QCon LONDON

# Events:

## Design, Catalogue, Discover and Use

Tom Fairbairn

March 2nd, 2020

solace

# Agenda

QCon LONDON

- Events & Event Driven Architectures
- Events as Assets
- Tools.  Or the lack of…
- Approach & Architecture
- Demo





solace

# Quiz

QCon LONDON

**Who:**

- **Has deployed an EDA?**
- **Is developing one?**
- **Is planning one?**
- **Has no idea about events?**
- **Wants to hear me drone on about why they're a good idea?**





solace

# What is an “event?”

- **A piece of data**
- **About something that’s happened**

  Usually a *change*

**Characteristics**

- **Has a source**
- **Has one or more sinks… we hope**
- **Relevant only in a *context***
- **Temporal (= real time?)**
- **Asynchronous**

# Event myths

- **New fangled**
- ***Only* asynchronous**
  - API design
  - Must have asynchronous programming model
- **The schema is the API/contract**
- **Can be implemented with REST point to point**

# BOLD STATEMENT

To be useful, deployable and manageable, *most* event driven architectures are:

- Distributed
- Micro–service orientated
- Asynchronous
- Use the Publish/Subscribe pattern
- **Make use of a *centralised,* distributed event distribution layer**

# BOLD STATEMENT

# Quiz

- **How did you felt about event driven when you first started to use it?**
- **Why?**

# Event non-myths

- **Difficult/hard to**
  - Understand
  - Govern
  - Monitor/Manage
  - Lack of orchestration (choreography) makes tracing, tracking and debug hard
- **“Pain in the behind’**
- **“Annoying”**
- **“Disruptive”**

# Quiz

QCon LONDON

- What event driven technologies/tools are you using?
- Do you have your own tooling?





