



The bridge to possible

Would You Be Mine? Could You Be Mine? Won't You Be My Data?

Cisco FSO Platform Data Pipeline

Geoffrey Hendrey, Chief Architect, Cisco AppDynamics

Cisco Webex App

Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live 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 until February 24, 2023.

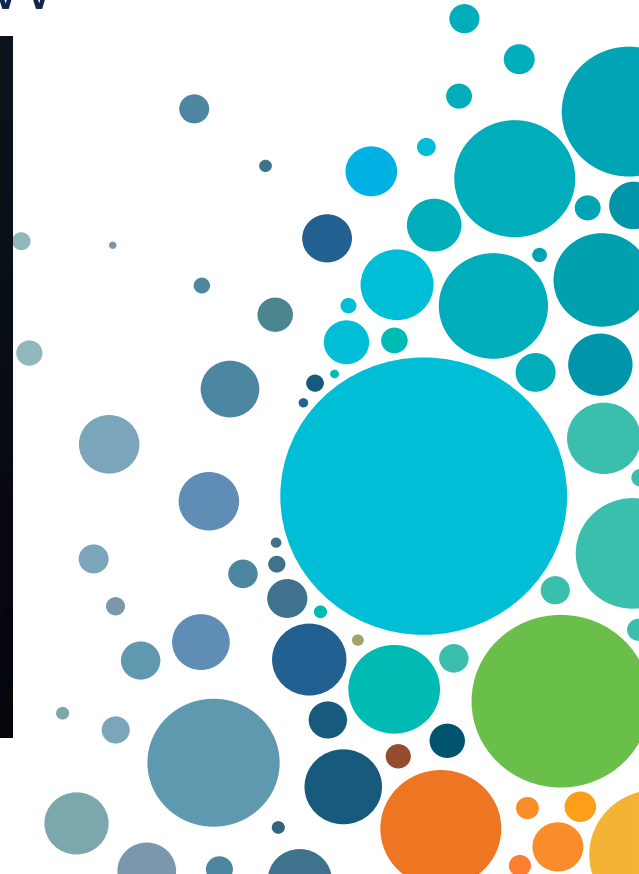




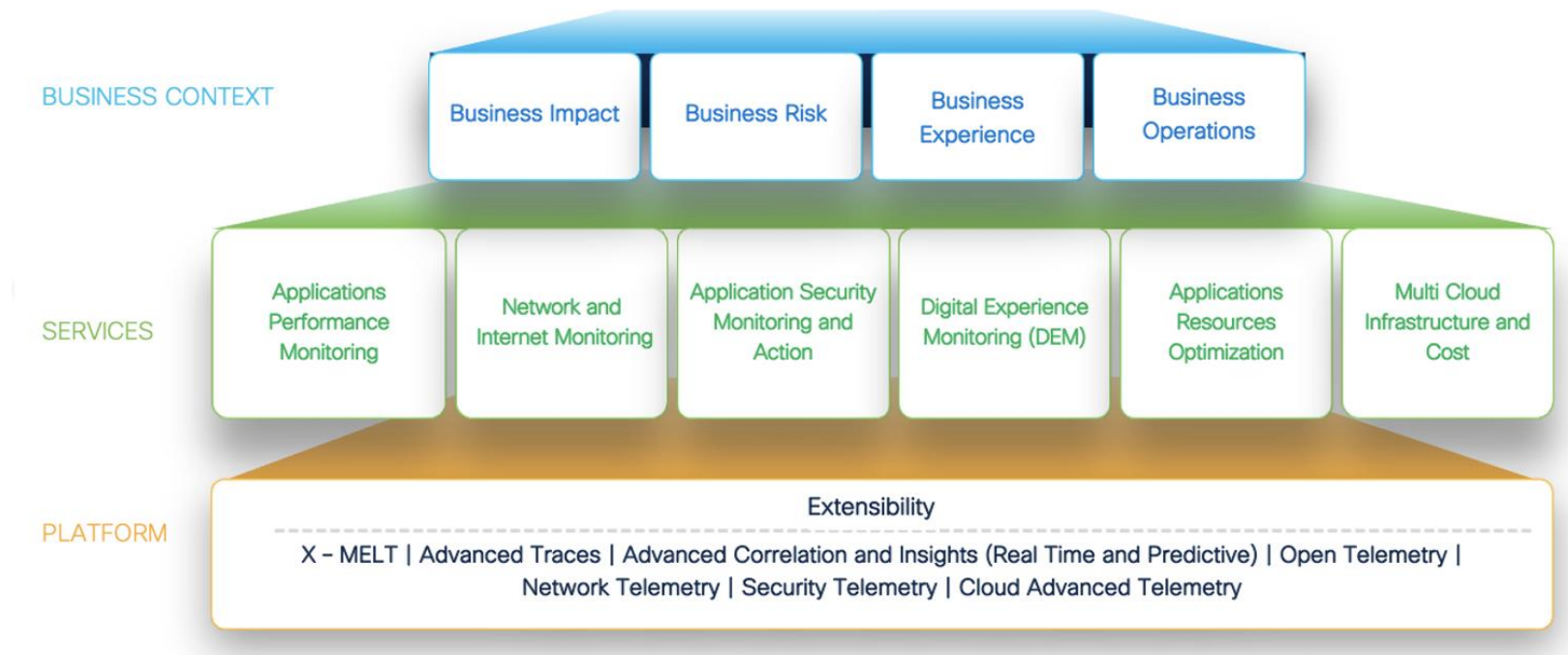
Agenda

- Introduction
- Platform Overview
- Data Processing Pipeline Overview
- Data Processing Language
- Deep Dive
- Conclusion

