WSO2CON2024

YELLOW ROOM

WSO2CON 2024

Mastering API Lifecycle: Design, Deployment, and Effective Consumption



Sanjeewa Malalgoda
Director - Engineering
WSO2



Thilini Shanika
Senior Technical Lead
WSO2

Agenda

- Introduction to WSO2 APIM
- Component Architecture
- API Design
- API Security
- API Policies
- Rate Limiting
- API Consumption
- API Versioning and Lifecycle Management
- API Governance
- Monitoring and Analytics
- Monetization
- CI/CD for API Management
- Beyond REST API Management
- Service Catalog
- Q&A

Why API Management?

- Enhanced Security
- Improved Scalability
- Streamlined Development
- Deeper Insights and Analytics
- Increased Developer Productivity



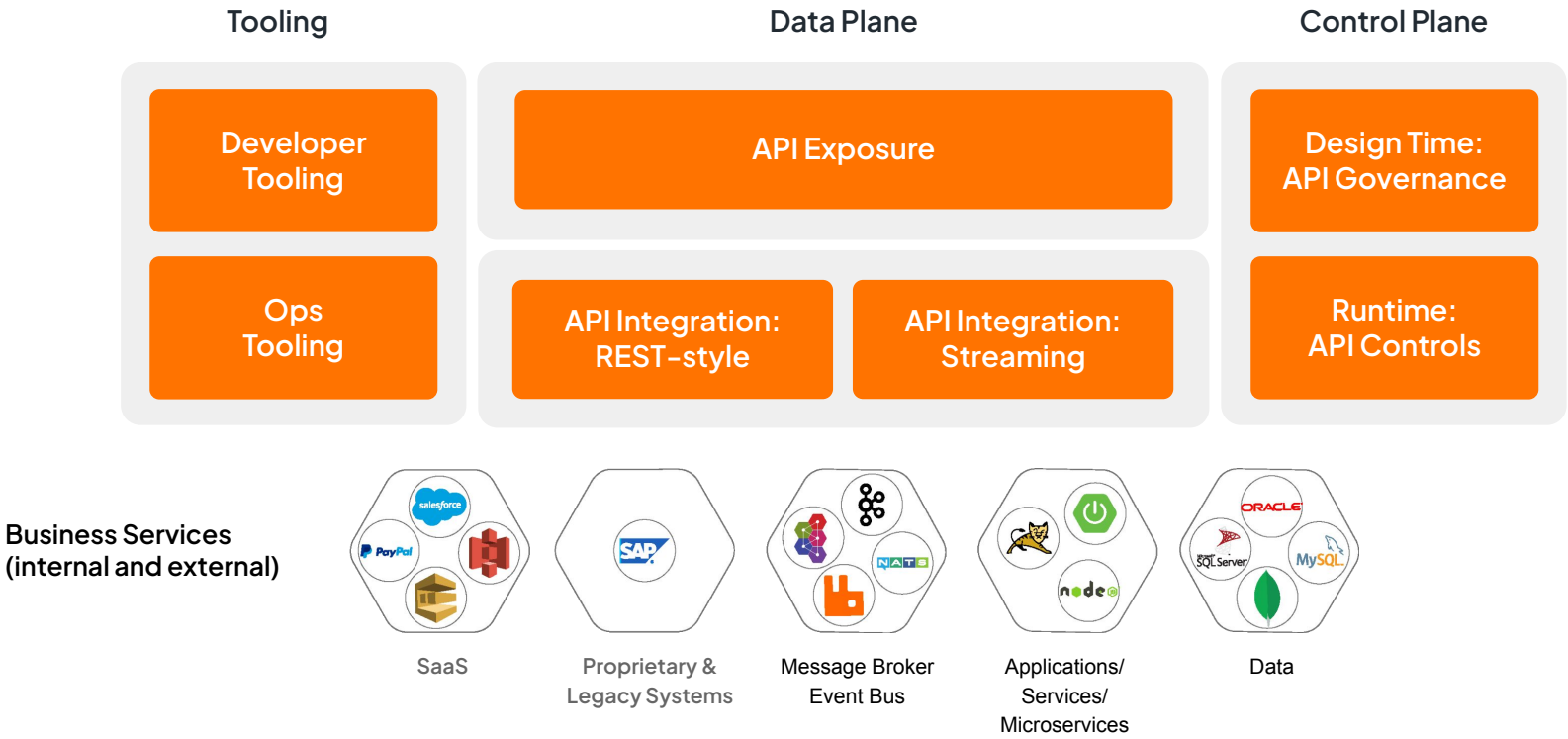
Introduction to WSO2 API Manager

WSO2 API Manager

- 100% open source
- Full Lifecycle API management
- Secure API Management: Secure APIs, enforce authorization rules and access policies
- API Discovery and Governance: Portals for API Design and Consumption, Approval workflows for API publishing and API subscription
- API Monetization
- Insightful Analytics and Observability
- Customisable Portals, Workflows and Extension points

Component Architecture

API Manager – Overview



API Manager - Components

Tooling

- API Controller
- Integration Studio
- Streaming API Editor

Data Plane

Regular Gateway

- REST
- SOAP
- GraphQL
- WebSockets
- WebHooks
- SSE

APK Gateway

- REST
- GraphQL

API Integration

- API Connectors
- Data-Source Connectors
- Streaming Connectors
- Integration Patterns
- Protocol Converters
- Data Mapping /Transformation

Control Plane

- Developer Portal
- API Publisher
- Service Catalog
- Business Insights
- Rate Limits
- Key Management
- AI Capabilities

Business Services (internal and external)



SaaS | Proprietary & Legacy Systems | Message Broker Event Bus | Applications/ Services/ Microservices | Data

Control Plane

Publisher

- Centrally Manage API Policies
- Access Control Configurations
- Lifecycle Management
- API Governance

Developer Portal

- API Marketplace
- Application Management
- Consumer Collaboration



Traffic Manager

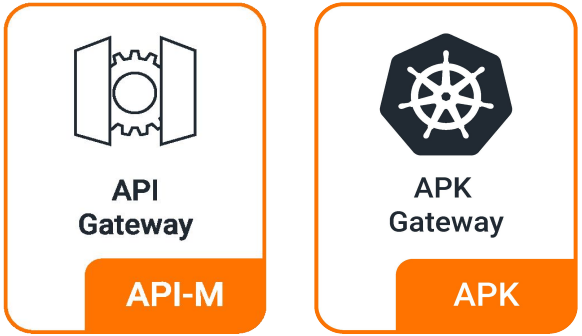
- Rate Limiting
- Distributed traffic policy evaluation

Key Manager

- Authentication/Authorization
- Token Management
- Identity Provider Integration

Data Plane

- Centralized API runtime
- Security/Rate limiting Policy Enforcement
- Protocol Switching
- Mediation
- Integrations



Architecture Flexibility

Centralized

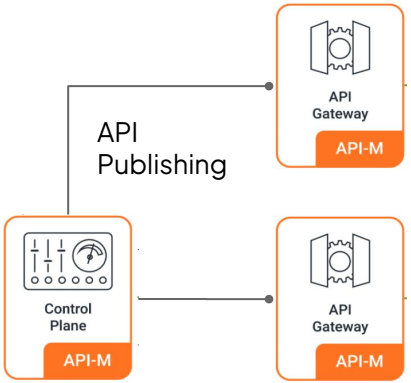


Gateway &
Control Plane

Easy to set up.

Ideal for Centralized and Monolithic Architectures.

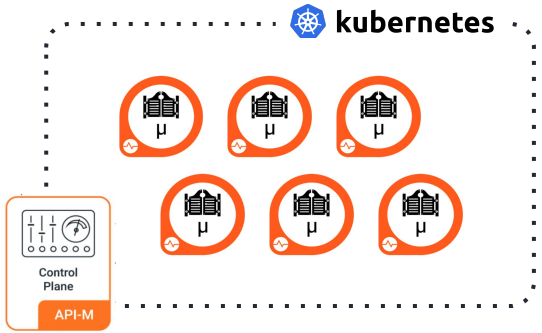
Multiple Gateways



More Flexible.

Gateways per Region, Use Case, Business Unit, etc.

Microservices Gateway Strategy



Highly Scalable.

Idealized for decentralized and Microservices architectures.

