

**DESCUBRIENDO
DYNATRACE:
LA PLATAFORMA
INTEGRAL PARA
OBSERVABILIDAD,
SEGURIDAD Y
ANÁLITICAS AVANZADAS**



Jose Manuel
Iglesias
SE MANAGER
DYNATRACE

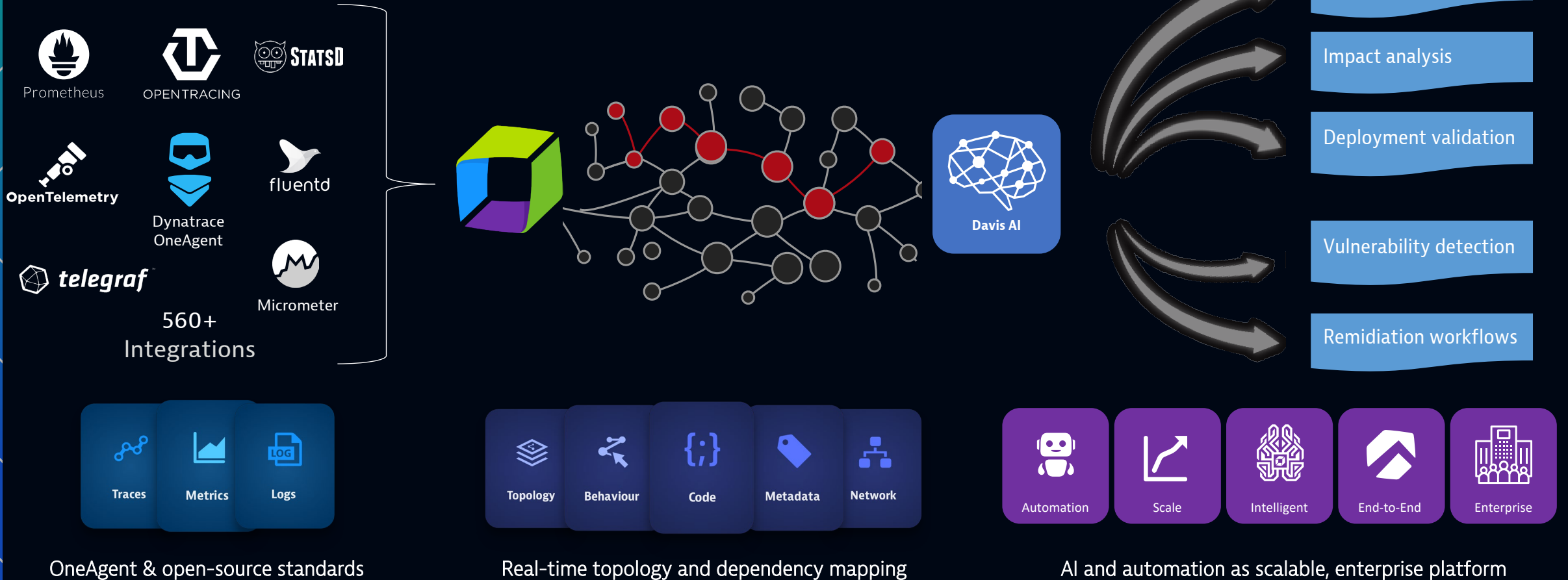
AGENDA

- The Platform
 - How it worked & Works
 - Answers & Intelligent automation from data
 - GRAIL
 - AppEngine
- Monitoring vs Observability
- Analytics



THE PLATFORM

HOW IT WORKS










Analytics and Automation for Unified Observability and Security **CLOUD DONE RIGHT.**

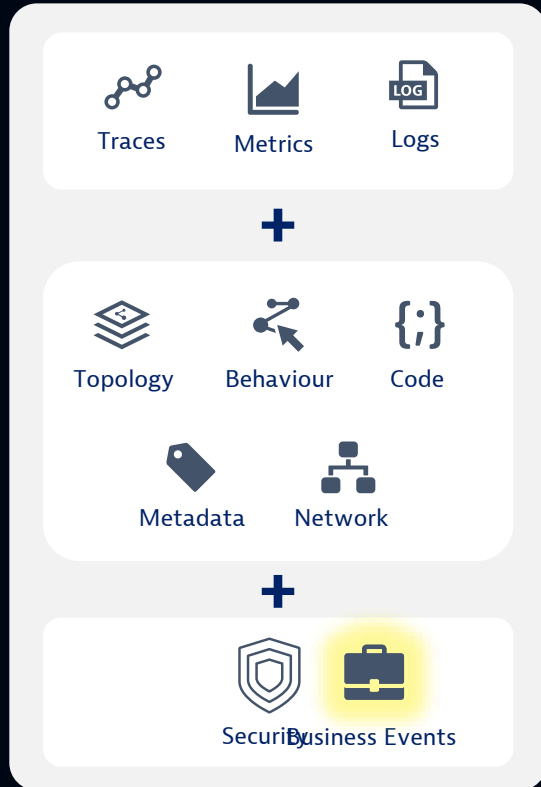
-  Infrastructure Observability
-  Application Observability
-  Security Protection
-  Security Analytics
-  Digital Experience
-  Business Analytics
-  Automations
-  Custom Solutions

Platform

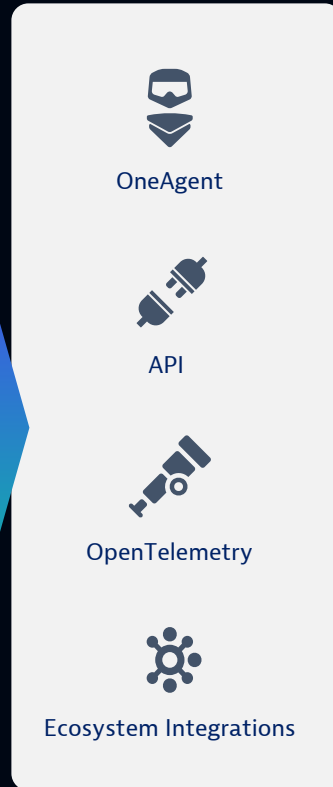
-  AutomationEngine
-  AppEngine
-  Smartscape®
-  Davis® AI
-  Grail™
-  Hub
-  Unified Ingest
-  PurePath®
-  OneAgent®

-  Topology
-  Traces
-  Metrics
-  Logs
-  Behaviour
-  Code
-  Metadata
-  Network

ANSWERS AND INTELLIGENT AUTOMATION FROM DATA



Deep context rich,
full stack beyond
observability
sources



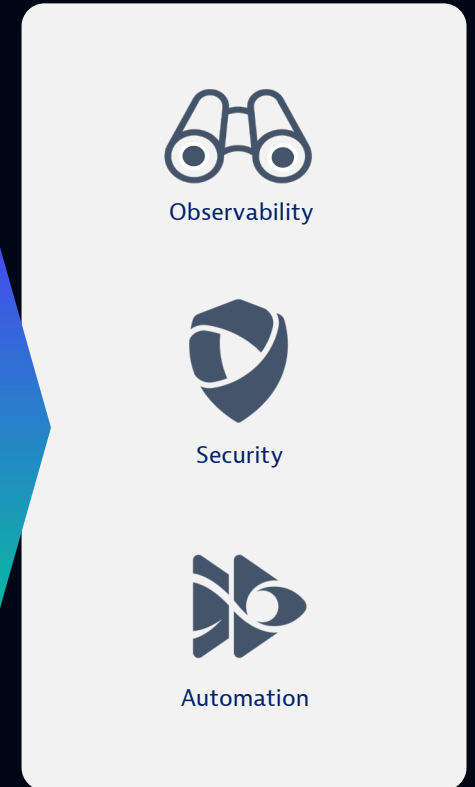
Automatically
captured in context
& pre-processed



Stored contextually with
massive processing and
retrieval capabilities



Accessed by our
causal AI for
analysis & answers



Powering automation,
orchestration, prevention
and protection

GRAIL – YOUR DATA GOLDMINE

Grail is a **causational data lakehouse** with a **massively parallel processing** (MPP) analytics engine. It leverages the new Dynatrace Query Language (DQL) for **context-rich** analytics.

