



# Develop and Connect Cloud Native Applications

Red Hat Integration Technical Overview

Presenter's Name

Title

# What we'll discuss today

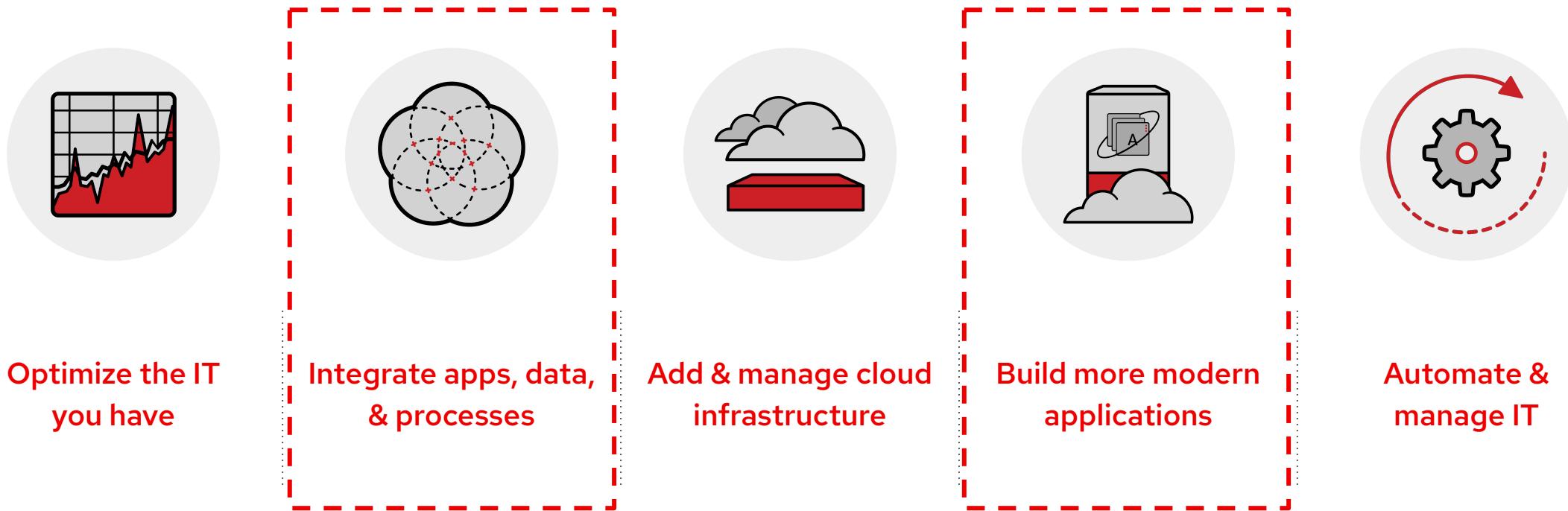
APIs

Events &  
Messaging

Enterprise  
Integration

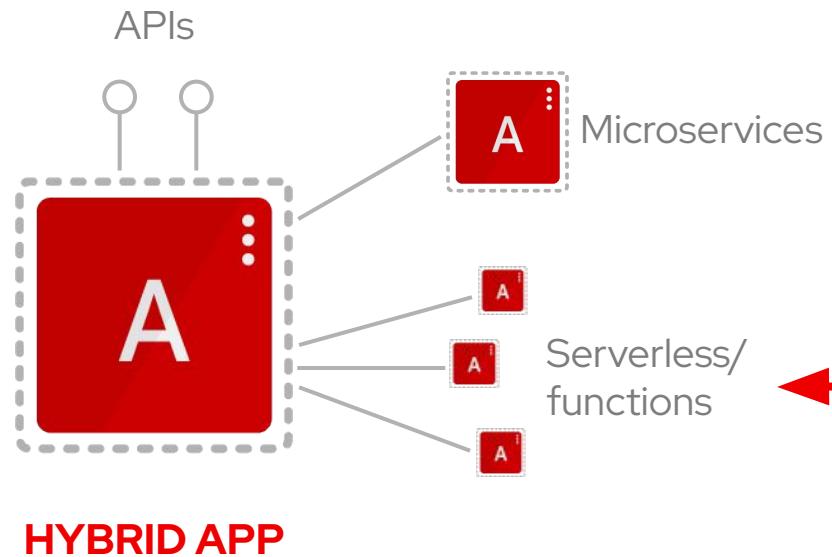
Data  
Integration

How do you drive innovation to meet business expectations while keeping the lights on?



**Leveraging the cloud becomes a key strategy for success**

# Common Modernization Patterns



Phased approach in migration

## Lift and Shift

Leave the architecture alone but modernize the deployment platform. Can be used for performance increases by allowing for deploying to better hardware. Can be used to accelerate deployments and improve processes by leveraging platform automation. **Fast Monoliths.**

