Red Hat
Managed Integration

API as a Product on RHMI

Dean Peterson
Full Stack Solution Architect
dpeterso@redhat.com

Red Hat

Agenda

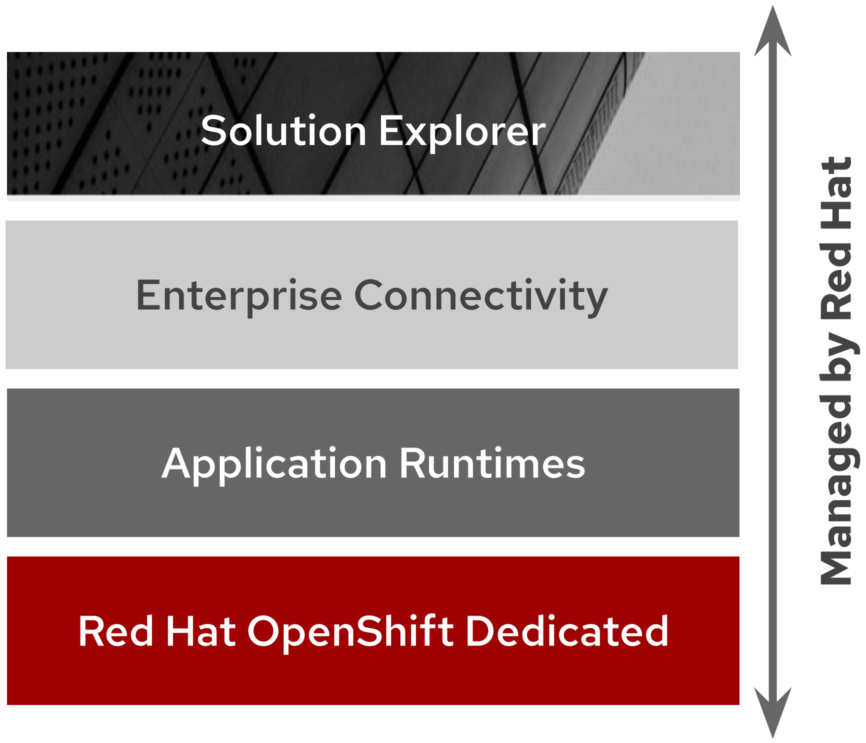
- Introducing RHMI
- API as a Product
- Demo
- Additional Resources
- Live Q&A

Red Hat Managed Integration

Hosted and managed platform for delivering cloud-native, integrated applications built on Kubernetes that reduces operational risk, cost, and time to value

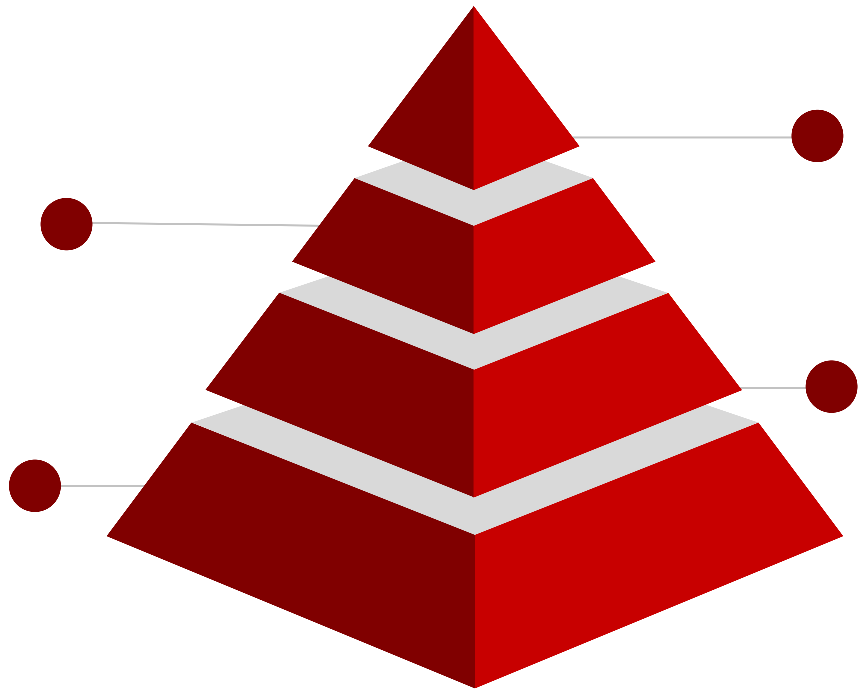
Platform for building, deploying, and scaling integrated enterprise applications, comprised by four layers:

- Streamlined development UX
- Enterprise connectivity
- Application Runtimes
- Hosted and Managed



