End to end Observability, Automation and DevOps

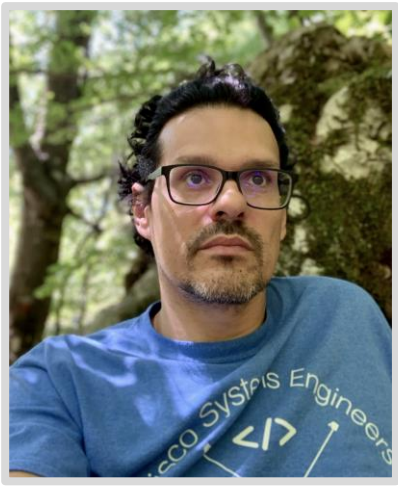
with Splunk AppDynamics and Observability Cloud

Luca Relandini - Principal Architect
Stefano Gioia - Technical Solution Architect

BRKOBS-2284

Some telemetry to... observe us

Metrics: 59yo, 85Kg, 183cm
Events: Joined Cisco in 2008
Married in 1996, 2 daughters, one dog
Won lots of Padel matches
Logs: Splunk Observability Strategists
Traces: Programming -> Architecture
DataCenter -> Cloud
Observability -> Automation



Metrics: 50yo, 94Kg, 185cm
Events: Joined Cisco in 2017
Married in 2009, 2 children, one cat
BBQ Judge
Logs: Part of Cloud & AI EMEA Team
Trace: DataCenter -> Cloud
Observability -> Automation

Webex App

Questions?

Use the Webex app to chat with the speaker after the session

How

1. Find this session in the Cisco Events mobile app
2. Click “Join the Discussion”
3. Install the Webex app or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until February 28, 2025.

By the end of this session, you will believe...

- Observability helps both IT operations and developers (DevOps)
- Automation increases efficiency and effectiveness

Agenda

- OpenTelemetry
- Splunk Appdynamics and Splunk Observability Cloud
- Automation – Observability as Code
- CI/CD Pipeline – Building applications and observability together

Collecting Open Telemetry with OTEL collectors and the AppDynamics Hybrid Agent

cisco Live!

slido

Please download and install the Slido app on all computers you use

How familiar are you with OpenTelemetry?

ⓘ Start presenting to display the poll results on this slide.

Why not standardize this?

Monitor (Detect)

Explore (Troubleshoot)

Discover (Root Cause)

Traditional to Hybrid:
The Petclinic Story

cisco Live!

Traditional N-tier applications

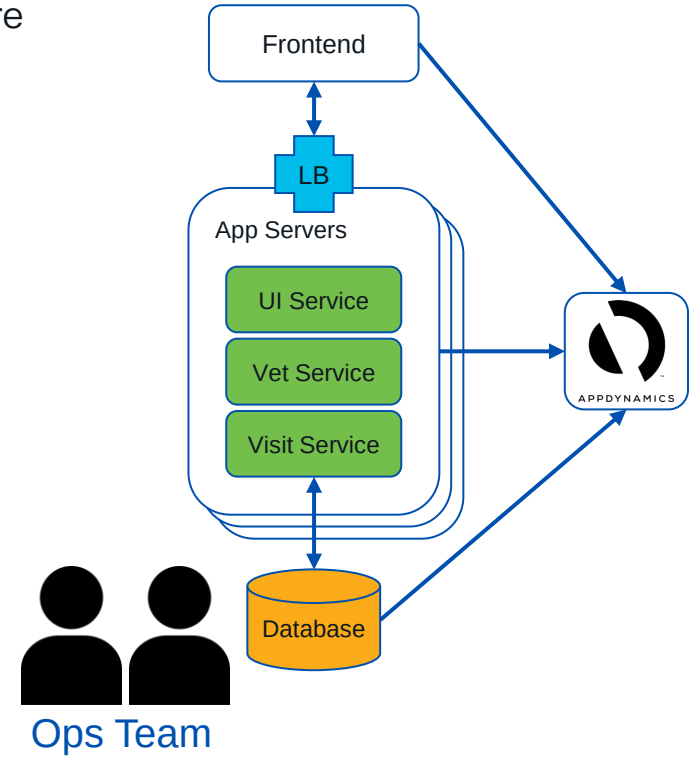
ACME Vet Co. has, until now, been using traditional software development and operated by their Operations team

Petclinic is one of their traditional N-tier applications
- Hosted on-premises

AppD's agent-based approach is simple
- Real User Monitoring agent for the frontend
- Application Agent for the backend services
- DB Agent for the Database

Serves the need of the Operations team
- User Experience
- Line of code visibility
- Application server and infrastructure metrics
- App to Database correlation

Simple straightforward deployment



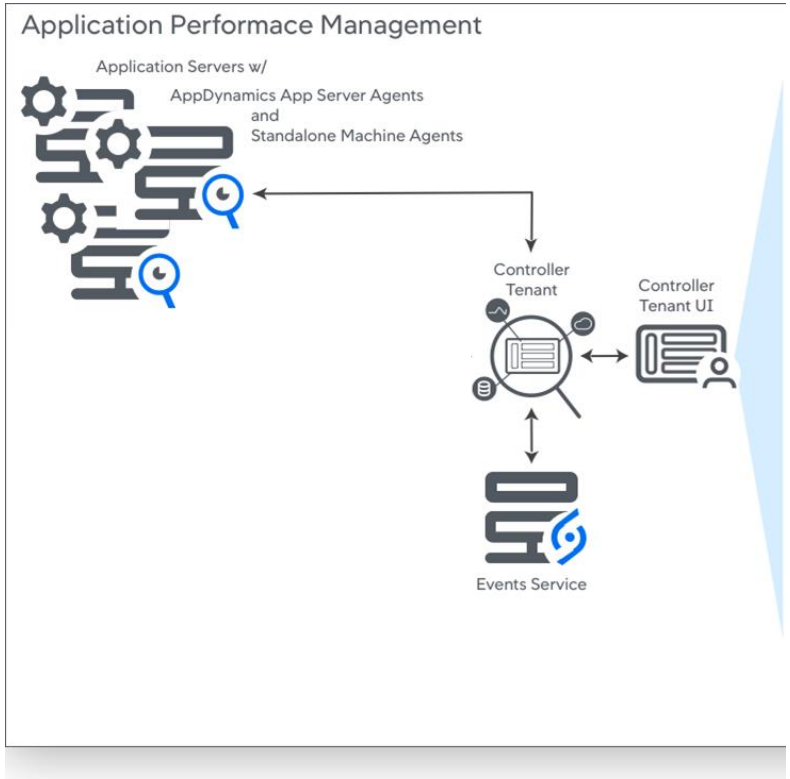
AppDynamics Agent Deployment

Example – add AppD parameters to the start up of the application

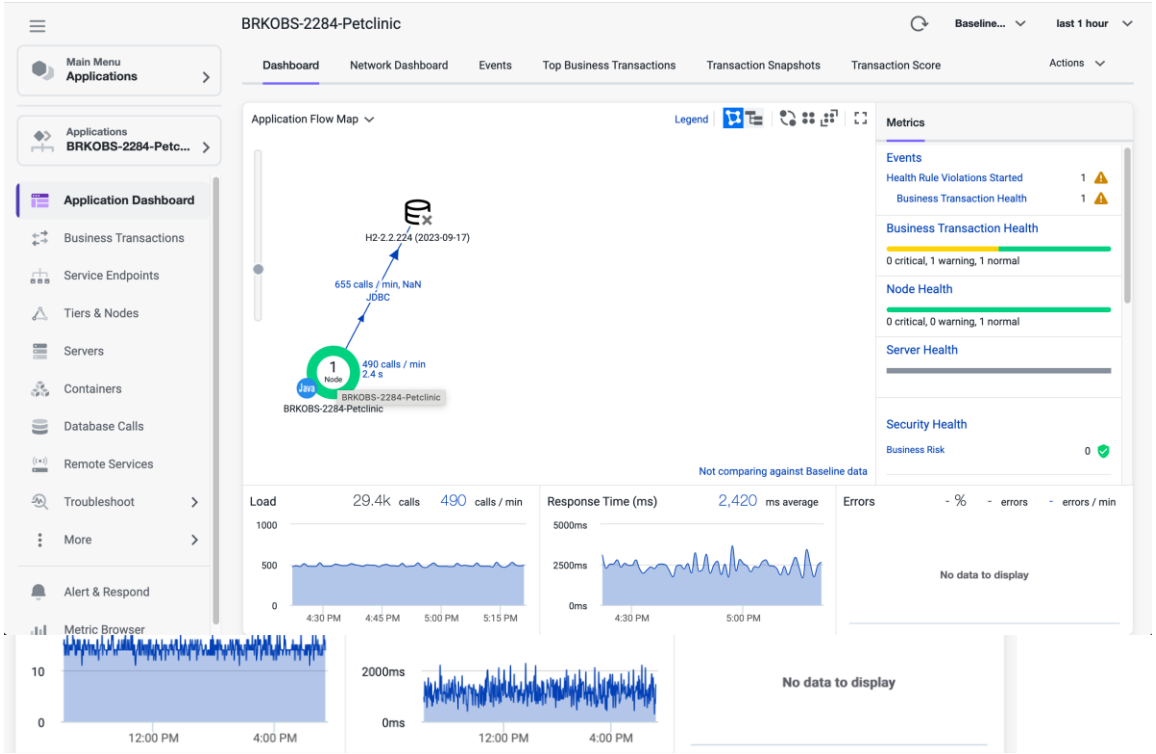
```
java -jar app.jar
-javaagent:/opt/appdynamics-java/javaagent.jar
-Dappdynamics.agent.accountAccessKey=<access-key>
-Dappdynamics.agent.applicationName=<application-name>
-Dappdynamics.agent.tierName=<tier-name>
…
```