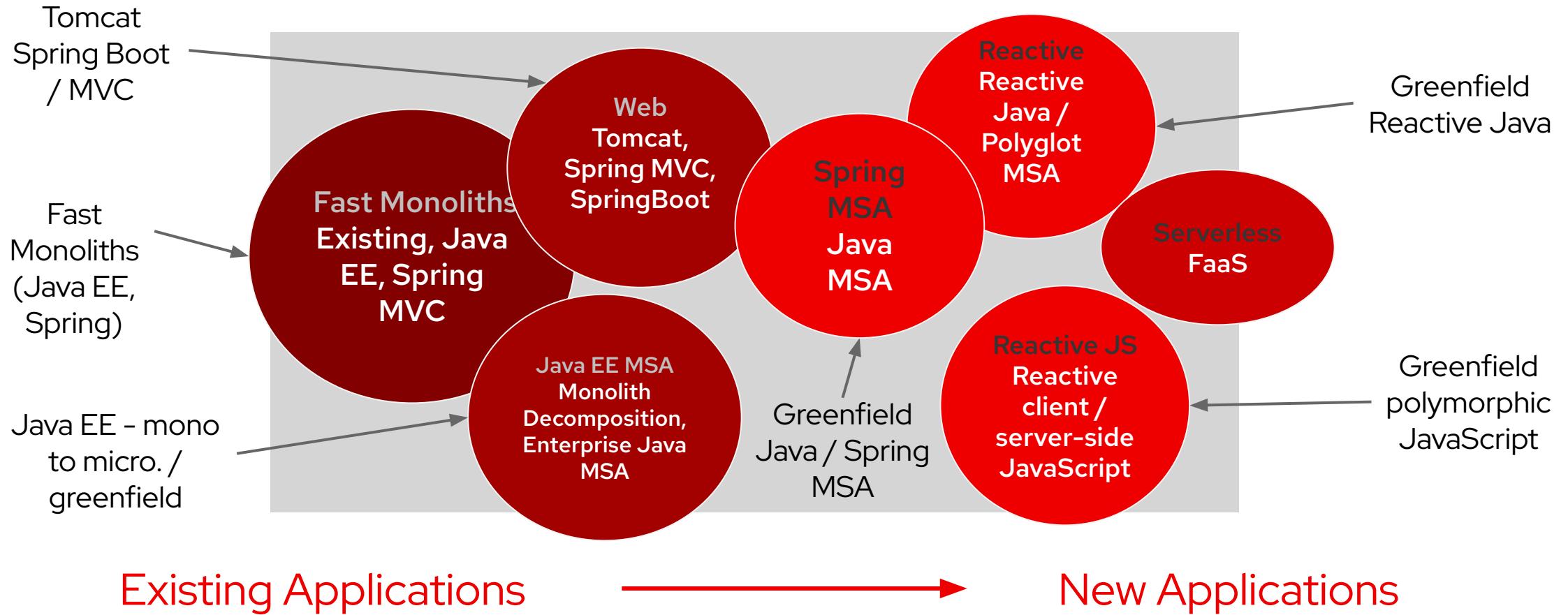
## Refactor and Augment/Extend

Find parts of the architecture that are sources of pain - refactor. Build **new capability in microservices** with well-defined APIs. Wrap legacy software too brittle to change with **adapter layers**.

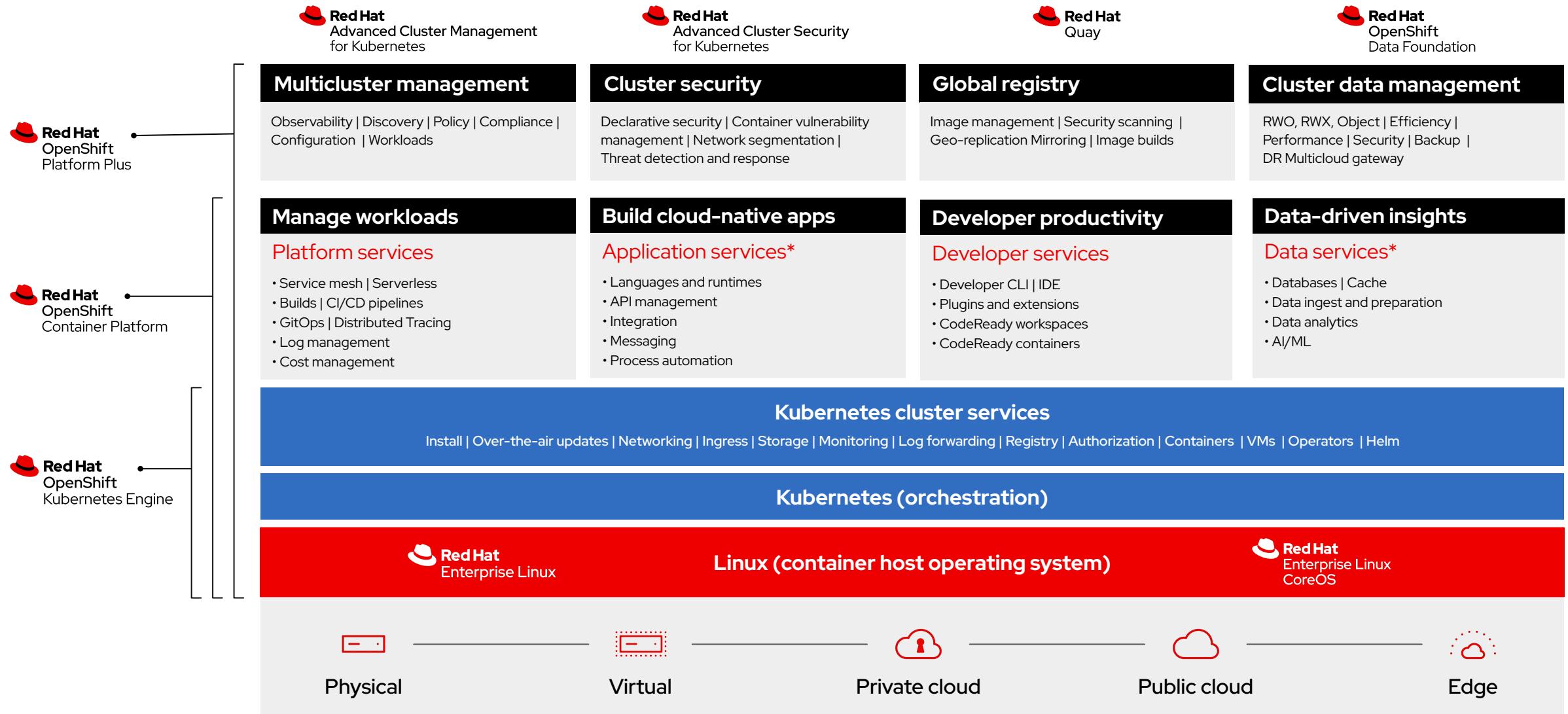
## Rewrite/Replace

Create new functionality to **replace existing functionality**. Likely expensive and time-consuming. Typically only recommended when legacy vendors go away or a major skills gap forces it.