RHMI: STREAMLINED UX

Cloud-native integration...simplified

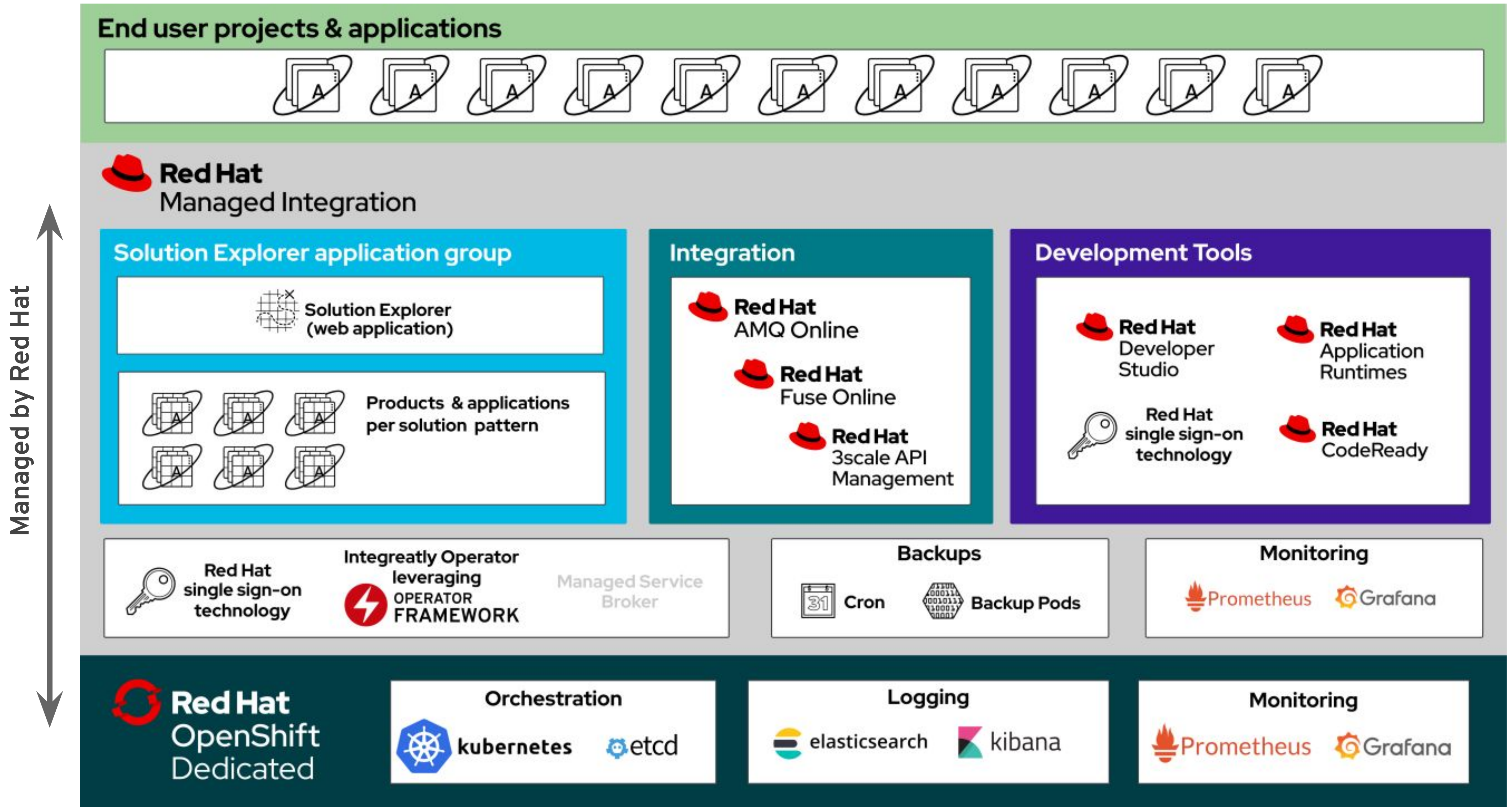


Solution Patterns
Pre-built templates shorten development time and encourage best practices

Solution Explorer
Web-based UI with automated provisioning, single sign-on, and integrated IDE

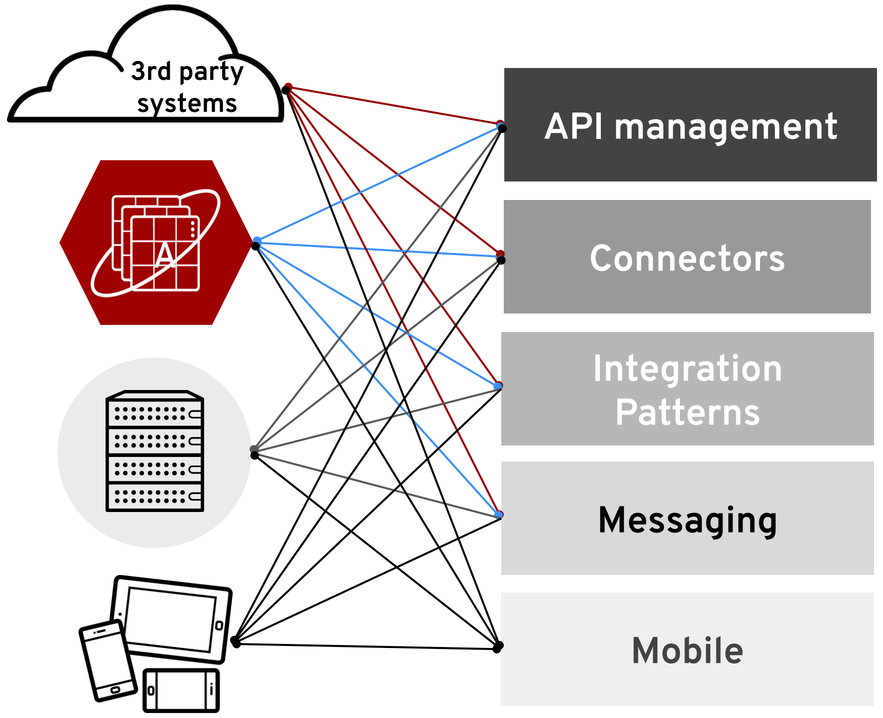
Recommended configurations
Installed infrastructure components are preconfigured for maximum performance

Technologies integrated
Development tools and services are unified for a friction-free experience


End user projects & applications
Red Hat Managed Integration
Solution Explorer application group
Solution Explorer (web application)
Products & applications per solution pattern
Integration
Red Hat AMQ Online
Red Hat Fuse Online
Red Hat 3scale API Management
Development Tools
Red Hat Developer Studio
Red Hat Application Runtimes
Red Hat single sign-on technology
Red Hat CodeReady
Red Hat single sign-on technology
Integreatly Operator leveraging OPERATOR FRAMEWORK
Managed Service Broker
Backups
Cron
Backup Pods
Monitoring
Prometheus
Grafana
Red Hat OpenShift Dedicated
Orchestration
kubernetes
etcd
Logging
elasticsearch
kibana
Monitoring
Prometheus
Grafana
Managed by Red Hat

RHMI: ENTERPRISE CONNECTIVITY

Easily build apps that connect to services, systems and data



API Management

Supports API-first development for agility

Enterprise Integration Patterns (EIPs)