solace

# It’s Hard to Build Event Driven Applications

QCon LONDON

**Where** do you discover events for reuse?

**Why** does an event exist?

**What** topic do you use to subscribe to it?

**How** do you determine the data structure of its payload?

**Who** should have access to it?

**Who** can tell you more about it?

**Is** your change backward compatible?

**Who** is impacted by your change?

**Does** it comply with security policy?

solace

# Event Assets

QCon LONDON

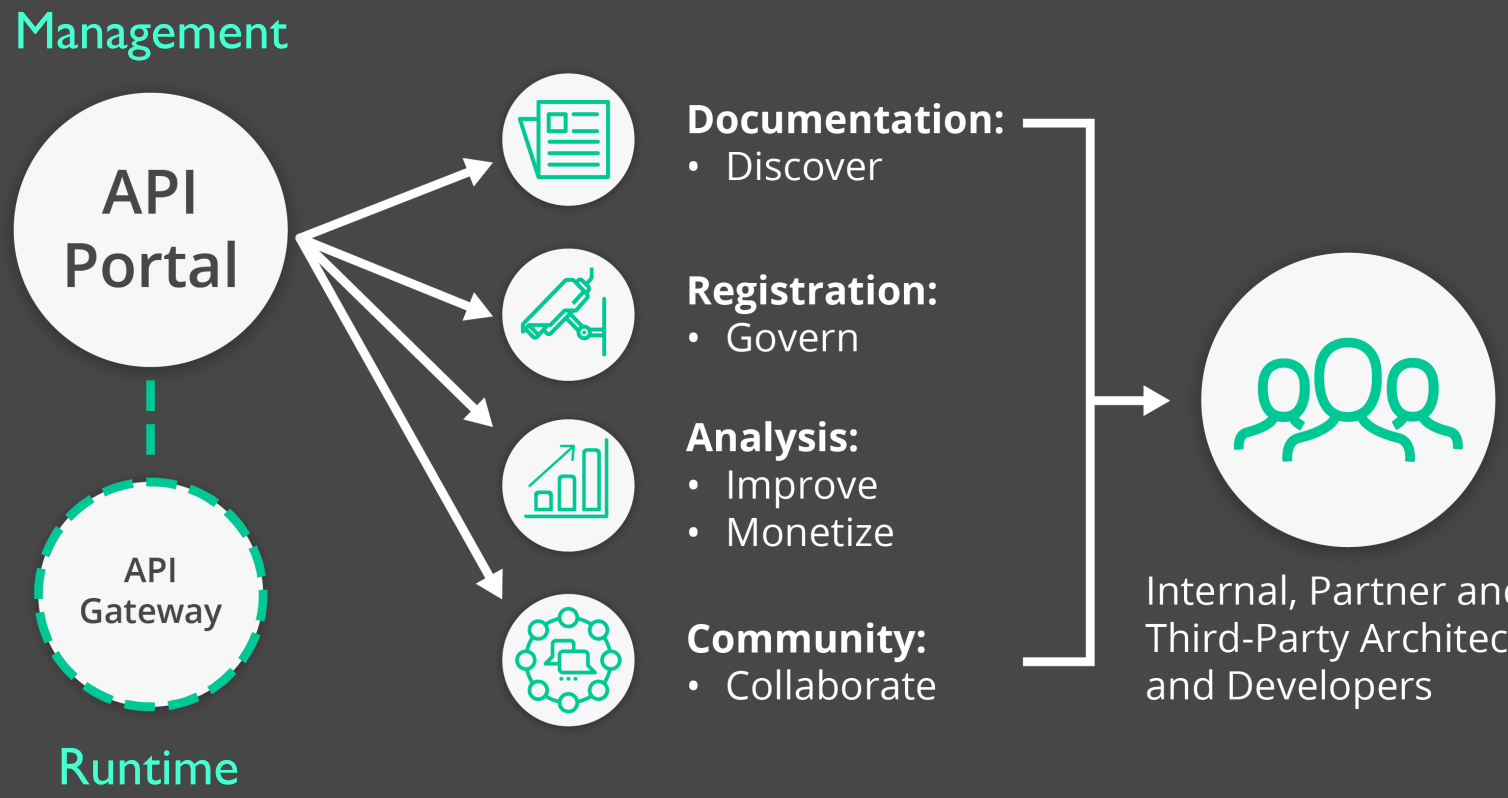
**Events form *part* of your event assets**

- **Context**
- **Documentation**
- **Skills**
- **Applications**
- **Data**
- **Systems**