Expanding the data lakehouse supporting more data types and contextual data mapping:

- Logs
- Business Events
- Graph (Smartscape)
- + Adding Metrics
- + Adding Traces



DYNATRACE QUERY LANGUAGE (DQL)

```
fetch logs, from:now()-20m
| filter endsWith(log.source,"/media/datastore/server-data/log/audit.config")
      and dt.host_group.id == "cluster_deve2e"
| parse content, "timestamp('yyyy-MM-dd HH:mm:ss'):ts
          id json:settings
          ipaddr:client_ip //IPv4/6"
| fields ts,
         type = settings[eventType],
         tenant = settings[tenantId],
         user = settings[userId],
         change = settings[jsonPatch]
| filter in(type,array("UPDATE","DELETE")) and user != "unknown"
| summarize creates = countIf(type=="CREATE"), upd = countIf(type=="UPDATE"),
             by:{tenant, user}
| fieldsAdd changes_per_min = (upd + del)/20
| sort changes_per_min desc
```



Purpose-built

for observability and security

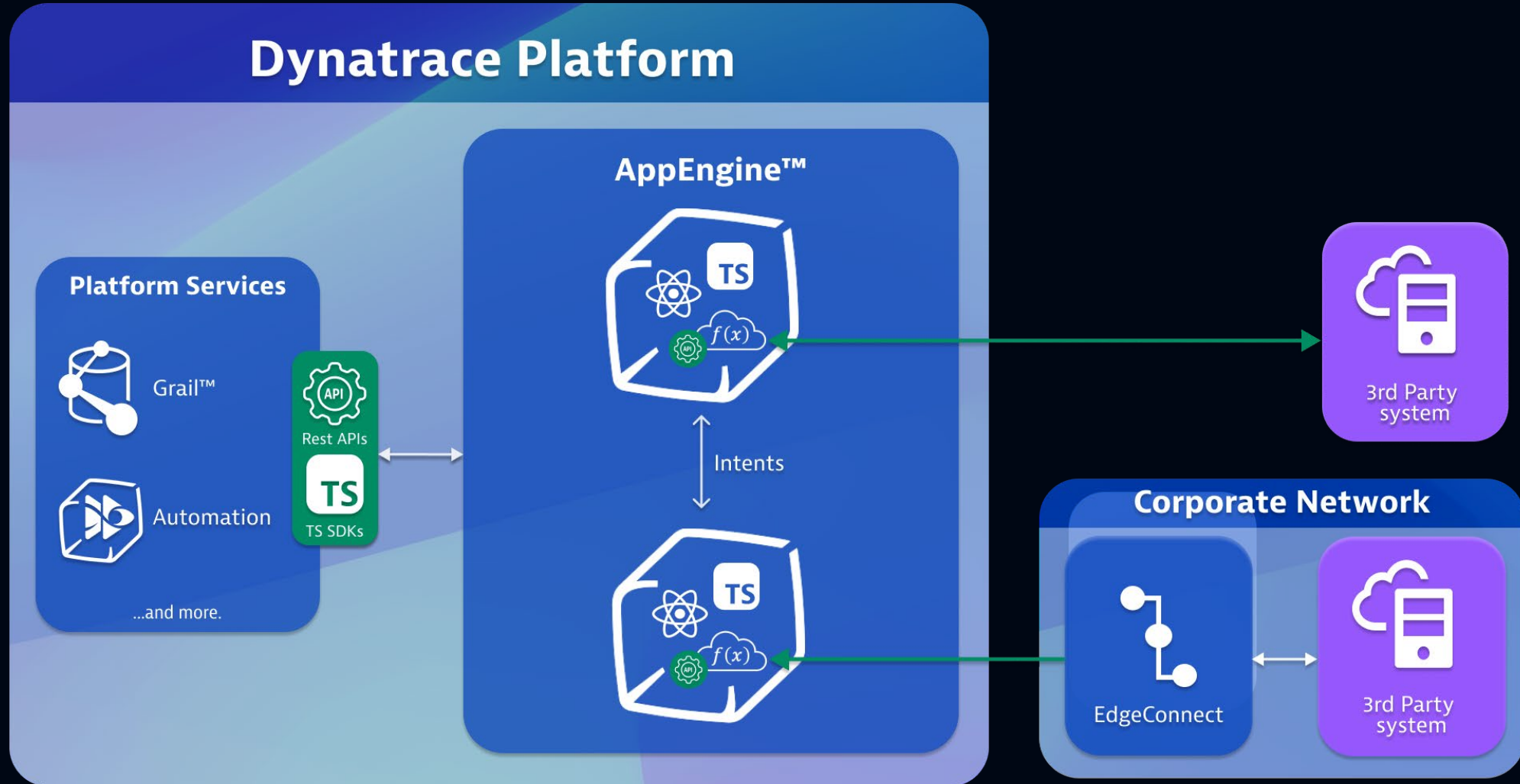
Powerful

*for even advanced use-cases,
parsing built-in*

Effortless

migration from e.g. Splunk

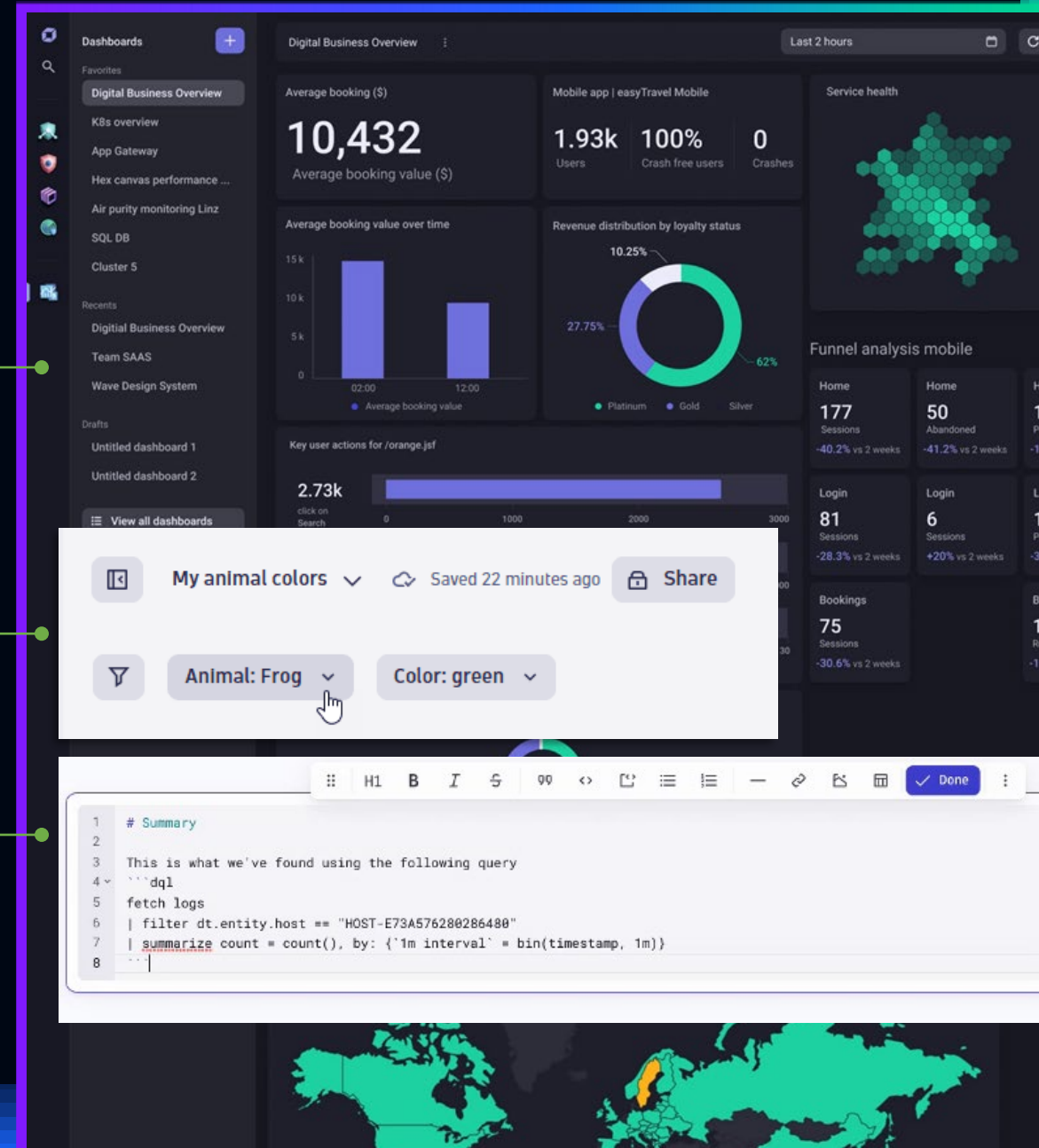
THE DYNATRACE PLATFORM



DASHBOARDS

Easily visualize metrics, trends, and anomalies with charts.

- Visualize data
 - From Grail using Dynatrace Query Language (DQL).
 - Even from external sources using Dynatrace functions.
- Filter data using variables.
 - Add comments using Markdown.