Product Portals

For API Developers & PMs



Publisher

API Configuration
Versioning and Lifecycle
Policy Management

For API Consumers



Devportal

API Catalog/Store
Tryout/Evaluation
Consumer Application Management

For Administrators



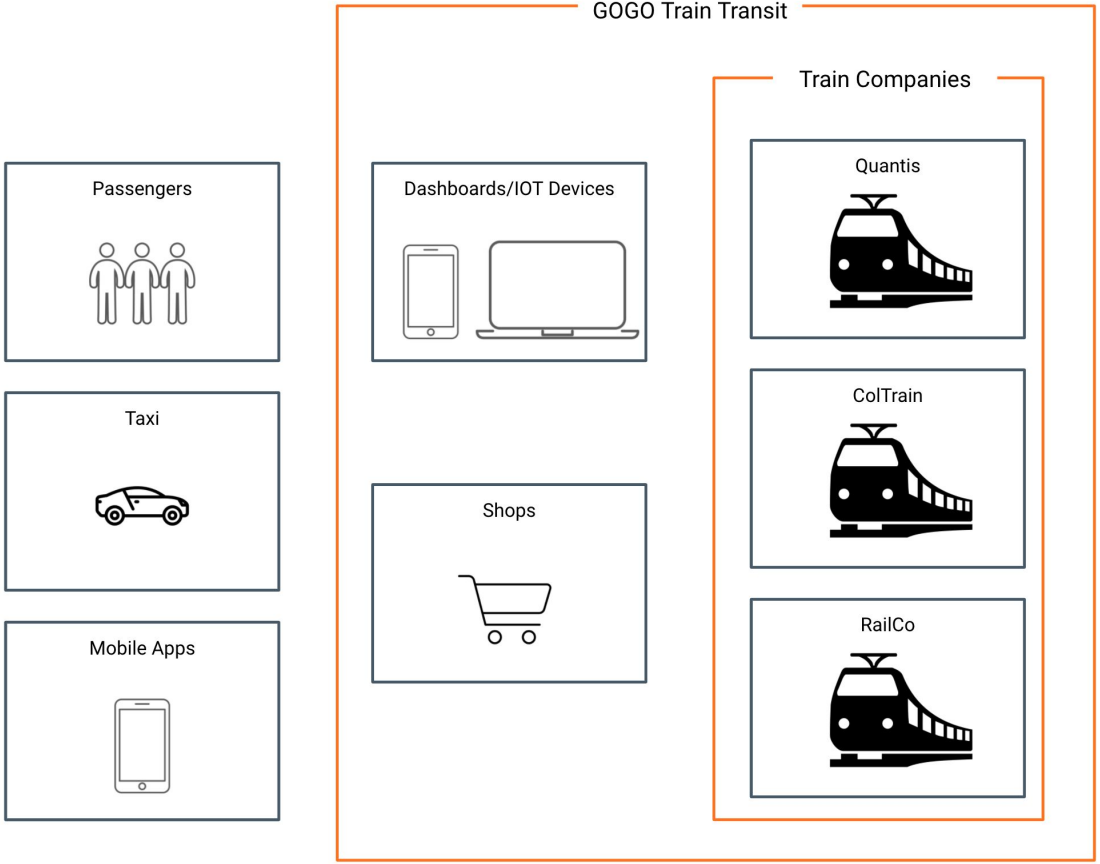
Admin Portals

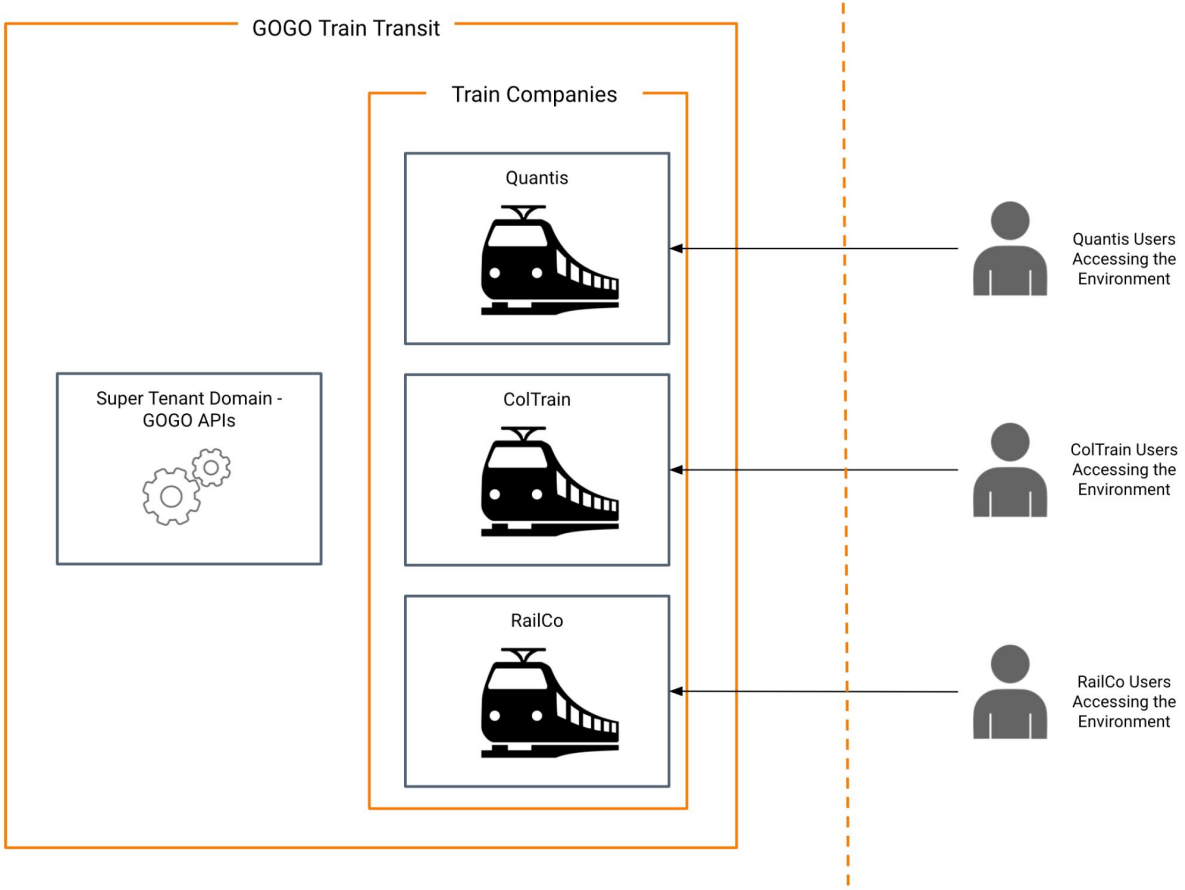
User/Role Management
Workflows
Approvals

API Design, Governance and Lifecycle Management

Business Use Case Overview

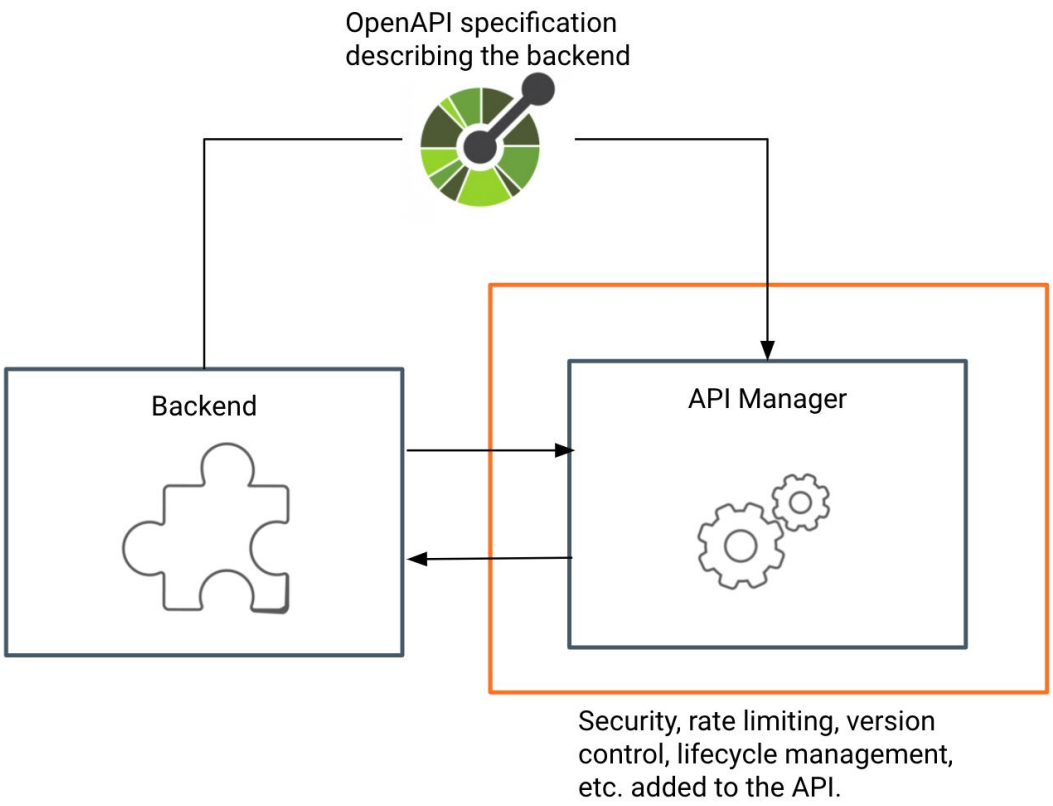
- **Union Station** is a bustling transportation center, which serves thousands of passengers daily.
- **GOGO Transit** owns and manages train sheds, platforms, and tracks.
- Union Station accommodates three train companies operating in the region. Each train company operates independently, managing its fleet of trains, schedules, and bookings.


GOGO Train Transit
Train Companies
Passengers
Taxi
Mobile Apps
Dashboards/IOT Devices
Shops
Quantis
ColTrain
RailCo


GOGO Train Transit
Train Companies
Quantis
ColTrain
RailCo
Super Tenant Domain -
GOGO APIs
Quantis Users
Accessing the
Environment
ColTrain Users
Accessing the
Environment
RailCo Users
Accessing the
Environment

Personas

Username	Role	Responsibility
Henry	API Owner (Internal/creator)	Designing and Implementing APIs
Paul	API Product Manager (Internal/publisher)	Manage API lifecycle, documentation, and ensuring quality and security standards
Tina	Partner App Developer/ (Internal/subscriber)	Discover, subscribe and utilize APIs
Ron	Station Manager/ (Internal/subscriber)	Manage train schedules, platforms and stations
Admin	System Admin/ (admin)	Manage users and system administration


OpenAPI specification
describing the backend
Backend
API Manager
Security, rate limiting, version
control, lifecycle management,
etc. added to the API.