# The Spectrum Of Enterprise Apps



## DEVELOPING CONTAINER & CLOUD-NATIVE APPLICATIONS



# Red Hat Cloud-Native Application Platform

Our vision is to simplify the creation of cloud-native services and serverless functions with a rich set of components and tools to match the **workloads** of modern cloud native apps.

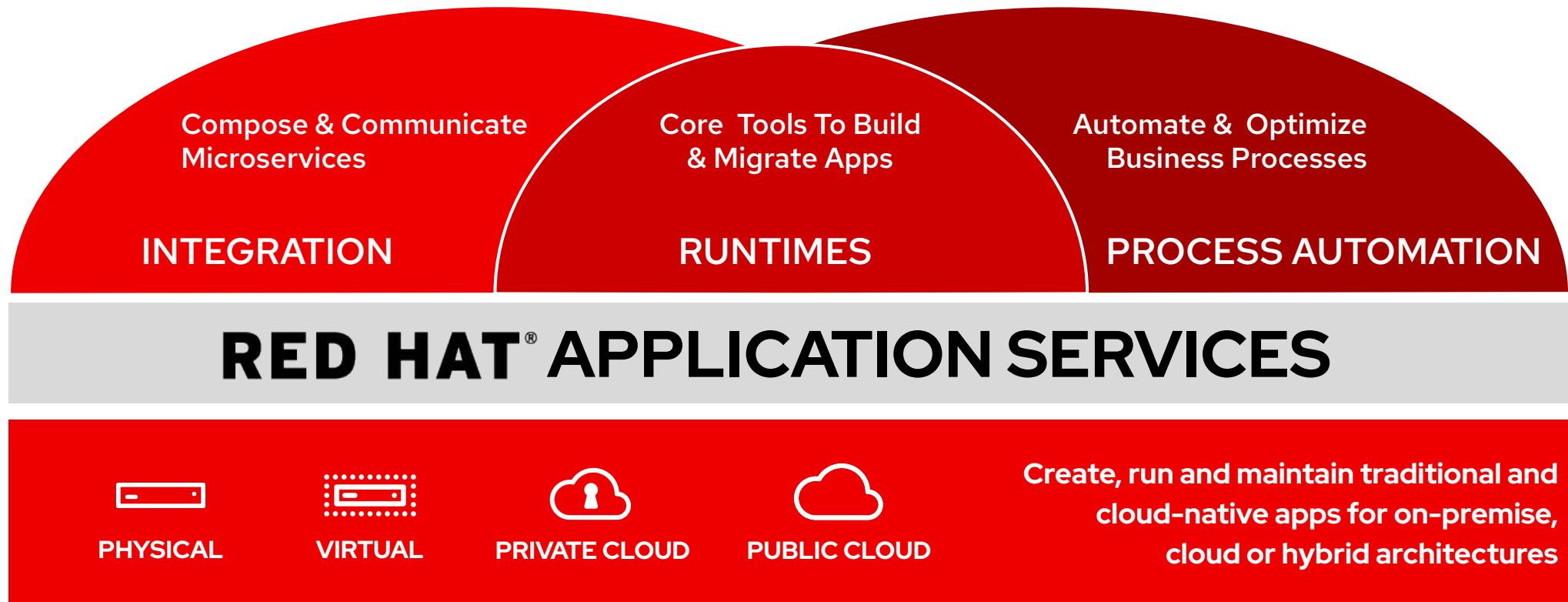
Automate Kubernetes application operations with DevOps in mind

Runtimes, frameworks and services to build applications natively on Kubernetes

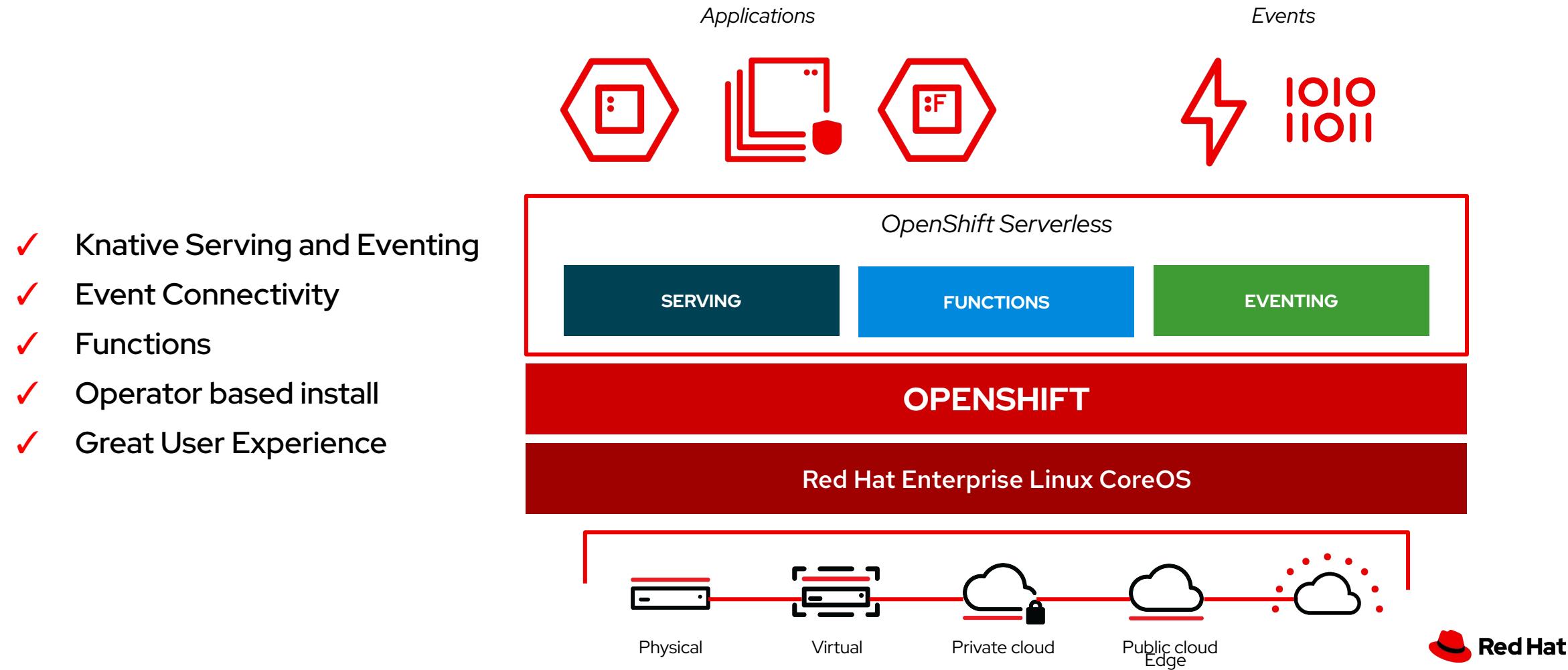
Tools and standard processes to increase developer productivity on Kubernetes