Low-code connectivity with predefined patterns

Messaging

Lightweight data sharing for IoT applications

Extend applications to reach across the enterprise and to hybrid clouds

MANAGED COMPONENTS

Managed and self-managed versions are released at the same time

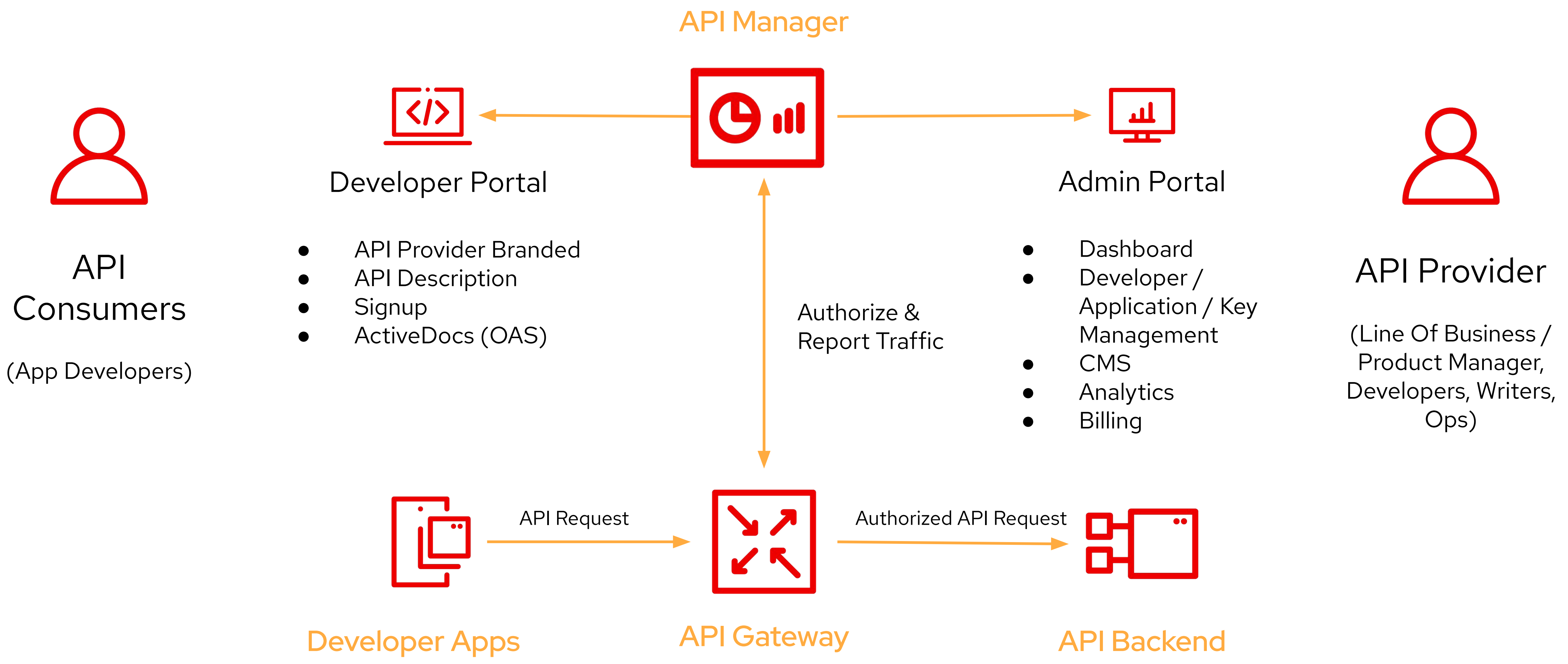
Component	RHMI 1.6 (Current)	RHMI 1.7 (Next)	RHMI 2.0 (Coming Soon)
OpenShift Dedicated	3.11	3.11	4.x
3Scale	2.7.1	2.8	2.8
CodeReady	1.2.0	2.0.0	2.0
AMQ Online	1.3.1	1.4.x	1.4.x
Fuse Online	1.8.x	1.9.x	1.9.x
Fuse on OpenShift	7.5	7.5	7.6
Apicurito	0.2.18.Final	1.5	1.5
Launcher	7224e23	N/A	N/A
UnifiedPush Server	2.3.2-1		
RH-SSO	7.3.x	7.3.x	8.0.x
Solution Explorer	2.20.8	2.21	2.21

Component versions are listed in the inventories/group_vars/all/manifest.yaml. Check this file for any release tag via https://github.com/integr8ly/installation/releases

We also maintain a version registry spreadsheet here.

Red Hat 3scale on RHMI

Red Hat 3scale API Management



Developer Portal

- API Provider Branded
- API Description
- Signup
- ActiveDocs (OAS)

Admin Portal

- Dashboard
- Developer / Application / Key Management
- CMS
- Analytics
- Billing

Red Hat 3scale API Management

Operational Aspects

- Multi-tenant 3scale API Manager and hosted Gateways
- Environment Managed by Red Hat
  - Scaling
  - High Availability
  - Backup/Restore
  - Upgrades
  - Patching
- Operator for install & upgrade
  - Managed by cluster admin
  - Namespace access restricted to cluster admin
- Cluster metrics and dashboard available to dedicated admin for monitoring
- SSO set up for Admin Portal access for all tenants

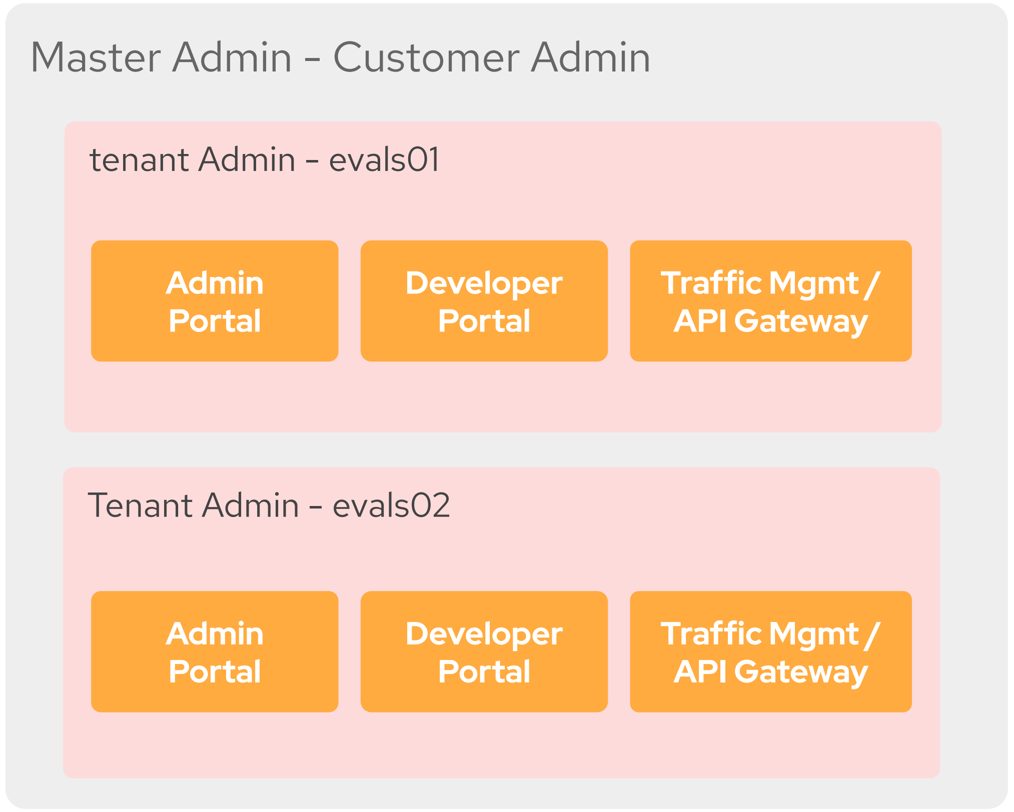
Red Hat 3scale API Management

Administrative Aspects

- Multitenancy:
  - One tenant for each user on RHPDS. Link to tenant admin portal available from Solution Explorer
  - Single Sign On using OCP credentials
  - NOTE: Single tenant mode in Production. Service request for additional tenants
- Tracing & Reporting:
  - Embedded APIcast configured to allow exporting metrics to Prometheus
  - Prometheus/Grafana for monitoring managed service workloads
- Single Sign On
  - Customer SSO instance available to set up SSO & OIDC for each tenant environment
- User namespaces:
  - Self managed Gateways (optional)
  - API implementations
  - Other deployments

Multi Tenancy

Logically separate environments using shared resources



Master Admin
- **Access for Customer Admin only**
- **Manage Tenants**
- **Impersonate Tenants**