API Design

API Design

- The API Publisher portal provides a variety of API design options
  - **API First** with built-in API graphical designer
  - Import existing API definitions
    - REST
      - OAS 3.1.0
      - OAS 3.0.0
      - Swagger 2.0
    - GraphQL
    - Async APIs
    - SOAP/SOAP to REST
  - **Integration First** with Service Catalog

API Security

API Security

Authentication

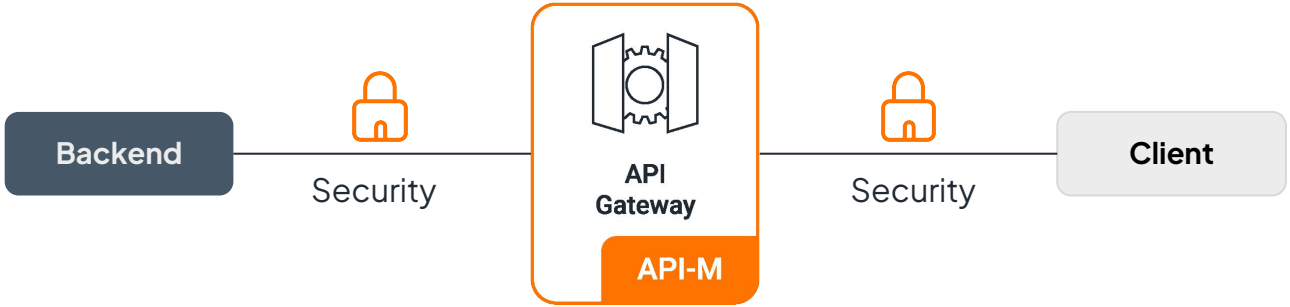
- Application Level
  - OIDC/OAuth 2.0
  - API Key
  - Basic Auth
- Transport Level
  - Mutual SSL

Authorization

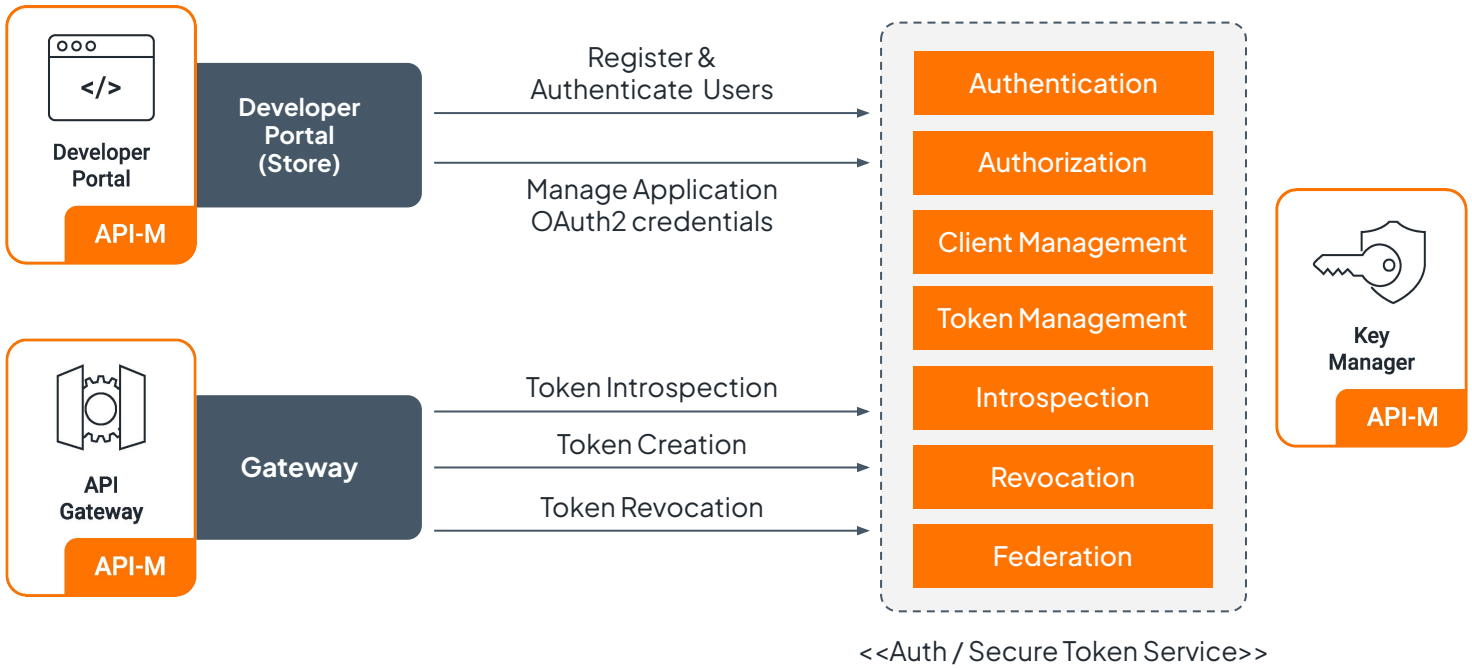
- Role Based Access Control with OAuth 2.0 Scopes
- OPA Policies

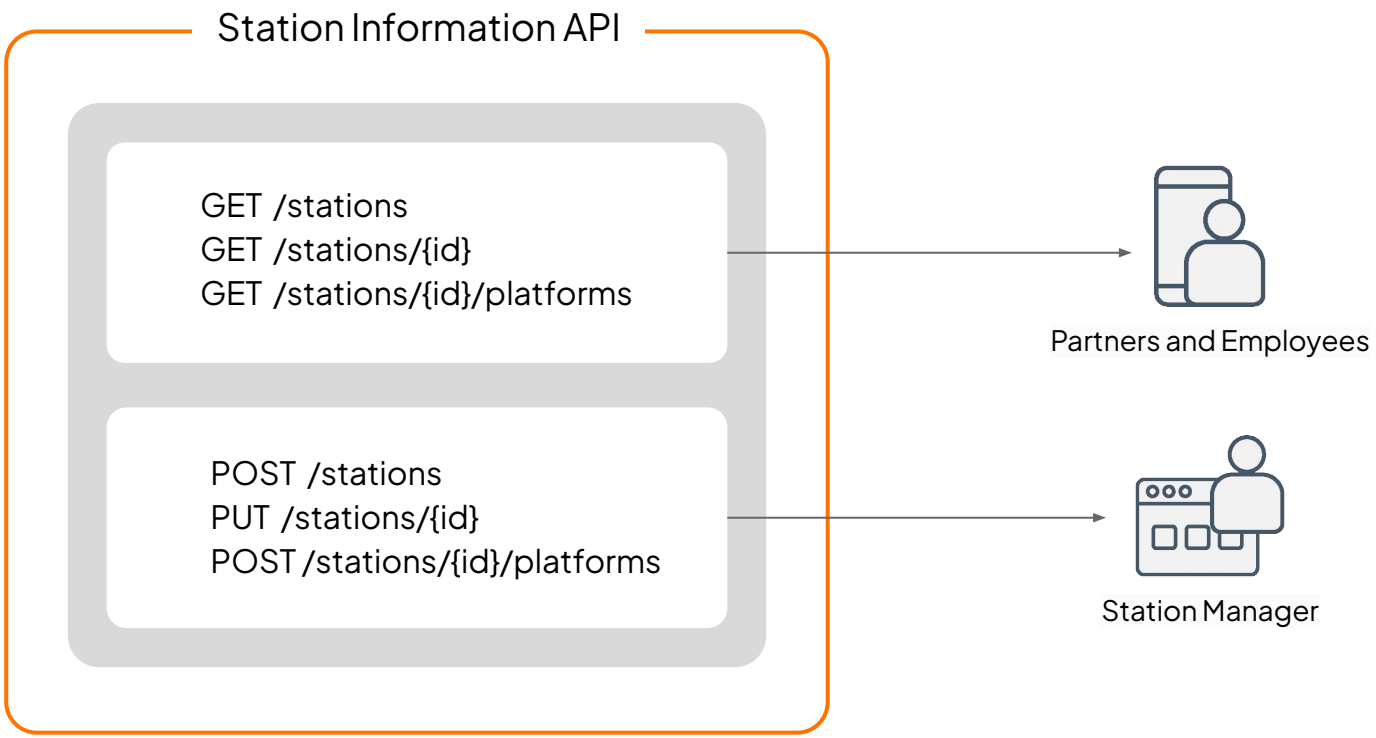
API Protection

- CORS
- Malicious Payload Scanning
  - Schema Validation
  - JSON/XML Threat Protectors
  - Injection Attack Detection



API Security – The Key Manager Component

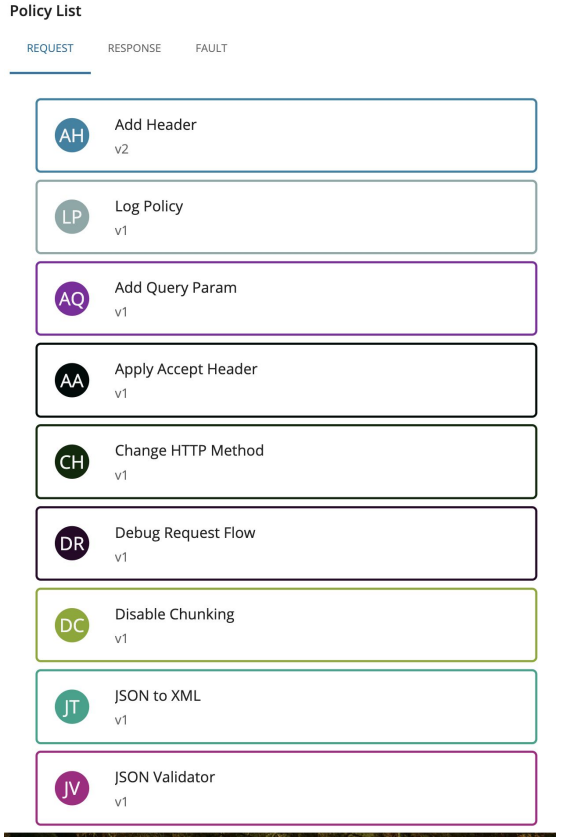



Station Information API
GET /stations
GET /stations/{id}
GET /stations/{id}/platforms
POST /stations
PUT /stations/{id}
POST /stations/{id}/platforms
Partners and Employees
Station Manager

API Policies

API Policies

- Applicable in different flows
  - Response
  - Request
  - Fault
- Applicable at different levels
  - API-wide
  - Per resource
  - Globally
- Mediation
  - Message Transformations/Rewrites
  - Header/Parameter Manipulations
  - Enrichments
  - Filtering/Validation
  - Debugging