NOTEBOOKS

Petabyte-scale data exploration & analytics for real-time insights.

- Notebooks are interactive apps
- Write a DQL (Dynatrace Query Language) query or code snippet
- Get the resulting table or chart immediately.
- Ask follow-on questions at the speed of your thoughts, get trend forecasts and assistance by Davis AI.
- Query, analyze, and visualize all data in Dynatrace and even integrate external data.
- Collaborate with teams using interactive, data-driven notebooks with comments.

DQL Query

Resulting chart

Dynatrace function

Resulting chart

Comments

The screenshot shows a notebook titled "My Analytic Notebook" with a search bar and "Rerun sections" button. The first cell contains a DQL query:

```
1 fetch logs, from:now()-90m
2 | filter loglevel == "WARN" or loglevel == "ERROR"
3 | summarize count = count(), by:{bin(timestamp, 1m), loglevel}
4 | sort count desc
```

 Below the query, it shows "162 records" and "Fetched: 8/31/2022, 1:19:50 PM, 51.5 kB". A "Record list" tab is selected, and a "Chart library" tab is active, displaying a bar chart of log counts over time. The second cell contains a Dynatrace function:

```
1 export default async () => {
2   const json = await fetch('https://api.openweathermap.org/data
3   return json.main.temp;
4 }
```

 Below the function, a "Chart library" tab is active, displaying a large temperature reading: "23.52°C".

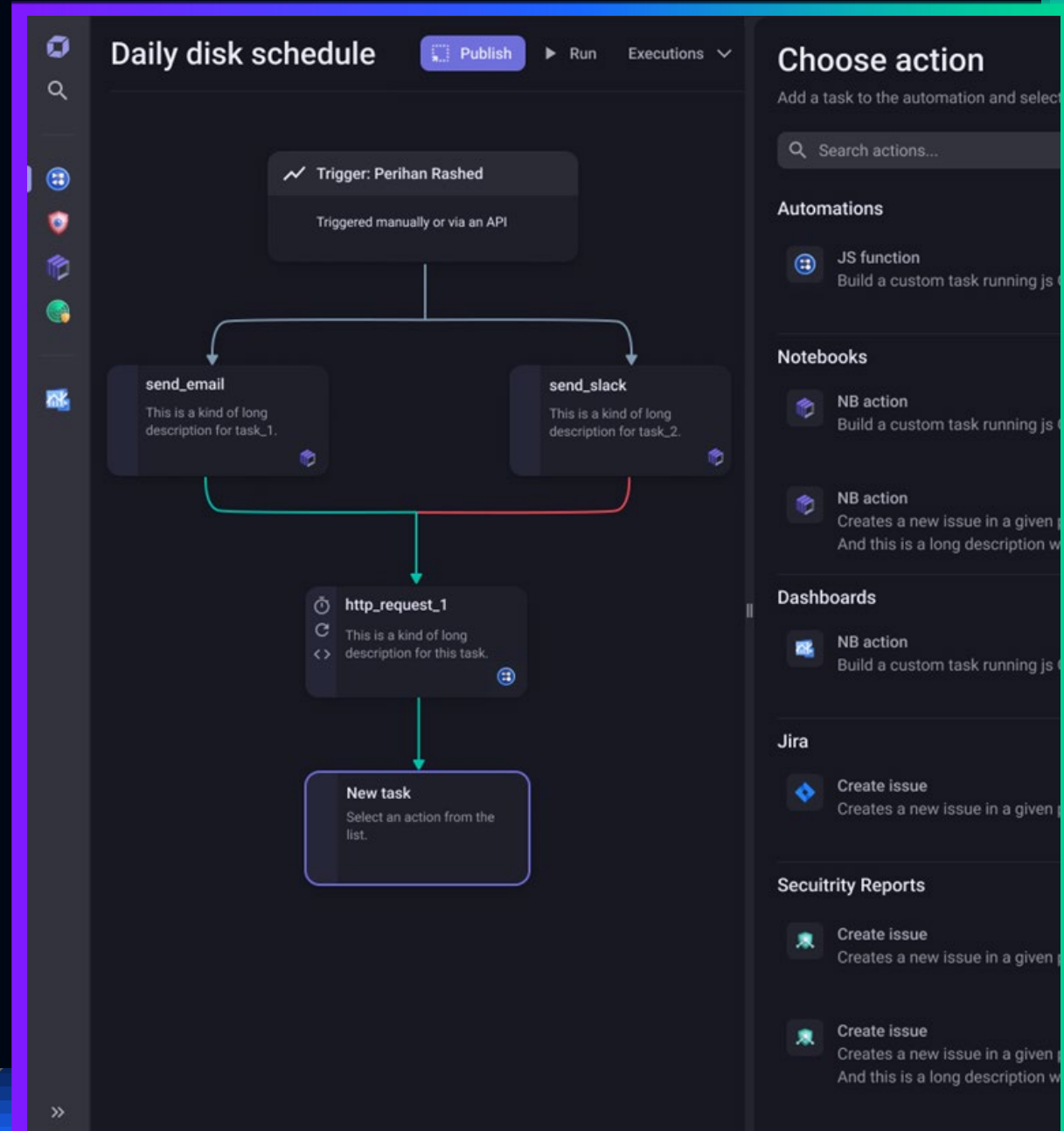
The screenshot shows a code editor with a summary comment:

```
1 # Summary
2
3 This is what we've found using the following query
4 ```dql
5 fetch logs
6 | filter dt.entity.host == "HOST-E73A576280286480"
7 | summarize count = count(), by: {'1m interval' = bin(timestamp, 1m)}
8 ...
```

WORKFLOWS

Use drag and drop to build powerful workflow automations

- Drag and drop tasks and draw dependencies in a no/low code graphical editor
- Example workflows:
 1. Query ticketing system
 2. Send a notification
- Automations are triggered based on
 - Schedule-triggered
Time-based execution according to flexible rules
 - Event-triggered
Automatically react to any Dynatrace Events or Problems
 - AI-triggered
Build and run based on Davis Analyzers
- Besides UI, configuration as code (Gitops) is possible as well