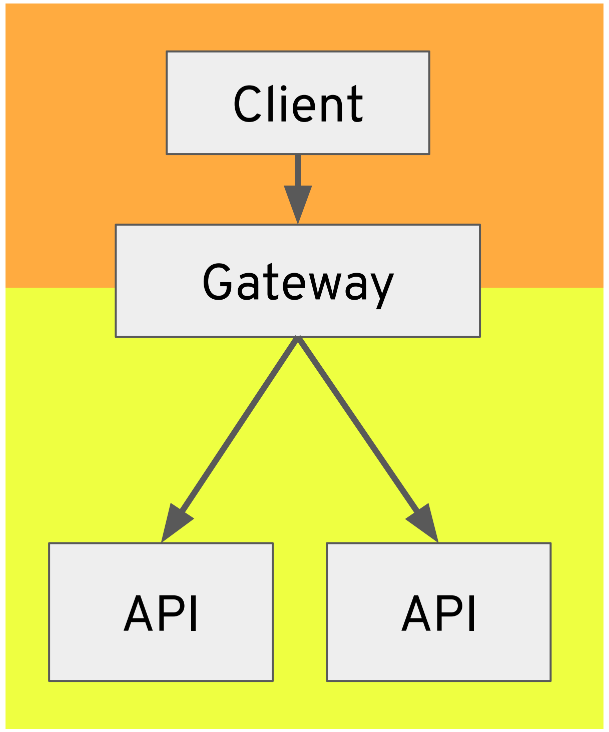
Tenant Admin
- **Access to each user**
- **Manage tenant admins / users**
- **Access APIs and Admin Portal**

Developers
- **Access to Developer Portal**
- **Access given to services / sections**
- **Managed by tenant admin**

API Packaging & API As A Product

API Management Traditional Flow

North-South Model for API Management



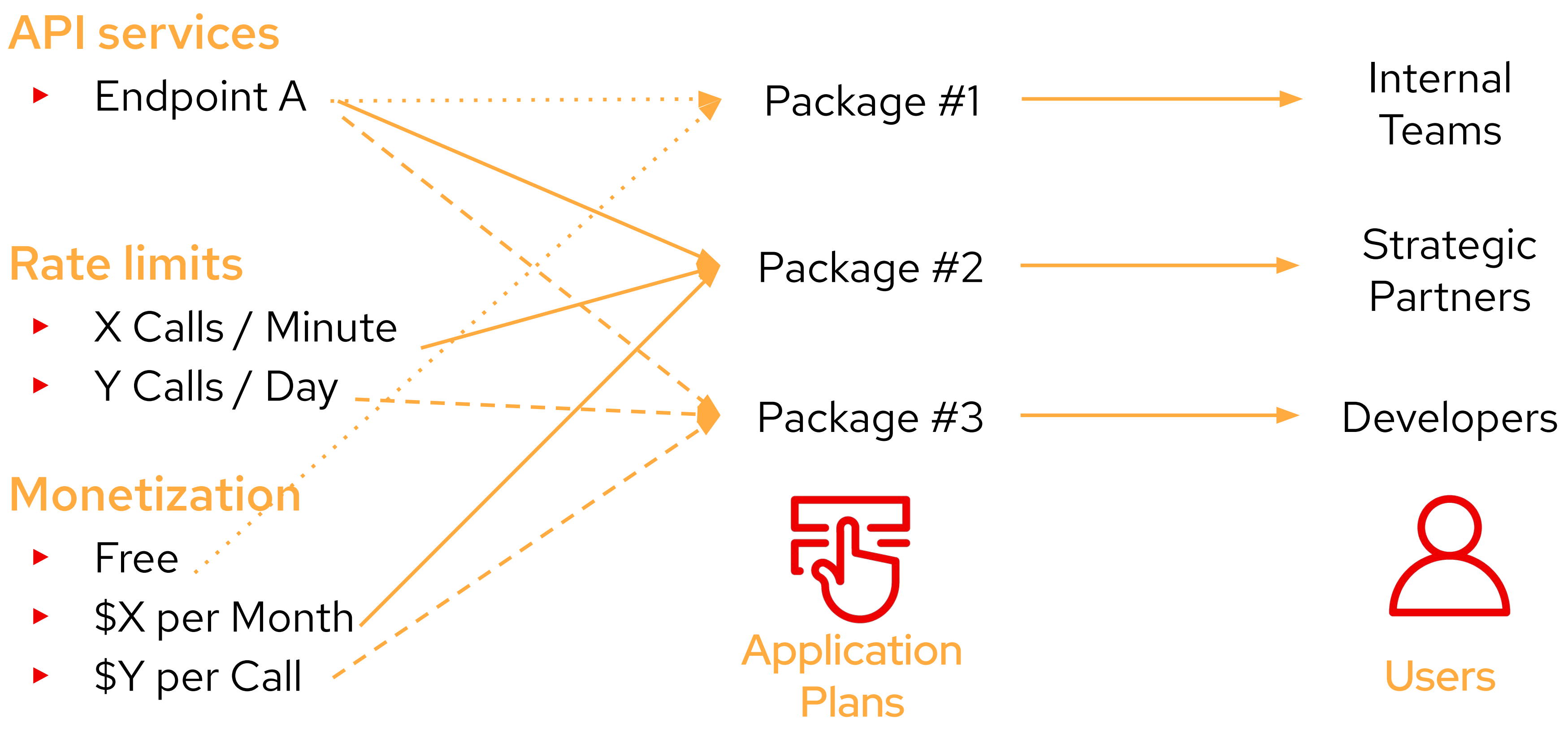
API Gateway

- APIs as a digital access point for your business
- Security, developer onboarding and analytics
- “North-South” service architecture pattern
- Requires traditional API management capabilities
- APIs As A Product