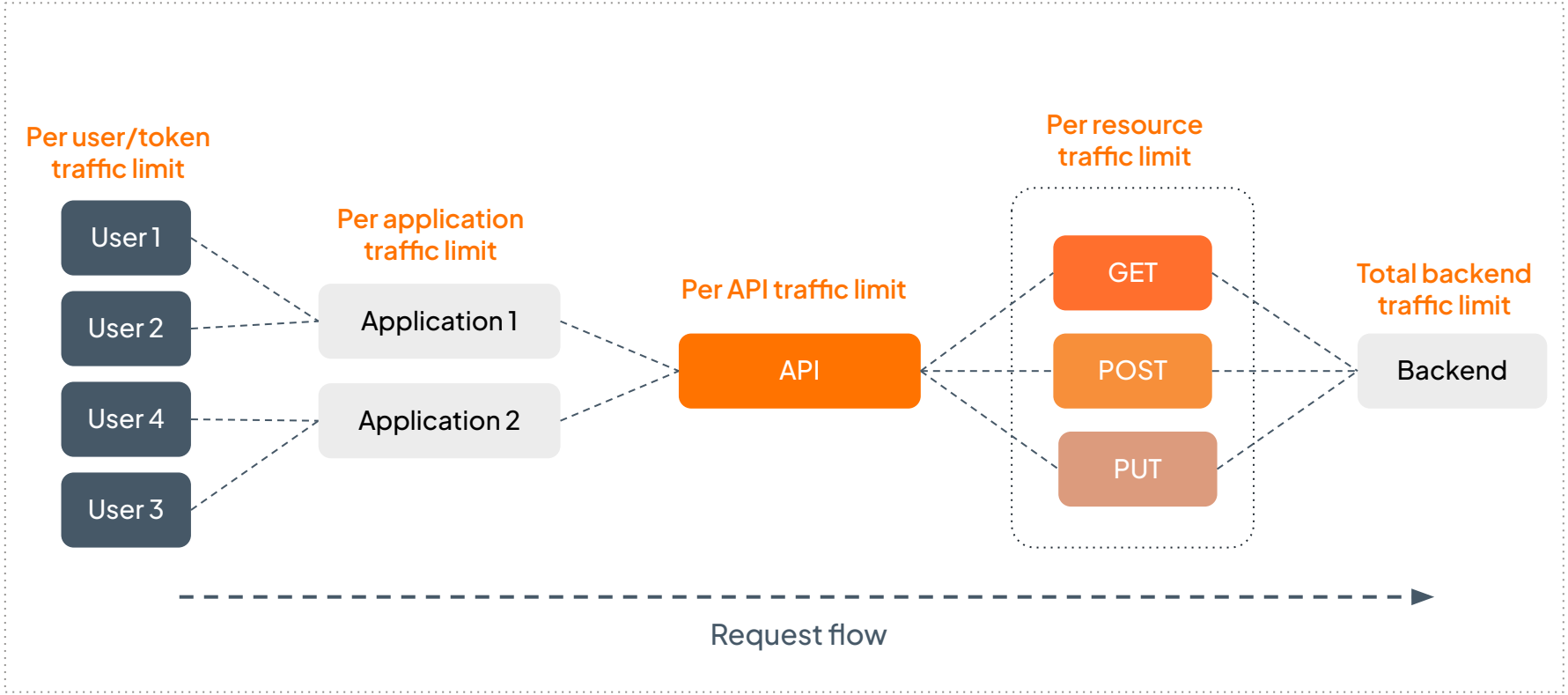
Use Case : API Mediation Policies

- Add custom header ‘X-Client-ID’ to request
- Transform GET /stations/{id} response



Rate Limiting

Traffic Management and Throttling on WSO2 API Manager



Rate Limiting Policies

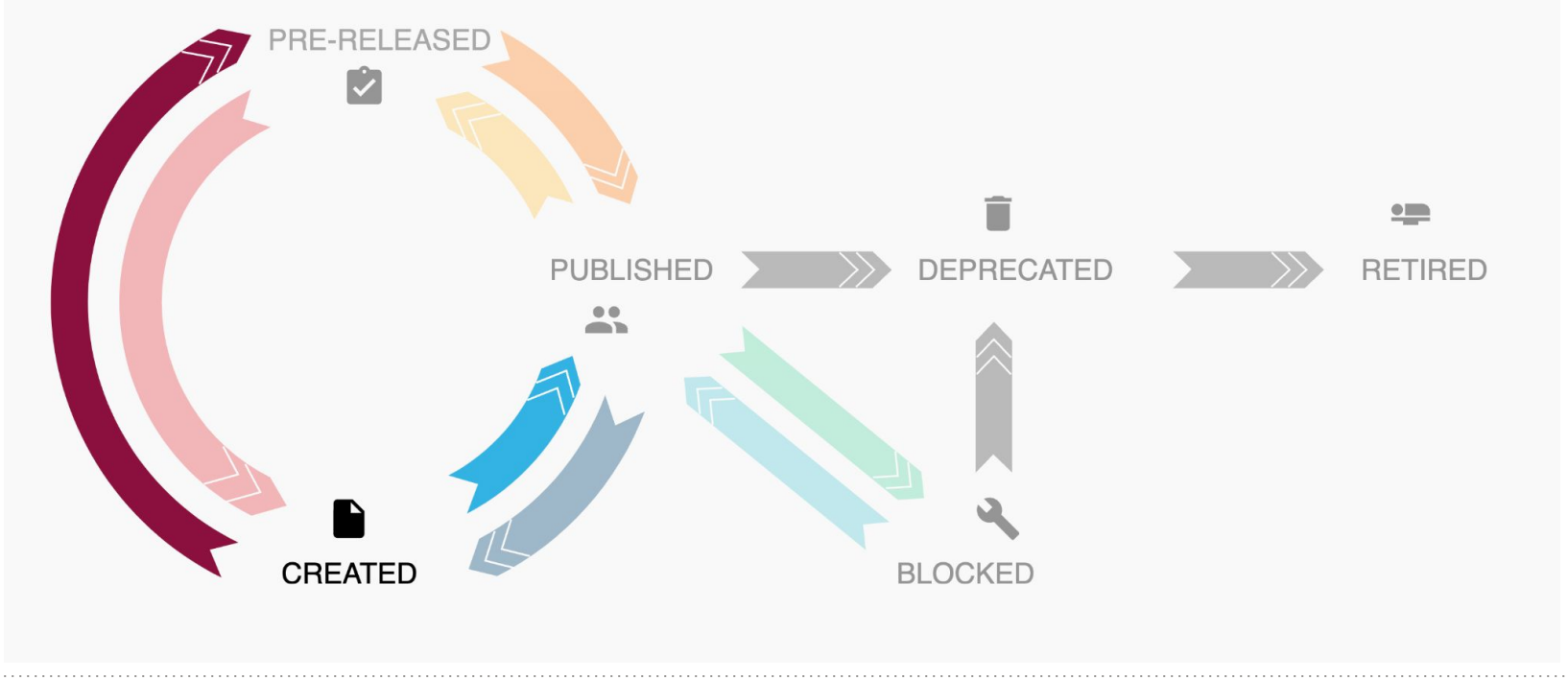
- Subscription Level (with burst control)
- Application Level (with burst control)
- Advanced Rate Limiting
  - API Level
  - Resource Level
  - Conditional rate limiting (IP/IP-Range/Header/JWT Claim)
- Deny Policy (API, IP, User, Application)
- Custom policies

API Deployment

API Deployment

- Makes the API available for invocation via a Gateway
- Revisions
- API developer testing console
- Multiple Gateways
  - External/Internal gateways
  - Multiple Environments

Lifecycle Management



API Consumption

API Developer Portal

- API Discovery
  - Marketplace Assistant - AI assisted search for Natural Language queries
  - API content search
  - Tags/API Categories
- User Self Signup
- Application Management
  - Subscriptions
  - Key Generation
  - Application Sharing
- API Testing
  - API Chat - AI-powered API testing for Natural Language queries
  - API Try-out
- SDK Generation
- Consumer Collaboration through Comments, Ratings and Social Sharing

API Versioning

API Versioning vs Revisioning

API Versions

Represents a specific release or iteration of an API with unique features, functionalities, or changes.

- Deep copy of an existing API with backward incompatible changes
- Managed independently
  - Lifecycle
  - Documentation
  - Policies
- Requires API consumer migration

API Revisions

Changes at a more granular level, such as modifying individual endpoints, parameters, or response formats.