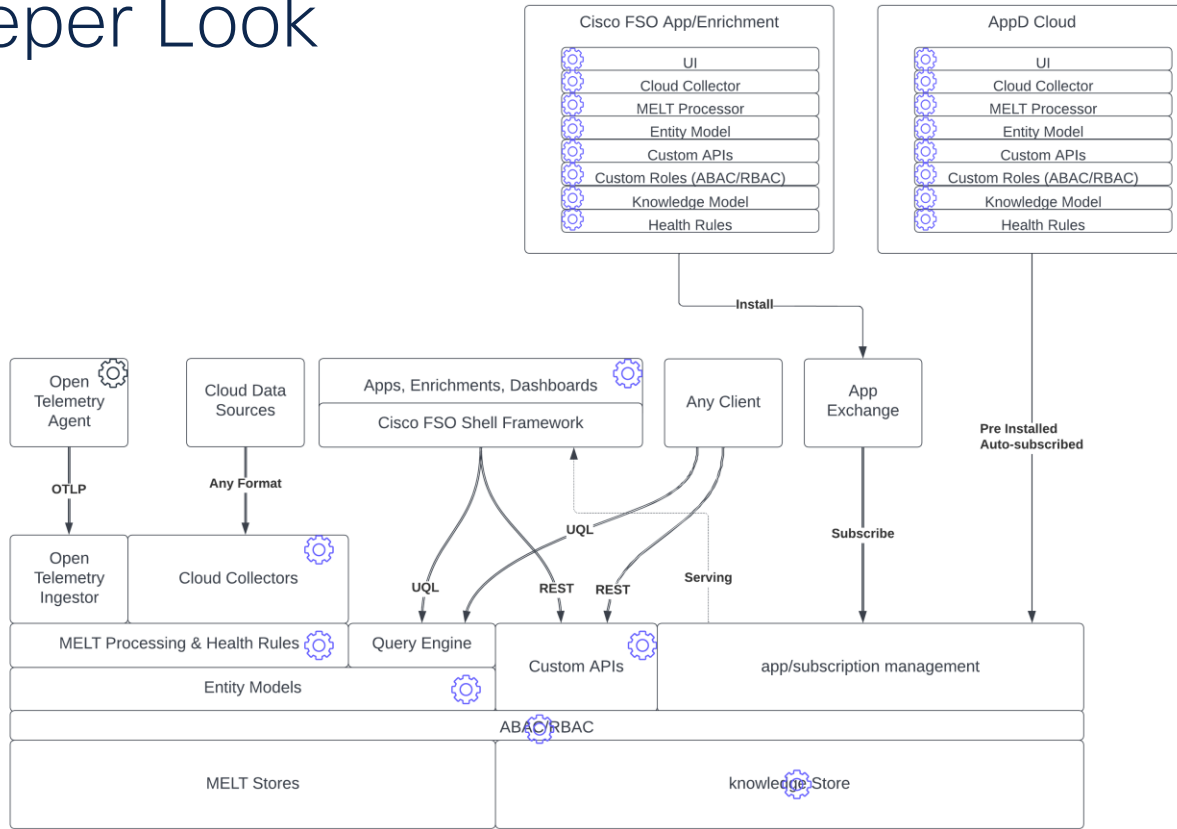
AppDynamics Cloud Overview



Cisco Full Stack Observability Architecture Foundation

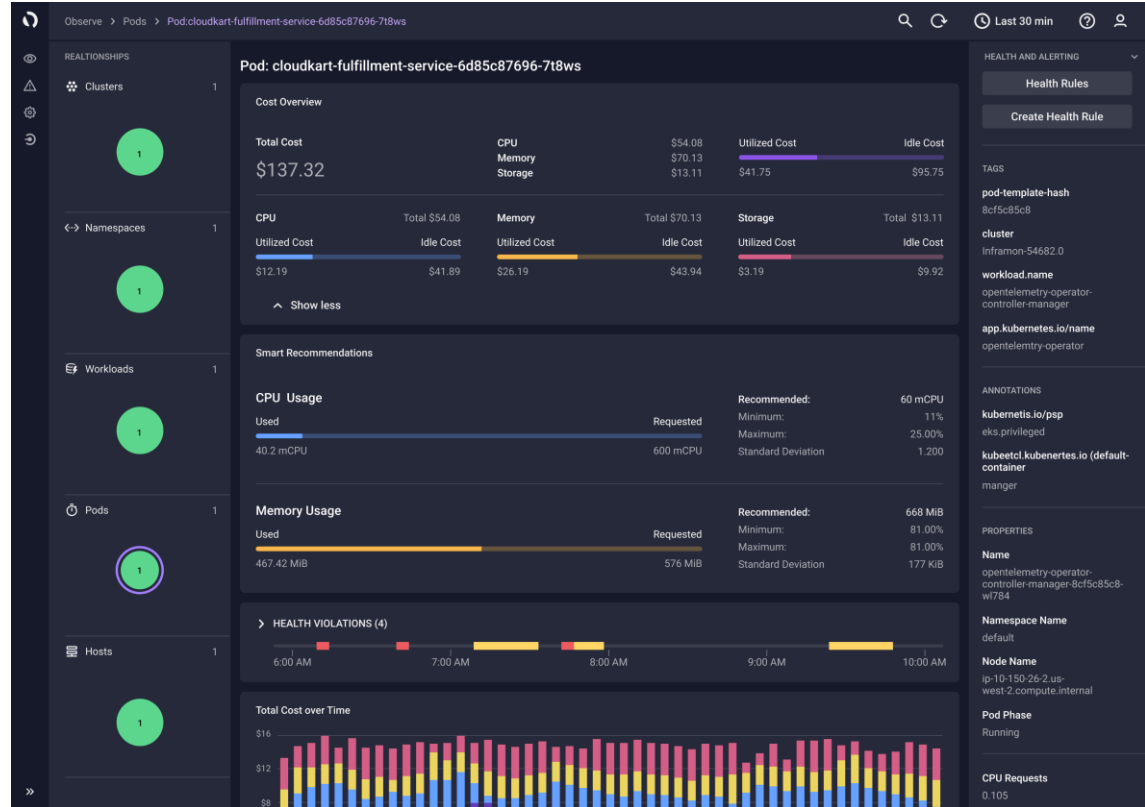


A Deeper Look

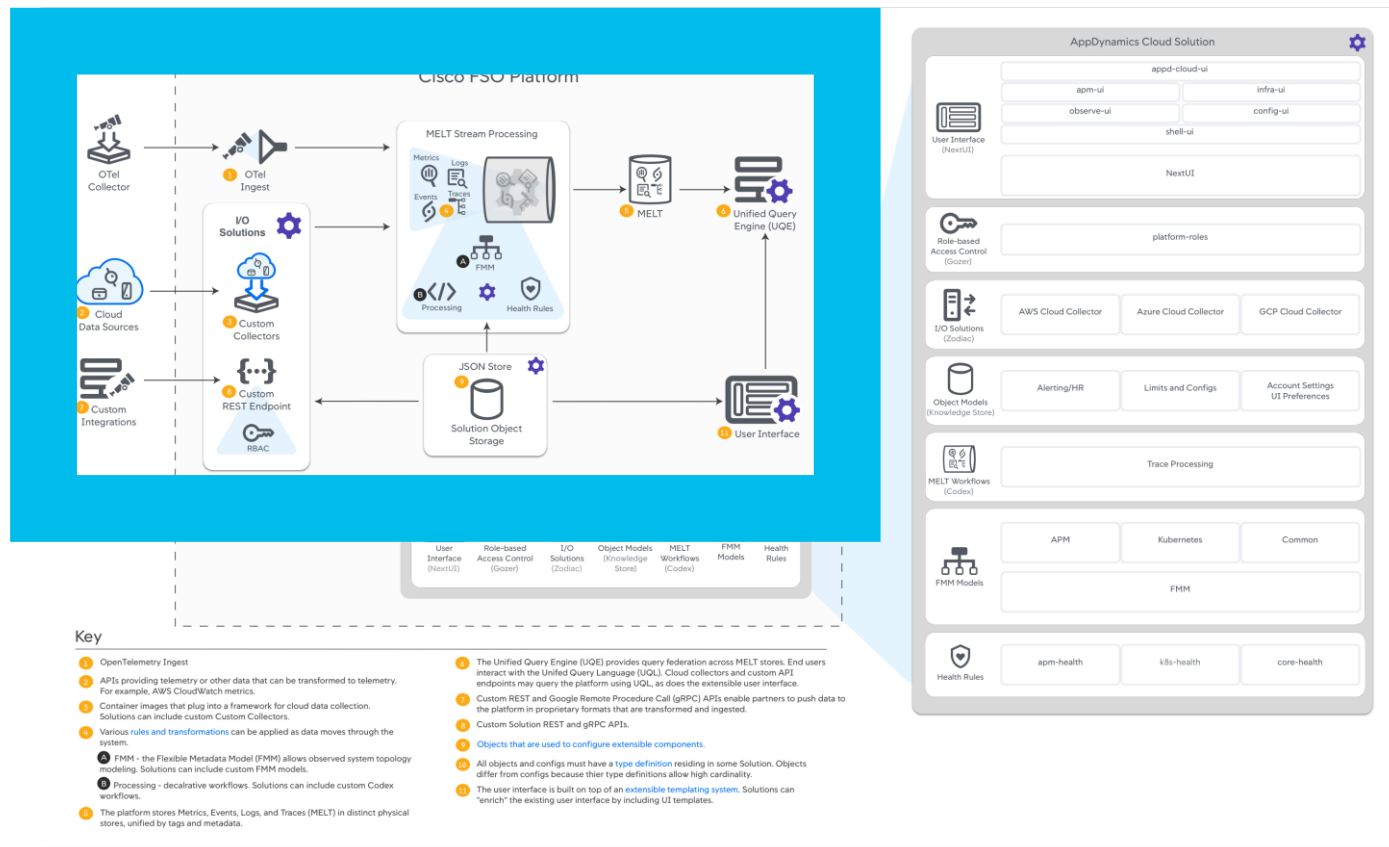


FSO Platform is Extensible

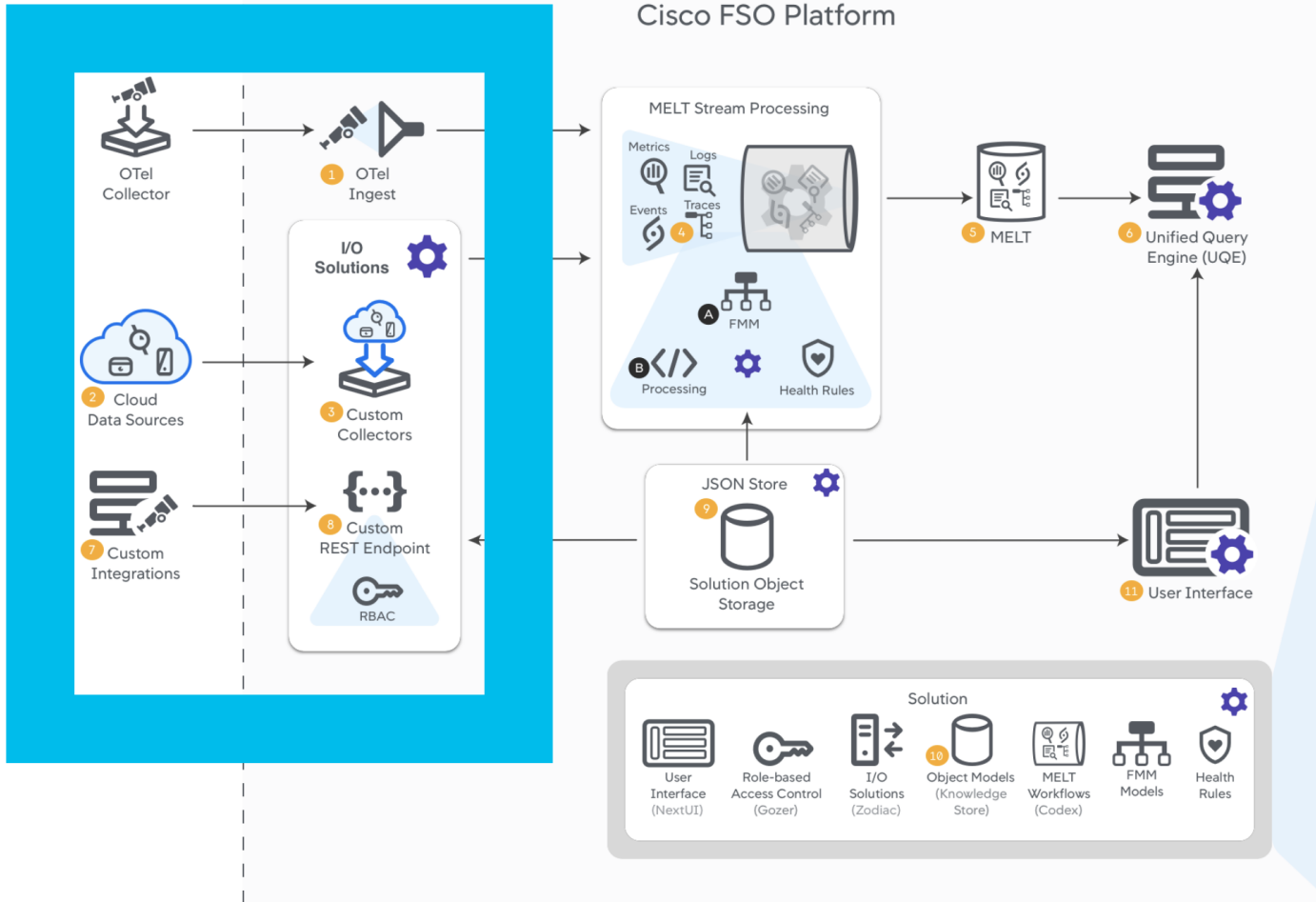
- cost optimizations and kubernetes settings optimizations
- Added to Appd Cloud as 3rd party subscription solutions

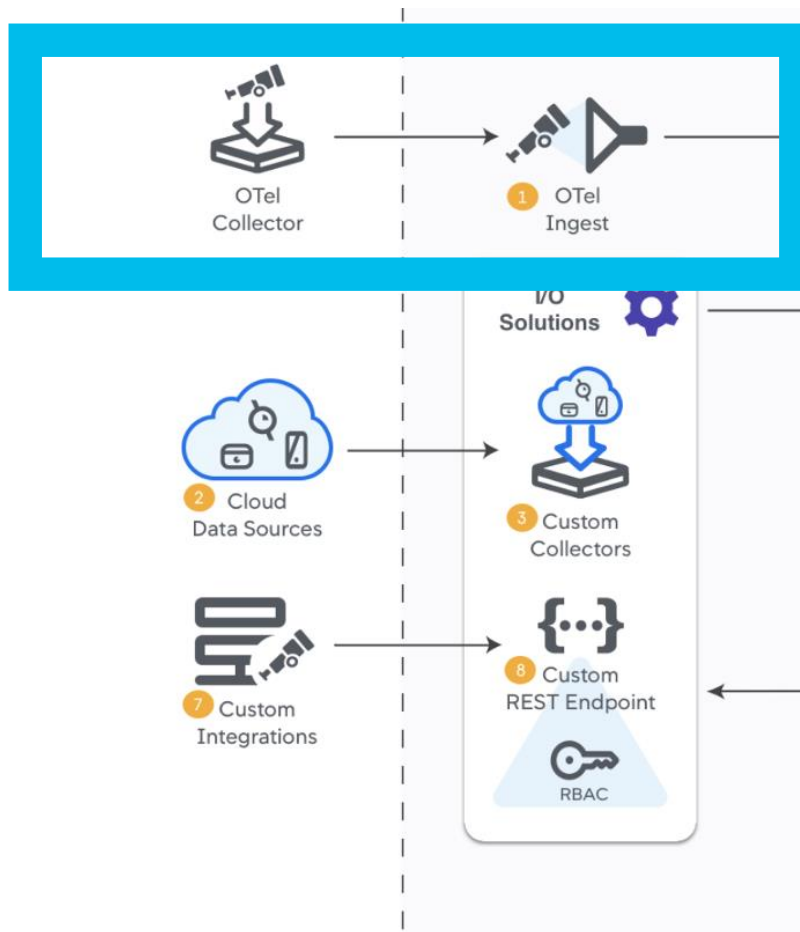


Cisco FSO Platform



Cisco FSO Platform





Getting Data In

- Open Telemetry (push)
- Cloud Collectors (pull)
- Custom REST Endpoints (push or pull)