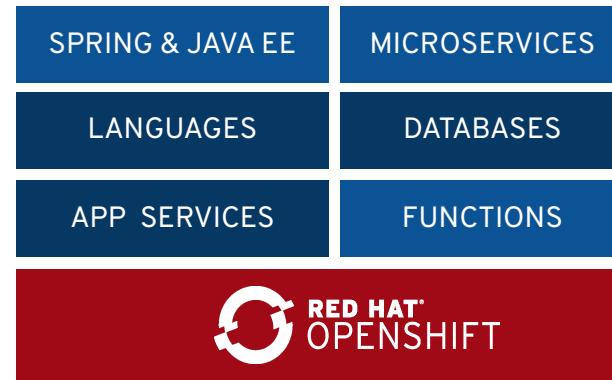
# Red Hat Cloud-Native Application Platform



# OpenShift Serverless

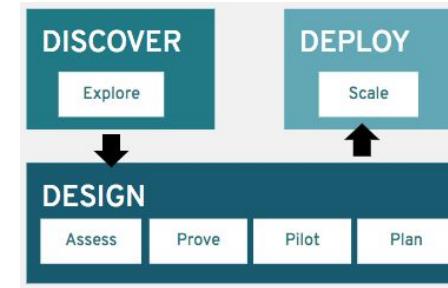


# Why is Red Hat the best choice for container and cloud-native applications?



Standardize on a set of best-of-breed open source runtimes and frameworks

Simplify development through native integration with OpenShift and Kubernetes Services



Support modernization initiatives at any speed with lift and shift, replatform and refactor



Match application requirements to polyglot runtime/framework ("right tool for the right job")

# Develop New Applications

Select The “Right Tool For The Right Task” For Your Applications



VERT.X



**Red Hat**  
JBoss Enterprise  
Application Platform



Apache Tomcat



**Red Hat**  
JBoss  
Web Server



QUARKUS



JAKARTA EE

- ▶ Established and emerging runtimes, frameworks, and languages
- ▶ Leverage your developers' enterprise Java expertise with minimal to no learning curve to microservices
- ▶ Prescriptive/guided development via missions and boosters

# Connect and Extend Applications Across the Hybrid Cloud

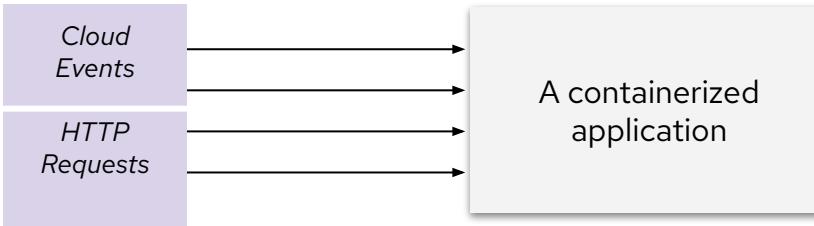
Select The “Right Tool For The Right Task” For Your Applications



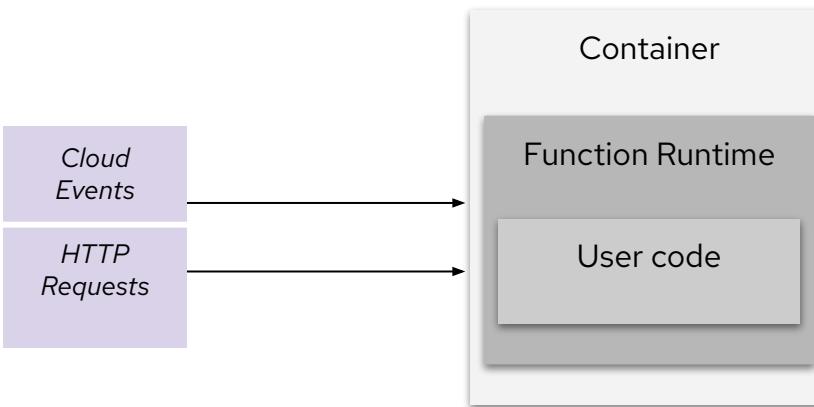
- ▶ Build applications with an API-first approach
- ▶ Build event driven apps leveraging tech like Apache Kafka for data streaming
- ▶ Compose and orchestrate microservices and serverless
- ▶ Create a multi cloud event mesh to connect the hybrid cloud
- ▶ Based on 100% open source community projects

# Serverless

## Serverless Containers



## Serverless Functions



The **Serverless** approach abstracts complexity delivering on the utility premise.