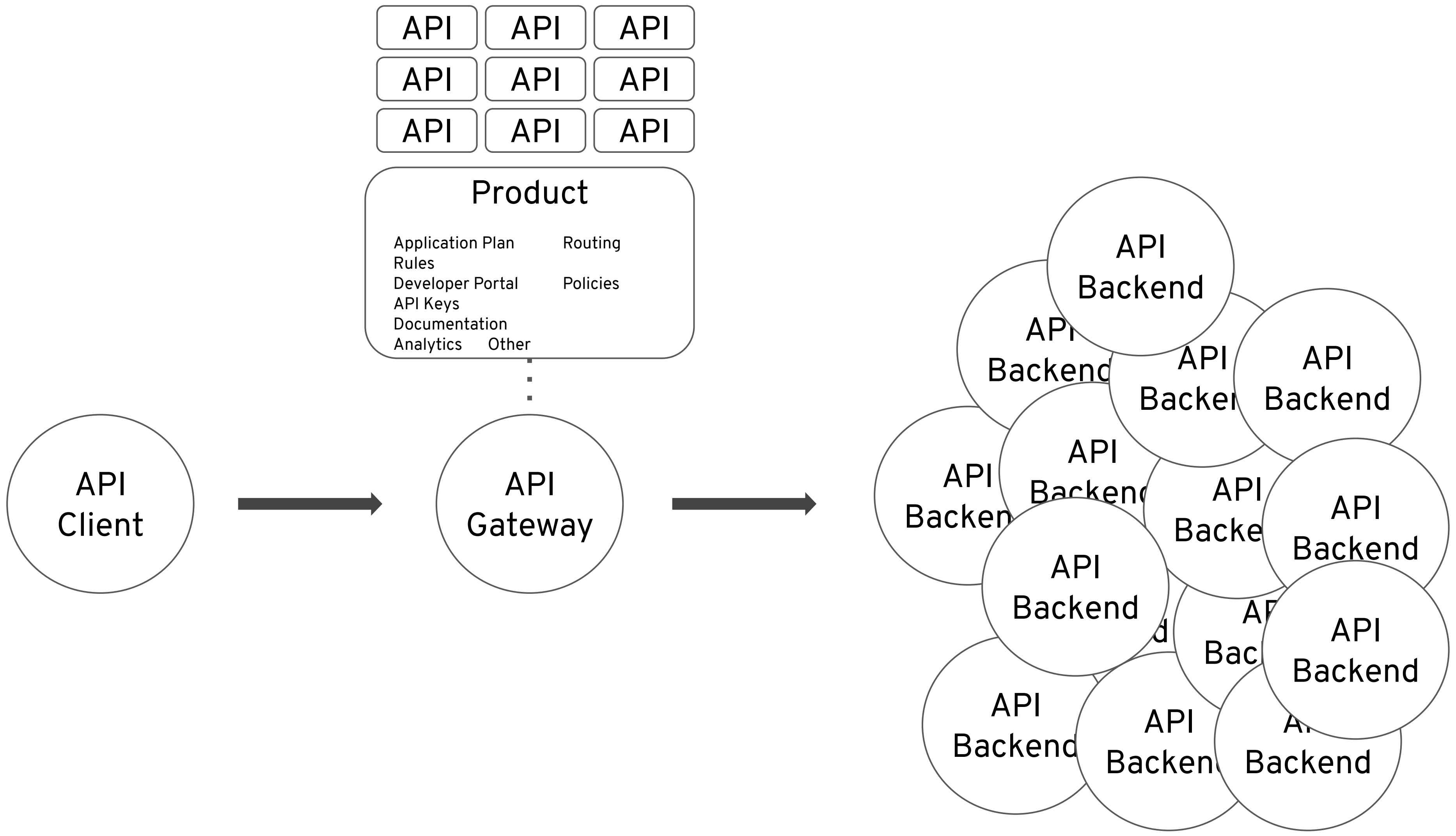
Red Hat

API Contracts, Throttling and Limits

Package your APIs. Create access tiers. Set rate limits.



APIs as a Product



APIs as a Product (APIP)

<= 2.6

Single-backend services (“APIs”)

Service exposes public managed endpoints and consumes in the background **one and only one private API** implementation behind.

2.7 ~ 2.8

API products + API backends

Service becomes **Product** (or “API product”)

Still exposes public managed endpoints and continues to have application plans, limits, monetization rules, etc.

Instead of only one backend API, it may use **multiple API backends**, with **path-based routing rules** that direct the traffic to either one or the other.

2.9+

API backend

Stats/analytics filtered by API backend and aggregated across API products

Backend Discovery

Full OAS3 support

API Products ✕ API Backends

API Product
public/managed

Business layer

- Public facing (“facade”) → URL structures
- Sign-up → API creds → AuthN
- App plans = limits and pricing rules
- (Global) metrics/mapping rules
- APIcast policies
- ActiveDocs
- Can use multiple API backends