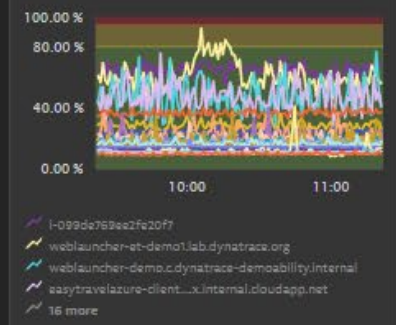
**MONITORIZACIÓN
VS
OBSERVABILIDAD**



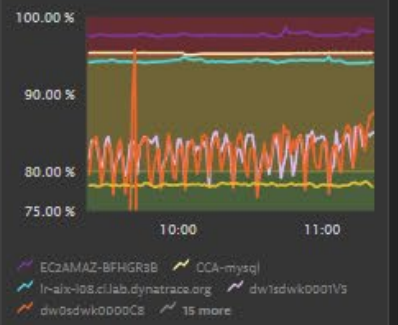


Host Metrics

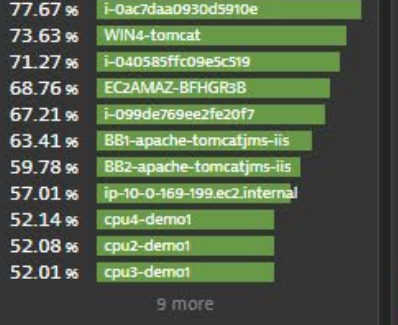
CPU Usage %



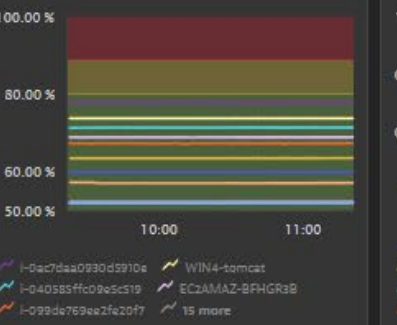
Memory Used %



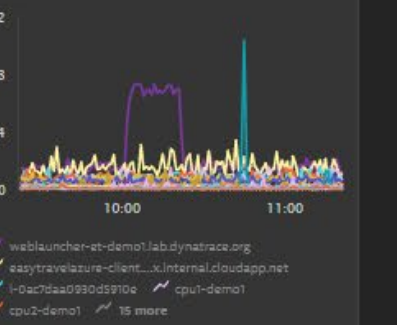
Disk Used %



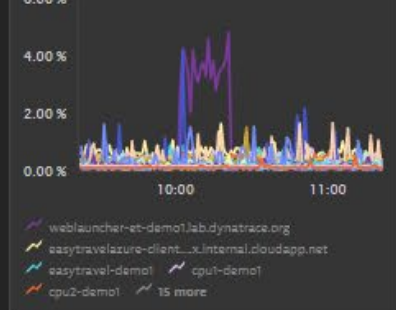
Disk Used %



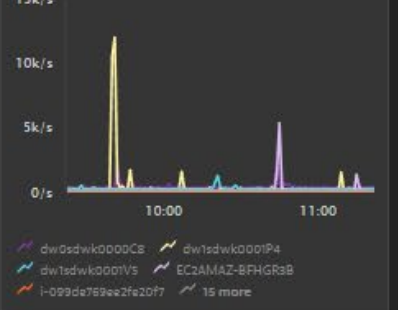
Disk Queue Length



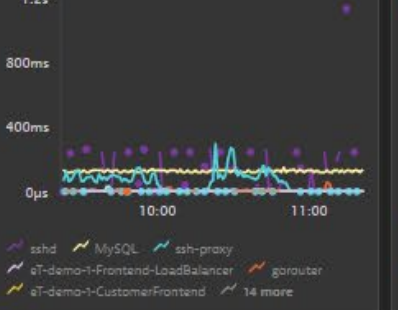
CPU IO Wait



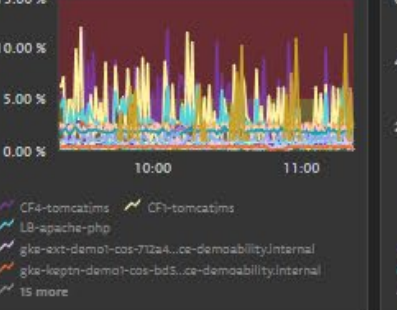
Page Faults



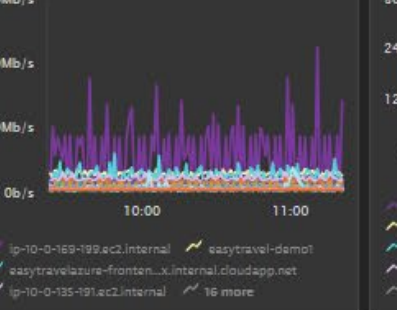
Network Latency



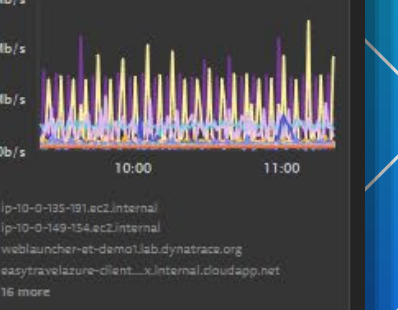
Network Retransmissions



Traffic Out

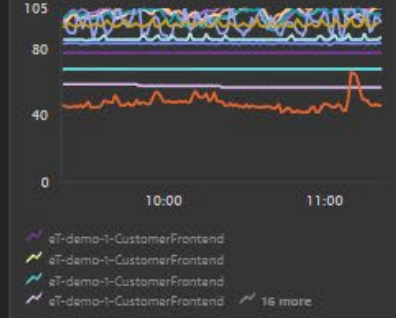


Traffic In



Process Metrics

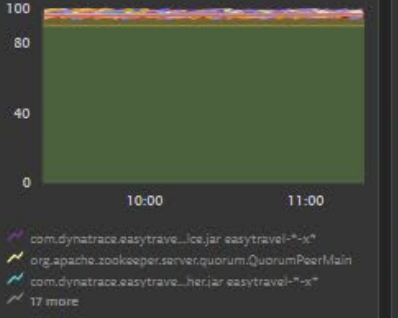
JVM Thread Count



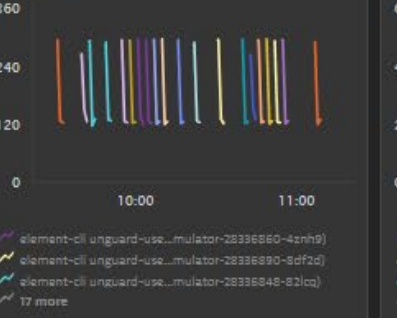
JVM total CPU time



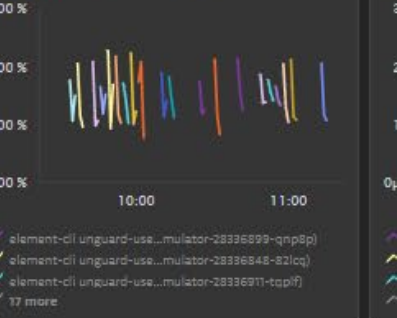
JVM % Memory Used



GC Count