Open Telemetry



OTel
Collector



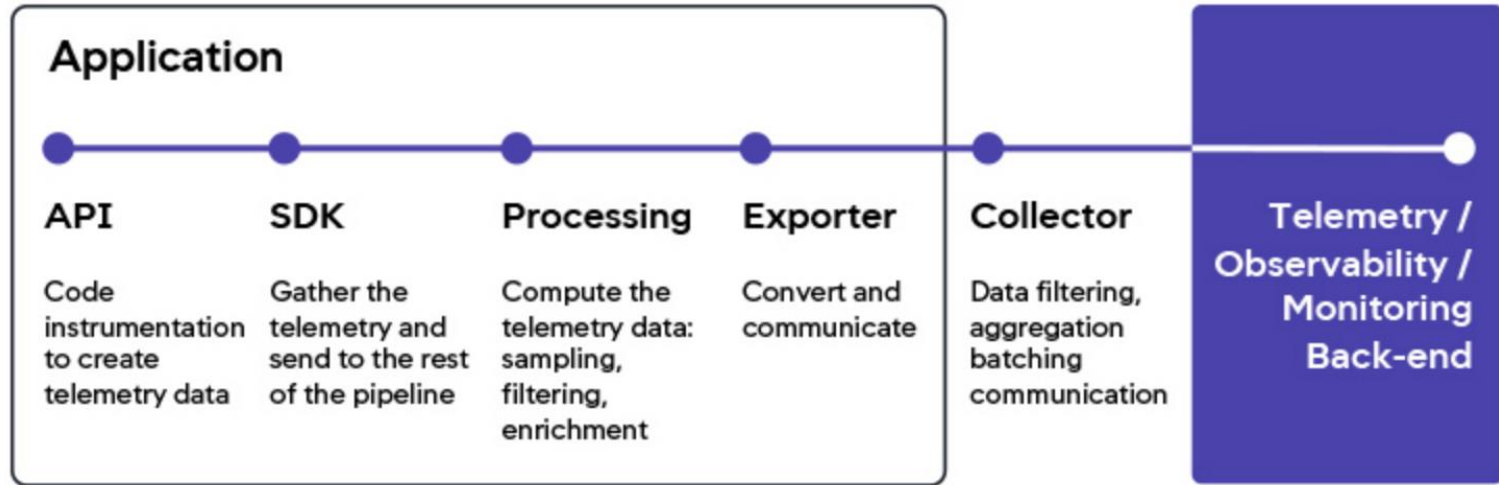
Open Telemetry Collector

OpenTelemetry is a collection of tools, APIs, and SDKs. Use it to instrument, generate, collect, and export telemetry data (metrics, logs, and traces) to help you analyze your software's performance and behavior.



OTel
Collector

Open Telemetry In a Nutshell



M-E-L-T

Metrics provide time-based numerical measurements on elements of the application ecosystems. For instance, there are an average of 255 transactions per second.

Events track individual actions, producing a time-stamped inventory of data related to operations that are defined by the user.

Logging collects application-generated structured or unstructured text added to help troubleshoot. There is much debate about the effectiveness of log data in a microservices context, as it is difficult to scale and often considered overly expensive.

Traces provide an understanding of how requests flow through the system. This can be used to understand interacting parts and their behaviors.

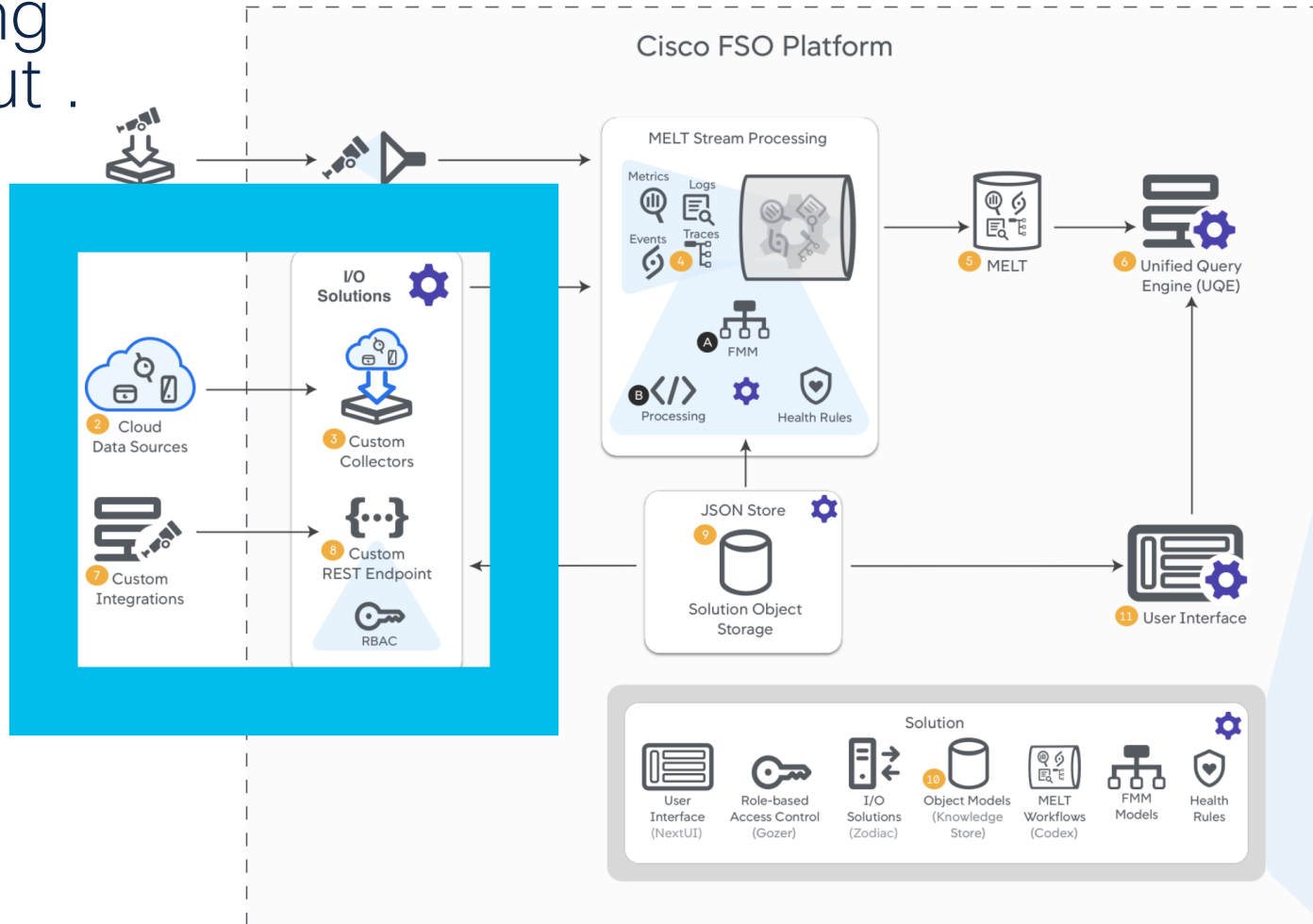
Open Telemetry Ingestion Service



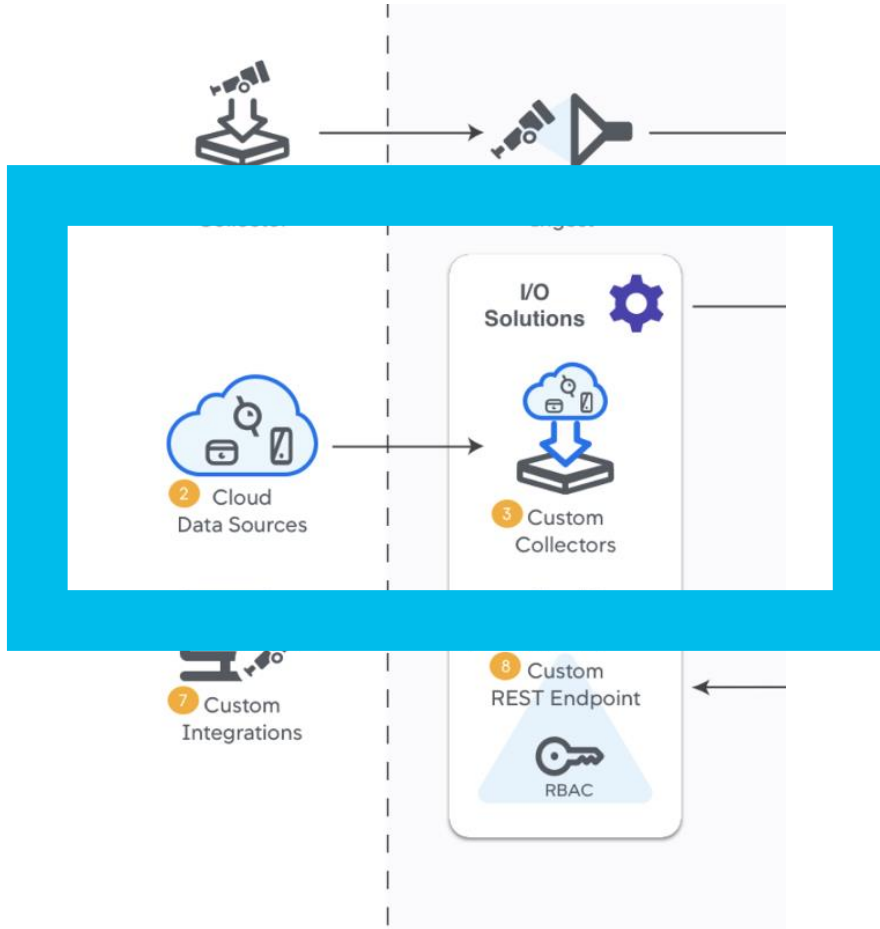
OTel
Collector



Zooming back out .