Gateway

Path-based routing

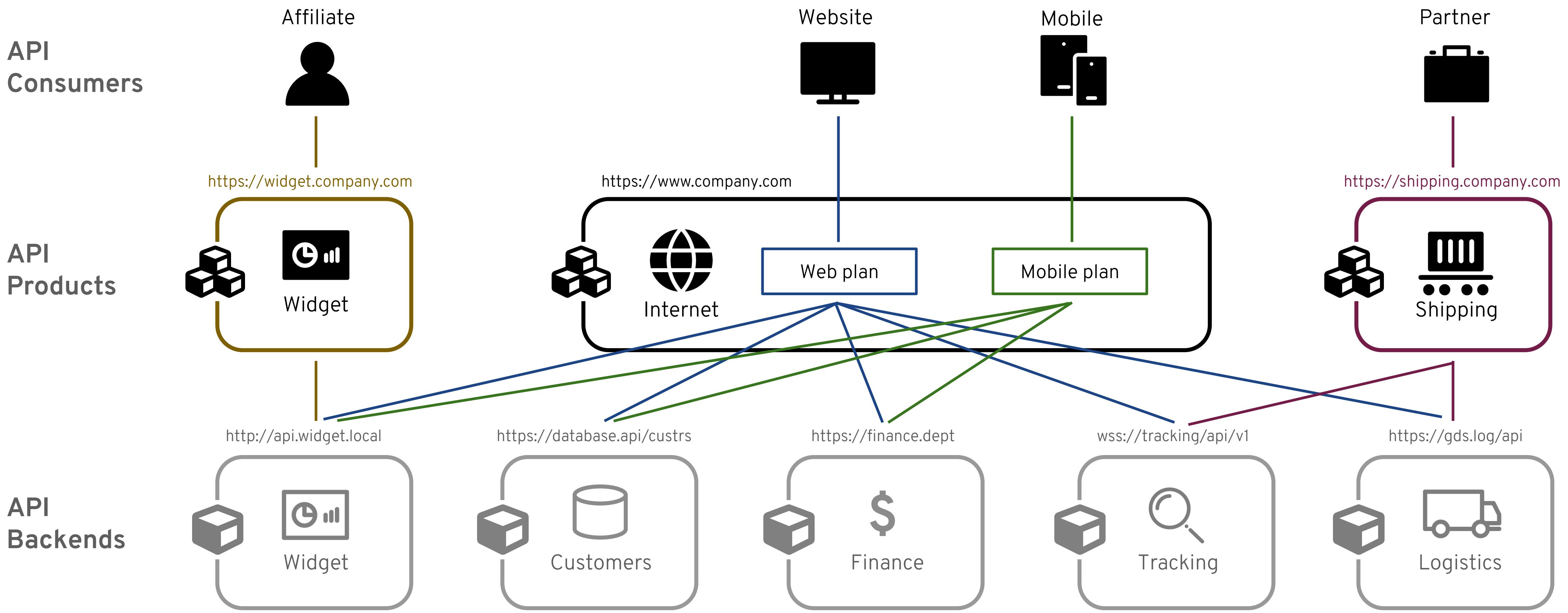
API Backend
private

Technical layer

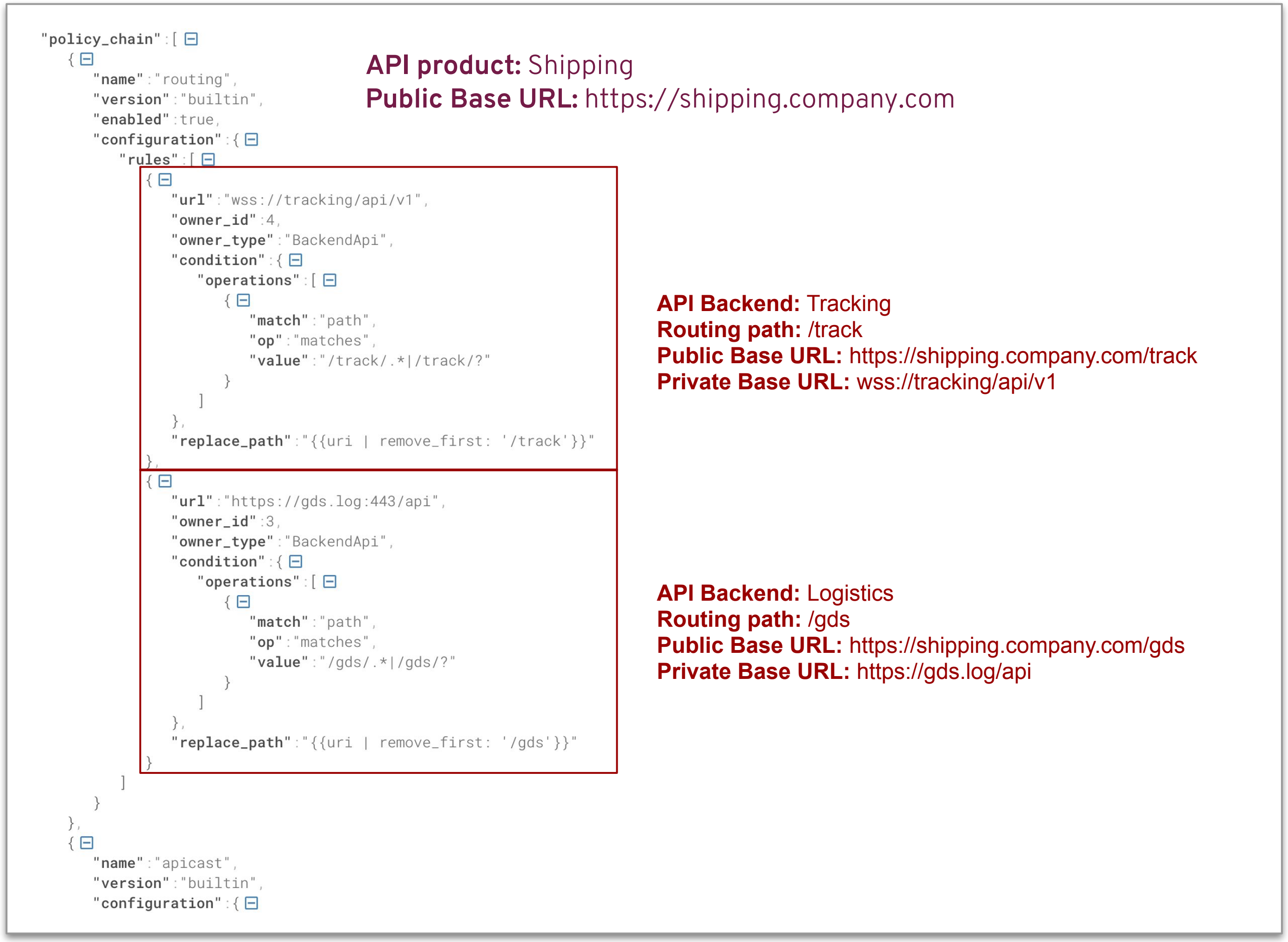
- Private base URL
- Private URL structures
- No AuthN/AuthZ → no plans
- (Local) metrics/mapping rules (by method)
- Can be used by multiple API products

Red Hat

Example


API Consumers
Affiliate
Website
Mobile
Partner
https://widget.company.com
https://www.company.com
https://shipping.company.com
API Products
Widget
Internet
Web plan
Mobile plan
Shipping
http://api.widget.local
https://database.api/custrs
https://finance.dept
wss://tracking/api/v1
https://gds.log/api
API Backends
Widget
Customers
Finance
Tracking
Logistics

Routing



```
"policy_chain":[
  {
    "name":"routing",
    "version":"builtin",
    "enabled":true,
    "configuration":{
      "rules":[
        {
          "url":"wss://tracking/api/v1",
          "owner_id":4,
          "owner_type":"BackendApi",
          "condition":{
            "operations":[
              {
                "match":"path",
                "op":"matches",
                "value":"/track/.*|/track/?"
              }
            ]
          },
          "replace_path":"{{uri | remove_first: '/track'}}"
        },
        {
          "url":"https://gds.log:443/api",
          "owner_id":3,
          "owner_type":"BackendApi",
          "condition":{
            "operations":[
              {
                "match":"path",
                "op":"matches",
                "value":"/gds/.*|/gds/?"
              }
            ]
          },
          "replace_path":"{{uri | remove_first: '/gds'}}"
        }
      ]
    }
  },
  {
    "name":"apicast",
    "version":"builtin",
    "configuration":{
```

API product: Shipping
Public Base URL: https://shipping.company.com

API Backend: Tracking
Routing path: /track
Public Base URL: https://shipping.company.com/track
Private Base URL: wss://tracking/api/v1

API Backend: Logistics
Routing path: /gds
Public Base URL: https://shipping.company.com/gds
Private Base URL: https://gds.log/api

- Each Backend bundled in a Product is "mounted" in a chosen path
- Path must be unique
- Root path ('/') is allowed
- A 'routing' policy is transparently injected into the proxy config
  - First policy in the chain
  - Longest path goes first
- Path is removed from the URL before redirecting the traffic