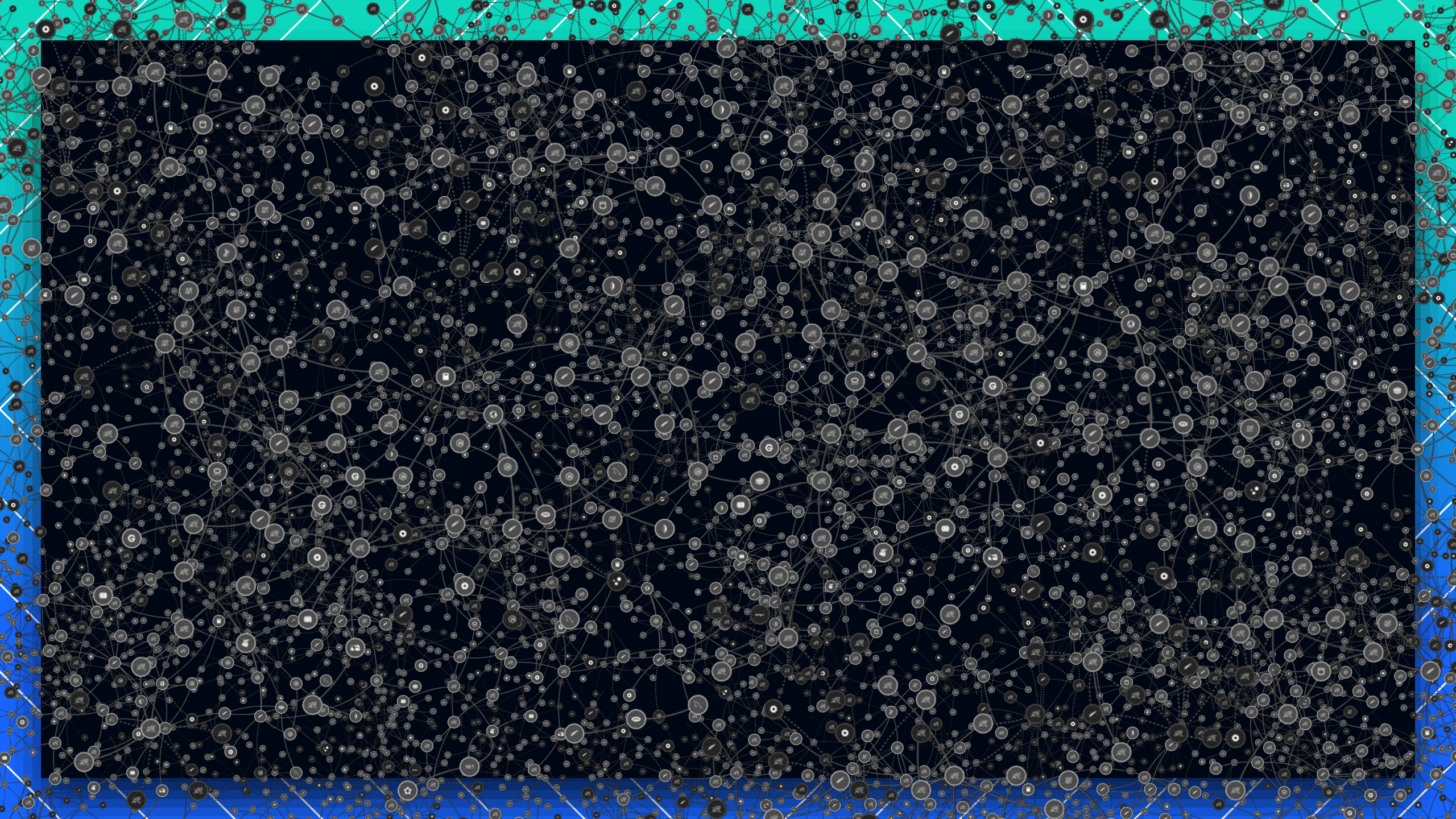


GC Suspension Time %



GC Time



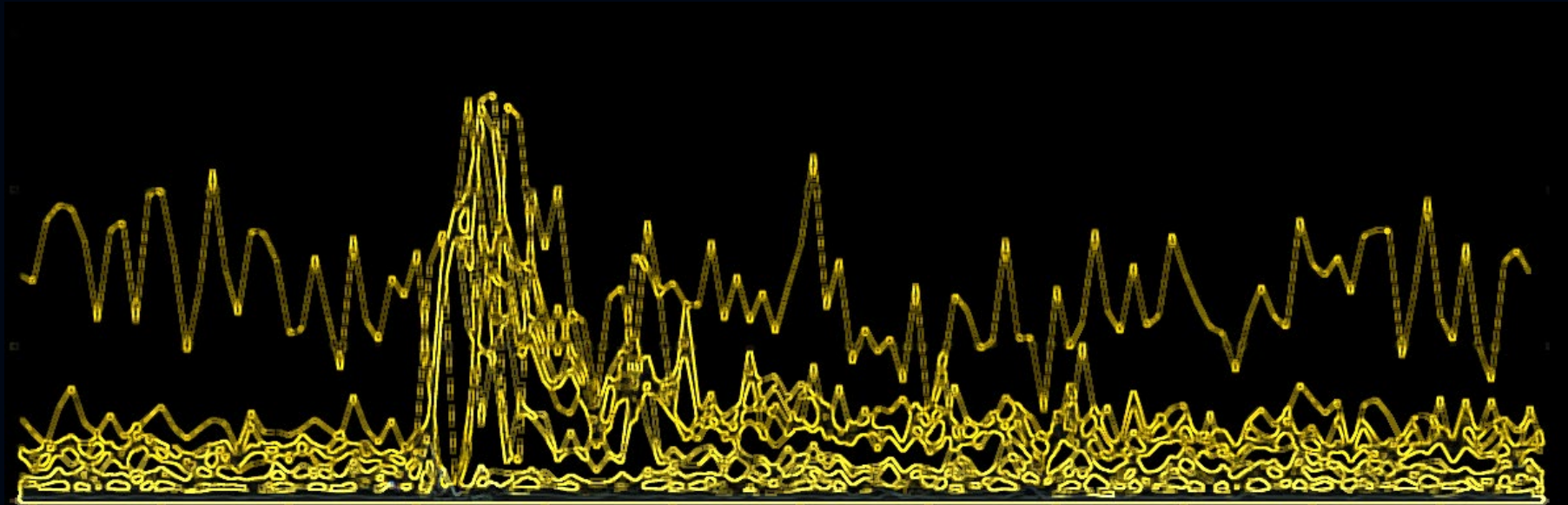


WHY CONTEXT MATTERS?

Let's do a quick example, in which we correlate process CPU metrics.



tomcat



In an environment with more than 100K timeseries, without context the AI will find **hundreds of correlating metrics**.

HIGHER PRECISION BY USING DYNATRACE CONTEXT

Same example but filter out not relevant signals by using Smartscape

context



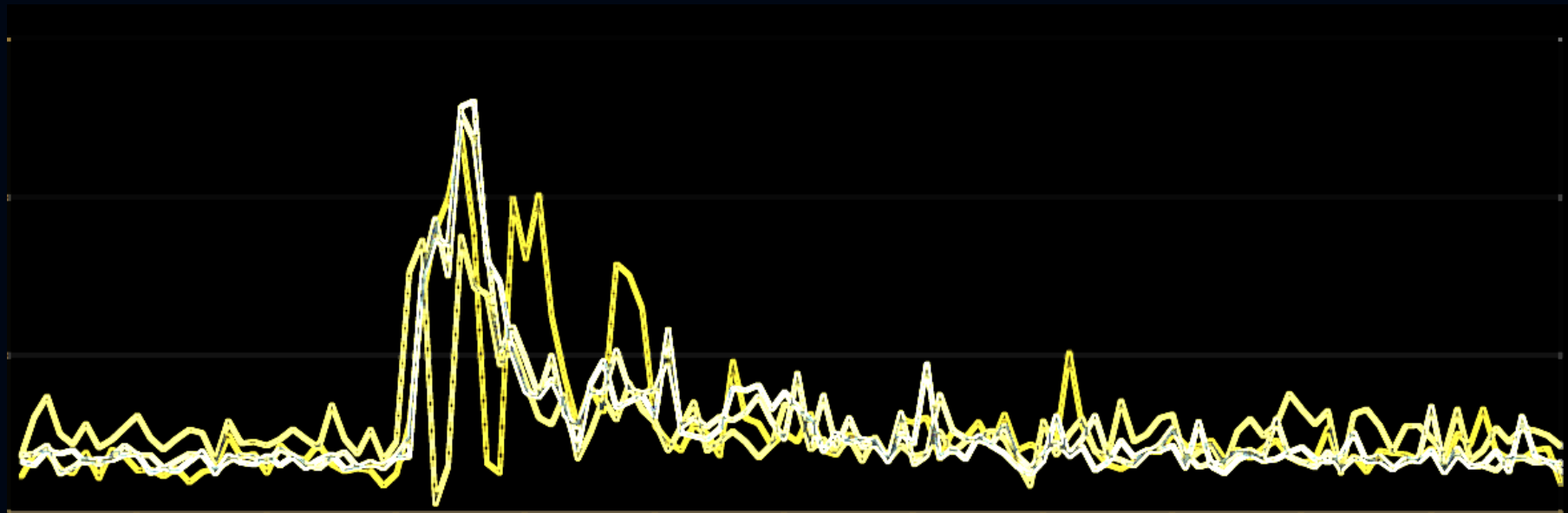
tomcat



pod



workload



The Dynatrace context introduces knowledge to **rule out misleading signals to increase precision.**

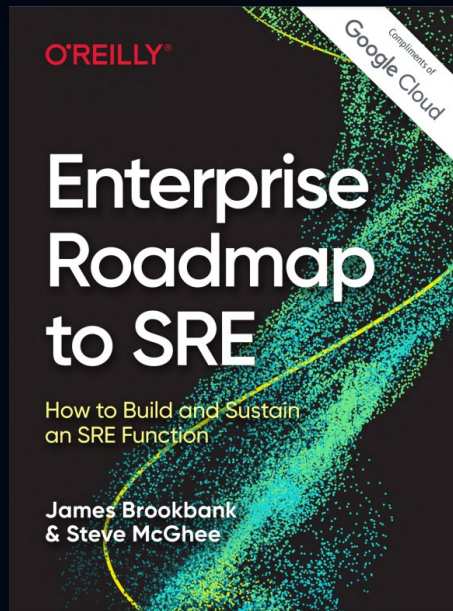
THE POWER OF THE SLO IS DRIVING TRANSFORMATIONS

SLOS ON TOP OF CTOS' LIST FOR 2022 & 2023

Business SLOs are bridging the gap between CEO`s and CTO`s

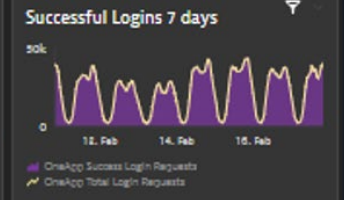
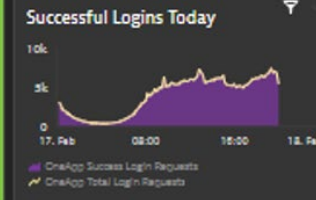
Focus on the user and all else will follow!