- Copy of an existing API with backward compatible changes
- Maintained within the same API
- Enable continuous improvement of the API
- Supports rollback
- No migration required

Use Case : API Versioning and Lifecycle Management

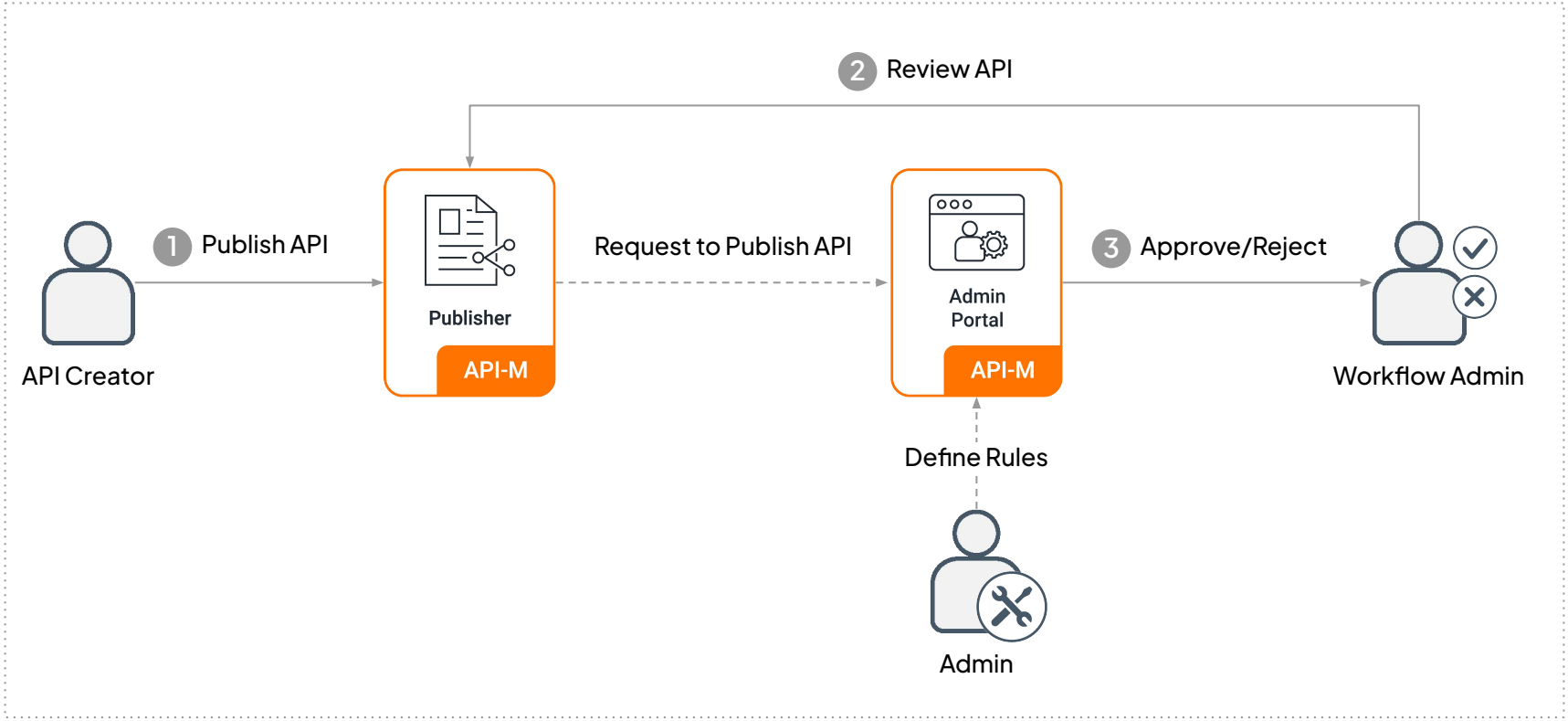
- The operations team at GOGO Transit recognizes the necessity to update the Train Information API to incorporate **train routes**.
- Existing developers should not be affected by these updates
- Requires seamless migration to the new version of the API by providing clear documentation and communication to developers regarding the changes and deprecated features

API Governance

Use Case : API Design-time Governance

- The administrators at Coltrain aim to enforce organization-specific rules in API definitions:
  - Contact email must belong to the organization's email domain.
  - APIs must include the Apache 2.0 License.
  - Paths must be in kebab case format.
- Only APIs adhering to these conformance rules will be permitted to publish
- Workflow-admin reviews and approves API for publishing (He has limited permissions to **only** access workflow information, review them and approve/reject)

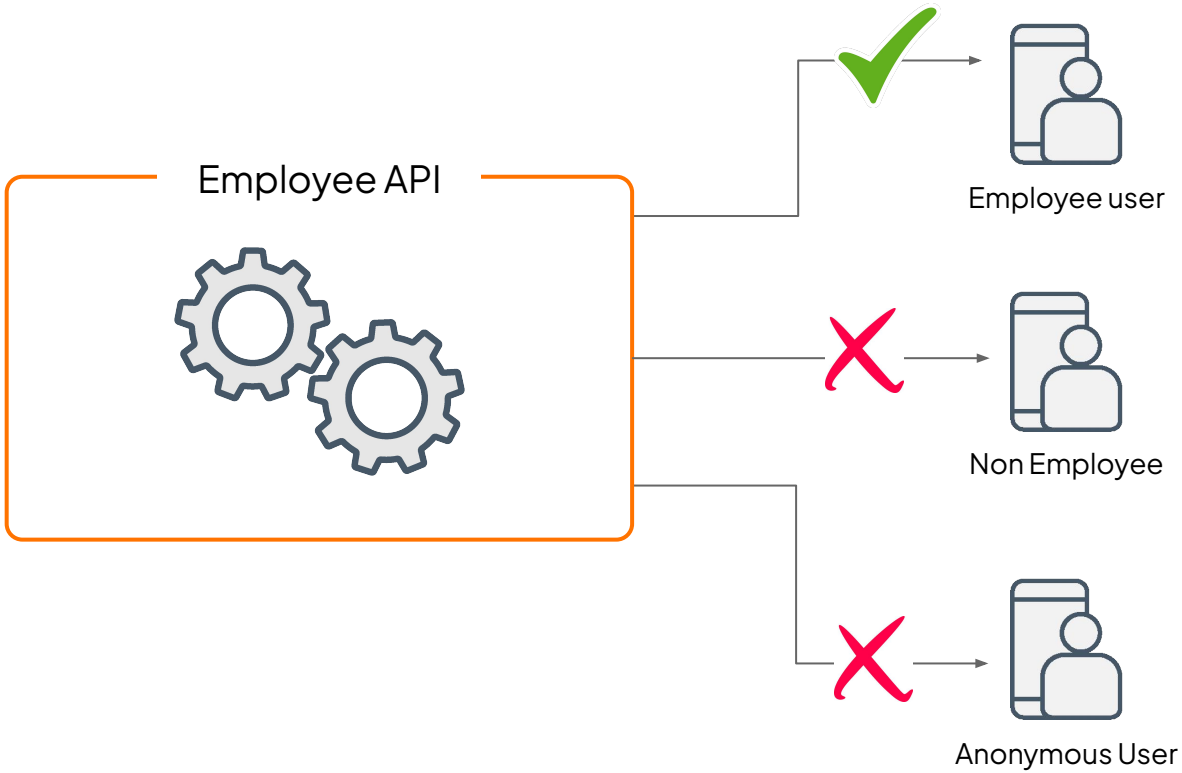
API State Change Workflow Approval



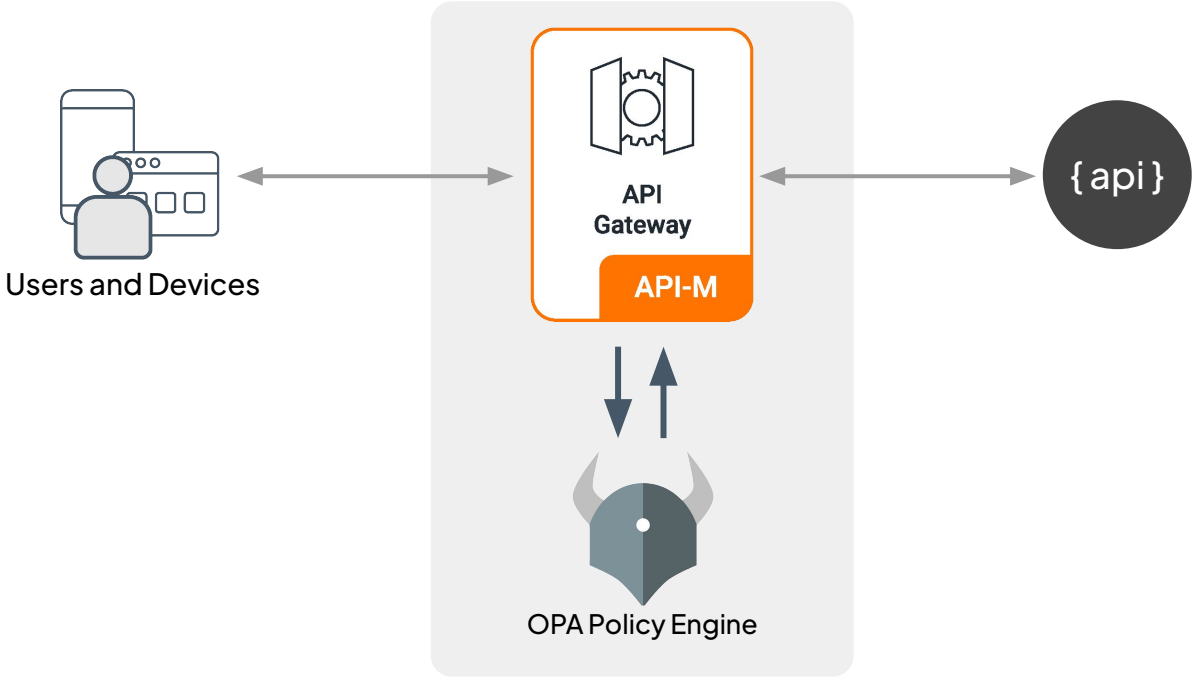
Employee API

Coltrain developers are planning to expose “Employee” API to internal staff

- The API should only be accessible to employees
- The API allows the retrieval of an employee's contract exclusively. Access to contracts of other employees is restricted.
- To ensure data integrity and security, all incoming data must adhere to a predefined schema