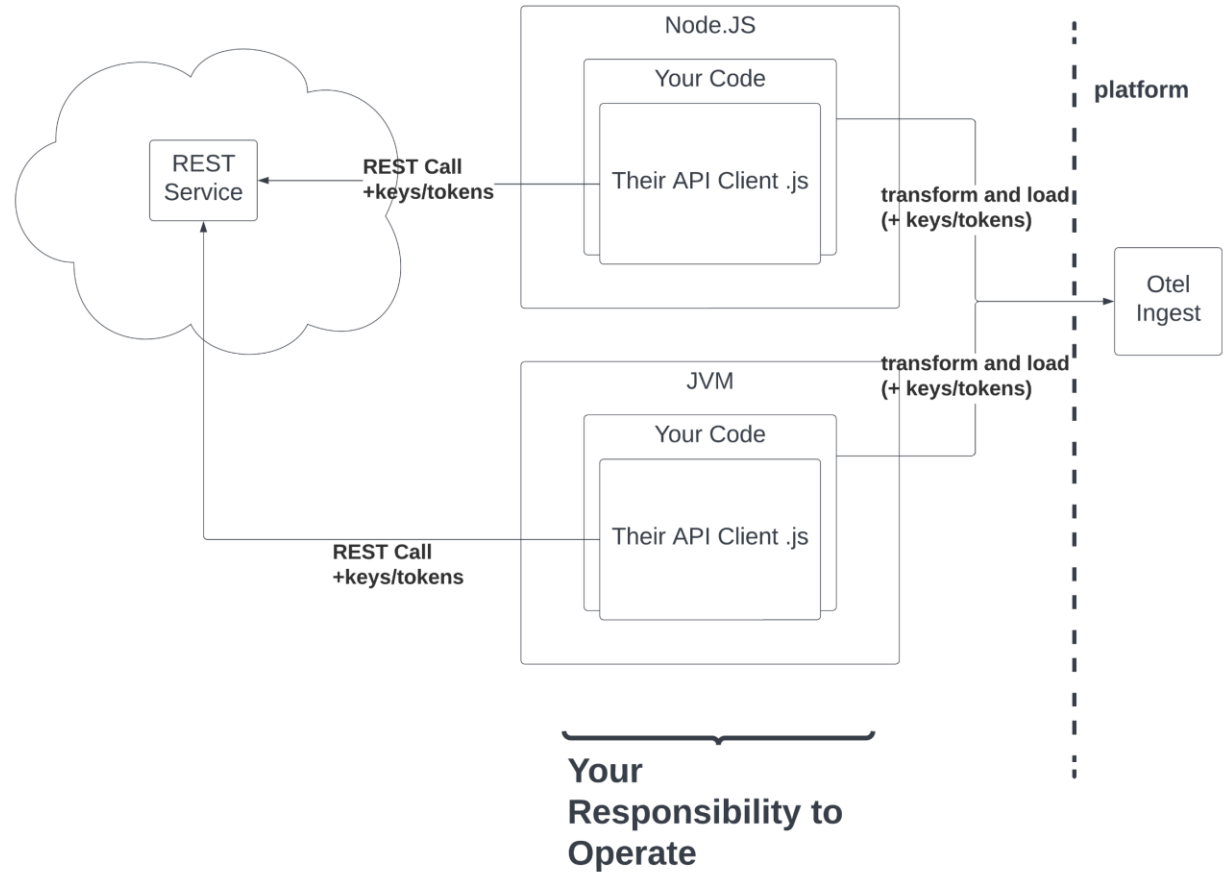


Getting Data In

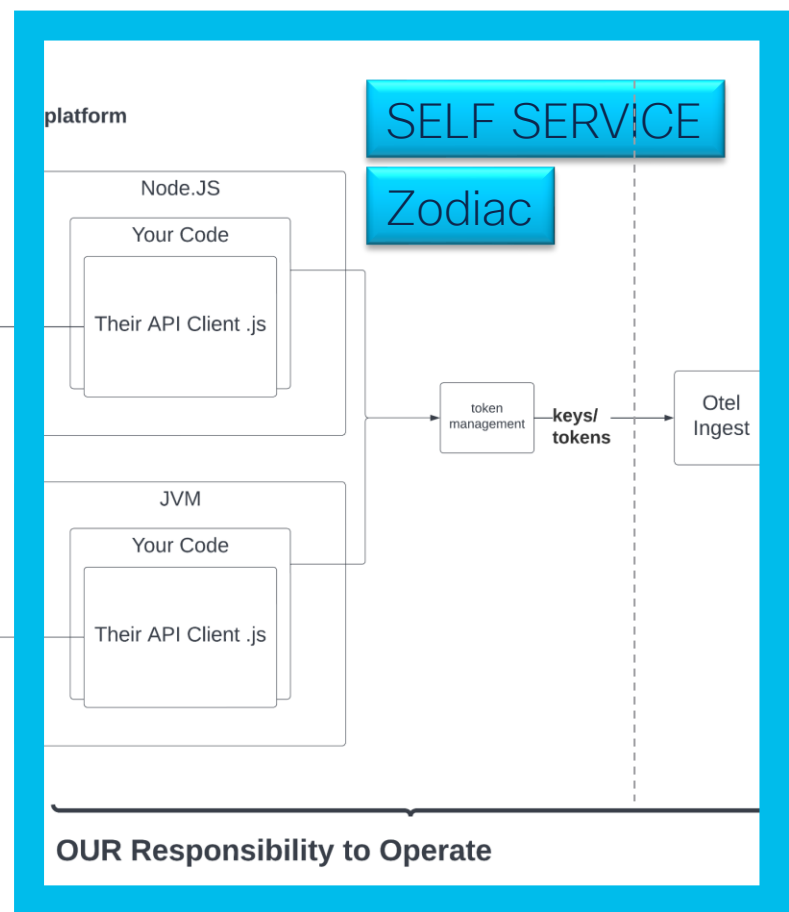
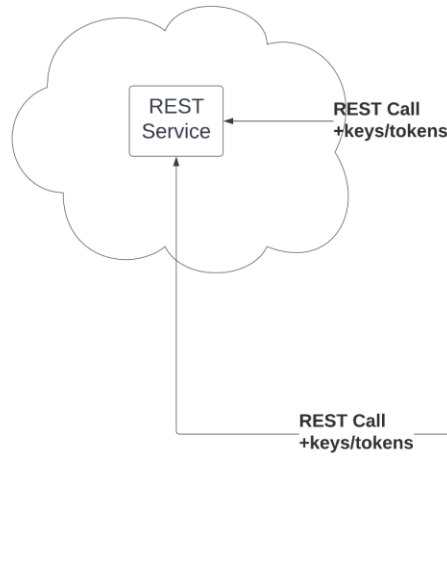


- Open Telemetry (push)
- **Cloud Collectors (pull)**
- Custom REST Endpoints (push or pull)

The Traditional API integration approach to getting data in



The Platform Approach To Getting Data In

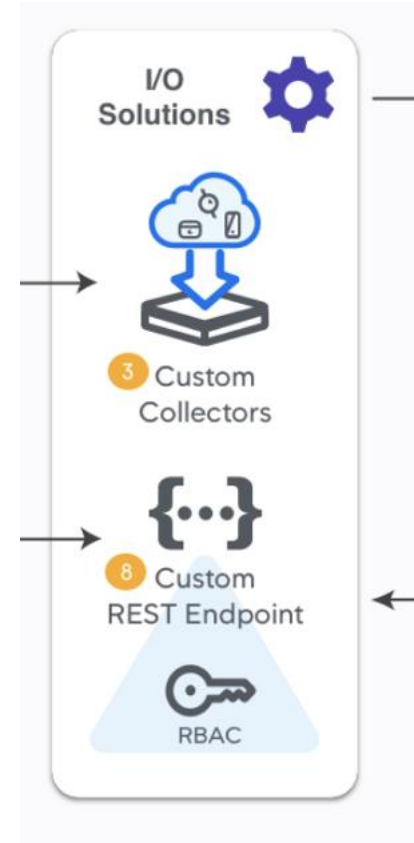


Zodiac containerization

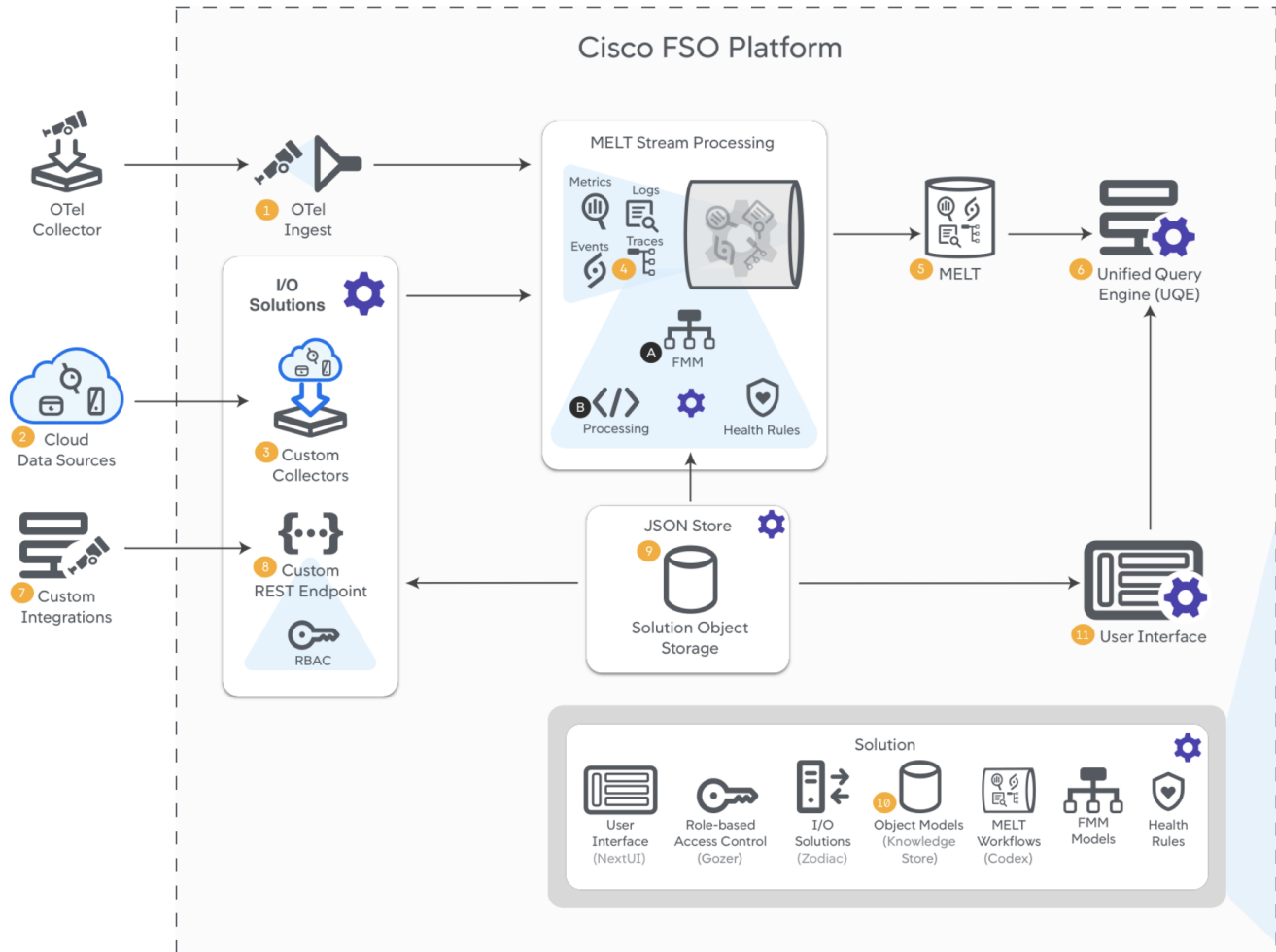
- *Containerized*
- *Secure*
- *Workload Scheduling*
- *Secret Management*

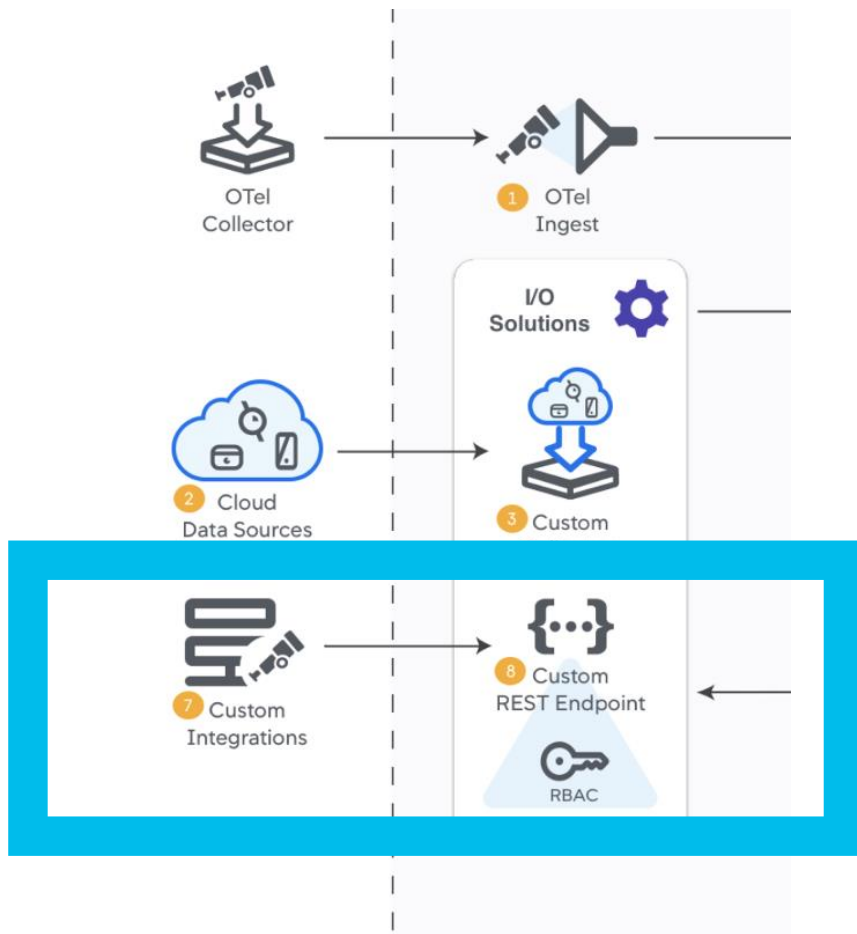
Zodiac is a unique and technically differentiated cisco technology developed specifically to handle your containerized code for getting data in and out of the platform

Zodiac



Zooming back out ...