solace

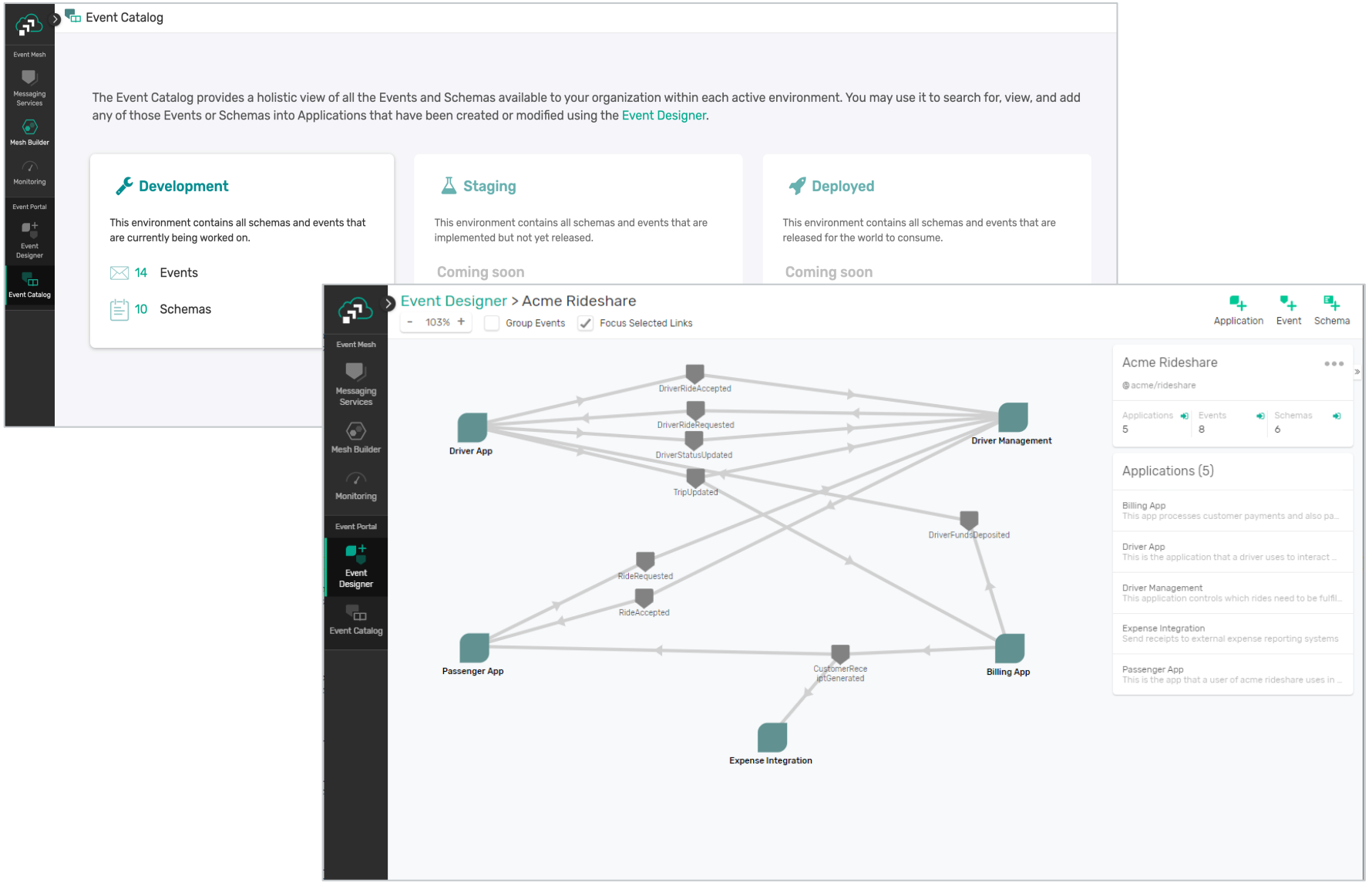
# Do we have a precedent?

**API Management platforms answer *who, what, when, where, why, how* for RESTful APIs**



# PubSub+ Event Portal

**Single place to design, create, discover, share, secure and manage your event assets**



# The Elements of Event Portal

## Application Domain

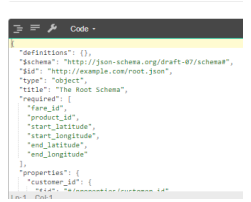
Team, LOB, process
(i.e. HR, Inventory, Billing)