# Red Hat Integration

## Data Integration

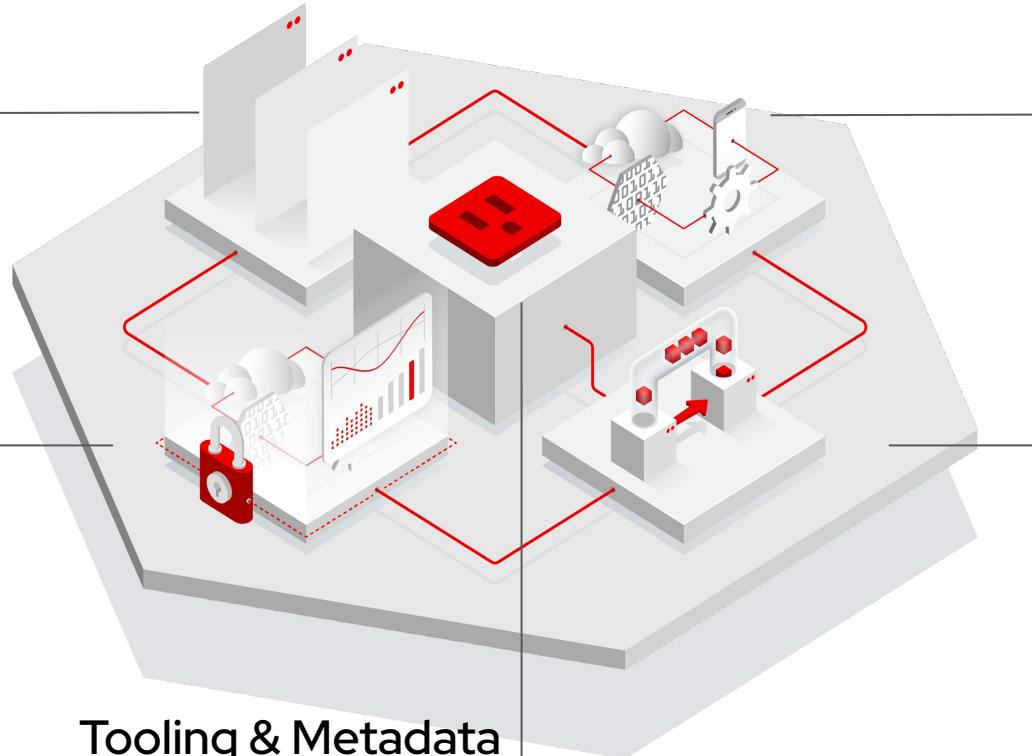
- ▶ Change Data Capture with Debezium

## API Management

- ▶ API Manager
- ▶ API Gateway
- ▶ Istio Service Mesh Adapter

## Tooling & Metadata

- ▶ Service Registry
- ▶ API Designer
- ▶ Integration Operator



## Enterprise Integration

- ▶ Comprehensive connectors
- ▶ Microservices orchestration
- ▶ Data Transformation
- ▶ Low-code iPaaS
- ▶ Kubernetes connectivity with Camel K

## Events & Messaging

- ▶ JMS Message Broker
- ▶ Wide Area Routing
- ▶ Data Streaming with Apache Kafka

Interface

Application concerns

Microservices Connectivity

Infrastructure concerns

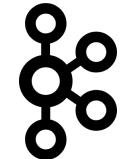
#### API

- ▶ API Contracts
- ▶ Monetisation
- ▶ Business strategic policy enforcement
- ▶ Partner ecosystem



#### APPLICATION

- ▶ Choice of language/framework
- ▶ Self-service / productivity
- ▶ Low mem, fast startup
- ▶ Integrations framework



#### SERVICE TO SERVICE

- ▶ Network resilience
- ▶ Service security
- ▶ Policy enforcement
- ▶ Metrics/Observability
- ▶ Load balancing



#### DEPLOYMENT PLATFORM

- ▶ Reliability
- ▶ Instance placement
- ▶ Scaling/autoscaling
- ▶ Resource usage
- ▶ Job scheduling
- ▶ Distributed Logging



Interface

Application concerns

Microservices Connectivity

Infrastructure concerns

#### API\*\*

- ▶ API Contracts
- ▶ Monetisation
- ▶ Business strategic policy enforcement
- ▶ Partner ecosystem

#### APPLICATION

- ▶ Choice of language/framework
- ▶ Self-service / productivity
- ▶ Low mem, fast startup
- ▶ Integrations framework

#### SERVICE TO SERVICE

- ▶ Network resilience
- ▶ Service security
- ▶ Policy enforcement
- ▶ Metrics/Observability
- ▶ Load balancing

#### DEPLOYMENT PLATFORM

- ▶ Reliability
- ▶ Instance placement
- ▶ Scaling/autoscaling
- ▶ Resource usage
- ▶ Job scheduling
- ▶ Distributed Logging



# Red Hat Application Services



Interface

Application concerns

Microservices Connectivity

Infrastructure concerns

- API\*\***
  - ▶ API Contracts
  - ▶ Monetisation
  - ▶ Business strategic policy enforcement
  - ▶ Partner ecosystem
- APPLICATION**
  - ▶ Choice of language/framework
  - ▶ Self-service / productivity
  - ▶ Low mem, fast startup
  - ▶ Integrations framework
- SERVICE TO SERVICE**
  - ▶ Network resilience
  - ▶ Service security
  - ▶ Policy enforcement
  - ▶ Metrics/Observability
  - ▶ Load balancing
- DEPLOYMENT PLATFORM**
  - ▶ Reliability
  - ▶ Instance placement
  - ▶ Scaling/autoscaling
  - ▶ Resource usage
  - ▶ Job scheduling
  - ▶ Distributed Logging