Business-driven SLOs with Dynatrace



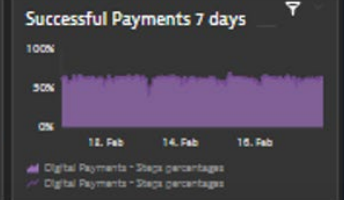
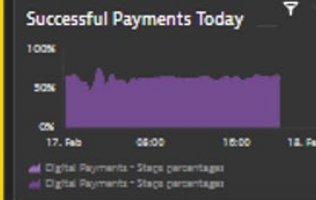
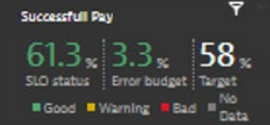
Login

Successful Logins



Payments

Successful Payments

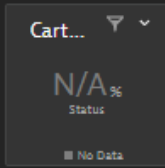
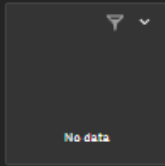
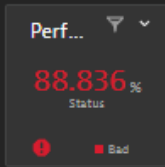


Orders

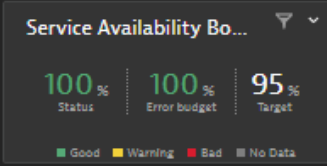
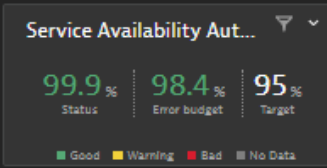
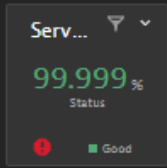
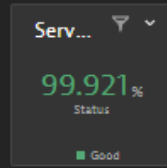
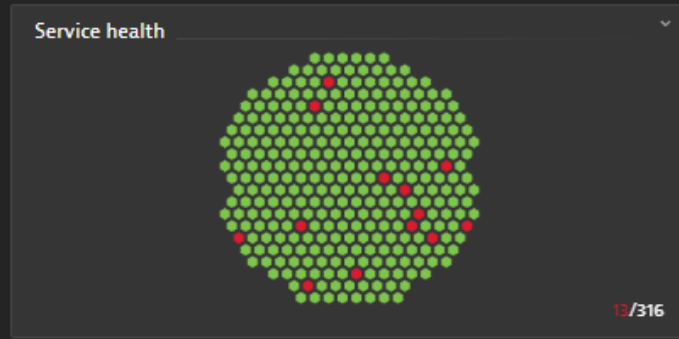
Orders



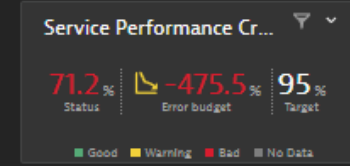
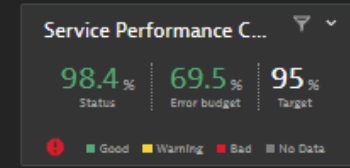
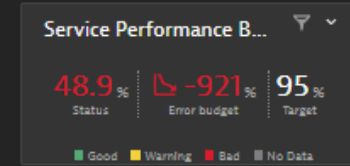
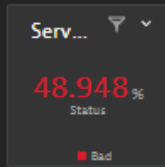
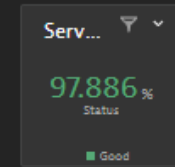
Application performance



Service availability



Service performance



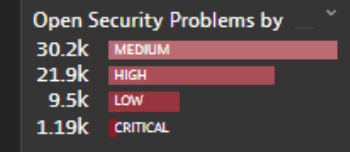
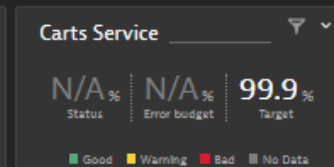
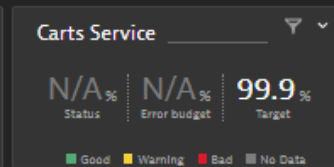
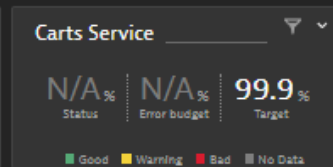
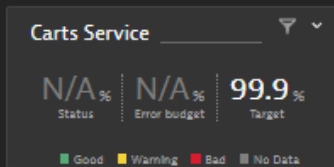
Shopping Cart Service Performance SLO

Live

24h

7 days

30 days



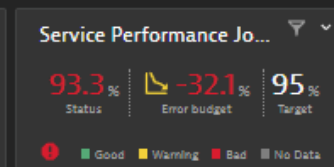
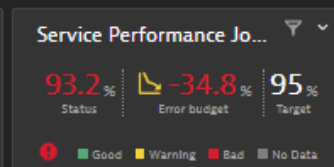
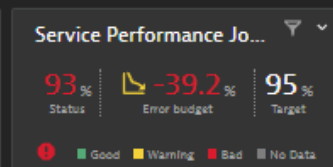
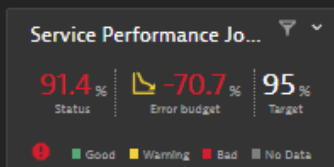
Journey Search Service Availability SLO

Current

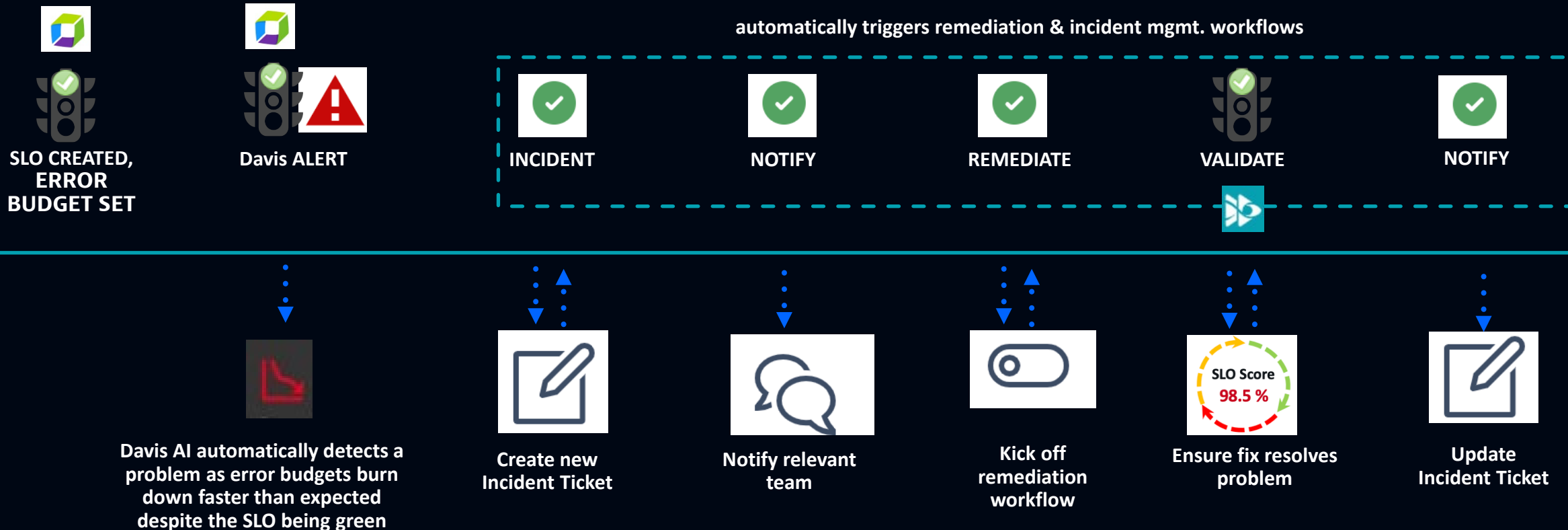
24h

7 days

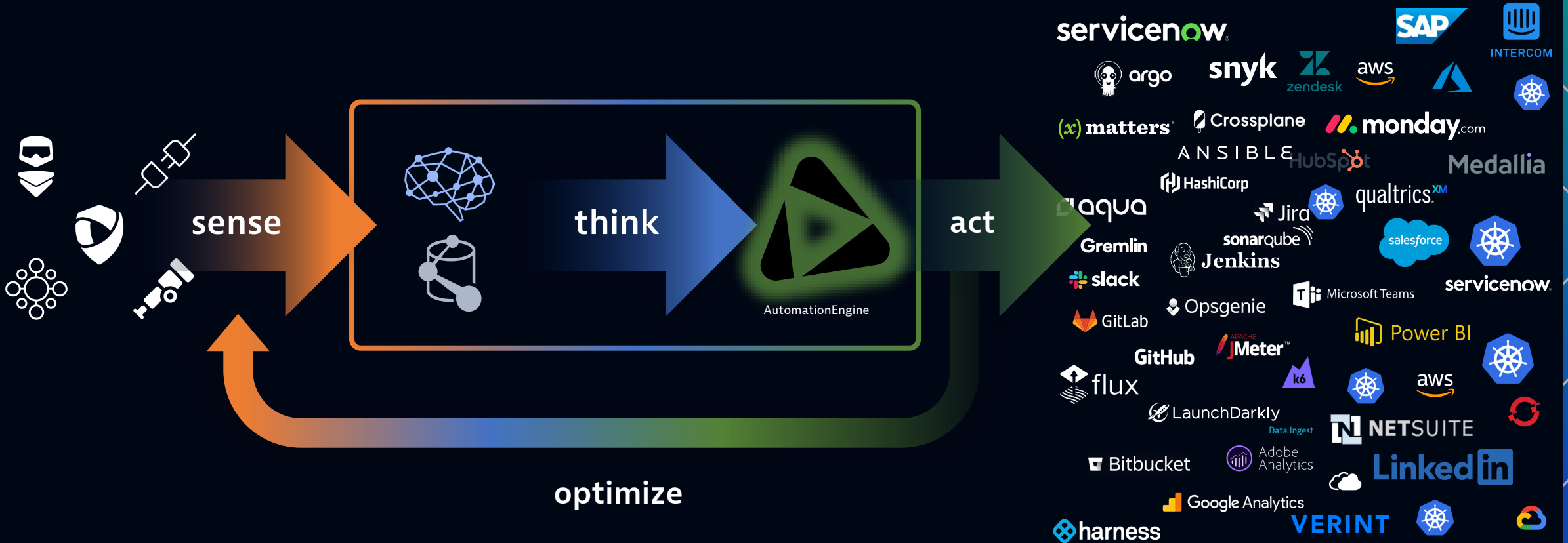
30 days



Proactive SLO Management with Closed Loop Problem Remediation



ANSWER-DRIVEN AUTOMATION





ANALYTICS



Data lakes are beautiful ...