Mapping rules

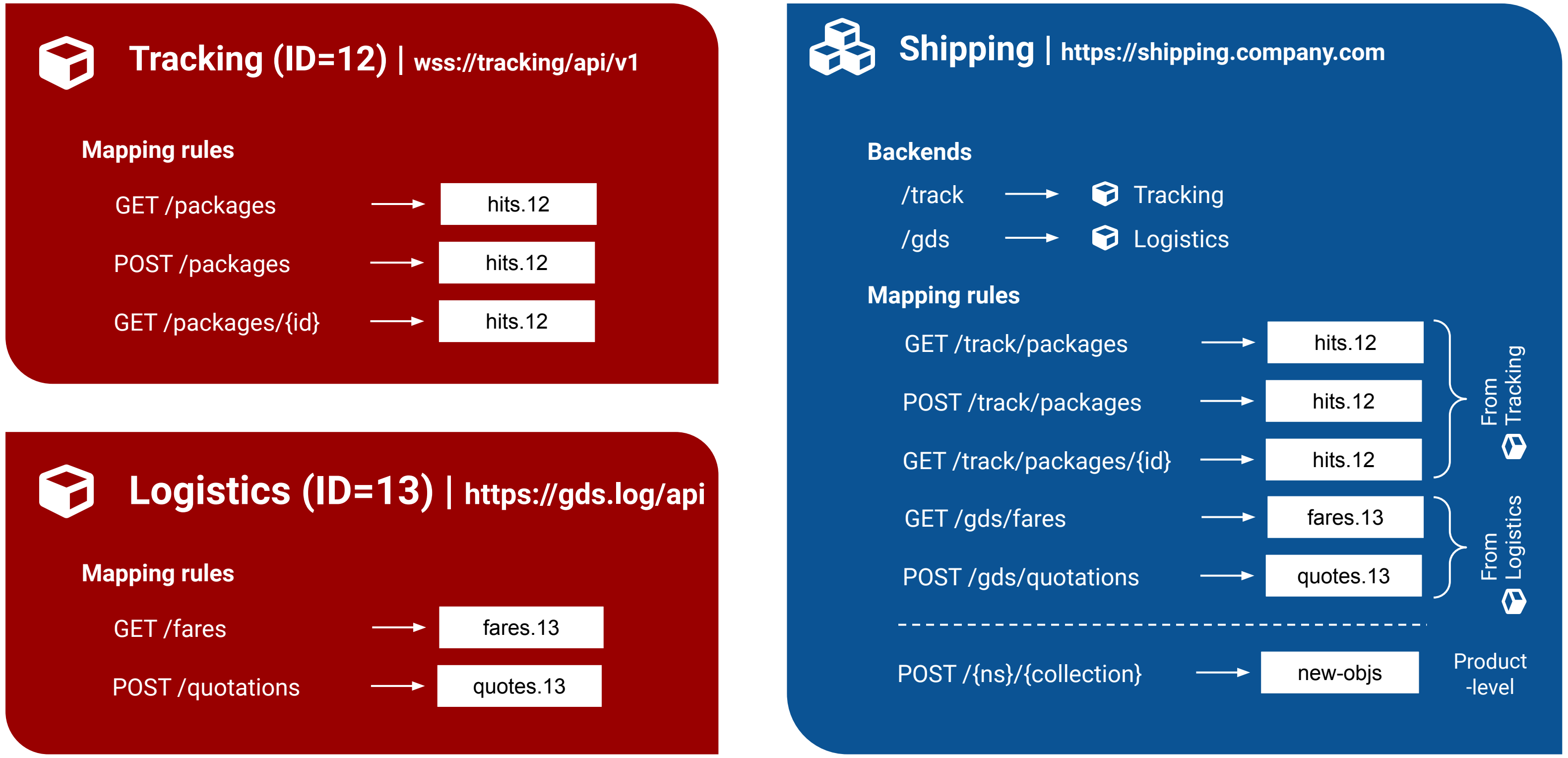
Product-level Mapping Rules

- Take precedence – i.e. go on top of all MRs
- Always evaluated, no matter to which Backend the traffic will be routed

Backend-level Mapping Rules

- Evaluated after Product-level MRs
- Evaluated only if the traffic is being routed to the same Backend the Mapping Rule belongs to
- Same set of MRs exists in all Products using the Backend
- The path of the Backend in a given Product is automatically and transparently prepended to each MR of the Backend in that Product

Mapping rules - example



Tracking (ID=12) | wss://tracking/api/v1

Mapping rules

GET /packages →	hits.12
POST /packages →	hits.12
GET /packages/{id} →	hits.12

Logistics (ID=13) | https://gds.log/api

Mapping rules

GET /fares →	fares.13
POST /quotations →	quotes.13

Shipping | https://shipping.company.com

Backends

/track →	Tracking
/gds →	Logistics

Mapping rules

GET /track/packages →	hits.12	From Tracking
POST /track/packages →	hits.12	From Tracking
GET /track/packages/{id} →	hits.12	From Tracking
GET /gds/fares →	fares.13	From Logistics
POST /gds/quotations →	quotes.13	From Logistics
POST /{ns}/{collection} →	new-objs	Product -level

POST https://shipping.company.com/track/packages → wss://tracking/api/v1/packages

GET https://shipping.company.com/gds/fares → https://gds.log/api/fares

Mapping rules - example

Tracking (ID=12) | wss://tracking/api/v1

Mapping rules

GET /packages →	hits.12
POST /packages →	hits.12
GET /packages/{id} →	hits.12

Logistics (ID=13) | https://gds.log/api

Mapping rules

GET /fares →	fares.13
POST /quotations →	quotes.13

Shipping | https://shipping.company.com

Backends

/track →	Tracking
/gds →	Logistics

Mapping rules

GET /track/packages →	hits.12	From Tracking
POST /track/packages →	hits.12	From Tracking
GET /track/packages/{id} →	hits.12	From Tracking
GET /gds/fares →	fares.13	From Logistics
POST /gds/quotations →	quotes.13	From Logistics
POST /{ns}/{collection} →	new-objs	Product -level

POST https://shipping.company.com/track/packages → wss://tracking/api/v1/packages