## Application

A Publisher
and/or Subscriber

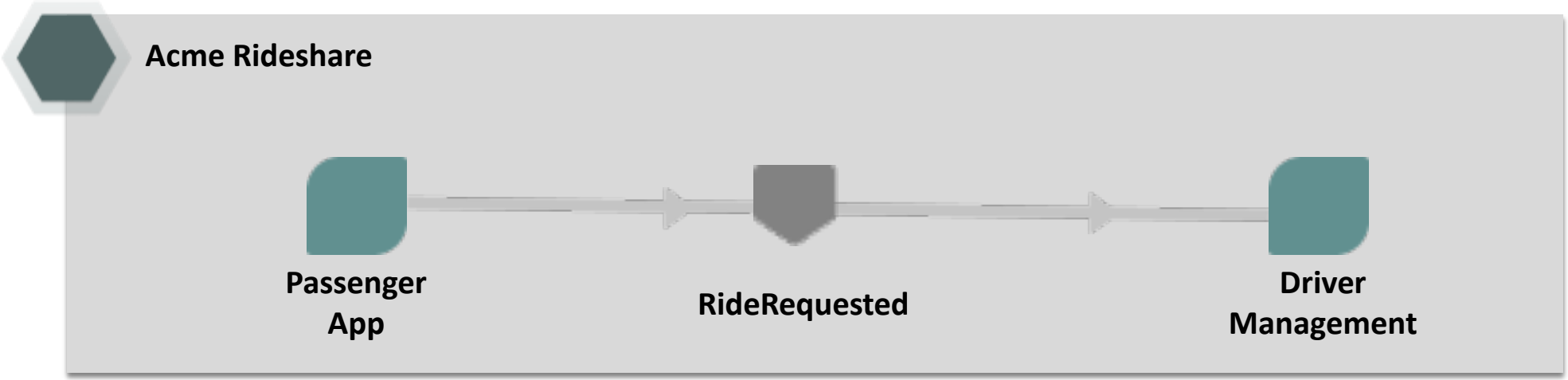
## Event

Topic address + metadata that
references a payload schema



## Schema

Payload object definition
JSON, Text, Binary, XML, Avro



# The Demo!

QCon LONDON





solace

# Design Principles

QCon LONDON

**Independent of:**

- **Architecture**
- **Event Distribution mechanism/tools**
- **Language**
- **Protocol**
- **Schema/serialisation**





solace

# Warning!

QCon LONDON

- **Early Days**
- **We want your feedback**





solace

# Specify and Generate Async Applications

QCon LONDON

| Management | Spec | Codegen | Generated Code | Runtime Protocol |
|---|---|---|---|---|

## Synchronous

API Mgmt → OPENAPI INITIATIVE → Swagger → Generated Code → Runtime Protocol

```
Available Clients: [ akka-scala,
  android, async-scala,clojure,cpprest,csharp,CsharpDotNet2,
  cwiki,dart,dynamic-html,flash,go,groovy,html,
  html2,java,javascript,javascript-closure-angular,
  jaxrs-cxf-client,jmeter,objc,perl,php,python,
  qt5cpp,ruby,scala,swagger,swagger-yaml,swift,
  swift3,tizen,typescript-angular,typescript-angular2,
  typescript-fetch,typescript-node],
```

```
Available Servers: [ aspnet5,aspnetcore,
  erlang-server,go-server,haskell,inflector,
  jaxrs,jaxrs-cxf,jaxrs-cxf-cdi,jaxrs-resteasy,
  "jaxrs-spec","lumen","msf4j","nancyfx","nodejs-server",
  python-flask,rails5,scalatra,silex-PHP, sinatra,
  slim,spring,undertow]
```

HTTP

**Open Standard for asynchronous API definition**

**https://www.asyncapi.com/**

**"OpenAPI or RAML for asynchronous design"**

## Asynchronous

Event Portal → AsyncAPI → AsyncAPI → node, Java, … → AMQP, AMQPS, MQTT, MQTTS, WebSocket, WSS