**Red Hat**  
Application  
Services



**Red Hat**  
OpenShift

# Cloud Services Managed by Red Hat

Managed OpenShift + Application Services + Data Services



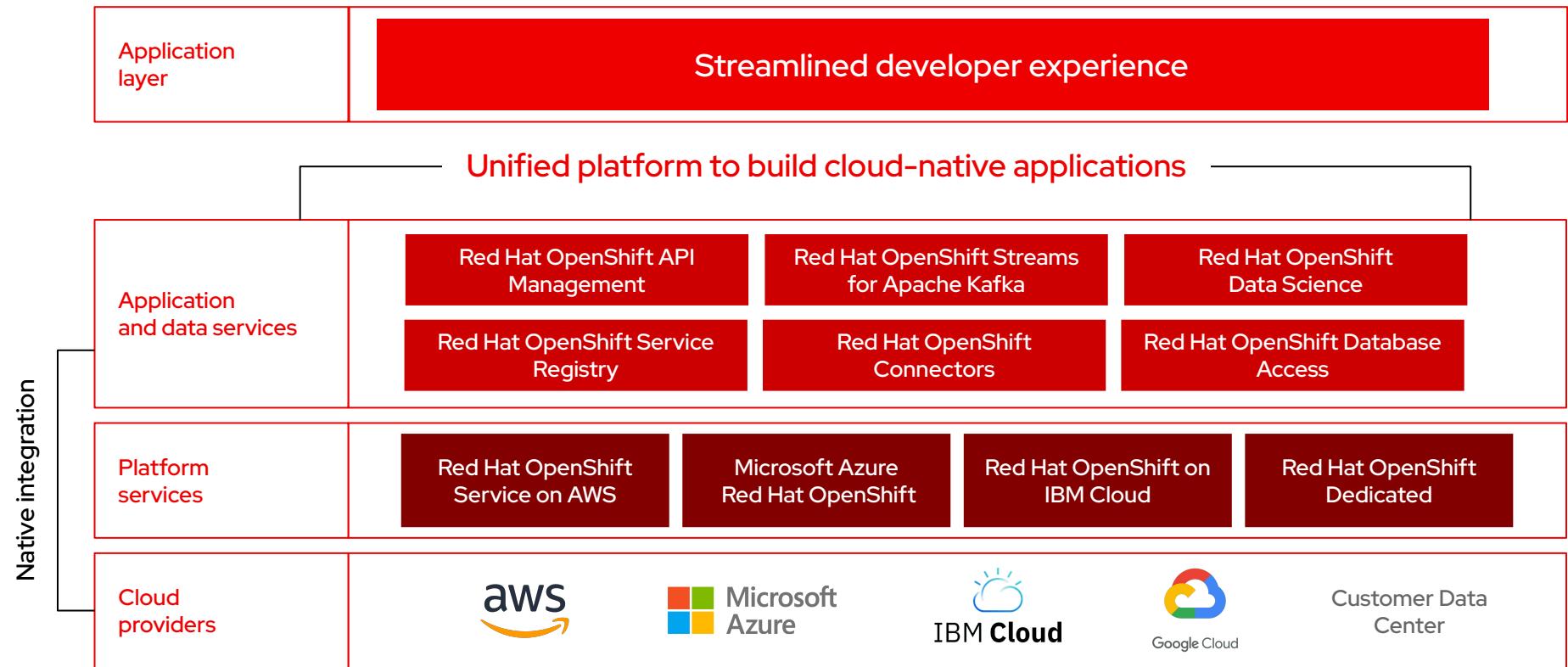
Full stack management  
and unified experience