Example – DB agent standalone: java service start

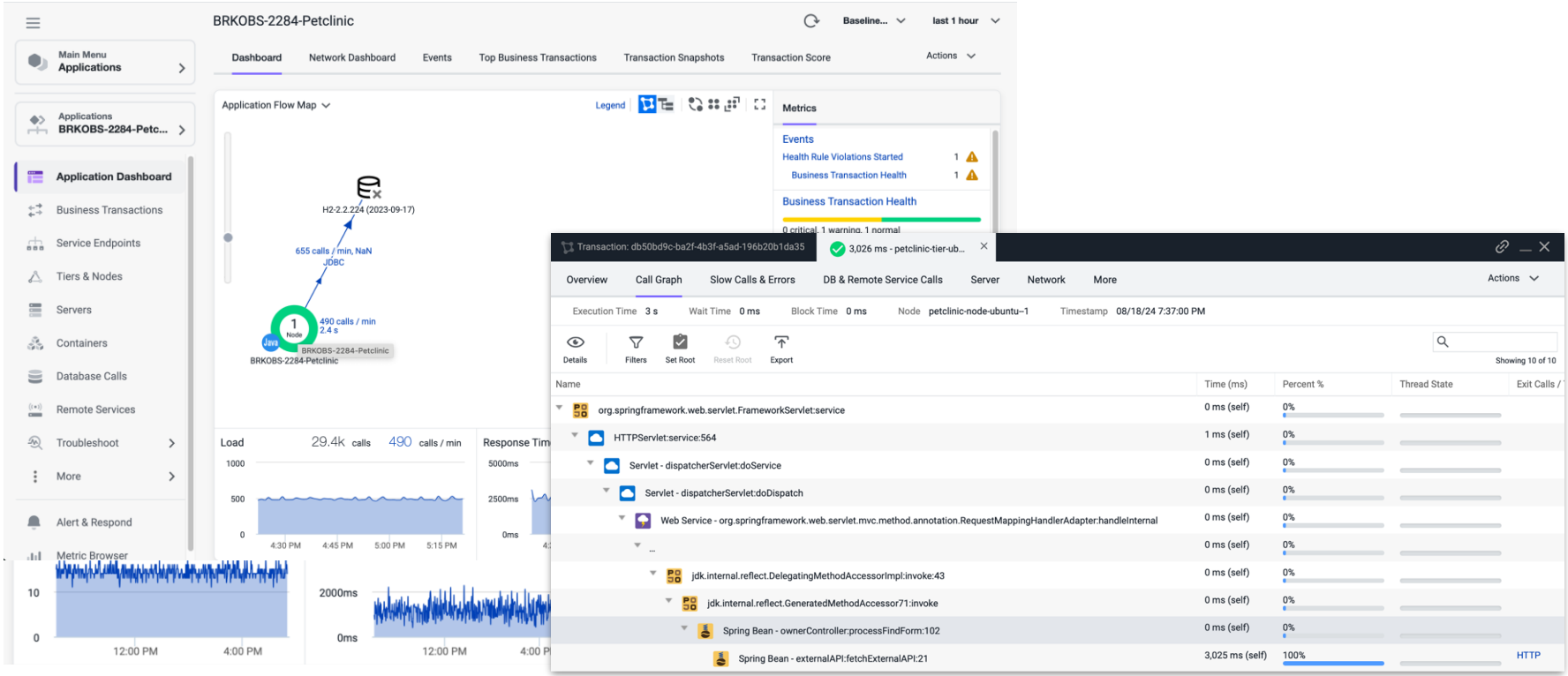
```
./start-dbagent
-Xms<min_heap_size> -Xmx<max_heap_size>
-Dappdynamics.agent.accountAccessKey=<access-key>
-Dappdynamics.controller.hostname=<controller-hostname>
-Dappdynamics.controller.port=<controller-port>
…
```



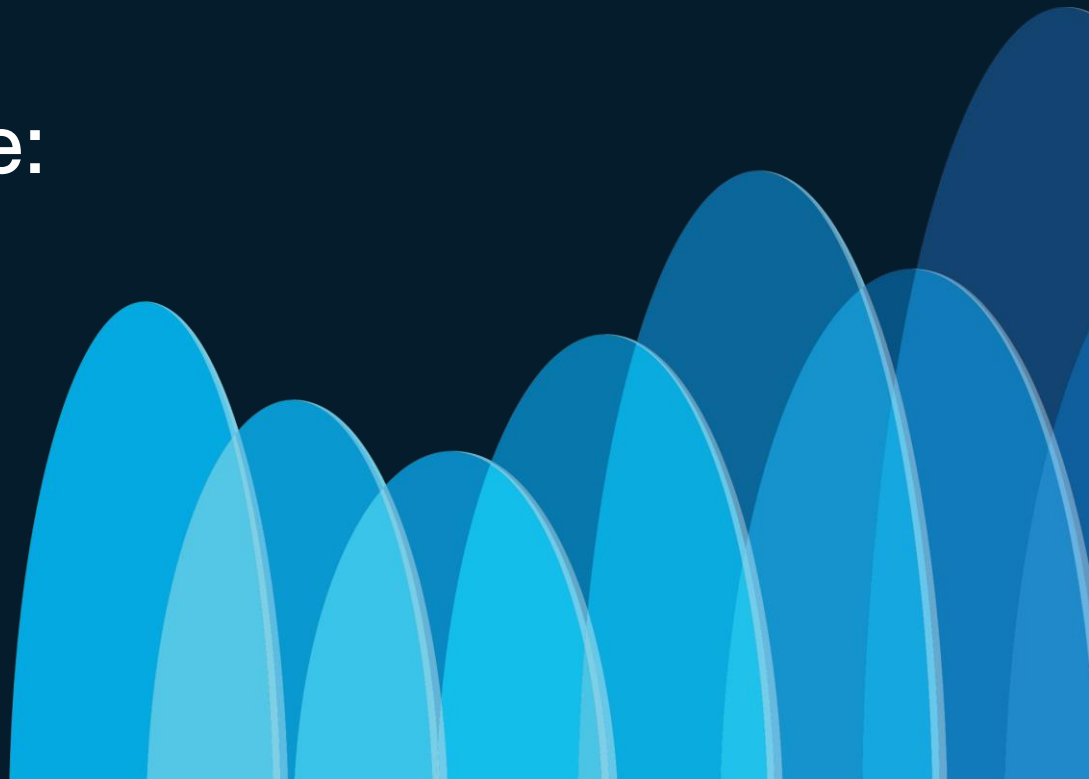
Petclinic + AppDynamics Java Agent



Petclinic + AppDynamics Java Agent



Shift To Cloud Native: Microservices and Standardization



cisco Live!

DevOps and Microservices

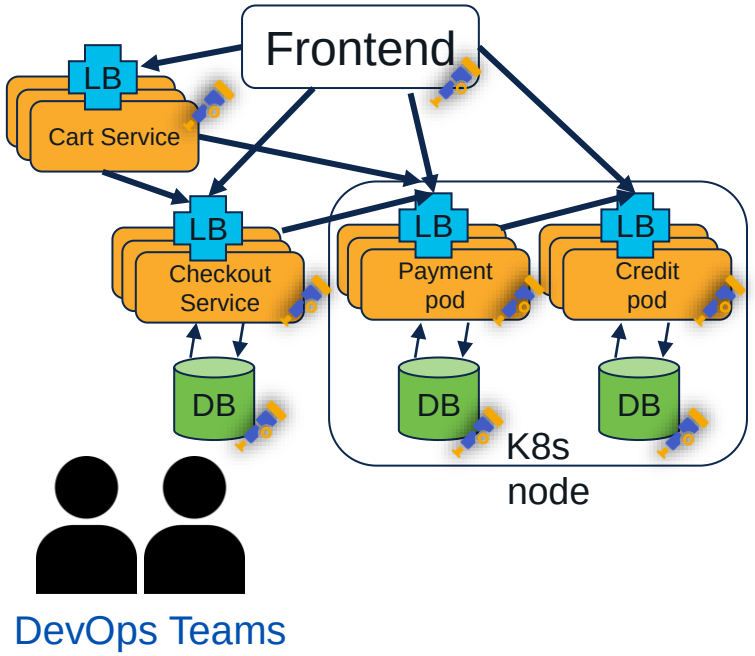
As **ACME Vet Co**. evolved it started developing microservices for its new Doggy Daycare App. These services lived in the cloud and followed a DevOps support model.

Why in cloud?

- Cost
- Scalability
- Development speed

Why Splunk Observability & OpenTelemetry?

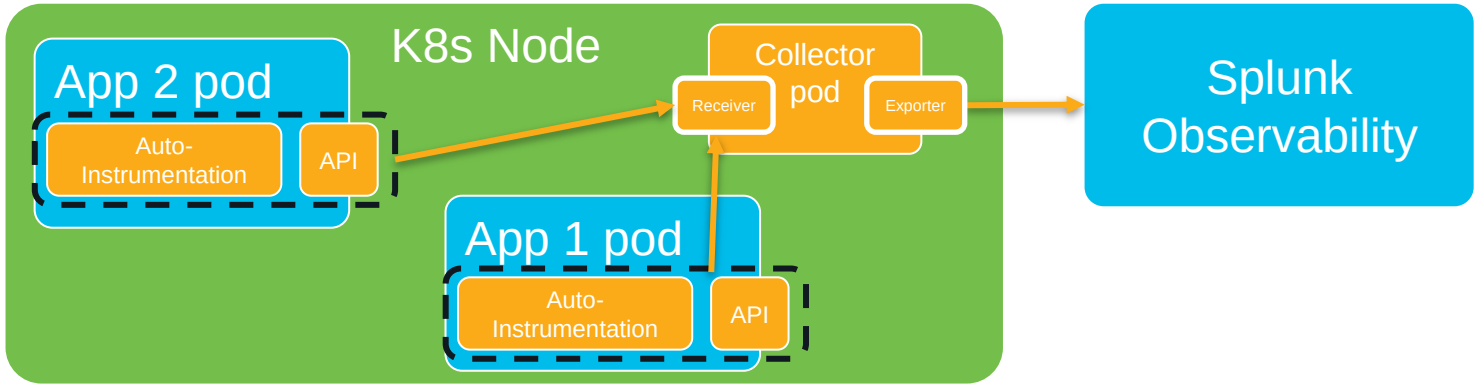
- Cloud Native Infrastructure Monitoring & APM
  What is broken? (infra / service availability)
  Where? (traffic shapes, status codes, etc)
- "Zero config" OpenTelemetry APM instrumentation
- Scripted OTel Collector deployment & installation
- No-Sample APM engine
- Standardized Telemetry