- Operations is bound to realtime data
- Organizing huge amounts of data is fun
- without proper context and model

... Data lakes become garbage heaps

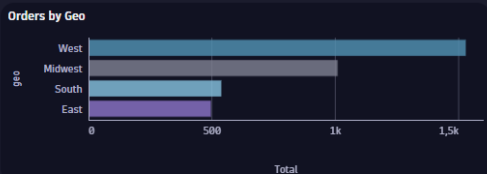
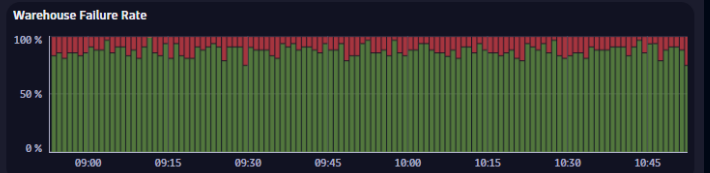
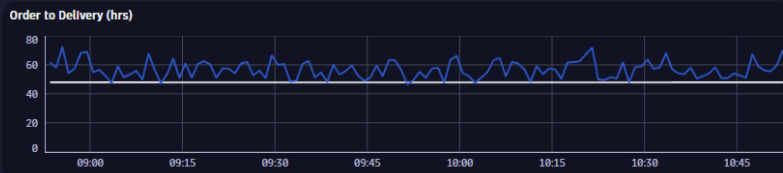
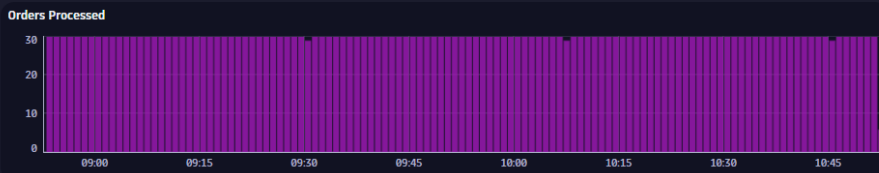
- Expensive to maintain
- Lots of redundancy
- Lack the necessary context

LOG ANALYTICS

Bobbleneers Dashboard Saved 16 seconds ago Share

Orders Processed: **3583** ✓ Orders Sales: **\$35651.0** ✓ Net Promoter Score (NPS): **81.0** ✓

Order to Warehouse: **0.46 hrs** ✓ Warehouse to Shipment: **33.28 hrs** ✗ Shipped to Delivered: **22.95 hrs** ✓ Order to Delivery: **56.69 hrs** ✗ Warehouse Failure Rate: **12.44 %** ✗

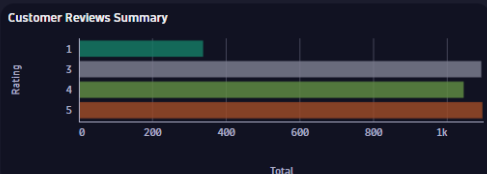


Top Bobbleheads

Bobblehead	Total
Francisco Merino Menares	69
Raymundo Nunez Pena	42
Troy S. Igney	42
Brian J. Wilson	41
Gabriel Marques	41
Brendan Steiner	40
Joshua Wood	40

Warehouse

Warehouse
Boston
Californ
Florida
Michiga
Minnes
Nevada
Oregon



1 Star Reviews

Review
Confidence totally shaken in an otherwise outstanding company. Their new shipping servic...
Wow, way to ruin my week. I told everyone at work I'd bring in Brendan Steiner to show the...
LightningBobble, more like Lame-o-Bobble. Took forever to arrive.
I lightningBobble, more like Lame-o-Bobble. Took forever to arrive.

Warehouse Status

Query Visualize

```
1 fetch logs
2 | filter company == "Bobbleneers" and (serviceevent == "warehousedistribution" or serviceevent == "shipped" )
3 | fieldsAdd errors = if(status == "ERROR", "1", else:"0")
4 | fieldsAdd warehouse_distribution_ts = if(serviceevent == "warehousedistribution", toTimestamp(event))
5 | fieldsAdd shipped_ts = if(serviceevent == "shipped", toTimestamp(event))
6 | summarize {warehouse_events = countIf(serviceevent == "warehousedistribution"),warehouse_errors = countIf(errors
7 | fieldsAdd time_from_warehouse_distribution_to_shipped = toLong((first_shipped_ts - first_warehouse_distribution_ts
8 | summarize {Orders = toDouble(countDistinct(orderid)), ErrorRate=(todouble(sum(warehouse_errors))/todouble(sum(ware
9 | fieldsAdd Penalty = if(Hours < 24, 0, else: ((Hours)-24) *5000)
10 | fieldsAdd SLO = if(Hours <= 24, "✓", else:"✗")
11 | fields Warehouse = warehouse,SLO,Penalty = concat("$", toLong(Penalty)),Orders,TIWD = concat(toLong(Hours)," ", "hr
12
```

Run query Discard changes

7 records Executed at: 20/11/2023, 10:52:39, Timeframe: 08:52:38 - 10:52:38 ⓘ

BUSINESS EVENTS ANALYTICS

Trade Volume

241

258

Trade Dollar Volume

64,8M

26,8M

Trade Performance

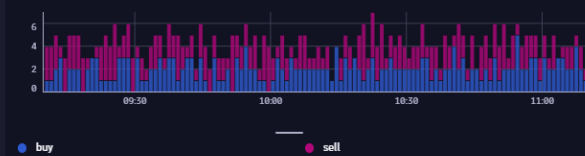
269,1k

103,8k

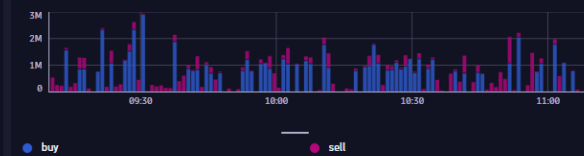
12.82 ms

12.38 ms

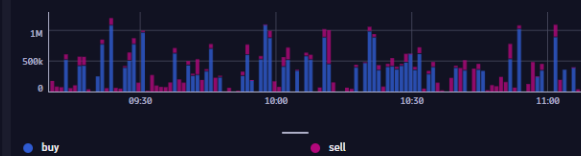
Trades by TradeType



Volume by TradeType (Total)



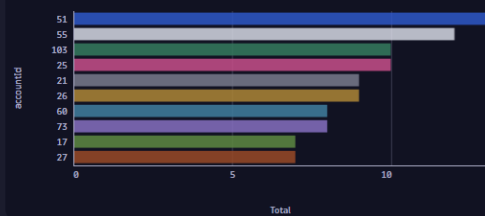
Volume by TradeType (Avg)



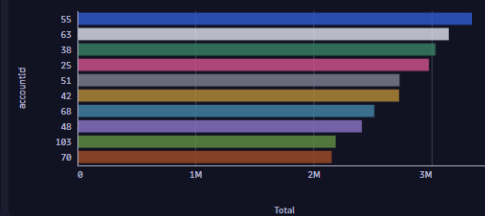
Latency (Avg)



Top Trade Volume by Account ID



Top Trade Dollar Volume by Account ID



Top In



Time from deposit to first Sell

Query Visualize

Query

```

1  fetch bizevents
2  | filter event.provider == "www.easytrade.com"
3  | sort timestamp, direction:"descending"
4  | filter event.type == "com.easytrade.long-sell" OR event.type == "com.easytrade.deposit"
5  | fieldsAdd deposit_ts = if(event.type == "com.easytrade.deposit", toLong(timestamp))
6  | fieldsAdd sell_asset_ts = if(event.type == "com.easytrade.long-sell", toLong(timestamp))
7  | summarize first_deposit_ts = takeFirst(deposit_ts), first_sell_asset_ts = takeFirst(sell_asset_ts), by:{accountId}
8  | fieldsAdd time_from_deposit_to_first_sell = (first_sell_asset_ts - first_deposit_ts)/(1000000000.0)
9  | filter time_from_deposit_to_first_sell > 5
10 | limit 5
11 | fields AccountID = accountId, time_from_deposit_to_first_sell, alias:{Seconds}
12 | sort Seconds desc
13
14
15
    
```