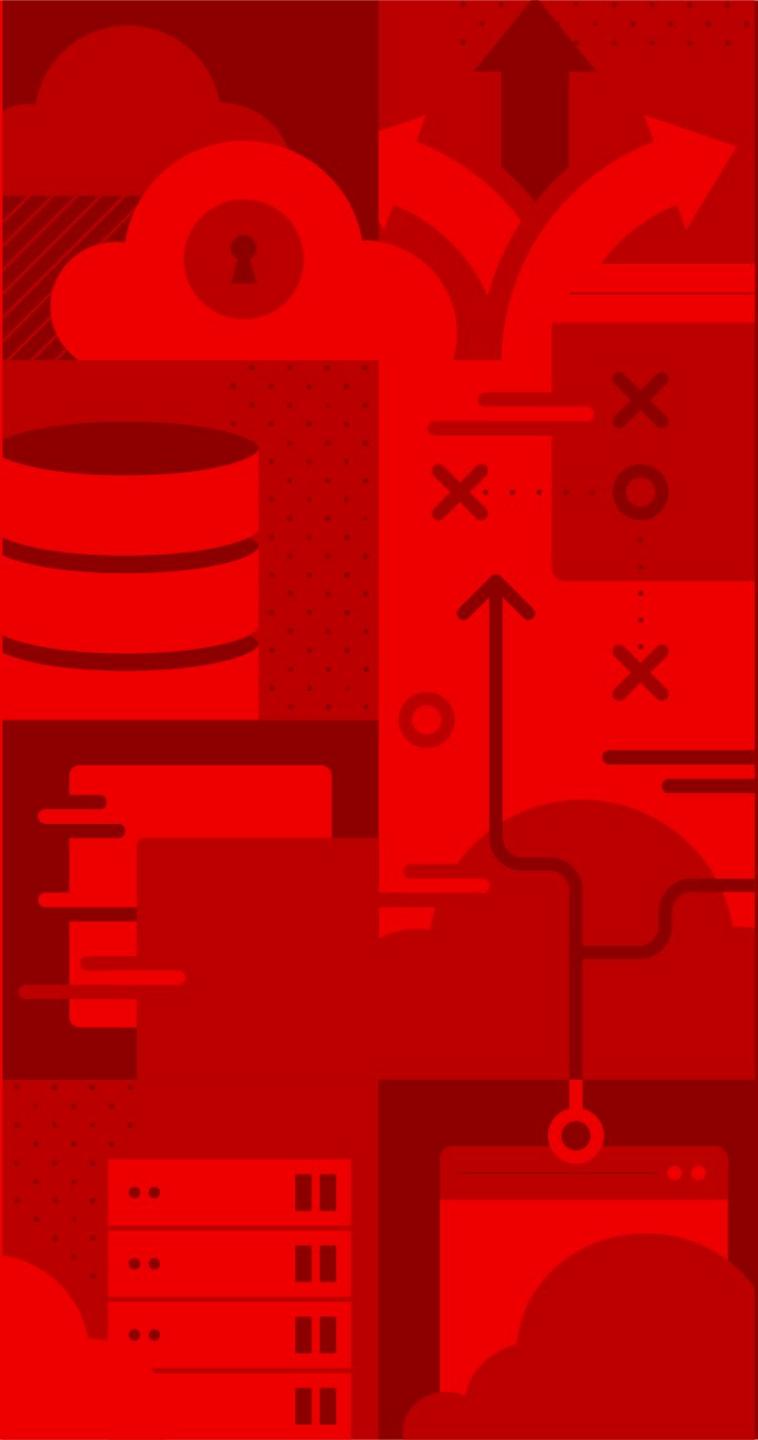


Maximize full value of  
Red Hat OpenShift



Hybrid cloud flexibility





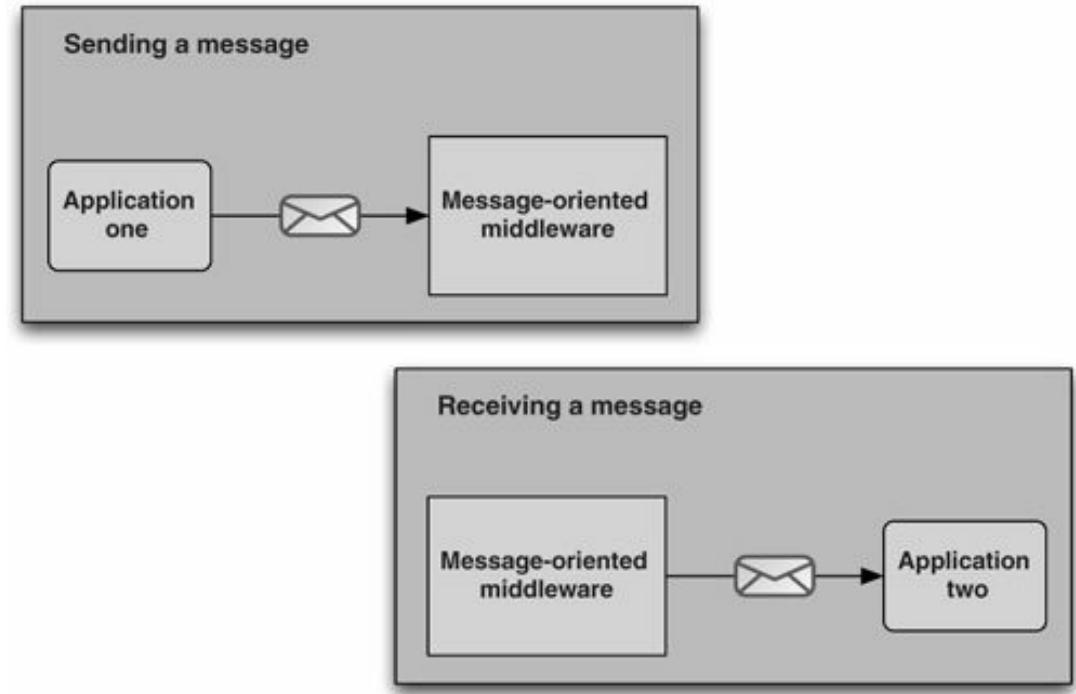
# Event Driven Architecture

---

# Introduction: Events

# What is Event-Driven Architecture?

Event-Driven Architecture (EDA) is a way of designing applications and services to respond to real-time information based on the sending and receiving of event notifications



## What is an event?

**Event** an action or occurrence recognized by software, often originating asynchronously from the external environment, that may be handled by the software

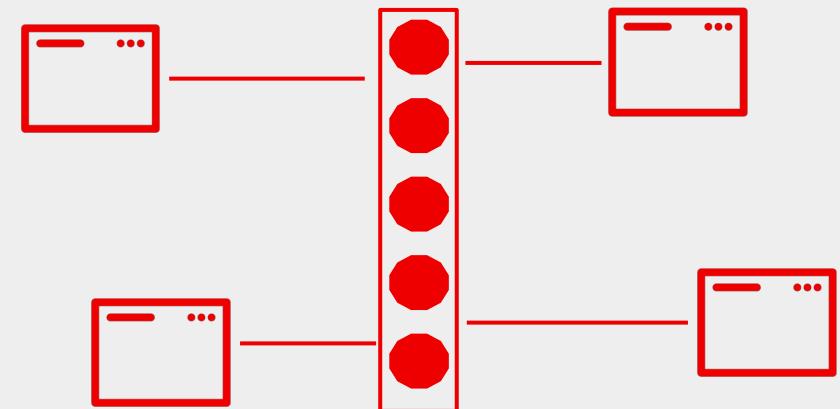
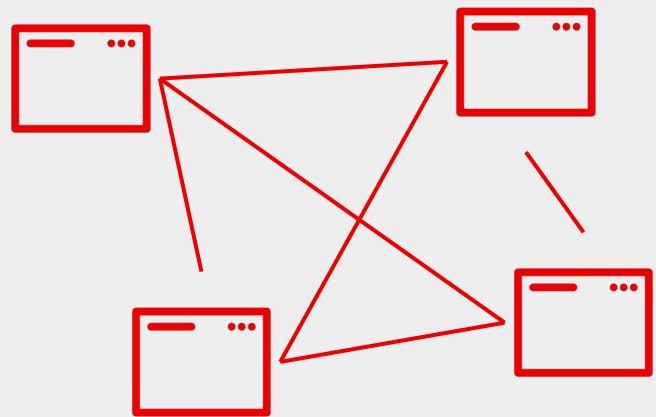
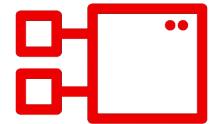