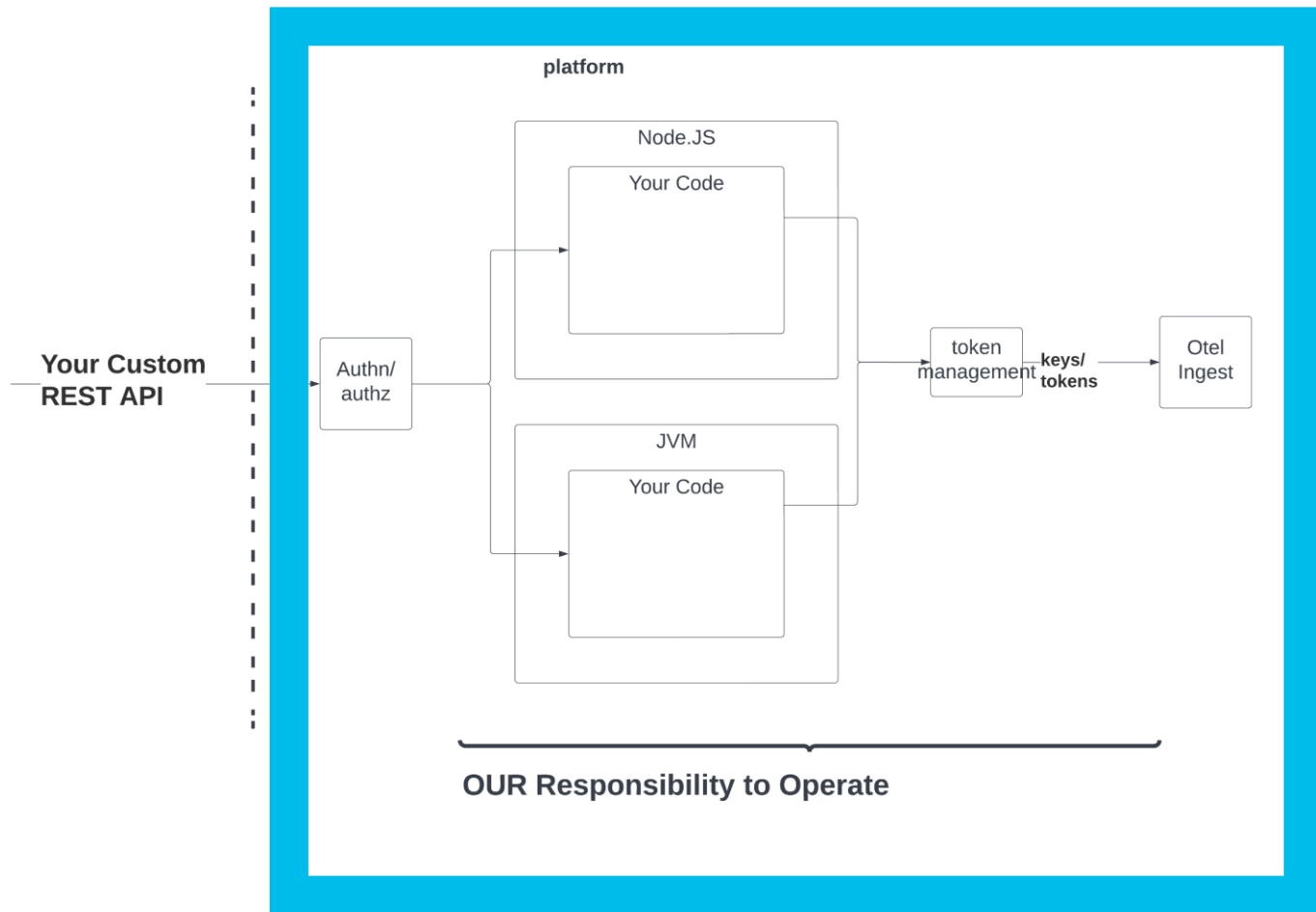


Custom REST APIs

- Open Telemetry (push)
- Cloud Collectors (pull)
- Custom REST Endpoints (push or pull)

Zodiac: self-service for extending the platform

Zodiac
enables
custom
REST
services



FSOC



FSOC

The platform CLI

- Developer friendly command line tool for managing solutions
- Open Source
- <https://github.com/cisco-open/fsoc>

The screenshot shows the GitHub repository page for `cisco-open/fsoc`. The repository is marked as 'Public'. The navigation bar includes links for Product, Solutions, Open Source, and Pricing. Below the repository name, there are tabs for Code, Issues, Pull requests (1), Actions, Projects, Security, and Insights. The 'Releases' tab is selected, showing a list of releases. The latest release is `v0.26.4`, marked as 'Latest', and it includes a note: 'Fixed config file update bug'. Below the release information, there is a list of assets (8 total): `fsoc-darwin-amd64`, `fsoc-darwin-arm64`, `fsoc-linux-amd64`, `fsoc-linux-arm64`, `fsoc-windows-amd64.exe`, `fsoc_0.26.4_checksums.txt`, `Source code (zip)`, and `Source code (tar.gz)`.

FSOC – it's like Kubectl

```
ghendrey@GHENDREY-M-C5F5 ~ % curl -fSL -o fsoc "https://github.com/cisco-open/fsoc/releases/latest/download/fsoc-darwin-amd64"
chmod +x fsoc
sudo mv fsoc /usr/local/bin

% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed

  0     0    0     0    0     0      0      0  --:--:--  --:--:--  --:--:--    0
  0     0    0     0    0     0      0      0  --:--:--  --:--:--  --:--:--    0
100 11.1M 100 11.1M    0     0  564k      0  0:00:20  0:00:20  --:--:--  584k

Password:
ghendrey@GHENDREY-M-C5F5 ~ % fsoc
fsoc is an internal Cisco utility that serves as an entry point for developers on the
Full Stack Observability (FSO) Platform.
It allows developers to interact with the product environments--developer, test and production--in a
uniform way and to perform common tasks. fsoc targets developers building the platform itself, as well
as developers building solutions on the platform.

Examples:
$ fsoc login
$ fsoc uql query "FETCH id, type, attributes FROM entities(k8s:workload)"
$ fsoc solution list
$ fsoc solution list -o json
```

FSOC solution list

```
ghendrey@GHENDREY-M-C5F5 ~ % fsoc solution list
```

NAME	ISSYSTEM	ISSUBSCRIBED	DEPENDENCIES
extensibility	true	true	[]
environment	true	true	["extensibility"]
k8scost-demo	true	true	["zodiac", "fmm"]
gitrepo	true	true	["zodiac"]
apm	true	true	["fmm"]
fmm-demo	true	true	["fmm"]
sapi-solution-mock	true	true	["zodiac"]
sum-solution	true	true	[]
zodiac	true	true	[]
k8s	true	true	["dashui", "fmm", "infra", "aws", "azure"]
fmm-extension-demo	true	null	["fmm-demo"]
fmm	true	true	[]
codex	true	true	[]
preferences	true	true	[]
dashui	true	true	[]
dashui-deploy	true	true	["zodiac"]
brostatus	true	true	["apm", "codex", "fmm", "zodiac"]
greetings	false	false	[]
replex-v1	true	true	["zodiac", "fmm", "dashui"]
bro-status-test	false	null	[]
bro-status	false	null	[]
brostatus-test	true	null	[]

FSOC solution --help

```
ghendrey@GHENDREY-M-C5F5 ~ % fsoc solution --help
```

This command has a list of subcommands that allow you to execute solution lifecycle operations with the FS0 Platform.

Usage:

```
fsoc solution
```

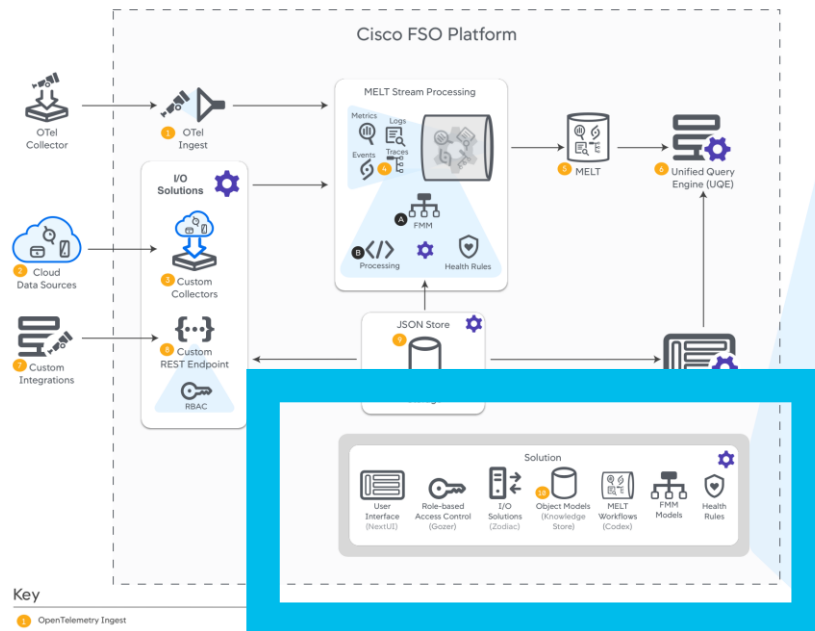
Usage:

```
fsoc solution [command]
```

Available Commands:

check	Validate your solution component definitions
download	Download solution
extend	Extends your solution package by adding new components
fork	Fork a solution in the specified folder
init	Creates a new solution package
list	List all solutions available in this tenant
package	Build and package a solution
push	Deploy your solution
status	Get the installation/upload status of a solution
subscribe	Subscribe to a solution
unsubscribe	Unsubscribe from a solution
validate	Validate your solution package