Employee API
Employee user
Non Employee
Anonymous User

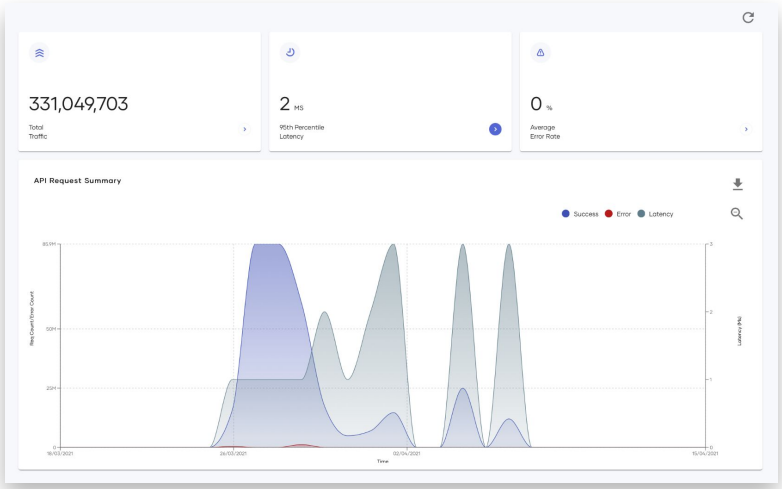
OPA Policies



API Analytics and Monetization

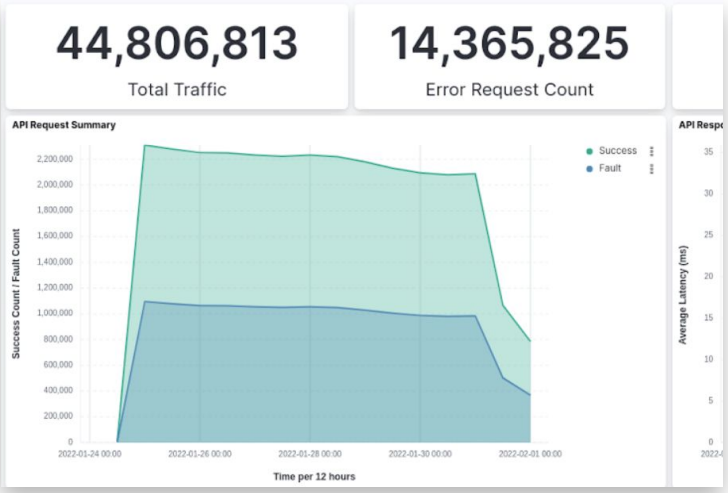
API Analytics

Choreo Insights Analytics

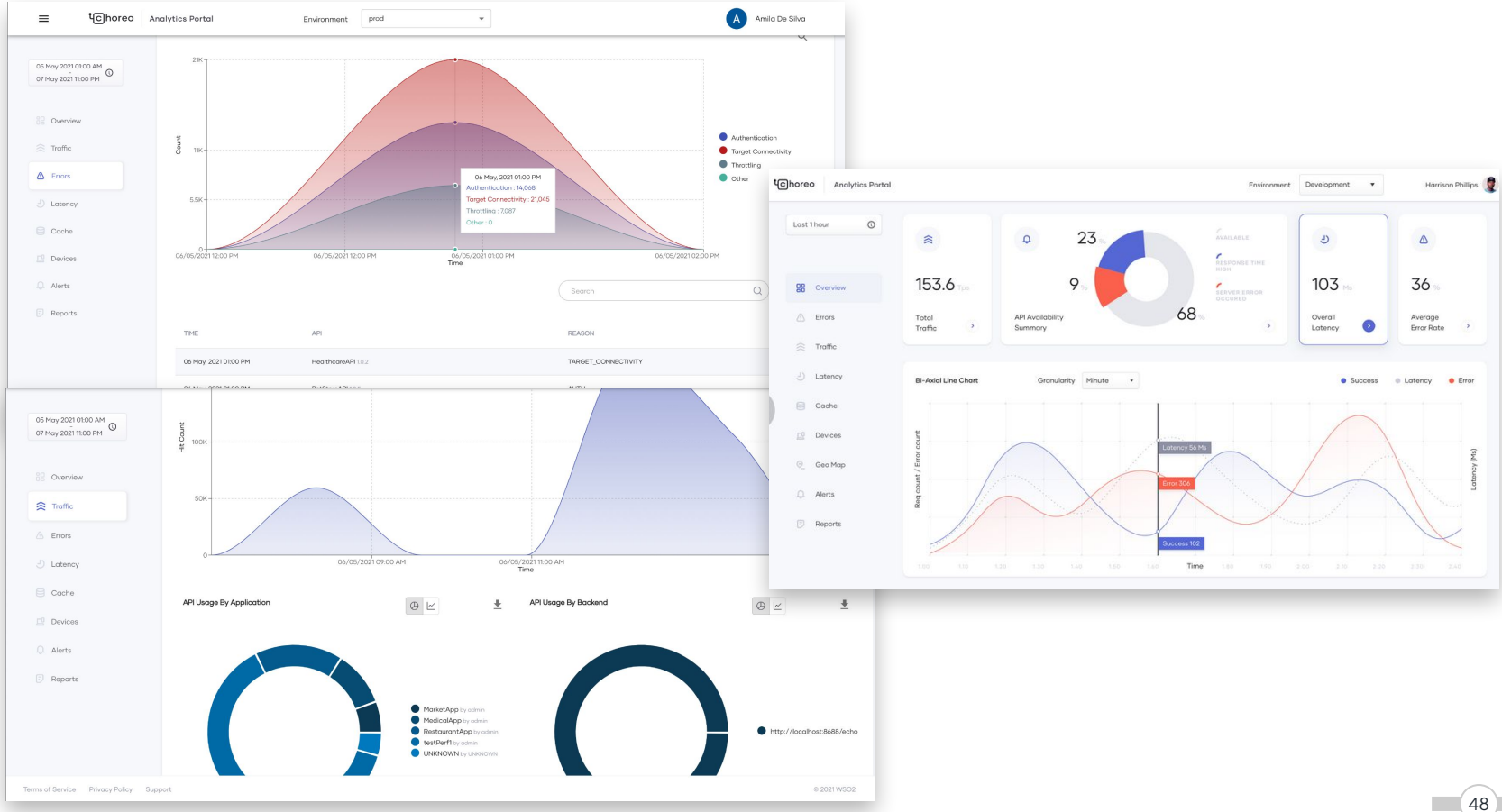


Available by default with WSO2 Support Subscription
Plug and play solution
Limited customizations

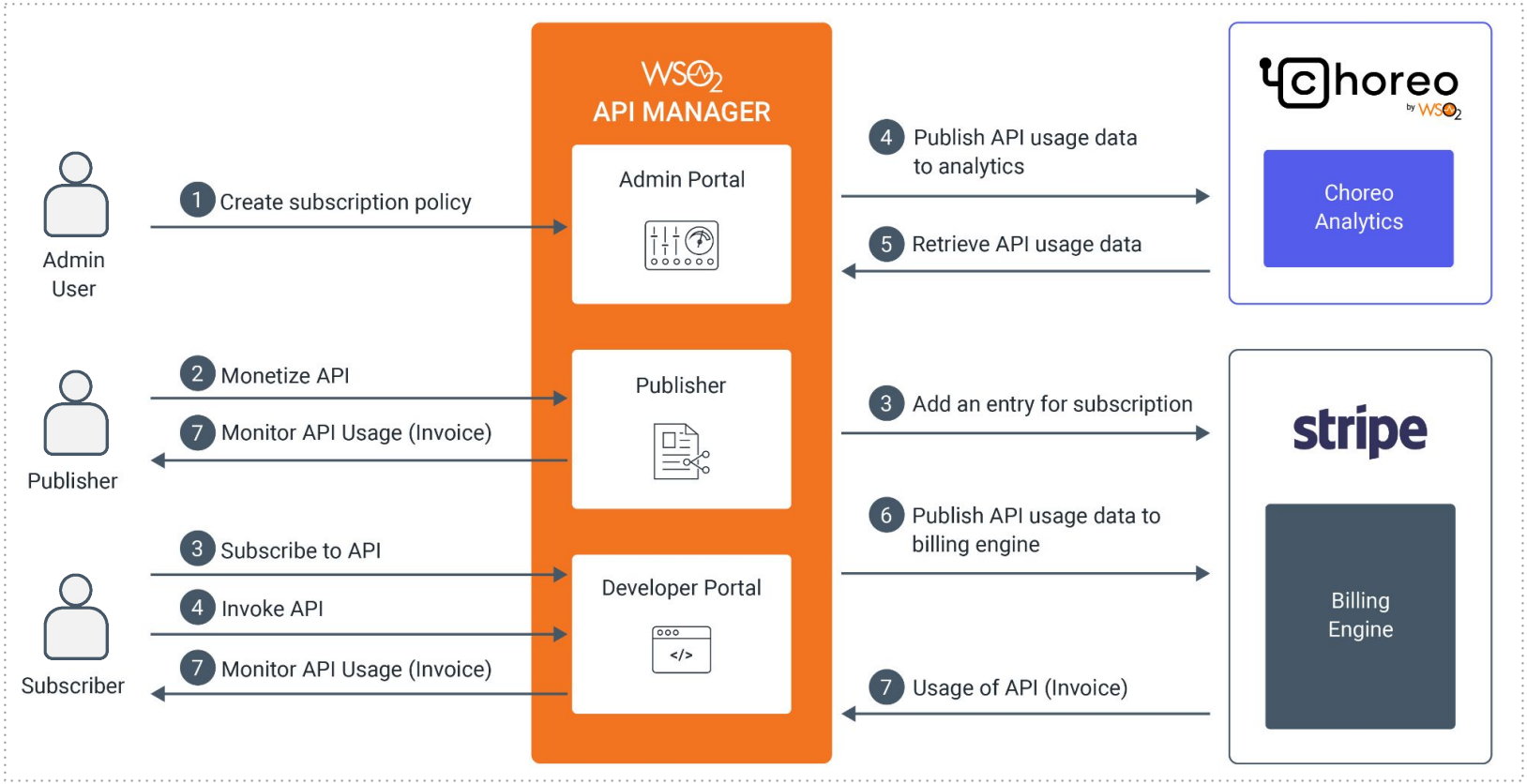
ELK Based Analytics



Open Source Option
Maintained by the customer
Highly customizable


Analytics Portal
Environment
prod
Amila De Silva
05 May 2021 01:00 AM
07 May 2021 11:00 PM
Overview
Traffic
Errors
Latency
Cache
Devices
Alerts
Reports
Count
Time
06 May, 2021 01:00 PM
Authentication : 14,068
Target Connectivity : 21,045
Throttling : 7,087
Other : 0
Authentication
Target Connectivity
Throttling
Other
Search
TIME
API
REASON
06 May, 2021 01:00 PM
HealthcareAPI 1.0.2
TARGET_CONNECTIVITY
Hit Count
API Usage By Application
API Usage By Backend
MarketApp by admin
MedicalApp by admin
RestaurantApp by admin
testPerf1 by admin
UNKNOWN by UNKNOWN
http://localhost:8688/echo
Terms of Service
Privacy Policy
Support
© 2021 WSO2
Analytics Portal
Environment
Development
Harrison Phillips
Last 1 hour
Overview
Errors
Traffic
Latency
Cache
Devices
Geo Map
Alerts
Reports
153.6 Tps
Total Traffic
23
9 %
68 %
AVAILABLE
RESPONSE TIME HIGH
SERVER ERROR OCCURED
API Availability Summary
103 Ms
Overall Latency
36 %
Average Error Rate
Bi-Axial Line Chart
Granularity
Minute
Success
Latency
Error
Req count / Error count
Latency (Ms)
Latency 56 Ms
Error 306
Success 102
Time

