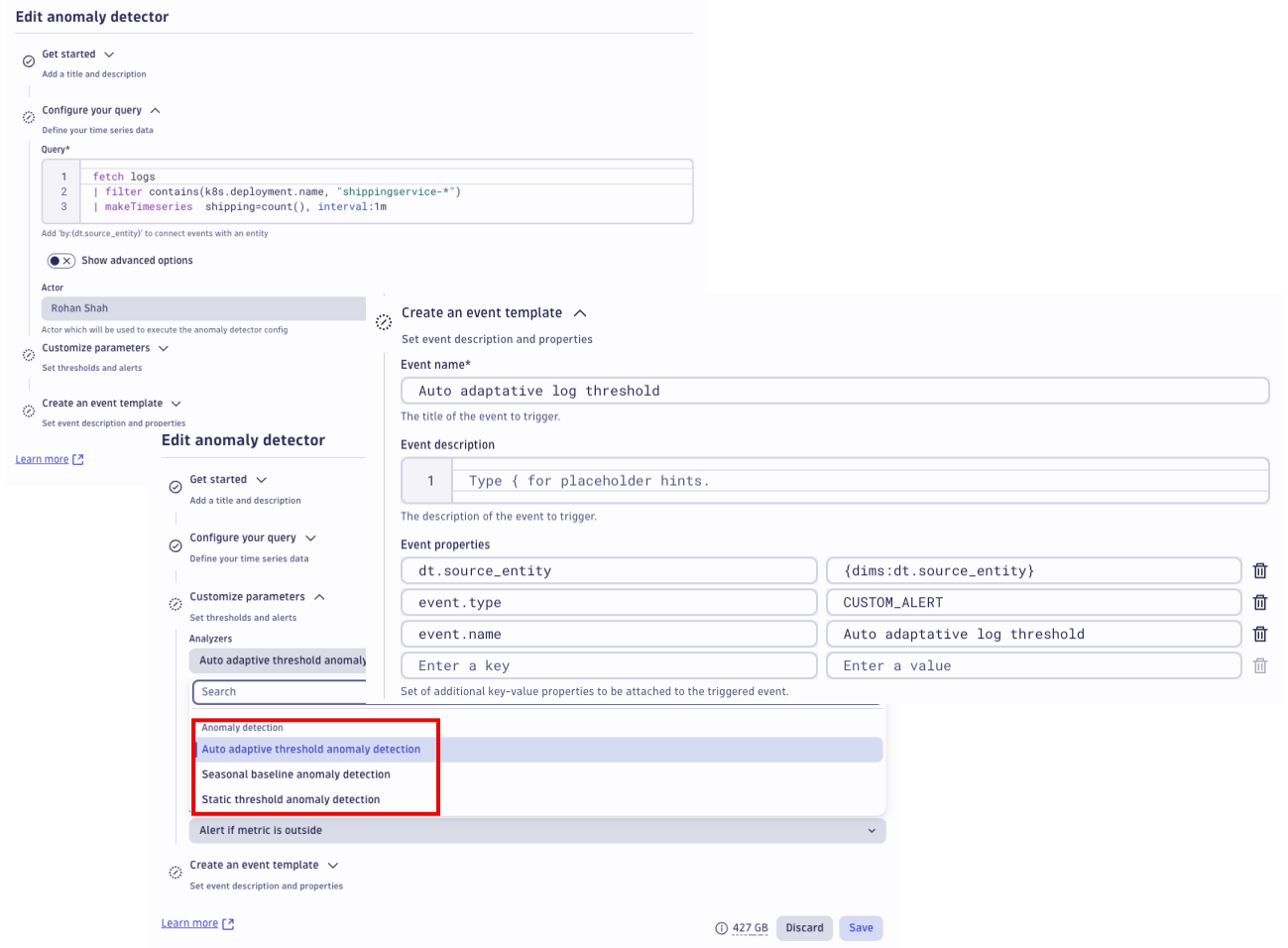




Alerting

- **Use logs when:**
  - Infrequent ingest of log records
  - Complex query
  - Detailed investigations

- **Use metrics when:**
  - Frequent data points
  - Dynamic thresholds
  - Performance tracking



Thank you!