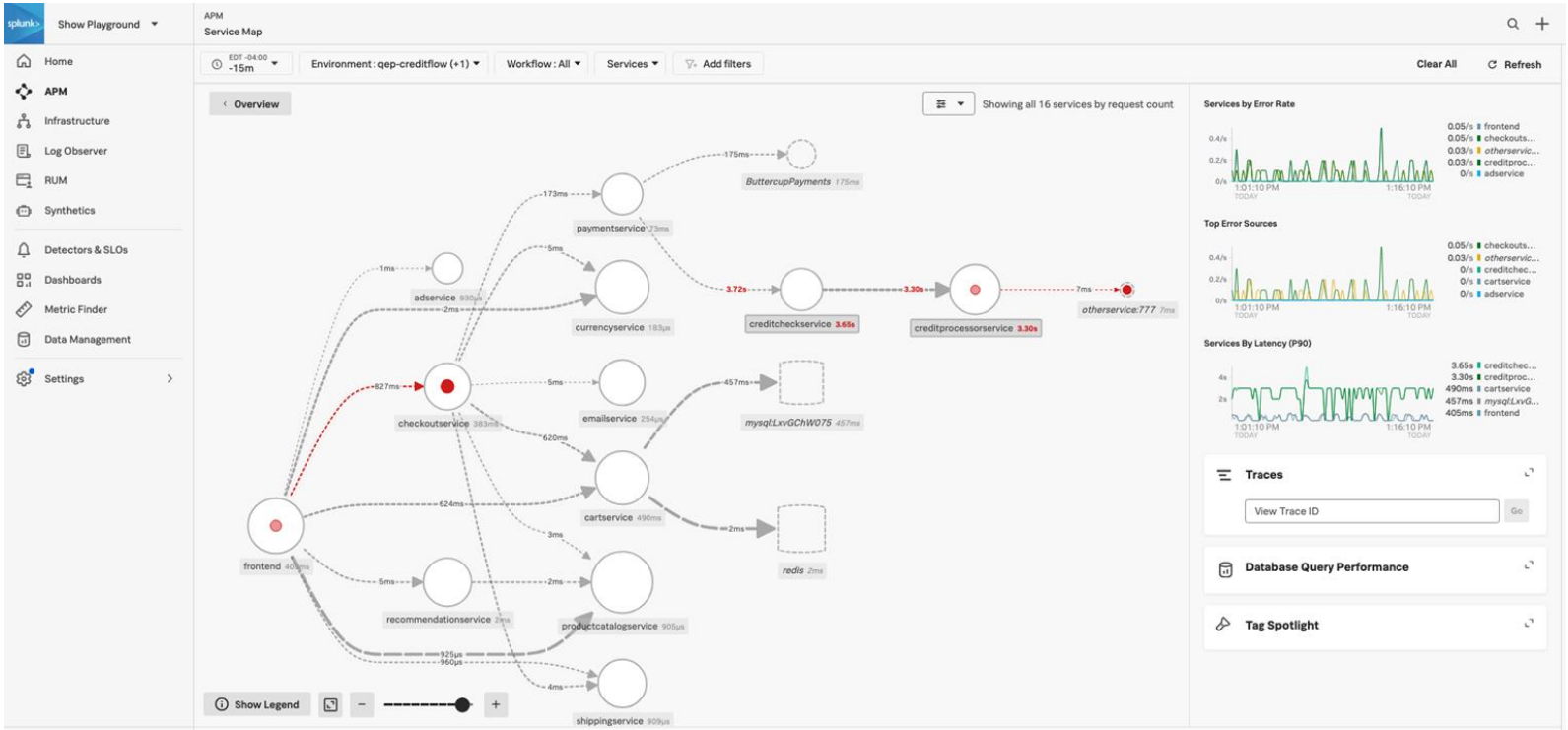
Installing an OpenTelemetry Collector

Example: Kubernetes install

```
helm repo add splunk-otel-collector-chart https://signalfx.github.io/splunk-otel-collector-chart
helm install my-splunk-otel-collector \
    --set="splunkObservability.realm=us0,splunkObservability.accessToken=xxxxxx,clusterName=my-cluster" \
    splunk-otel-collector-chart/splunk-otel-collector
```



Apps and Services in Splunk Observability Cloud



A single pane of glass for Traditional Apps and Microservices

How can Petclinic leverage Doggy Daycare's Payment Flow?

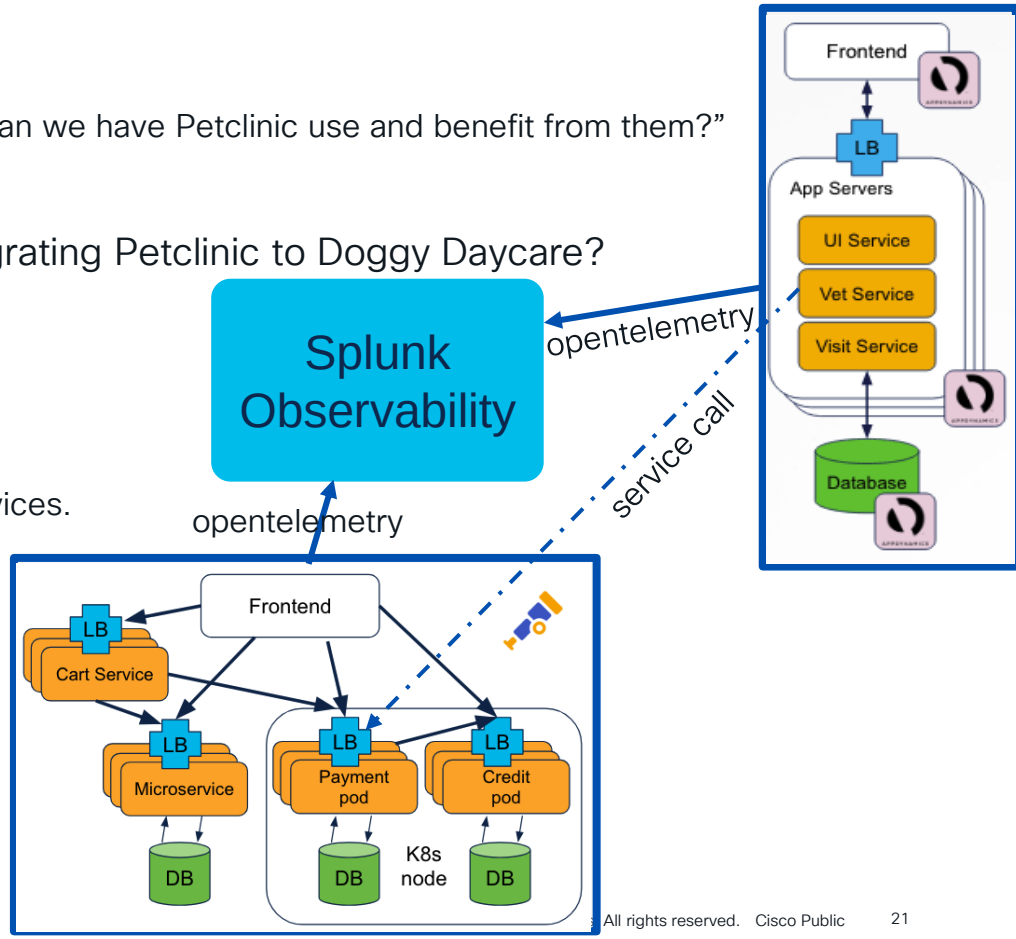
"Now that we have all these great microservices, how can we have Petclinic use and benefit from them?"

What were ACME Vet Co's concerns with integrating Petclinic to Doggy Daycare?

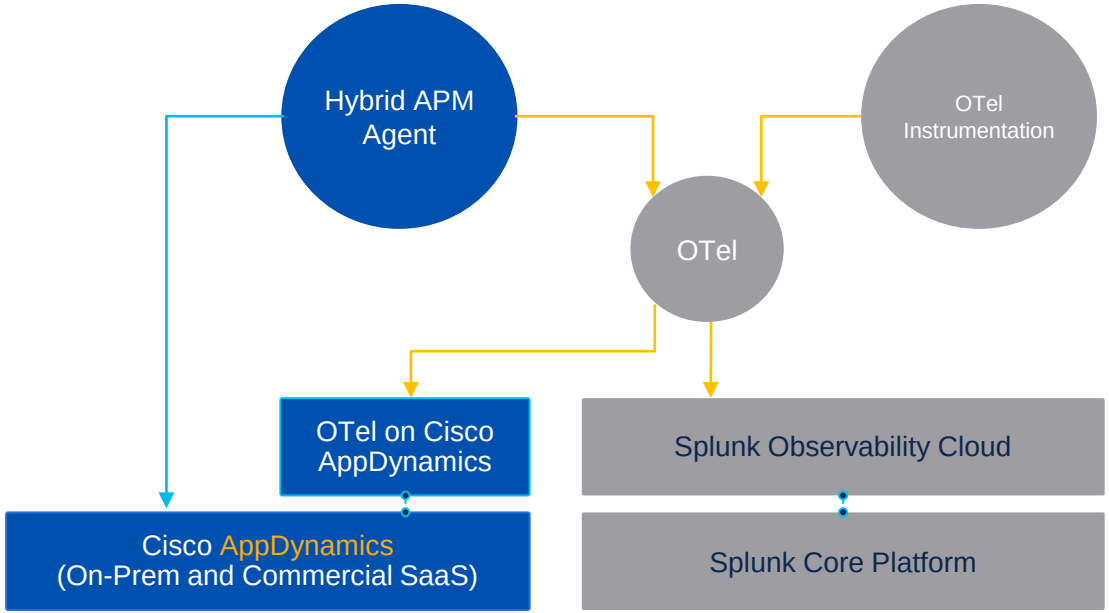
- Want to keep AppD instrumentation on Petclinic
- AppD Hybrid Agent OTel Export
- Minimize disruption to Ops Teams
- Continue to use OpenTelemetry for microservices
- Require visibility between our n-tier and our microservices.
- Maintain context of Petclinic calls to microservices

Solution:

- Standardization (OpenTelemetry)



AppDynamics Hybrid Agent



End-to-end visibility for mixed instrumentation environments

Output OTel for Cloud Native Application Observability and 3rd party consumption

Simplify your path from Cisco AppDynamics to Splunk Observability Cloud

Gain OTel experience without manually re-instrumenting

How to Use OTel with AppDynamics Hybrid Agents

Example – adding OTLP parameters to the regular agent injection

```
java -jar app.jar
-javaagent:/opt/appdynamics-java/javaagent.jar
-Dappdynamics.agent.accountAccessKey=<access-key>
-Dappdynamics.agent.applicationName=<application-name>
-Dappdynamics.agent.tierName=<tier-name>
…
-Dappdynamics.opentelemetry.enabled=true
-Dotel.resource.attributes=service.name=<tier-name>,
deployment.environment=<application-name>
-Dotel.traces.exporter=otlp
-Dotel.propagators=tracecontext,baggage,b3multi
-Dotel.exporter.otlp.traces.endpoint=http://otel-collector:4317
```