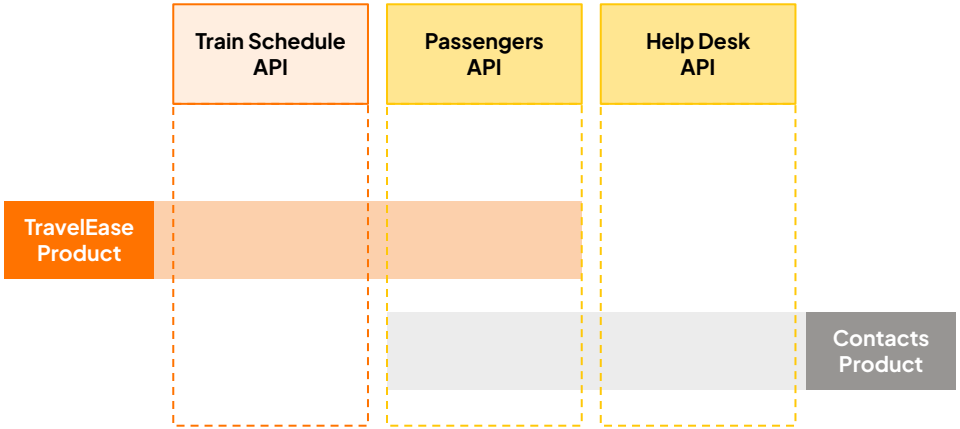
Monetization



API Products

API Products

- Serves as a packaging mechanism, bundling a preferred set of resources from multiple APIs
- A unified interface for developers to access multiple APIs
- Can be monetized as a single offering
- Supports full lifecycle management and versioning



Use Case : API Products

TravelEase, a leading travel app company, seeks to enhance its platform by integrating real-time train schedule and passenger count information. GOGO Transit, plans to expose following resources to TravelEase developers.

- ScheduleAPI → GET /schedules
- PassengerAPI → GET /passenger-count

To prevent overuse, strict rate-limiting policies are enforced

- Maximum quota: 500 req/minute
- Maximum requests per second: 15
- Subscription tier: Exclusive to TravelEase developers
- Usage-Based Pricing : pay-as-you-go model

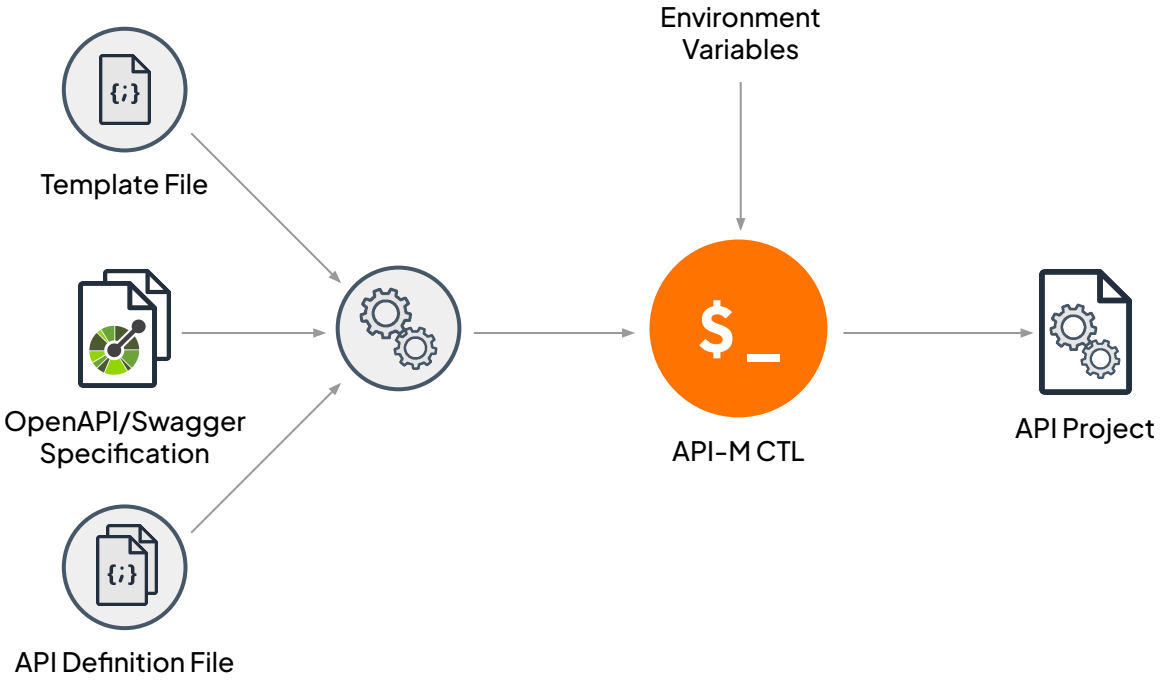
Travelease developers shares the same application

API Continuous Development & Integration (CICD)

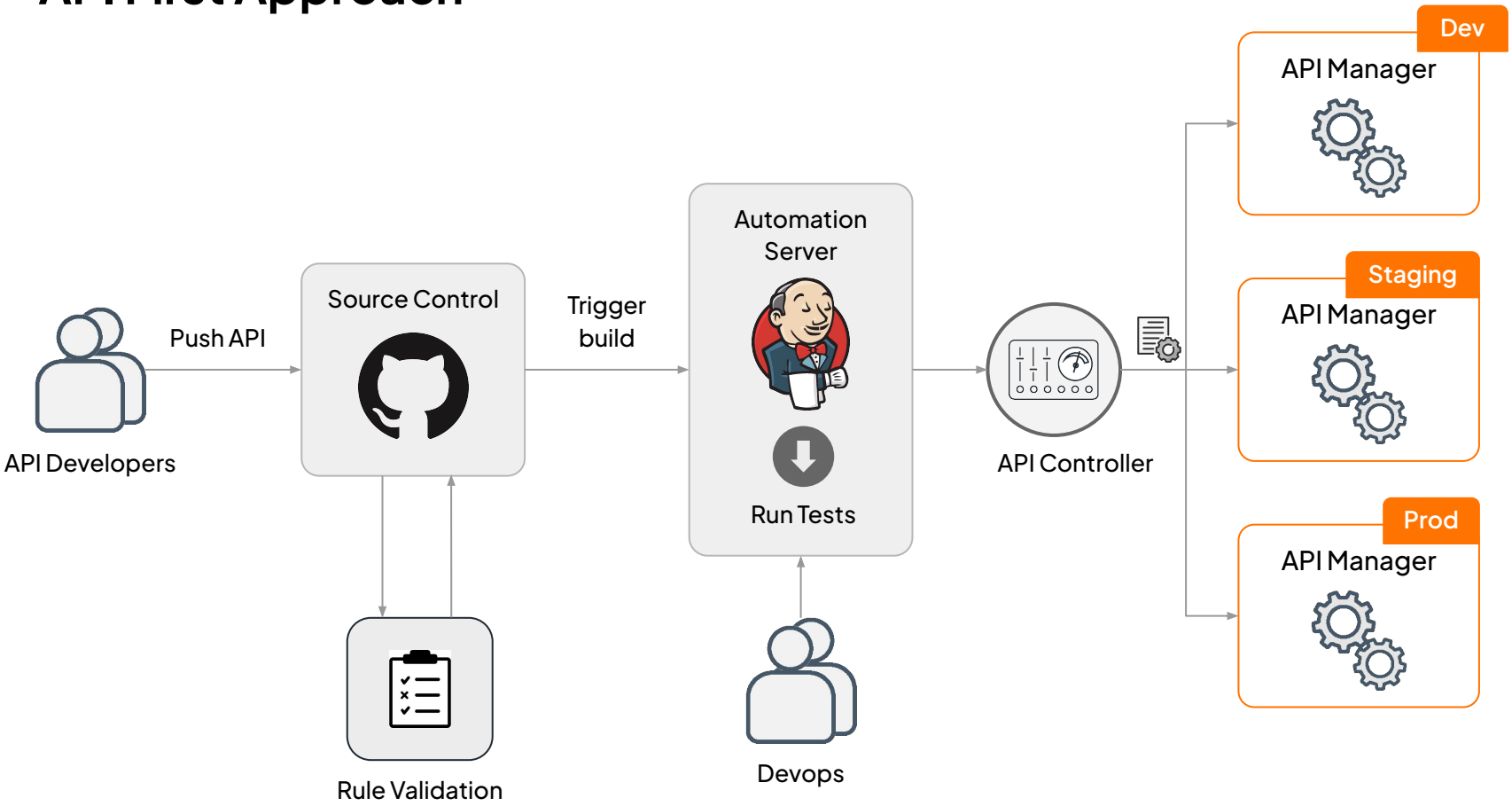
WSO2 API Controller (APICTL)