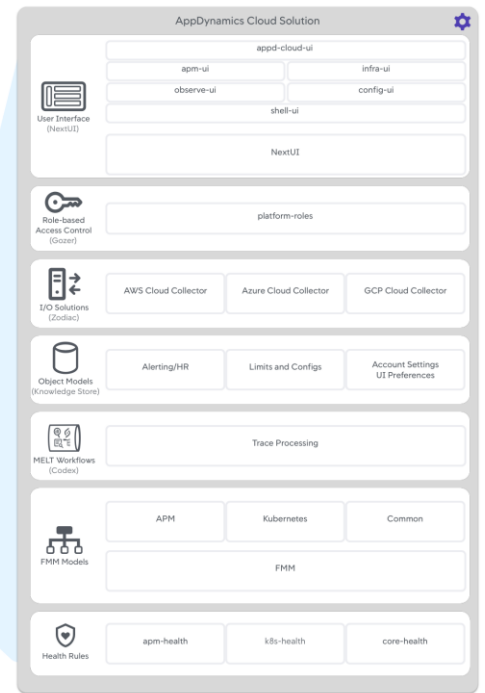
Solution Bundle



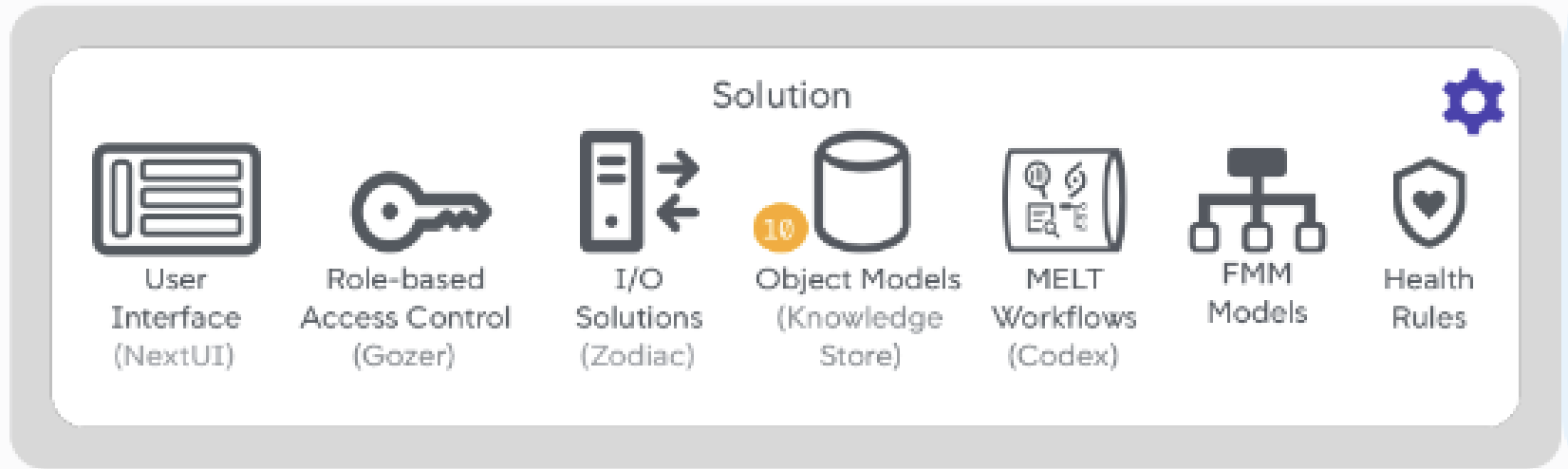
Key

1. OpenTelemetry Ingest
2. APIs providing telemetry or other data that can be transformed to telemetry. For example, AWS CloudWatch metrics.
3. Container images that plug into a framework for cloud data collection. Solutions can include custom Custom Collectors.
4. Various rules and transformations can be applied as data moves through the system.
5. FMM - the Flexible Metadata Model (FMM) allows observed system topology modeling. Solutions can include custom FMM models.
6. Processing - declarative workflows. Solutions can include custom Codex workflows.
7. The platform stores Metrics, Events, Logs, and Traces (MELT) in distinct physical stores, unified by tags and metadata.

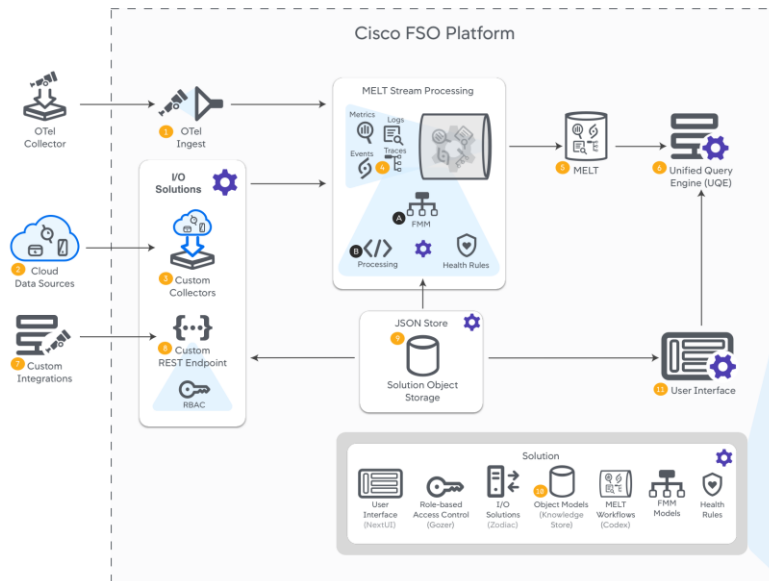
8. endpoints may query the platform using UQL, as does the extensible user interface.
9. Custom REST and Google Remote Procedure Call (gRPC) APIs enable partners to push data to the platform in proprietary formats that are transformed and ingested.
10. Custom Solution REST and gRPC APIs.
11. Objects that are used to configure extensible components.
12. All objects and configs must have a type definition residing in some Solution. Objects differ from configs because their type definitions allow high cardinality.
13. The user interface is built on top of an extensible templating system. Solutions can "enrich" the existing user interface by including UI templates.



Artifacts of a Solution Bundle



AppDynamics Cloud Is a Solution Bundle on Cisco FSO Platform

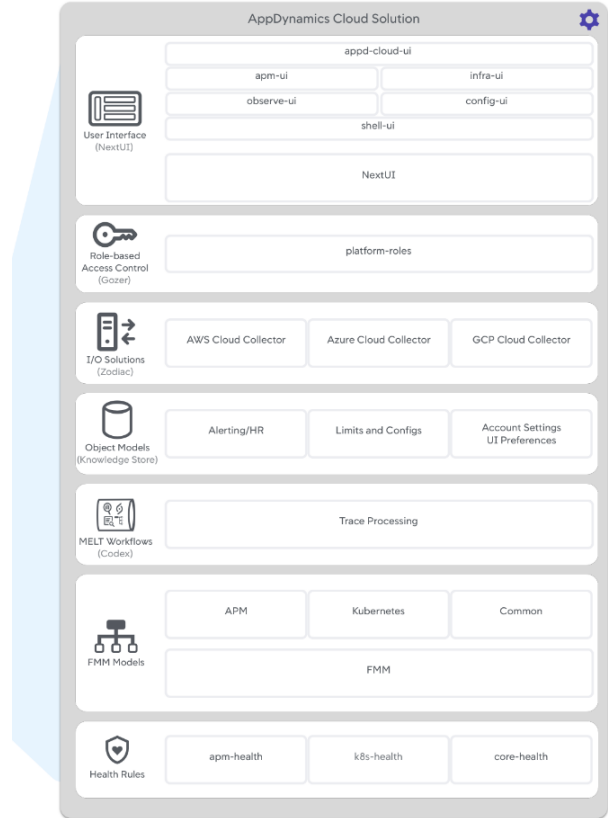


Key

1. OpenTelemetry Ingest
2. APIs providing telemetry or other data that can be transformed to telemetry. For example, AWS CloudWatch metrics.
3. Container images that plug into a framework for cloud data collection. Solutions can include custom Custom Collectors.
4. Various rules and transformations can be applied as data moves through the system.
5. FMM - the Flexible Metadata Model (FMM) allows observed system topology modeling. Solutions can include custom FMM models.
6. Processing - declarative workflows. Solutions can include custom Codex workflows.
7. The platform stores Metrics, Events, Logs, and Traces (MELT) in distinct physical stores, unified by tags and metadata.
8. The Unified Query Engine (UQE) provides query federation across MELT stores. End users interact with the Unified Query Language (UQL). Cloud collectors and custom API endpoints may query the platform using UQL, as does the extensible user interface.
9. Custom REST and Google Remote Procedure Call (gRPC) APIs enable partners to push data the platform in proprietary formats that are transformed and ingested.
10. Custom Solution REST and gRPC APIs.
11. Objects that are used to configure extensible components.
12. All objects and configs must have a type definition residing in some Solution. Objects differ from configs because their type definitions allow high cardinality.
13. The user interface is built on top of an extensible templating system. Solutions can "enrich" the existing user interface by including UI templates.