Regular parameters

OTel parameters

Our Observability Journey

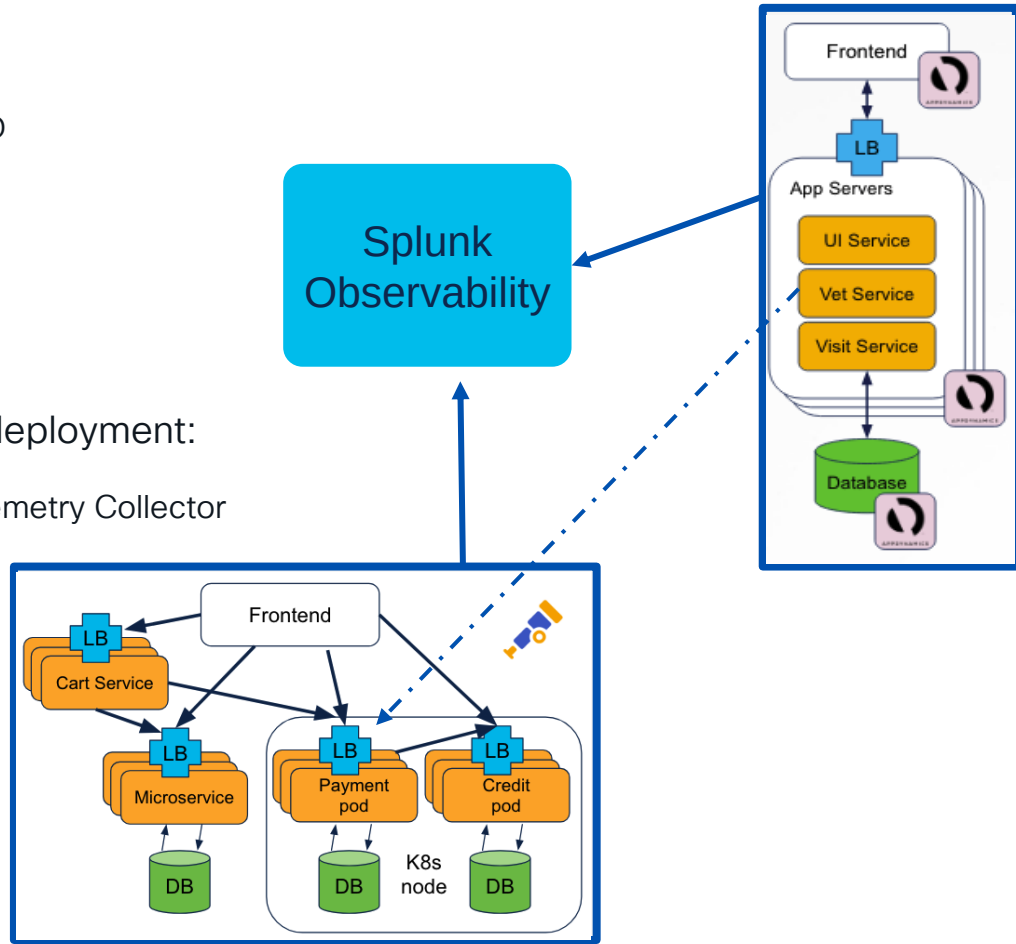
Where we were - Petclinic:
- Monolithic n-tier service and DBs monitored with AppD
- Traditional Ops concerns

Where we were heading - Doggy Daycare:
- Cloud native microservices and DB offerings
- DevOps culture (You build it. You own it.)
- Complex environments (SaaS, K8s, etc)

Unified context and observability in our hybrid deployment:

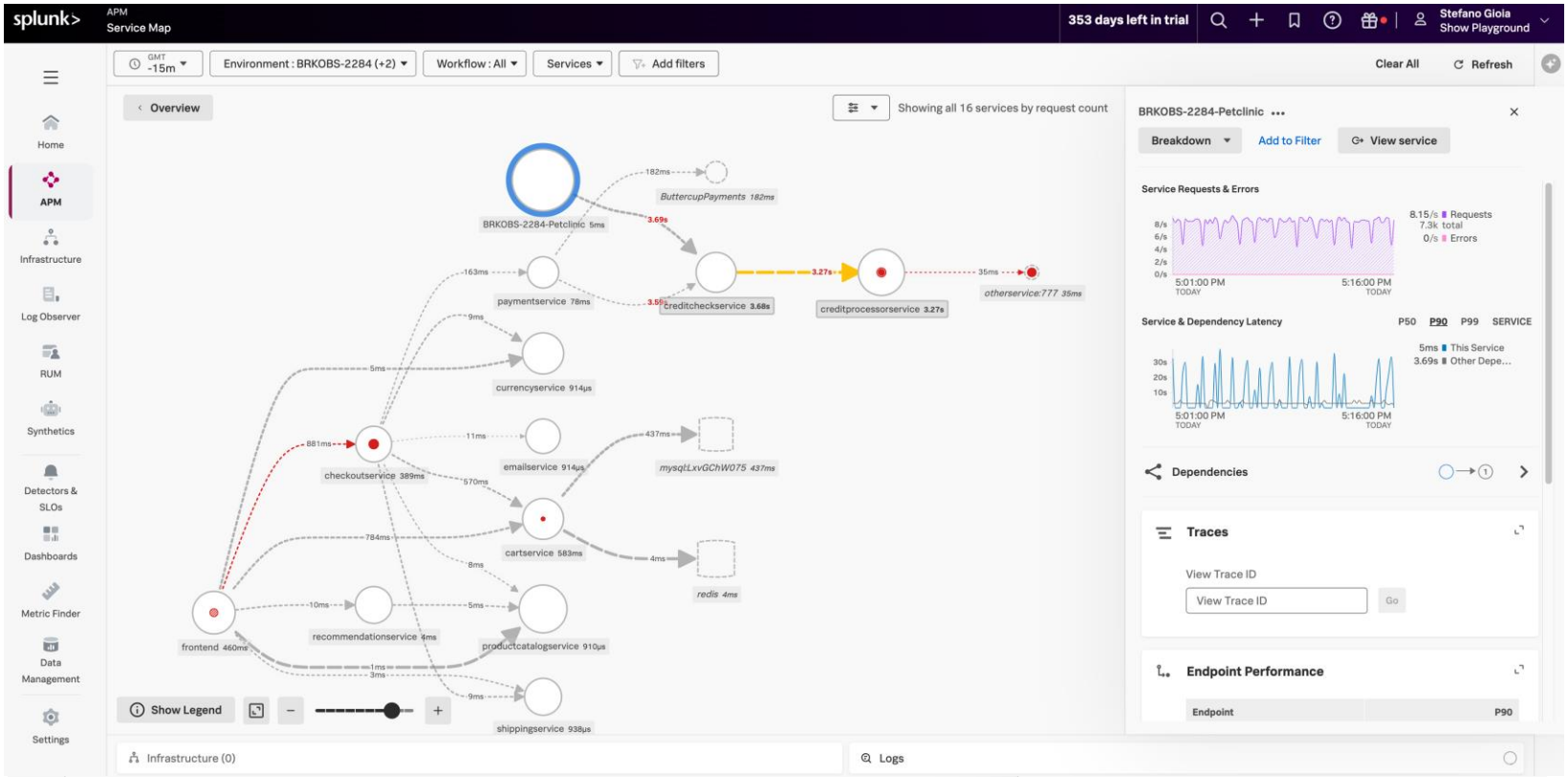
AppDynamics
- Hybrid Agent sends OpenTelemetry data to OpenTelemetry Collector

Splunk Observability
- Leverage OTel Collector and SDKs
- Contextualize Petclinic APM data
- Provide 3 clicks to AppD investigation