- APICTL is a simple Command Line Interface (CLI) tool that serves as an abstraction to internal product APIs.
- Facilitates API/Application migration across different environments. ie: Production, Staging, Dev
- Supports,
  - Importing and exporting,
    - APIs/API Products
    - Applications
    - Rate limiting Policies
  - Manage Lifecycle and versioning
  - Importing APIs Via Dev First Approach

API Project


Environment Variables
Template File
OpenAPI/Swagger Specification
API Definition File
$ _
API-M CTL
API Project

API First Approach



Beyond REST API Management

GraphQL APIs

- Importing GraphQL SDL and expose as a managed API
- Supports GraphQL,
  - Queries
  - Mutations
  - Subscriptions - Enables real-time data updates to be pushed to clients
- Rate Limiting via Query Analyzer
  - Query depth limit
  - Query complexity limit
- Built-in try-out tool

Streaming APIs (Async)

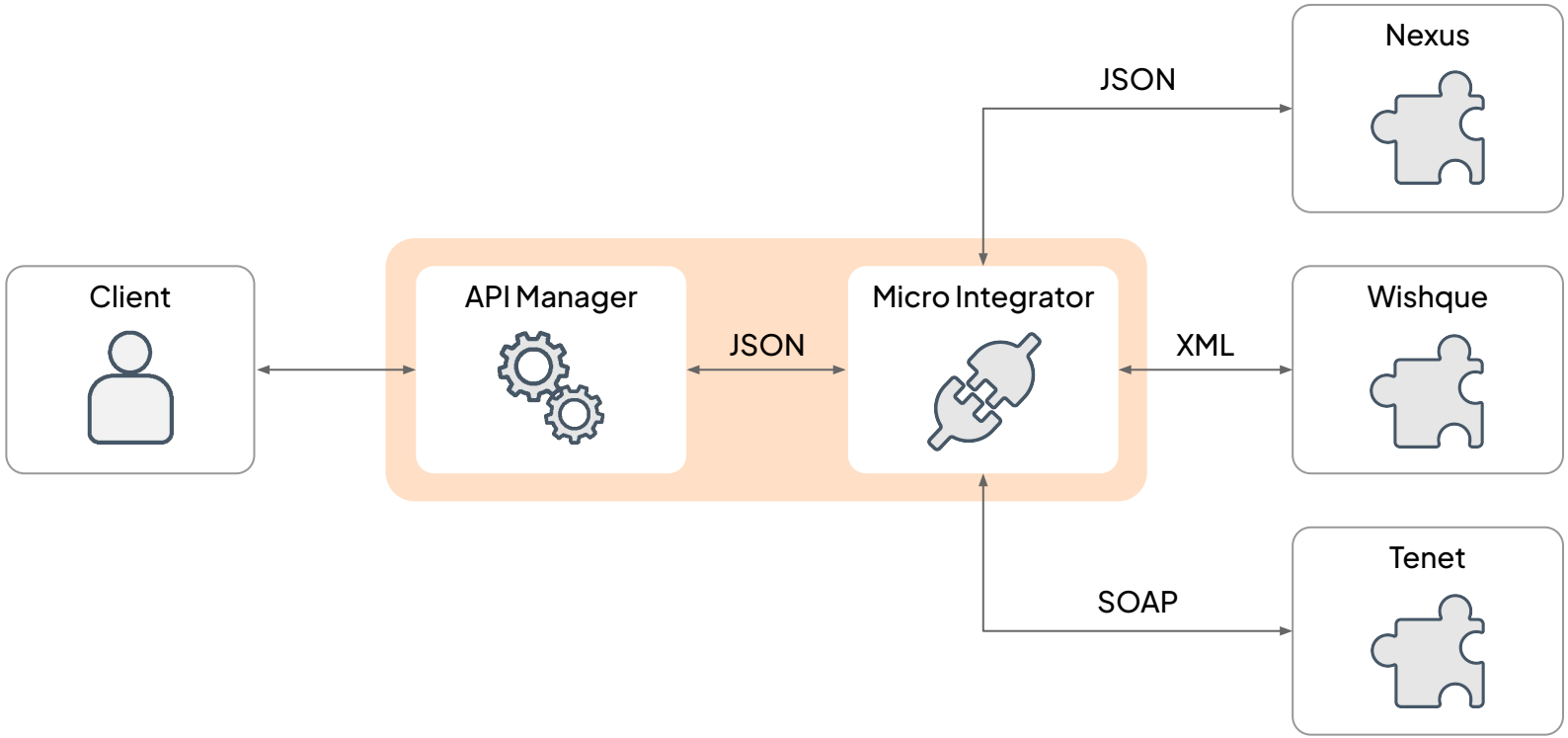
- WebSocket
- WebSub (WebHook)
- Server Sent Events (SSE)
- Authorization
- Rate Limiting
  - Number of active connections
  - Total number of subscribers
  - Total data transferred

Service Catalog

Service Catalog

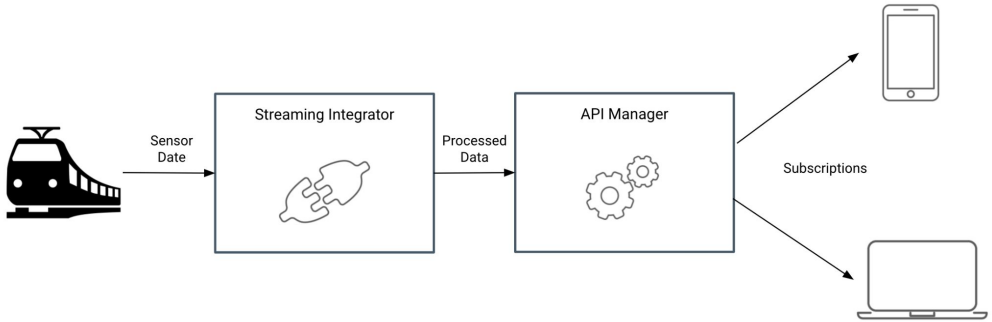
- Integration First approach
- Developers can register REST/Streaming services
  - Micro Integrator - REST services
  - Streaming Integrator - Streaming APIs
- Registered service can be converted to a Managed API from a single click

API Implementation

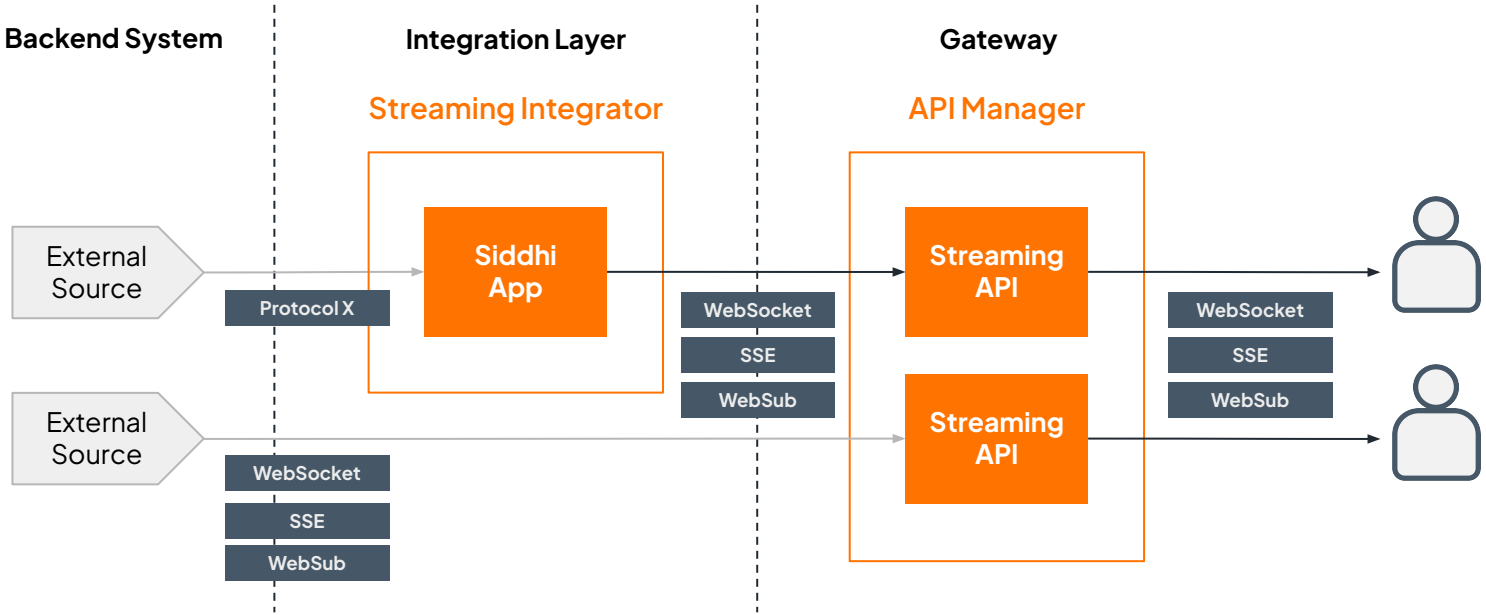


Use Case : Realtime Data with Streaming APIs

- GoGo Transit wants to provide real time location of their trains to their customers.
- The sensors in the train will be providing real time events and Quantis wants to convert it to WebSockets and expose it as a Streaming API so that the client mobile applications can subscribe to it and receive real time events.



Streaming APIs



Observability

Logs

- Correlation Logs
- HTTP Access Logs
- Audit Logs
- Per API Logs
- Wire Logs

Traces

- OpenTracing and OpenTelemetry
  - Jaeger
  - Zipkin
  - Log
  - OTLP

Metrics

- JMX Monitoring
- Healthcheck endpoints

APK Workshop –
https://docs.google.com/presentation/d/1e8AXcEghL-30nr5jmVTz200atSrcBAKpgoEeHX0aDq4/edit#slide=id.g26efb48413e_0_0

Question Time!



Provide Feedback



Thank You!