Run query

Discard changes

5 records Executed at: 20/11/2023, 11:09:40, Timeframe: 09:09:39 - 11:09:39

Most recent trades

Time	AccountID	TradeType	Amount	Price	TotalVolume	trace_id
20/11/2023, 11:09:24	37,000	buy	860,000	0,304	261,740	024850827e4e29c8b9532347cd7b8fca
20/11/2023, 11:09:21	28,000	sell	16,060,000	4,058	65,167,180	1062bd5e485f196f5e877d3c0eeb6403
20/11/2023, 11:08:57	54,000	sell	26,000	83,905	2181,540	0c41d42845dd14aa2bf5bb235f66dbb9
20/11/2023, 11:08:42	14,000	buy	265,000	4,958	1313,900	7c0789e663ca8999af00b033def5470

Problems

SECURITY ANALYTICS

AppSec - Multiple vulnerabilities exposure Saved 10 minutes ago Share

ScanGBLimit: 1000 Status: ANY RiskLevel: ANY ComponentName: ANY ManagementZone: ANY StackLevel: ANY Type: ANY

Risk

Risk state

🔥 **Vulnerable**

Exploitation Risk Score (0-100)

92

Exploitation Risk Score (ERS) is a dynamic, weighted score that leverages the Davis Security Score (DSS) of each vulnerability and the actual number of vulnerabilities per risk level. ERS represents the likelihood of a potential attacker exploiting open vulnerabilities from the filtered scope. How ERS is calculated:

- DSS average and vulnerability counts are calculated per risk level (critical, high, etc).
- A weighted function is applied per risk level to adjust the averages to the respective vulnerability counts.
- All scores are aggregated in an iterative manner starting with the highest risk level to derive a single score out of 100.

Note: Higher scores represent higher risk. Scores above 90 involve critical vulnerabilities.

Vulnerabilities

Most severe open vulnerabilities

! **Critical: 2**

Open

404

Vulnerability count by risk level

Risk Level	Percentage
Critical	50%
High	27%
Medium	22%

Risk analysis

Query Visualize

```
1 fetch events
2 | filter event.provider=="Dynatrace"
3 | filter event.type=="VULNERABILITY_STATE_REPORT_EVENT"
4 | filter event.level=="ENTITY"
5 | filter ($Status=="ANY" OR ($Status==vulnerability.resolution.status AND vulnerability.mute.status=="NOT_MUTED" ) OR (vulnerability
6 | filter in("ANY", $RiskLevel) OR in(vulnerability.risk.level,$RiskLevel)
7 | filter "ANY"==$StackLevel OR vulnerability.stack==$StackLevel
8 | filter "ANY"==$Type OR vulnerability.type==$Type
9 | filter $ManagementZone=="ANY" OR in($ManagementZone,affected_entity.management_zones.names)
10 | filter $ComponentName=="ANY" OR contains(affected_entity.vulnerable_component.name,$ComponentName)
11 // filter only the latest snapshot data
12 | fieldsAdd matcher="match"
13 | lookup [
14 fetch events
15 | filter event.provider=="Dynatrace"
16 | filter event.type=="VULNERABILITY_STATE_REPORT_EVENT"
17 | filter event.level=="ENTITY"
18 | filter ($Status=="ANY" OR ($Status==vulnerability.resolution.status AND vulnerability.mute.status=="NOT_MUTED" ) OR (vulnerability
19 | filter in("ANY", $RiskLevel) OR in(vulnerability.risk.level,$RiskLevel)
20 | filter "ANY"==$StackLevel OR vulnerability.stack==$StackLevel
21 | filter "ANY"==$Type OR vulnerability.type==$Type
22 | filter $ManagementZone=="ANY" OR in($ManagementZone,affected_entity.management_zones.names)
23 | filter $ComponentName=="ANY" OR contains(affected_entity.vulnerable_component.name,$ComponentName)
24 | fields
25 maxTimestamp=timestamp,
26 matcher="match"
27 | limit 1
28 ], sourceField:matcher, lookupField:matcher
29 | filter timestamp==lookup.maxTimestamp
30 //end of filtering for the latest snapshot
31 | summarize
32 `Status`=takeFirst(vulnerability.resolution.status),
33 `Muted`=takeFirst(vulnerability.mute.status=="MUTED"),
34 by: {vulnerability.display_id,alias:'Problem id'}
35 | summarize
36 `Muted`=countIf(Muted==TRUE),
37 `Resolved`=countIf(Status!="OPEN" AND Muted==FALSE),
38 `Open`=countIf(Status=="OPEN" AND Muted==FALSE)
39 | fields `Risk state`=if(toLong(Open)>0,concat("🔥", " ", "Vulnerable"),else;if(Muted<1,"Not at risk", else;if(Resolved<1,concat("⚠️", " ", "At risk")))
```

Run query

DATAMODEL ANALYTICS

Problems

startTime	status	title	impactLevel	impactedEntities	rootCauseEntity	duration
20/11/2023, 10:41:40	CLOSED	Memory saturation	INFRASTRUCTURE	EC2AMAZ-BFHGR3B		13,000
20/11/2023, 10:02:12	CLOSED	Browser monitor local outage	APPLICATION	angular easytravel booking, angular easytravel abandon		61,000
20/11/2023, 9:04	CLOSED	Failure rate increase	SERVICES	DTMD on CICS	DTMD on CICS	21,000
20/11/2023, 8:03	CLOSED	User action duration degradation	APPLICATION	www.easytravel.com	CheckDestination	25,000
20/11/2023, 7:03	CLOSED	Browser monitor global outage	APPLICATION	OpenTelemetry	Third Party Currency Converter	64,000

Deployments Overview

stage	status	Deployments
production	✗	3

Hosts oversized calculation CPU (last 7 days)

Server	cpu max	type	oversized
credhub/8ea6cb53-4f1b-4aa4-a4c2-4611038f30e1	1,675	r5.large	✗
nfs_server/4abe199b-f042-4155-999e-4163049ca9f7	1,765	m5.large	✗
backup_restore/bbaf1efb-862e-459e-b755-cdd6180e964f	2,629	t3.micro	✗
mysql_monitor/a833414a-7d35-4e20-b5ba-d39028d5a978	2,743	t3.micro	✗
nats/47813f0e-663b-457c-b606-2025bd89a207	3,413	t3.micro	✗
diego_brain/088e39f8-d3df-4fe8-904f-315451a999ff	4,405	t3.small	✗
mysql_monitor/d5074e6a-4100-40f7-a37b-0e1e4164d5ed	4,550	t3.micro	✗

Contextualized Metrics

nameSpace	bucketName	ucanaMemAvg	ucanaCpuAvg	ucanaMemAlert	ucanaCpuAlert
-----------	------------	-------------	-------------	---------------	---------------

Hosts oversized calculation CPU (last 7 days)

Query Visualize

Query

```
1 //fetch metrics, from:now()-5m | filter metric.key == "dt.host.cpu.user" | filter dt.entity.host == "HOST-54A9A1B91AD09994"
2 fetch dt.entity.host | fieldsAdd ipAddress | fieldsAdd entity=id | fieldsAdd entity.name
3 | fields ip=arrayFirst(ipAddress), entity, entity.name
4 | lookup [fetch dt.entity.ec2_instance| fieldsAdd localIp,awsInstanceType],sourceField:ip, lookupField:localIp
5 | filter isNotNull(lookup.id)
6 | fields ip,entity, entity.name,type=lookup.awsInstanceType
7 | lookup [timeseries from:now()-7d, to:now(), cpu=max(dt.host.cpu.usage),by:{dt.entity.host} | fields cpu_max=arrayAvg(cpu),dt.entity
8 | fieldsRename `cpu max`=`lookup.cpu_max`
9 | fields Server=entity.name, `cpu max`,type
10 | filter isNotNull(`cpu max`)
11 | fieldsAdd oversized=if(`cpu max` > 50, "✔", else:"✗")
12 | sort `cpu max`,direction:"ascending"
```

▶ Run query

↶ Discard changes

31 records Executed at: 20/11/2023, 11:24:18, Timeframe: 13/11/2023, 11:00:00 - 20/11/2023, 12:00:00 ⓘ



IMMERSE

📍 MADRID

📅 21.11.23