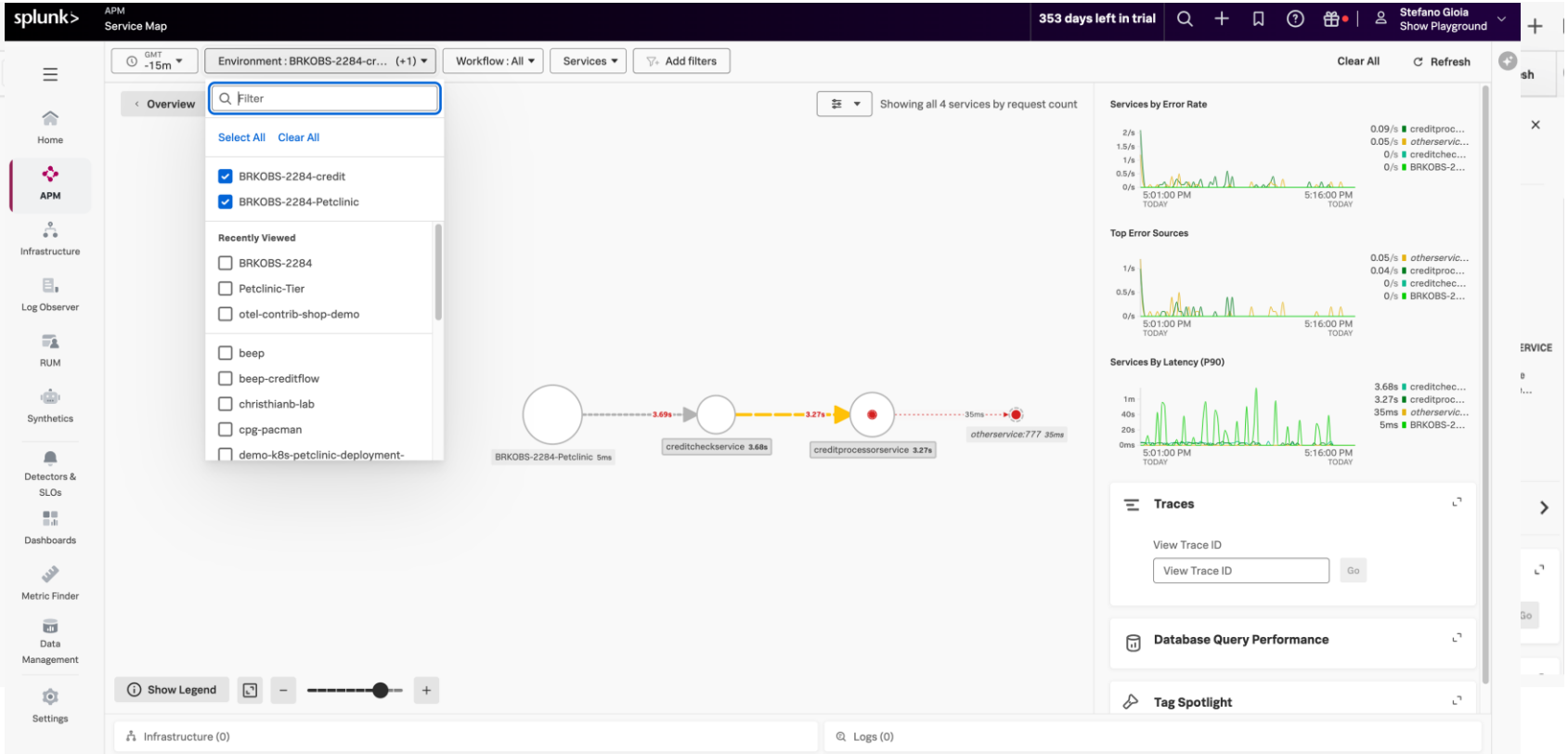


Bringing it all together

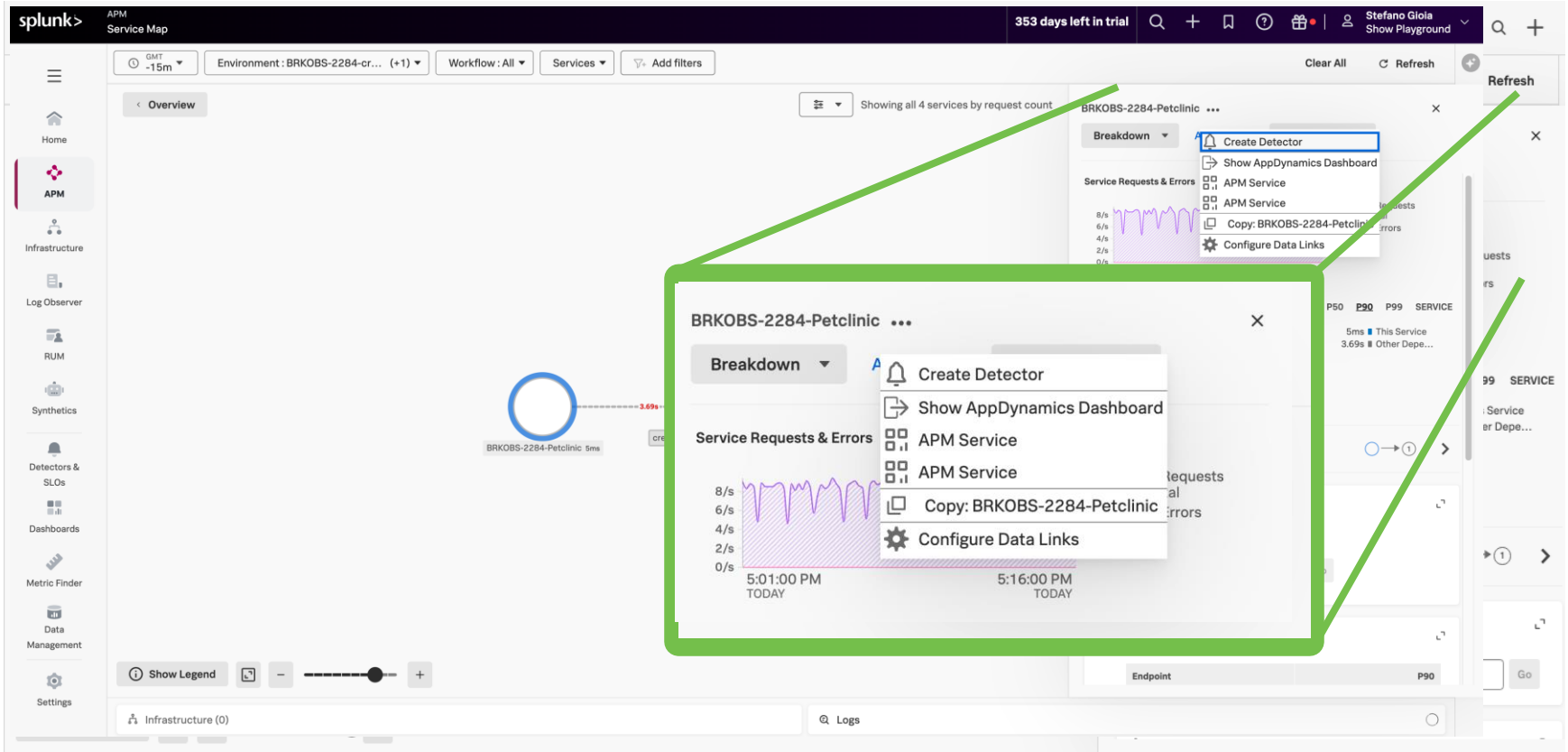
Apps in Cloud – Splunk Observability



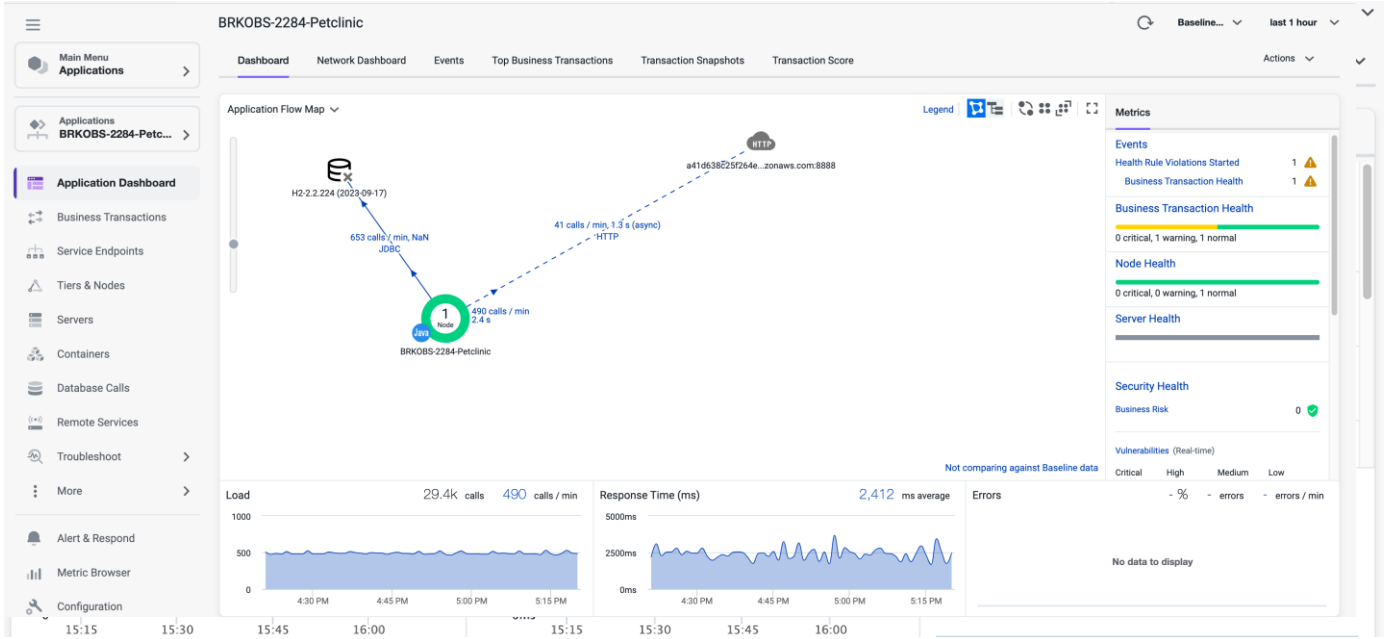
OpenTelemetry Native and AppD Hybrid Agent