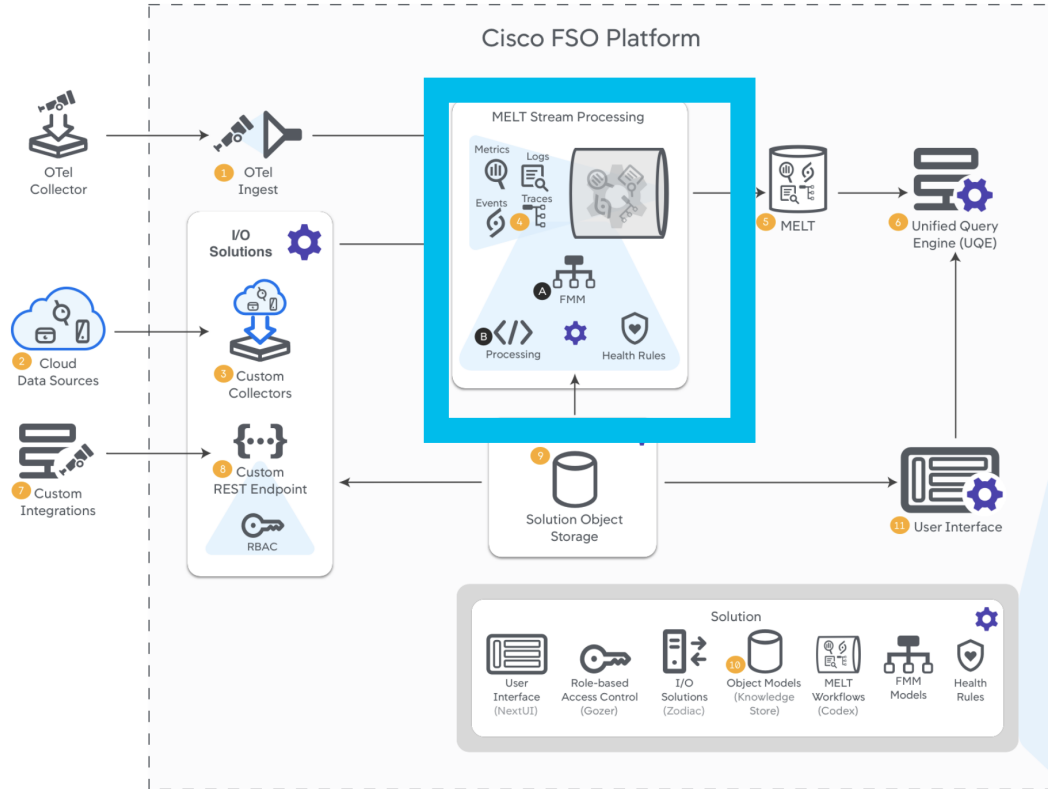


AppDynamics Cloud Solution Bundle

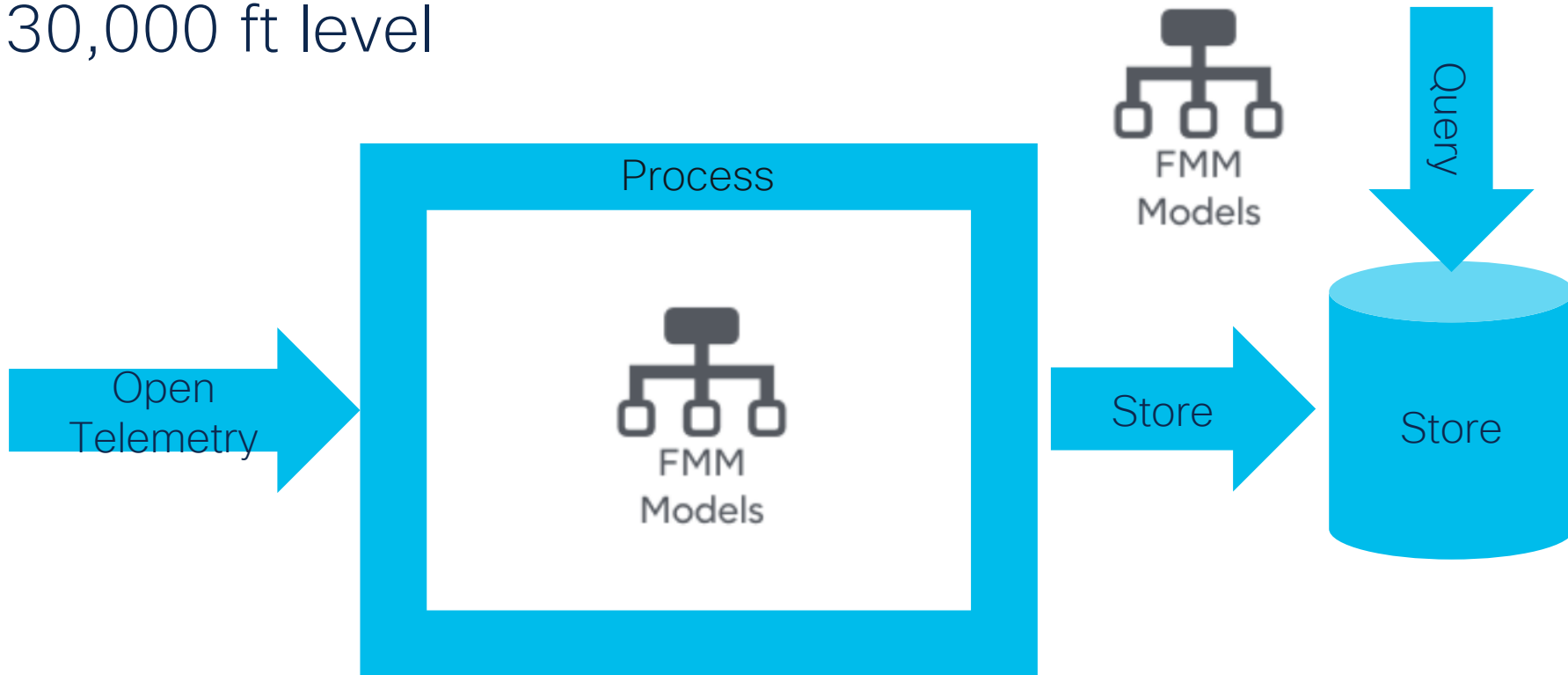


Data Pipeline

But how do I process MELT data?



30,000 ft level



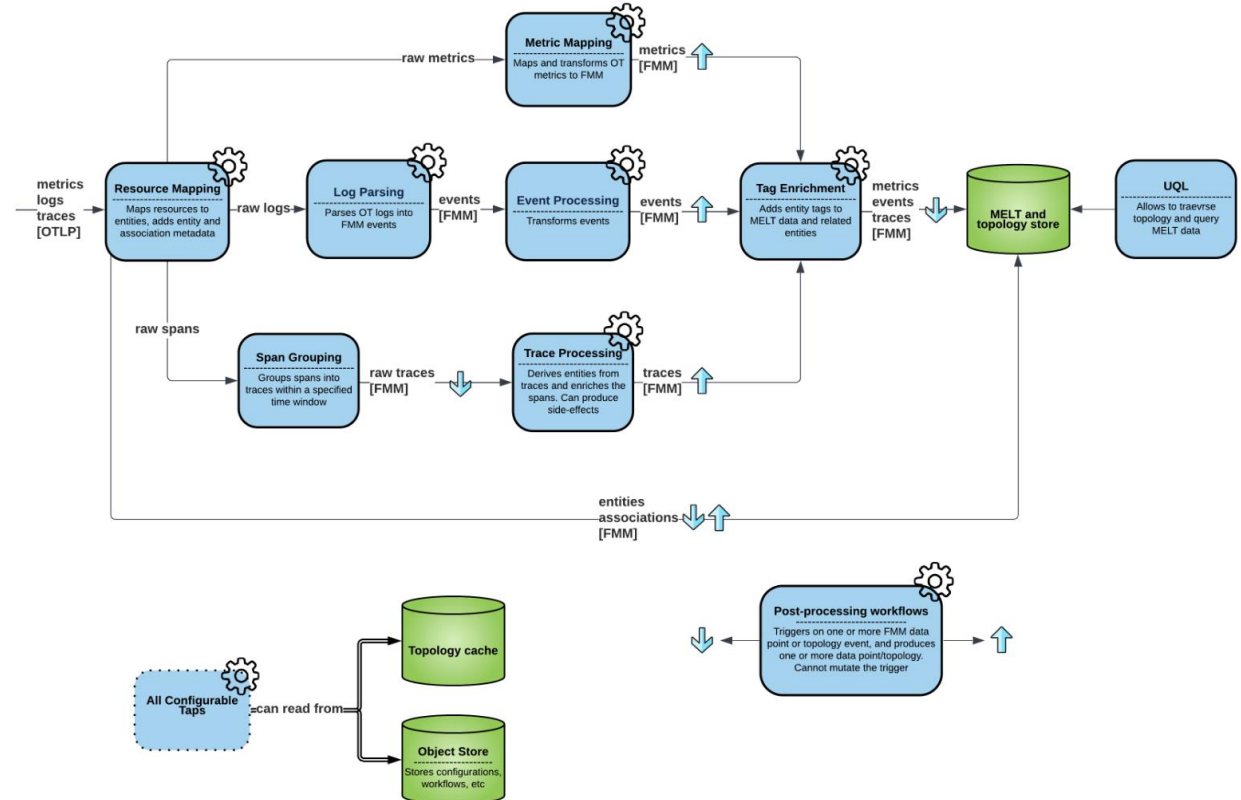
What is this FMM you speak of?

- “Flexible Metadata Models” ...or maybe “flexible MELT Modeling”?
- An Entity Centric MELT modeling framework
- Signals are anchored to entities
- Entities have complex relationships
- Example: Kubernetes Node Entity is one-to-many relationship to Pod Entity



Data Pipeline

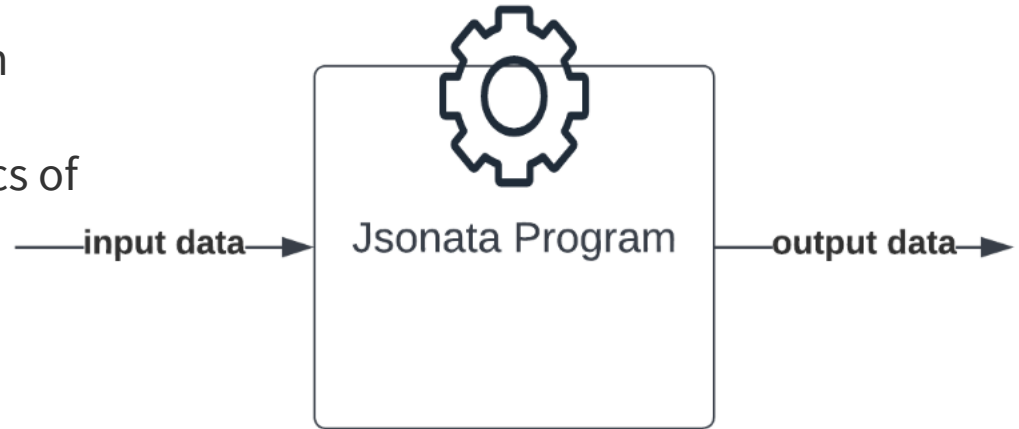
- Resource Mapping
- Log parsing
- Event processing
- Metric mapping
- Span grouping
- Trace processing
- Custom Workflows



Configurable Pipeline Processor

<https://jsonata.org/>

- Lightweight query and transformation language for JSON data
- Inspired by the *location path* semantics of XPath 3.1
- Sophisticated query expressions with minimal syntax
- Built in operators and functions for manipulating and combining data
- Create user-defined functions
- Format query results into any JSON output structure



- Try and share jsonata expressions
- <https://try.jsonata.org/ecxVAoT53>

The screenshot shows the try.jsonata.org website interface. The browser address bar displays <https://try.jsonata.org/ecxVAoT53>. The page title is "JSONata Exerciser".