GET https://shipping.company.com/gds/fares → https://gds.log/api/fares

Shipping prices | https://shipping-prices.api

Backends

/ →	Logistics

Mapping rules

GET /fares →	fares.13	From Logistics
POST /quotations →	quotes.13	From Logistics

GET https://shipping-prices.api/fares→ https://gds.log/api/fares

POST https://shipping-prices.api/quotations → https://gds.log/api/quotations

Red Hat

Method/metrics

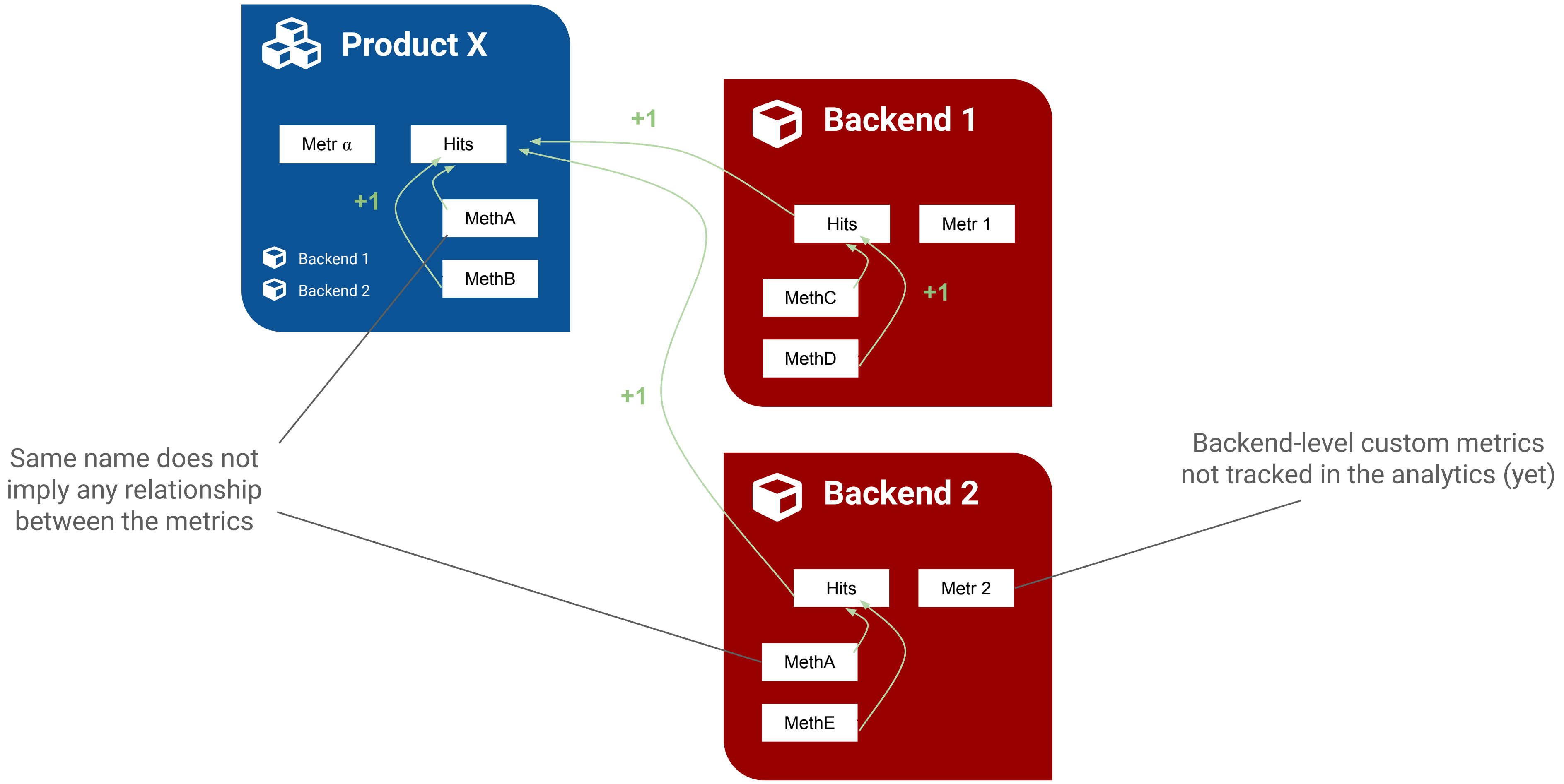
Product-level Methods & Metrics

- “Hits” metric counts hits mapped to “Hits” itself and to its methods + the hits mapped to all Backend-level “Hits” metrics and their methods.

Backend-level Method & Metrics

- Registered in APIcast as if they belonged to each Product using the Backend
- Automatically and transparently get the ID of the Backend appended to the `system_name` of the metric
- Limits and Pricing rules can be set upon Backend-level metrics in the Application Plans at the Product level

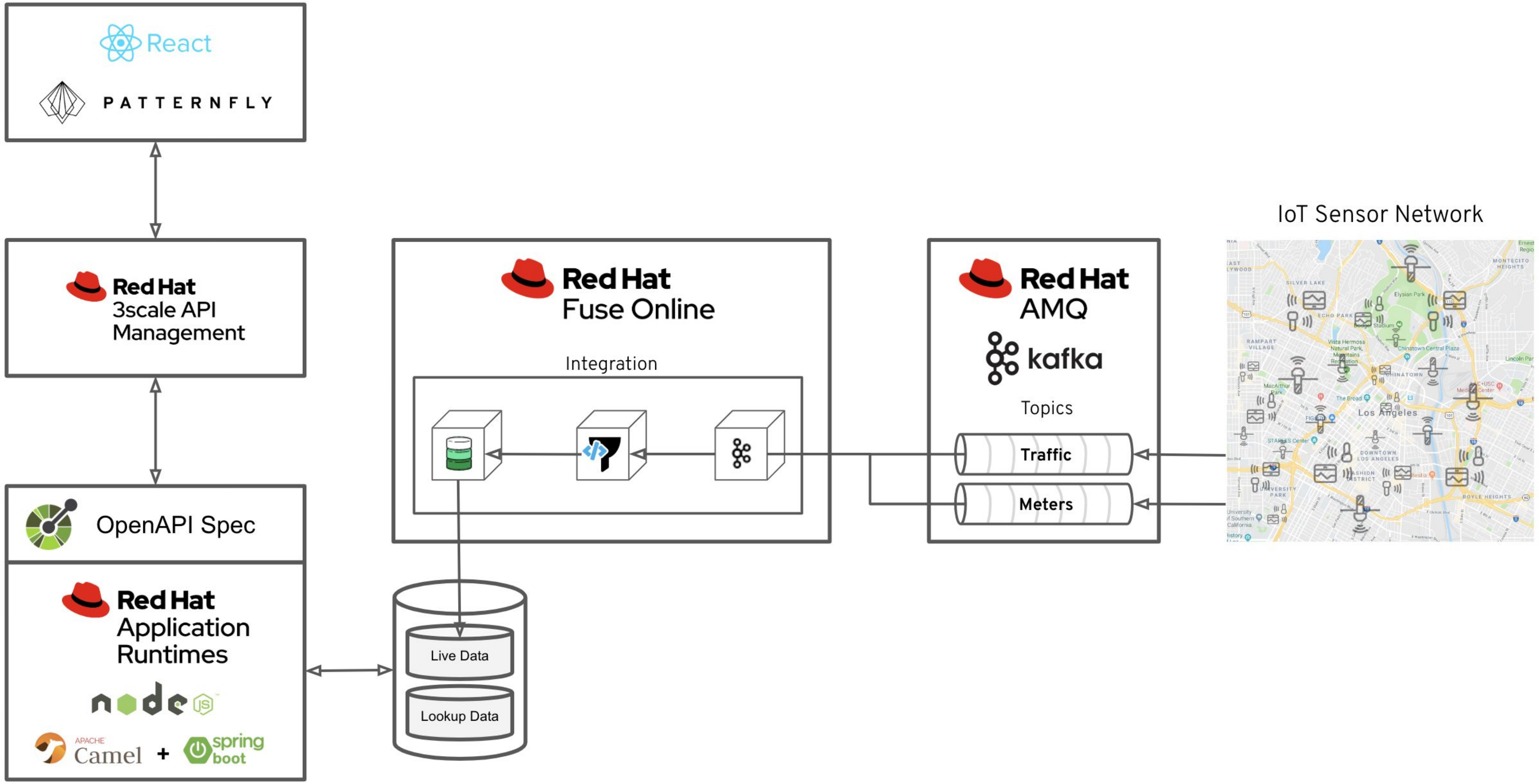
Method/metrics (3-level hierarchy)


Product X
Metr α
Hits
+1
MethA
MethB
Backend 1
Backend 2
+1
Backend 1
Hits
Metr 1
MethC
MethD
+1
+1
Backend 2
Hits
Metr 2
MethA
MethE
Same name does not imply any relationship between the metrics
Backend-level custom metrics not tracked in the analytics (yet)

Red Hat

Demo

DEMO OVERVIEW



Additional Resources

Red Hat Managed Integration - Resources

Where can I learn more?

- **OneStop gives you access to:**
  - Business, technical and pricing decks
  - General FAQs, Pricing FAQs, etc.
- **Customer stories:**
  - Mojo Page & Google Page
- **RHMI Webinar recordings:**
  - Mojo Page & Google Page
- **Allego Channel**
- **Datasheet**

Who to contact

- General questions? rhmi-info@redhat.com
- Provisioning requests? integreatly-provisioning@redhat.com
- Need support? integreatly-support@redhat.com
- ...or reach out to team members directly

?
Q
&
A
!

Thank you

Red Hat is the world's leading provider of

enterprise open source software solutions.

Award-winning support, training, and consulting

services make

Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

youtube.com/user/RedHatVideo
s

facebook.com/redhatinc

twitter.com/RedHat

Red Hat