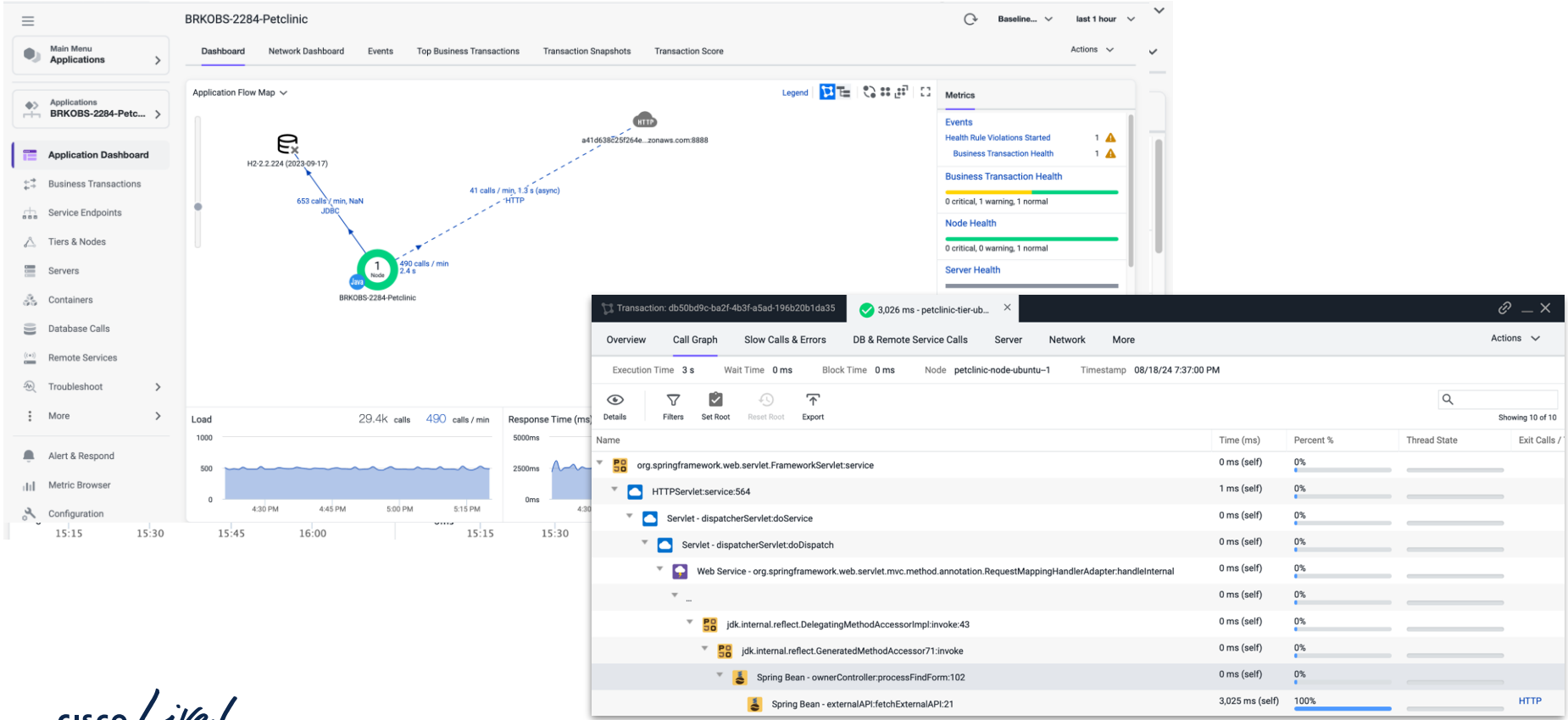
Direct navigation to AppDynamics



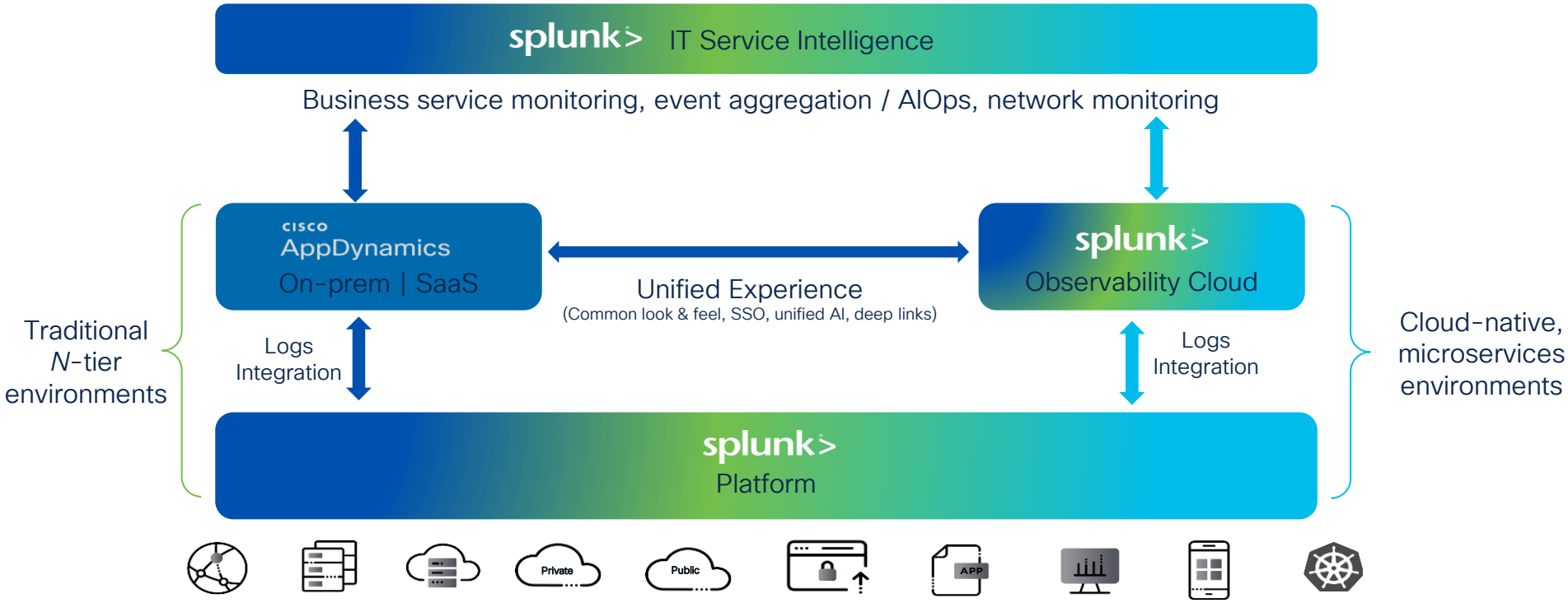
AppDynamics Hybrid Agent



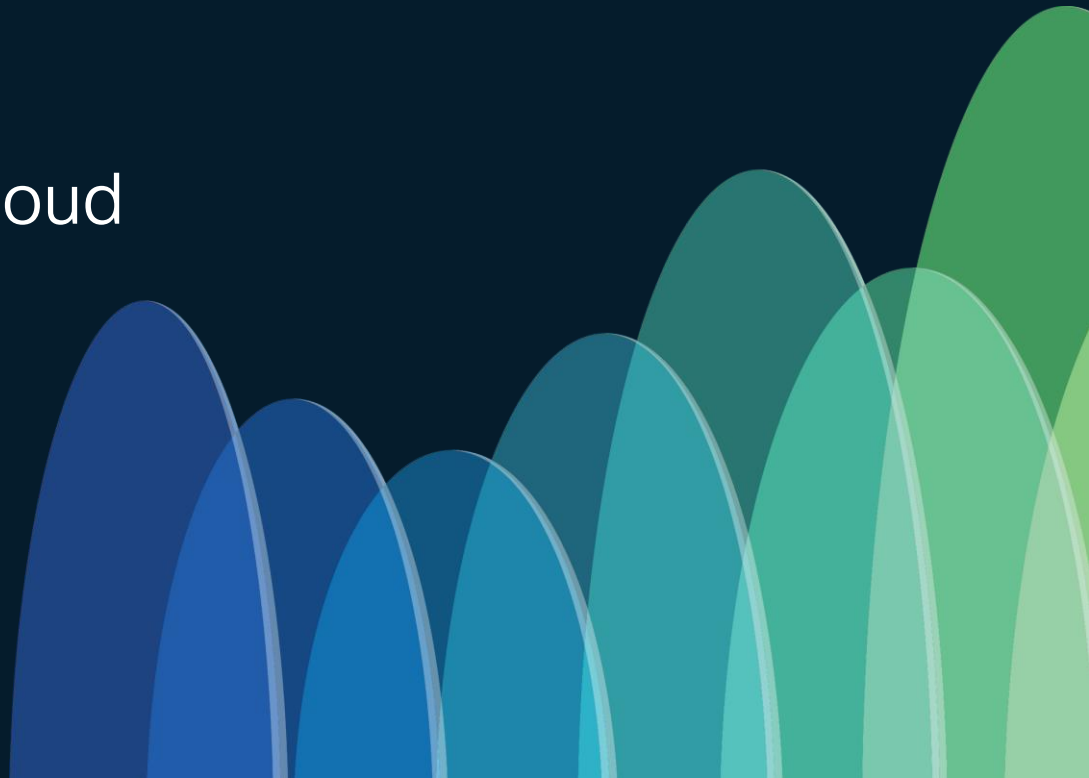
AppDynamics Hybrid Agent



Integrated Splunk Observability



Observability-as-Code with AppDynamics and Splunk Observability Cloud



cisco Live!

Observability-as-code

is the practice
of managing and automating observability tools and processes through code,
allowing for consistent, repeatable, and version-controlled observability
across environments and teams.

Why introducing automation?

ACME Vet Co. has the following challenges, impacting operational efficiency:

- Small operations team – responsible for company wide monitoring
- Manual tool deployment, application instrumentation and maintenance
- Developer productivity issues due to manual processes

Leading to high costs and delayed releases

Observability-as-Code in AppDynamics

cisco Live!

Introducing Smart Agent!

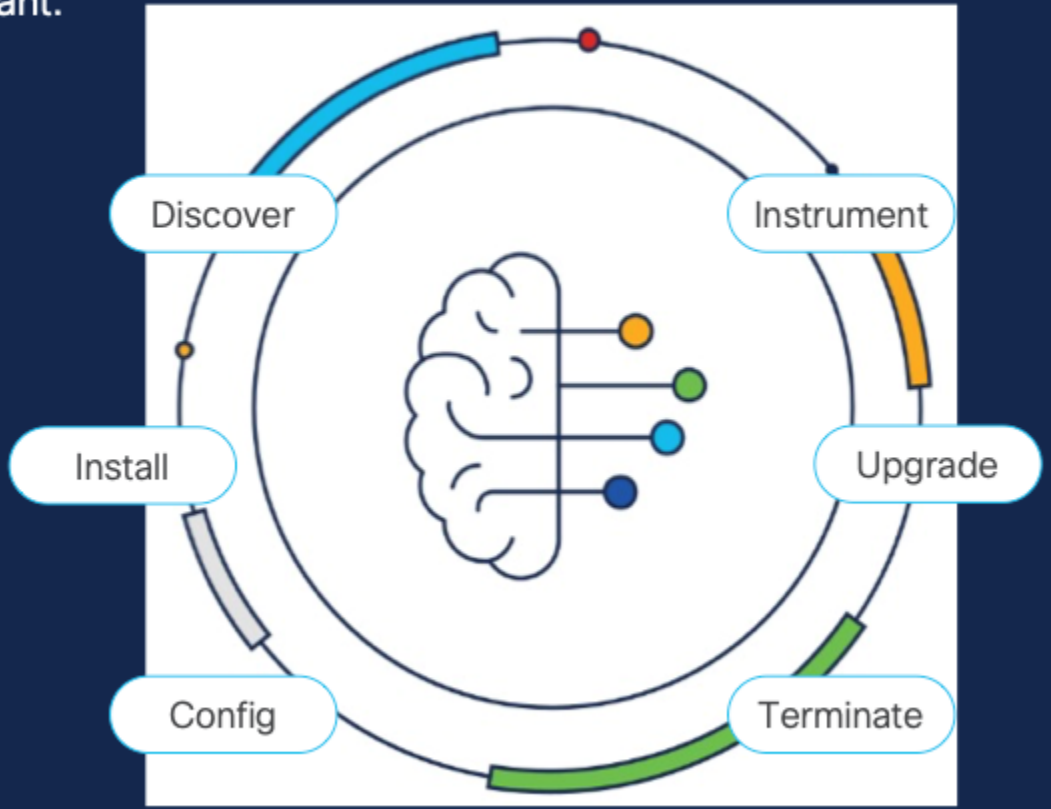
Your intelligent agent management assistant.

Orchestrate agent install, config and upgrades to make routine maintenance simpler and faster

Automate agent lifecycle to maintain compliance

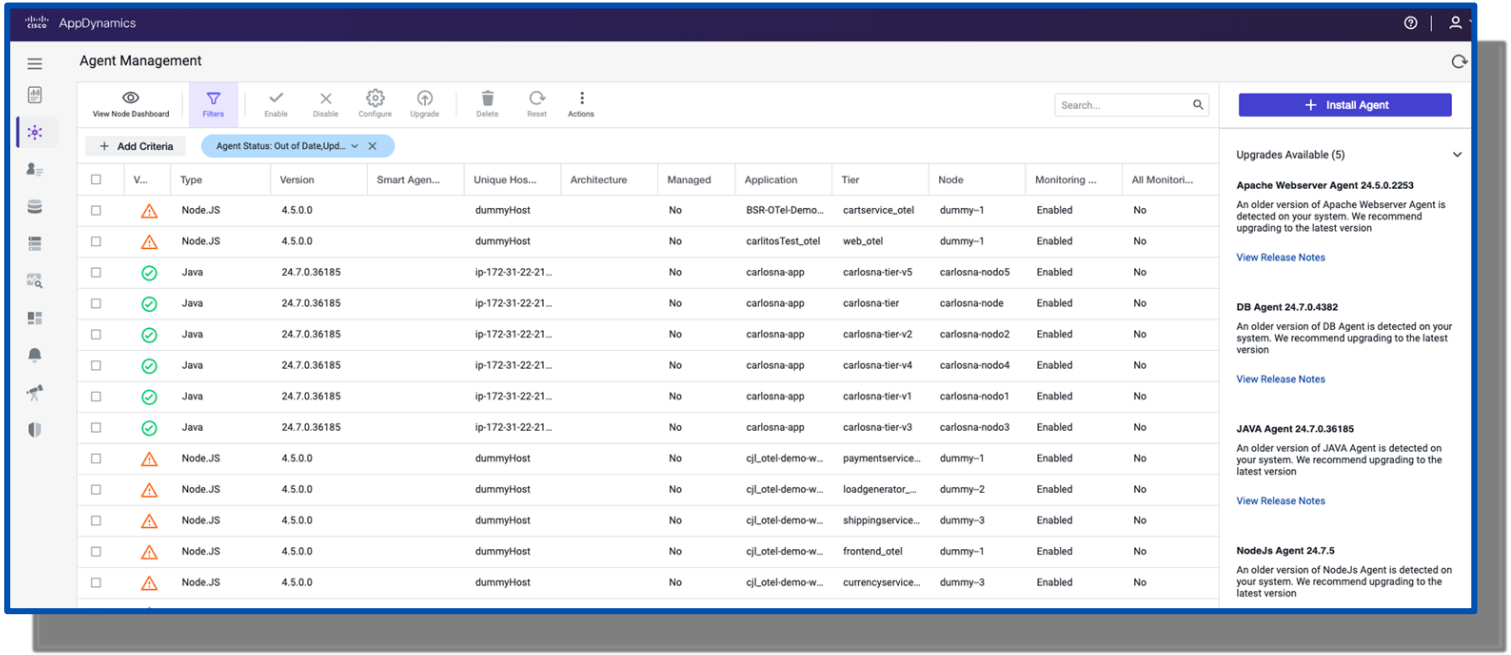
Simplify onboarding with a single Smart Agent installation

*Smart Agent required to drive upgrades from agent management console



Smart Agent

Smart Agent allows you to manage the agent operations (install, upgrade, and rollback) for agents by using the Controller user interface.