JSON Input (Left Panel):

```

{
  "resource": {
    "attributes": {
      {
        "key": "container.name",
        "value": {
          "string_value": "ot-resource-mapper"
        }
      },
      {
        "key": "k8s.namespace.name",
        "value": {
          "string_value": "dppl"
        }
      },
      {
        "key": "k8s.pod.name",
        "value": {
          "string_value": "ot-resource-mapper-0"
        }
      },
      {
        "key": "k8s.cluster.name",
        "value": {
          "string_value": "pdx-lev-p01-k8s-a1"
        }
      },
      {
        "key": "service.name",
        "value": {
          "string_value": "ot-resource-mapper-service"
        }
      },
      {
        "key": "telemetry.sdk.name",
        "value": {
          "string_value": "k8s-agent"
        }
      }
    }
  },
  "scope_metrics": {
    {
      "metrics": [
        {
          "description": "Current CPU usage",
          "gauge": {
            "data_points": [
              {
                "as_double": 98.3,
                "time_unix_nano": "1672832772000000000"
              }
            ]
          }
        }
      ]
    }
  }
}

```

JSONata Expression (Center Panel):

```

scopeFilter:=iskey('k8s.cluster.name') != null,
source:= $source,
entityType:= "k8s:cluster",
attributes:= {
  "k8s.cluster.name": $key('k8s.cluster.name')
}
}

/* ===== Get primary and context entities ===== */
$primaryEntity := $getPrimaryEntity($k8sEntityPriorities, $fmmEntities);
$contextEntities := $getContextEntities($primaryEntity, $fmmEntities);

/* ===== Output packet ===== */
{
  "entity": $primaryEntity,
  "contextEntities": $contextEntities,
  "metrics": $mapMetrics(scope_metrics),
  "logs": $mapLogs(scope_logs),
  "spans": $mapSpans(scope_spans)
}
}

```

JSON Output (Right Panel):

```

{
  "entity": {
    "id": "W0JnYaghs6eNB2Iy",
    "type": "k8s:pod",
    "source": "k8s-agent",
    "attributes": {
      "k8s.pod.name": "ot-resource-mapper-0",
      "k8s.namespace.name": "dppl",
      "k8s.cluster.name": "pdx-lev-p01-k8s-a1"
    },
    "tags": []
  },
  "contextEntities": [
    {
      "id": "01KoC7zehk7GRRM",
      "type": "k8s:namespace",
      "source": "k8s-agent",
      "attributes": {
        "k8s.namespace.name": "dppl",
        "k8s.cluster.name": "pdx-lev-p01-k8s-a1"
      },
      "tags": []
    },
    {
      "id": "z00celhuvcm6qPKJ",
      "type": "k8s:telemetry.sdk",
      "source": "k8s-agent",
      "attributes": {
        "k8s.telemetry.sdk.name": "k8s-agent"
      },
      "tags": []
    }
  ],
  "metrics": [
    {
      "description": "Current CPU usage",
      "gauge": {
        "data_points": [
          {
            "as_double": 98.3,
            "time_unix_nano": "1672832772000000000"
          }
        ]
      }
    }
  ]
}

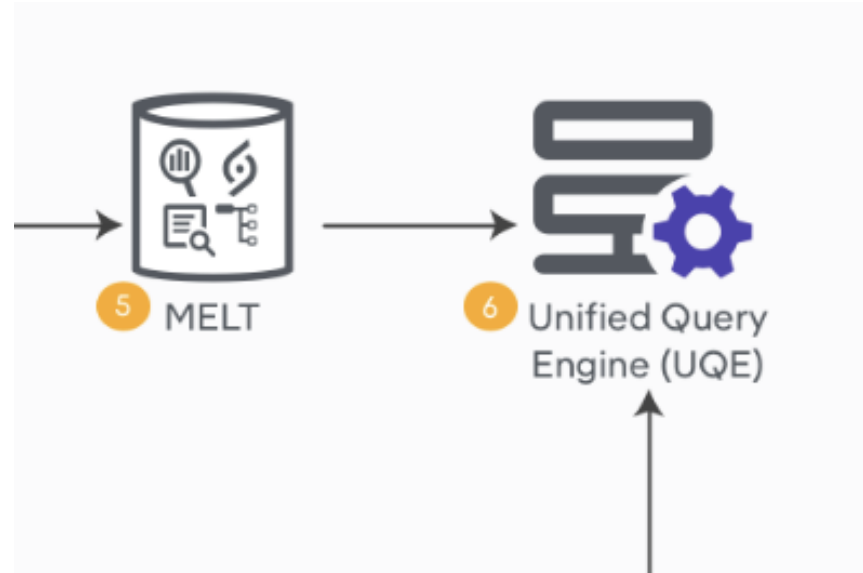
```

CISCO *Live!*



Unified Query Language

- It is a domain-specific language for the AppDynamics Cloud MELT (metrics, events, logs, traces) model.
- It is a declarative language
- It is a data query language
- It is read only
- It presents MELT data and State in the scope of monitored topology



Navigating Topology with UQL

```
fetch
  n.attributes(node.name),
  p.attributes(pod.name)
from
  n: entities(k8s:node),
  p?: n.out.to(k8s:pod)[attributes(pod.name) in ['pod123', 'pod456']]
```

Time

```
since -1h until 2021-02-15T16:30:00Z  
since 2021-02-15T16:30:00Z until +PT1H  
since -1h until now - 1h
```

Retrieving metrics with UQL

```
FETCH
  serviceId: id,
  metrics(calls_per_minute, infra_source)
    [attributes(region) IN ['pdx', 'fra']]
    {timestamp, value, max}
FROM
  entities(service_type)
LIMITS
  metrics.granularityDuration(PT5M)
```

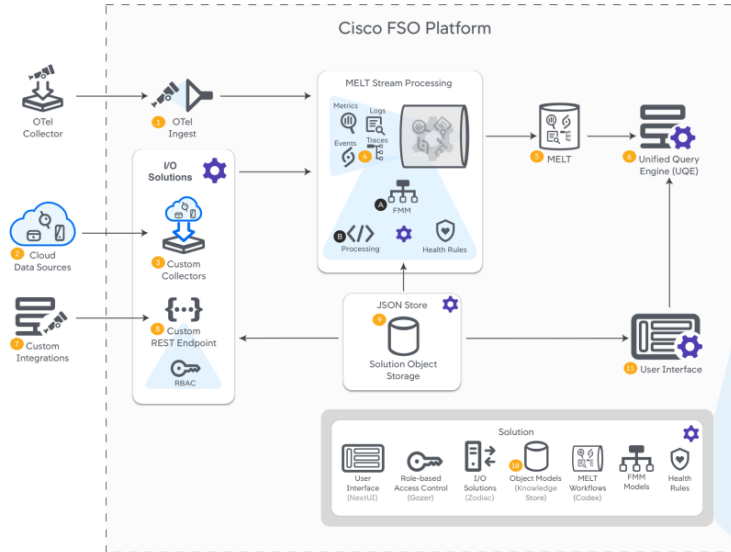
Logs/Events

```
FETCH
  id,
  events(logs)
    [attributes(severity) IN ['WARN', 'ERROR']]
    {timestamp, raw, attributes, eventFields}
FROM
  entities(entity.type)
```

Traces

```
FETCH
  traces
  [
    numErrors > 3
    && duration > 'PT1M'
    && originEntityIds(service) !IN ['service:abc', 'service:xyz']
    && entityIds >= ['service:1', 'service:2']
    && name ~ 'HTTP*'
    && source != 'apm'
  ] {
    traceId,
    duration,
    name,
    originEntityIds(service_instance)
  }
FROM
  entities(service)
```

Zooming Back Out

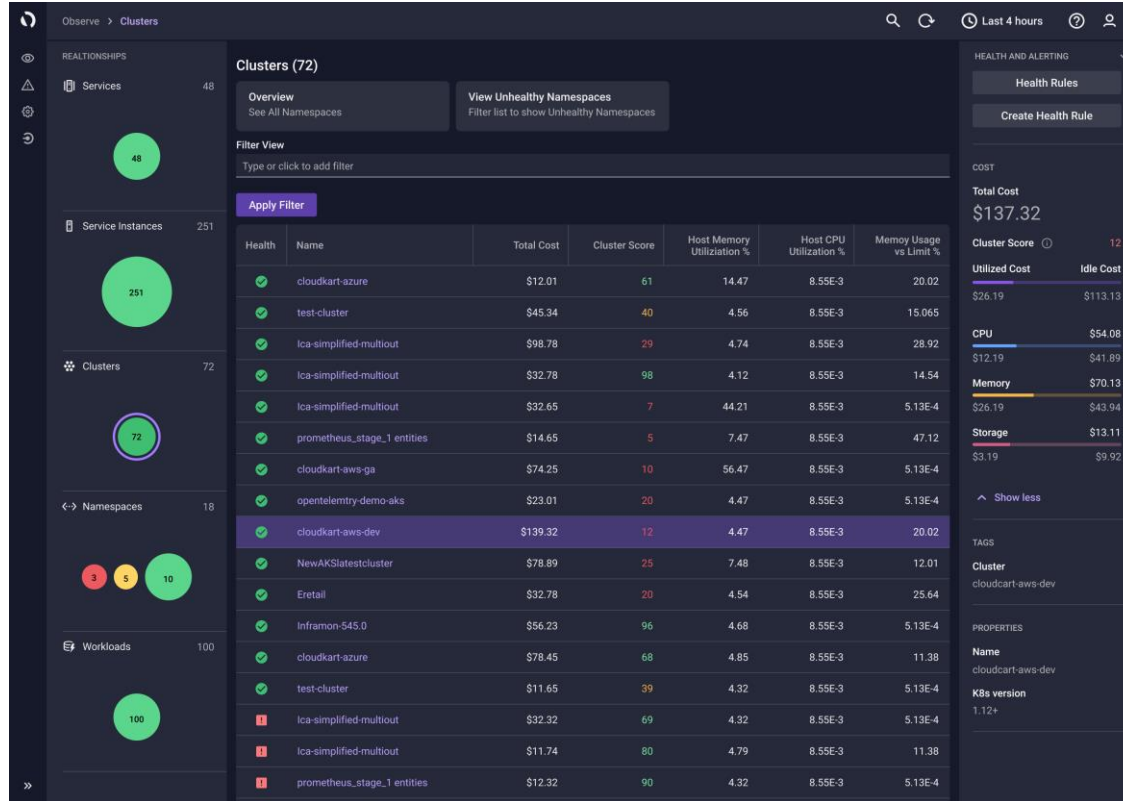


Key

- 1 OpenTelemetry Ingest
- 2 APIs providing telemetry or other data that can be transformed to telemetry. For example, AWS CloudWatch metrics.
- 3 Container images that plug into a framework for cloud data collection. Solutions can include custom Custom Collectors.
- 4 Various rules and transformations can be applied as data moves through the system.
- 5 FMM - the Flexible Metadata Model (FMM) allows observed system topology modeling. Solutions can include custom FMM models.
- 6 Processing - declarative workflows. Solutions can include custom Codex workflows.
- 7 The platform stores Metrics, Events, Logs, and Traces (MELT) in distinct physical stores, unified by tags and metadata.
- 8 The Unified Query Engine (UQE) provides query federation across MELT stores. End users interact with the Unified Query Language (UQL). Cloud collectors and custom API endpoints may query the platform using UQL, as does the extensible user interface.
- 9 Custom REST and Google Remote Procedure Call (gRPC) APIs enable partners to push data to the platform in proprietary formats that are transformed and ingested.
- 10 Custom Solution REST and gRPC APIs.
- 11 Objects that are used to configure extensible components.
- 12 All objects and configs must have a type definition residing in some Solution. Objects differ from configs because their type definitions allow high cardinality.
- 13 The user interface is built on top of an extensible templating system. Solutions can "enrich" the existing user interface by including UI templates.



Appd Cloud ... Built on Cisco FSO Platform



Complete your Session Survey

- Please complete your session survey after each session. Your feedback is important.
- Complete a minimum of 4 session surveys and the Overall Conference survey (open from Thursday) to receive your Cisco Live t-shirt.
- All surveys can be taken in the Cisco Events Mobile App or by logging in to the Session Catalog and clicking the "Attendee Dashboard" at <https://www.ciscolive.com/emea/learn/sessions/session-catalog.html>



Continue Your Education



Visit the Cisco Showcase for related demos.



Book your one-on-one Meet the Engineer meeting.



Attend any of the related sessions at the DevNet, Capture the Flag, and Walk-in Labs zones.



Visit the On-Demand Library for more sessions at ciscolive.com/on-demand.



The bridge to possible

Thank you

CISCO *Live!*

CISCO *Live!*

ALL IN