solace

# The Demo! Discovery

# Complete Event Lifecycle Management

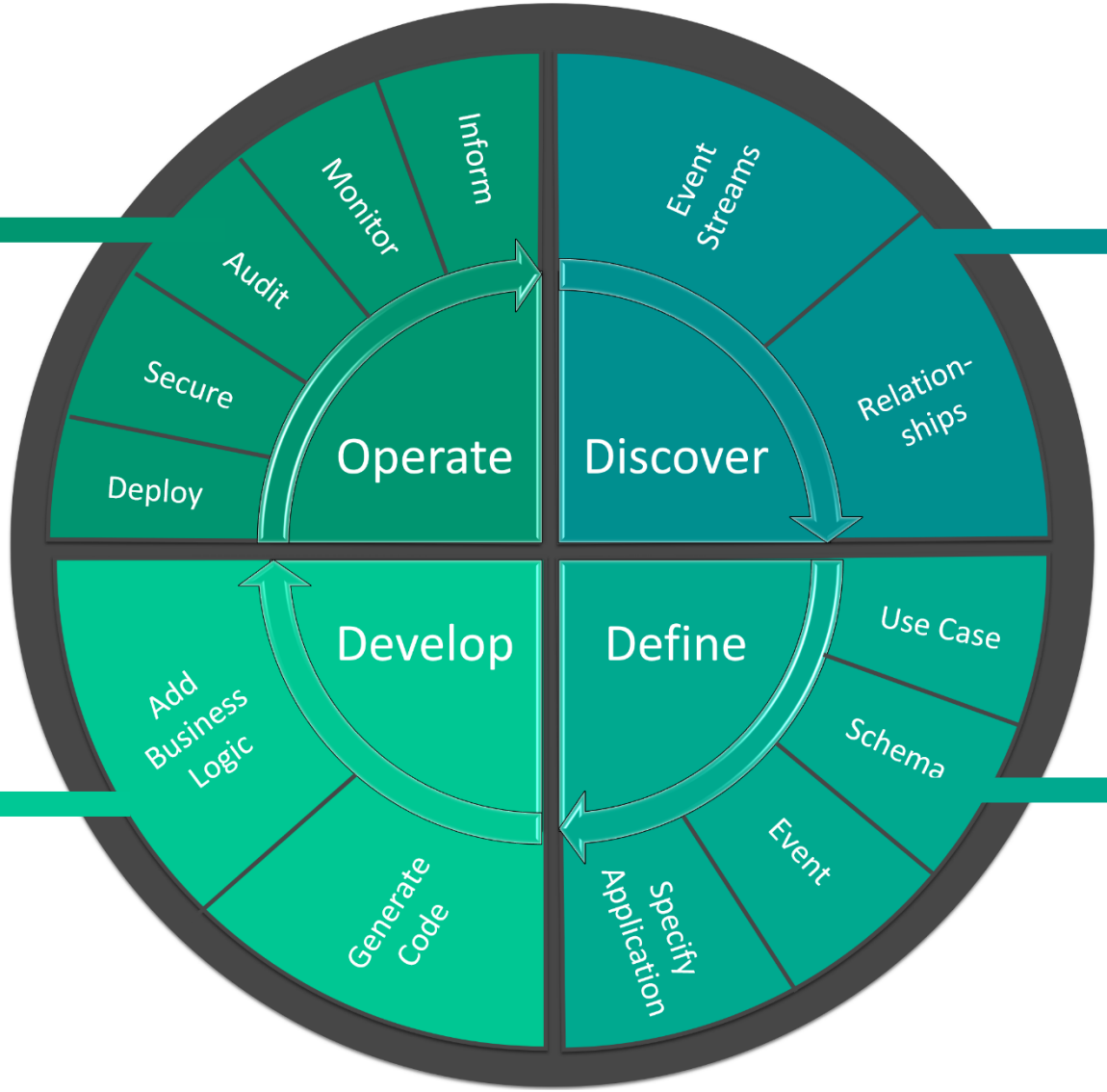
QCon LONDON

## Operate

- **Deploy:** start and connect to the broker
- **Secure**: enforce authentication and authorization
- **Audit:** identify runtime vs design time violations
- **Monitor:** understand utilization of events, schemas and apps
- **Inform**: discern insights to enhance apps, events and schemas; deprecate events that aren't being utilized

## Develop

- **Generate Code**: use code generators to create application scaffolding
- **Add Business Logic:** create apps that perform required business function