Auto-Discover Applications and Auto-Attach Agents to Applications

Smart Agent Command Line Utility

With the Smart Agent CLI, users can install, upgrade, config, uninstall Smart Agents, Infrastructure agents, and Language APM agents into multiple nodes/machines and/or containers.

CODE

Copy

```
appd install java --config /tmp/config.ini --app-name demo-java-app --tier-name demo-java-tier
appd install node -c /tmp/config.ini --app-name demo-node-app --node-name demo-node-name
appd install machine -c /tmp/config.ini --app-name demo-machine-app --install-agent-from /vagrant/appdynamics-machine-agent-23.9.10.zip
```

Auto-attach applications

CODE

```
appd configure smartagent --attach-configure-file { path_to_ld_preload_json }
```

Monitoring Configuration
Manage Health Rules using Terraform

Example: creating a Health Rule

```
resource "appd_health_rule" "my_baseline_rule" {
  name = "My Baseline Health Rule"
  application_id = var.application_id
  metric_aggregation_function = "VALUE"
  eval_detail_type = "SINGLE_METRIC"
  affected_entity_type = "BUSINESS_TRANSACTION_PERFORMANCE"
  business_transaction_scope = "ALL_BUSINESS_TRANSACTIONS"
  baseline_condition = "WITHIN_BASELINE"
  metric_eval_detail_type = "BASELINE_TYPE"
  baseline_name = "All data - Last 15 days"
  baseline_unit = "STANDARD_DEVIATIONS"
  metric_path = "95th Percentile Response Time (ms)"
  warn_compare_value = 1
  critical_compare_value = 2
}
```

Using techniques such as this help reduce toil and increase consistency and repeatability throughout the SDLC

Monitoring Configuration
Manage Policies using Terraform

Example: creating a Policy

```
resource "appd_policy" "my_policy" {
  name = "My Policy"
  application_id = var.application_id
  action_name = "my action"
  action_type = appd_action.my-first-action.action_type
  health_rule_event_types = [
    "HEALTH_RULE_OPEN_WARNING",
    "HEALTH_RULE_OPEN_CRITICAL"]
  health_rule_scope_type = "SPECIFIC_HEALTH_RULES"
  health_rules = ["my health rule"]
}
```

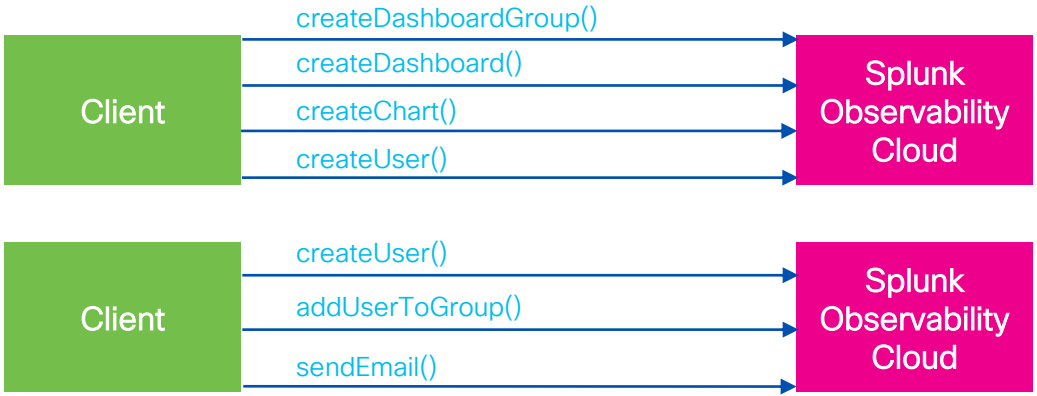
Correlate the appropriate Health Rules to the correct Actions!

Observability-as-Code in Splunk Observability Cloud

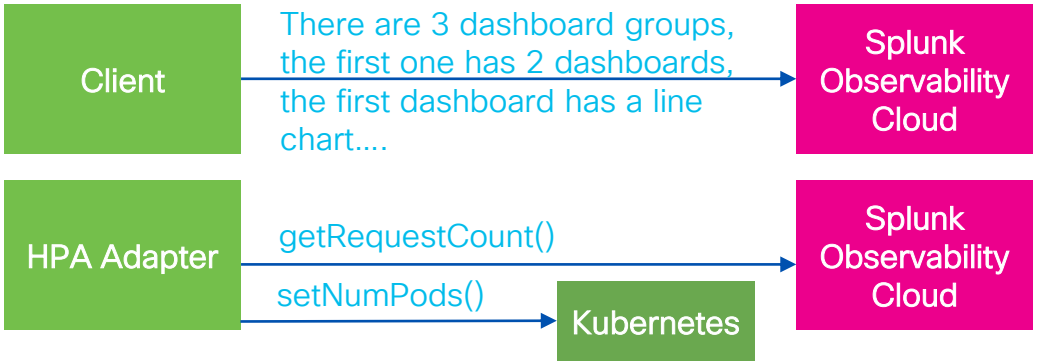
cisco Live!

Observability-as-Code in Splunk Observability Cloud

Imperative:
- API



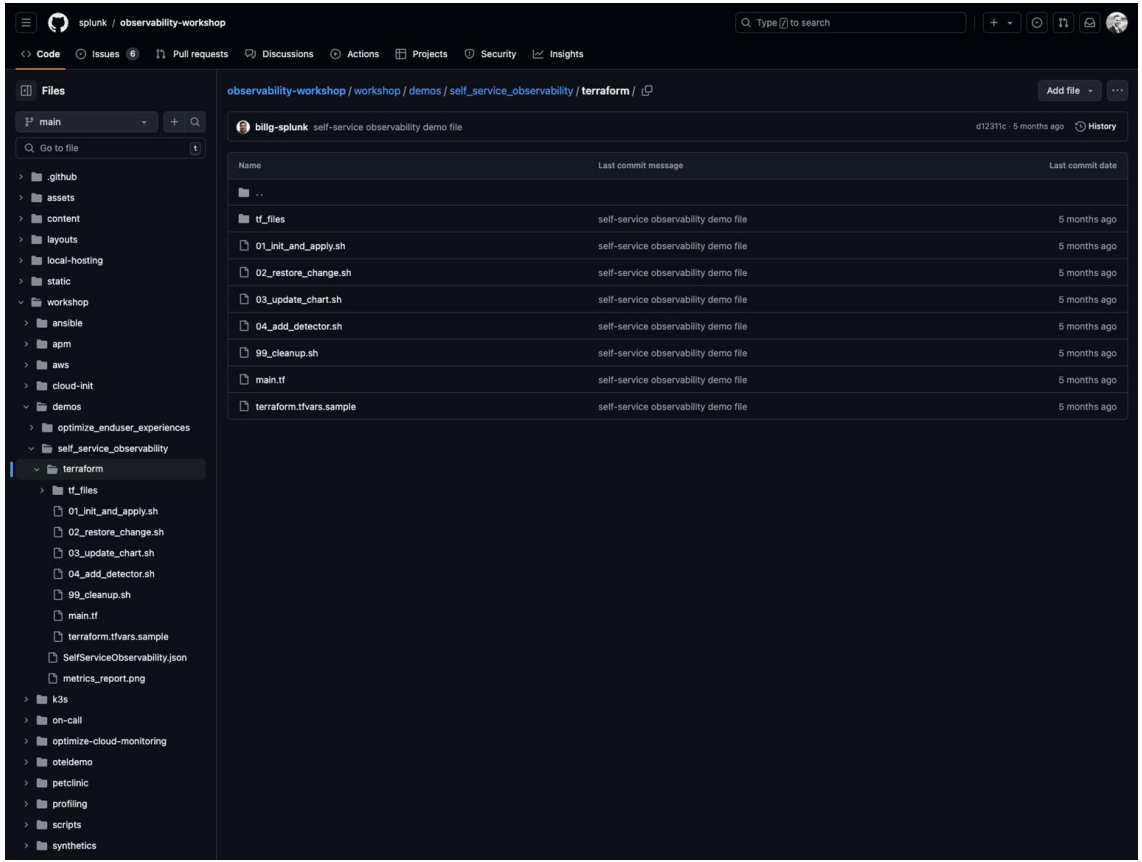
Declarative:
- Terraform



Standardized monitoring with Observability-as-Code

Charts, Dashboards, and Alert templates can be stored in code repositories (GitHub, GitLab, etc) enabling:

- Standardized Observability dashboards and alerts
- Automatic deployment for new teams and applications using automation (Terraform, etc)
- Version control



Observability-as-Code with Dashboards

Example (abbreviated Terraform)

```
resource "signalfx_detector" "cpu_detector" {
  name = "OAC - CPU over 90% for 1 minute"
  description = ""
  max_delay   = 30
  tags = ["prod"]
  program_text = <<-EOF
  signal = data('cpu.utilization').publish()
  detect(when(signal > 90, '1m')).publish('CPU over 90% for 1 min')
  EOF
  rule {
    description   = "CPU over 90% for 1 min"
    severity                    = "Critical"
    detect_label  = "CPU over 90% for 1 min"
    notifications = ["Email, foo-alerts@bar.com"]
  }
}
```

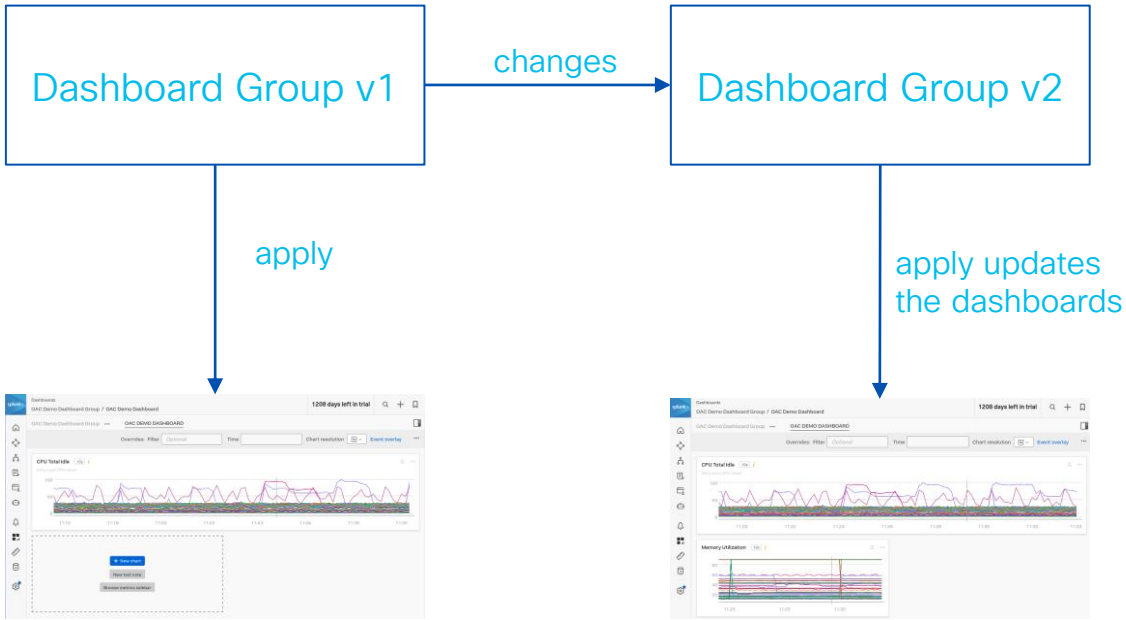
```
resource "signalfx_time_chart" "mychart0" {
  name = "CPU Utilization"
  description = "Very cool CPU chart"
  program_text = <<-EOF
  A=data("cpu.utilization").publish()
  B=alerts(detector_id="${signalfx_detector.cpu_detector.id}")
    .publish(label="B");
  EOF
}

resource "signalfx_dashboard_group" "mydashgroup0" {
  name = "OAC Demo Dashboard Group"
}

resource "signalfx_dashboard" "mydashboard0" {
  name = "OAC Demo Dashboard"
  dashboard_group = signalfx_dashboard_group.mydashgroup0.id
  chart {
    chart_id = signalfx_time_chart.mychart0.id
    width = 12
    height = 1
    row = 0
  }
}
```

Observability-as-Code with Dashboards

Maintaining State

Observability-as-Code with Dashboards

Maintaining State



Demo:

- Interoperability (OpenTelemetry)

- Automation (CI/CD pipeline, Observability as Code)

Demo 1: Interoperability via OTEL

- Traditional Application running in AppDynamics
- Enable OTEL and send traces to Splunk Observability Cloud

Demo 2: Automation & Observability as Code

- Automation: CI/CD Pipeline to:
  - Kubernetes cluster creation and instrumentation
  - Application Deployment on the cluster
  - Payment Service deployment and integration with existing App

- Observability as a Code:
  - Create Health Rule in AppDynamics
  - Dashboard Creation in Splunk Observability Cloud

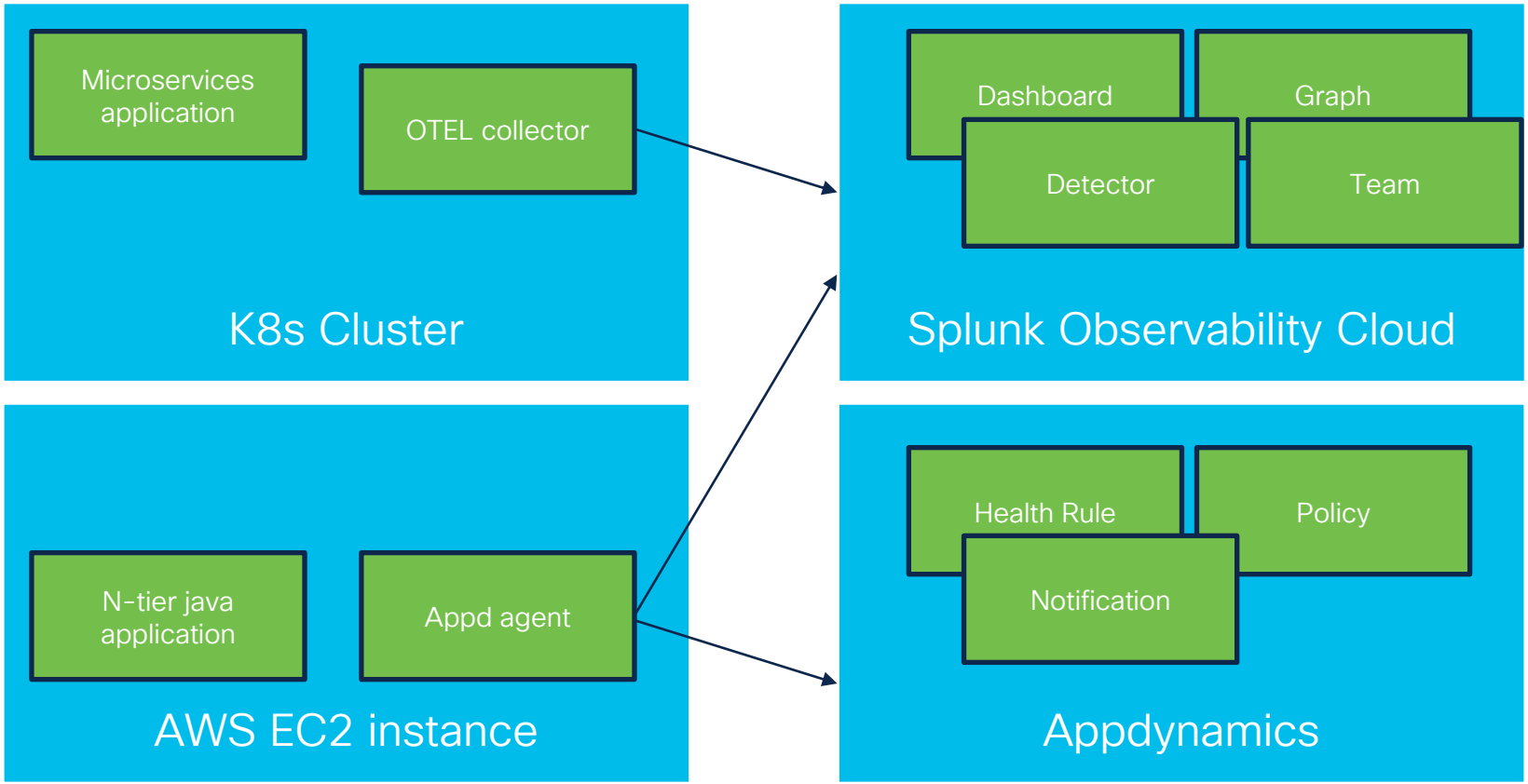
slido

Please download and install the Slido app on all computers you use

How familiar are you with CI/CD pipelines?

ⓘ Start presenting to display the poll results on this slide.

CI/CD Pipeline: what we build at each release



Call to Action

Start testing OpenTelemetry

Adopt automation,
starting with easy steps

Automate the entire release pipeline,
adding observability

And… speak with us 😃

Webex App

Questions?

Use the Webex app to chat with the speaker after the session

How

1. Find this session in the Cisco Events mobile app
2. Click “Join the Discussion”
3. Install the Webex app or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until February 28, 2025.

Fill Out Your Session Surveys

Participants who fill out a minimum of 4 session surveys and the overall event survey will get a unique Cisco Live t-shirt.

(from 11:30 on Thursday, while supplies last)

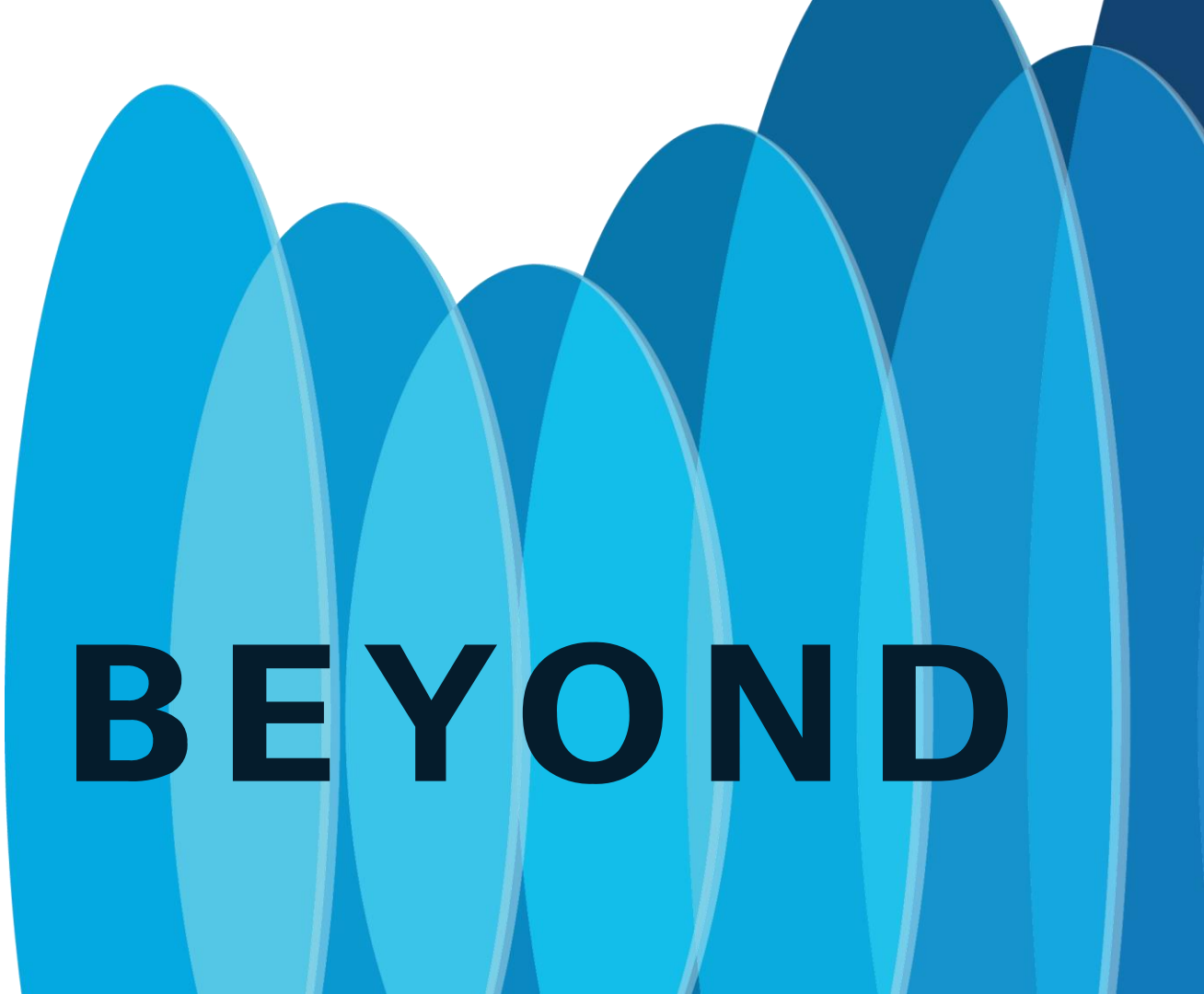
All surveys can be taken in the Cisco Events mobile app or by logging in to the Session Catalog and clicking the ‘Participant Dashboard’

Content Catalog

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at ciscolive.com/on-demand. Sessions from this event will be available from March 3.

cisco Live!



GO BEYOND