## Discover

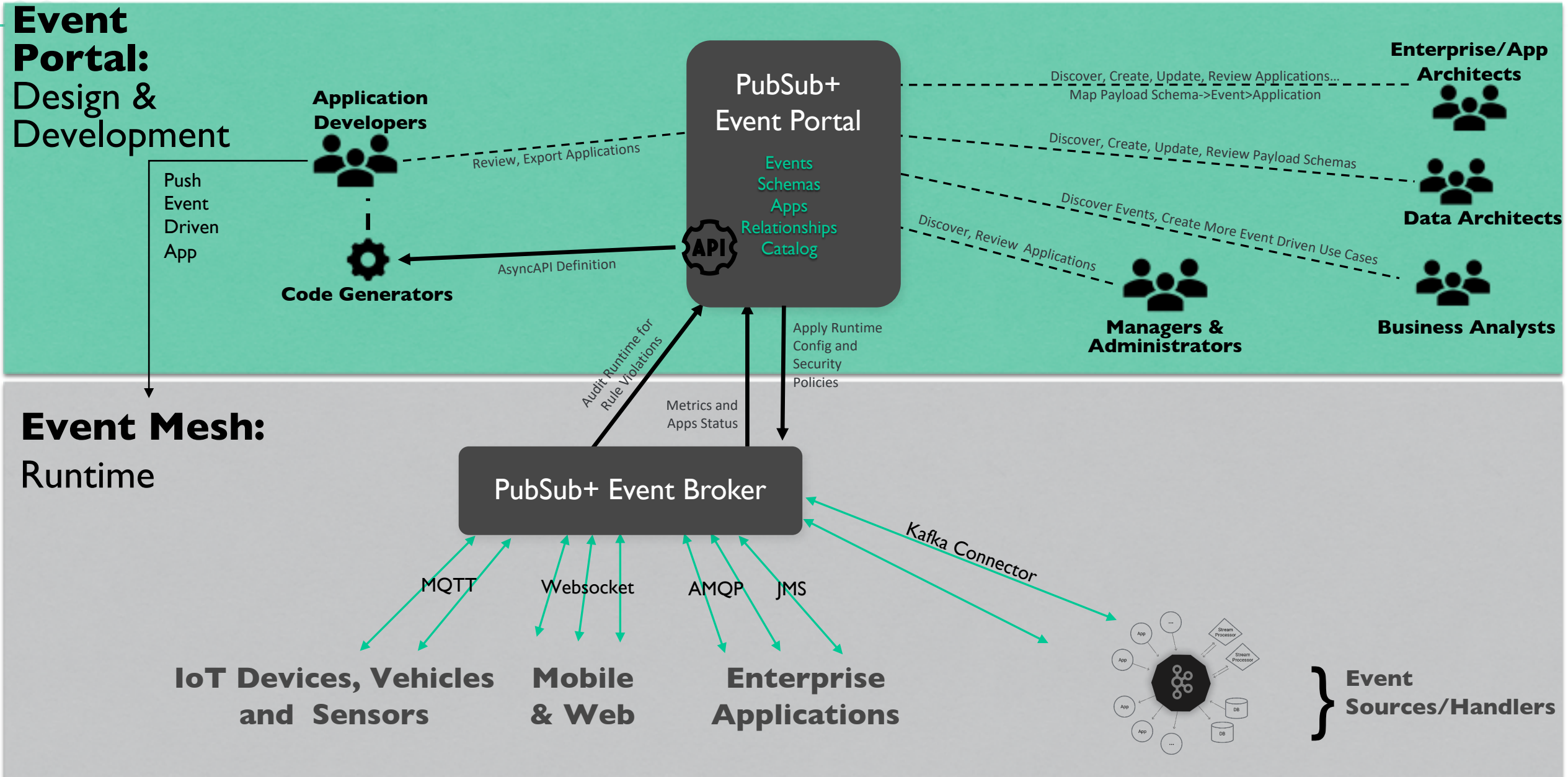
- **Event Streams:** Search and locate events which are of interest
- **Relationships:** Understand the relationship between events, their sources and who is consuming them

## Define

- **Use Case:** identify business outcome that can be improved by being event driven
- **Schema:** create business objects that represent the event context
- **Event:** create metadata that includes addressing (topic) and headers
- **Specify Application:** link events as inputs/outputs to the business logic

solace

# Unifying Design Time and Run Time



# solace.

That’s Possible