---

# Event-driven Microservices

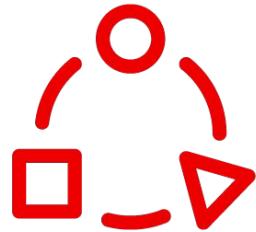
# Microservices Async Communication

Foundation for event-driven microservices

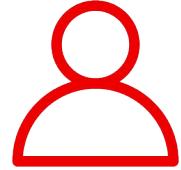


- High availability: No dependency on other services
- Autonomy in services with independent evolution

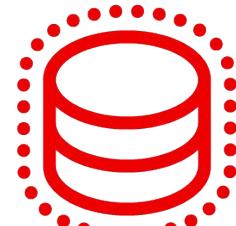
# Event-driven architecture use cases



Reactive notification



Behavior capture



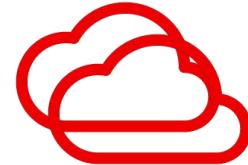
Cache store



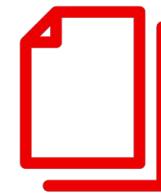
Complex event processing



Command query  
responsibility segregation  
(CQRS)

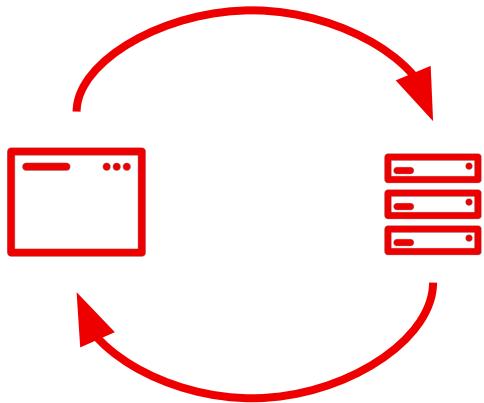


Streaming between data centers

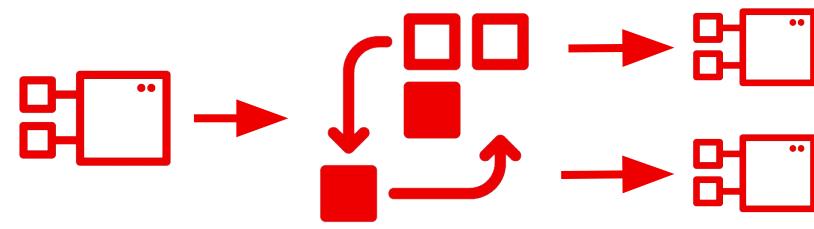


Auditing

## Request-reply & Event-driven



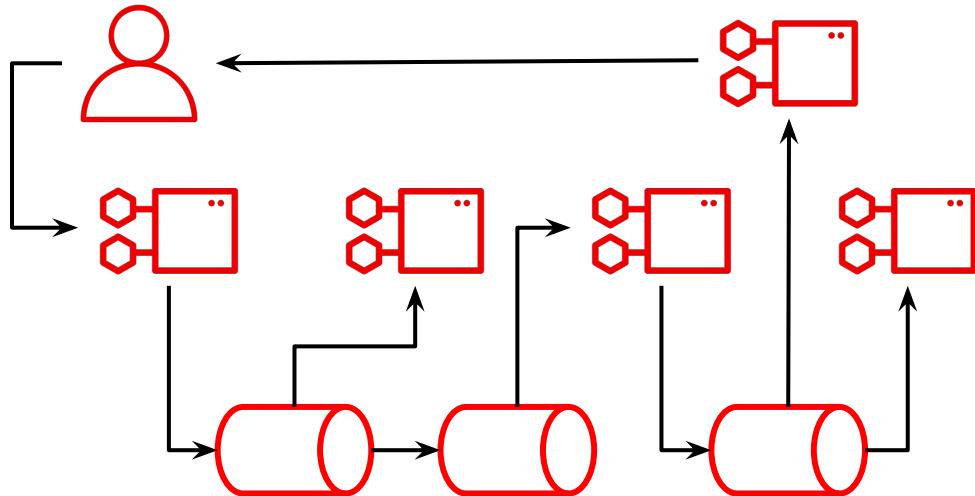
Synchronous & ephemeral  
Low composability  
Simplified model  
Low tolerance to failure  
Best practices evolved as REST



Asynchronous and persistent  
Decoupled  
Highly composable  
Complex model  
High tolerance to failure  
Best practices are still evolving

# Connect loosely-coupled microservices

Remain agile with event-centric microservice architecture



## Connect microservices and stay agile

- ▶ Publish events to Kafka brokers and decouple the data from the event-consuming services
- ▶ Meet event volumes by independently scaling up and down your microservices
- ▶ Avoid hard-coding integrations and connections between microservices applications

*The complete vision is delivered by Red Hat Integration*

Apache Kafka  
Community

Red Hat  
AMQ  
streams on RHEL

Apache Kafka +  
Bridge +  
Cruise Control

Red Hat  
AMQ  
streams on OCP

AMQ Streams +  
OpenShift  
Operators

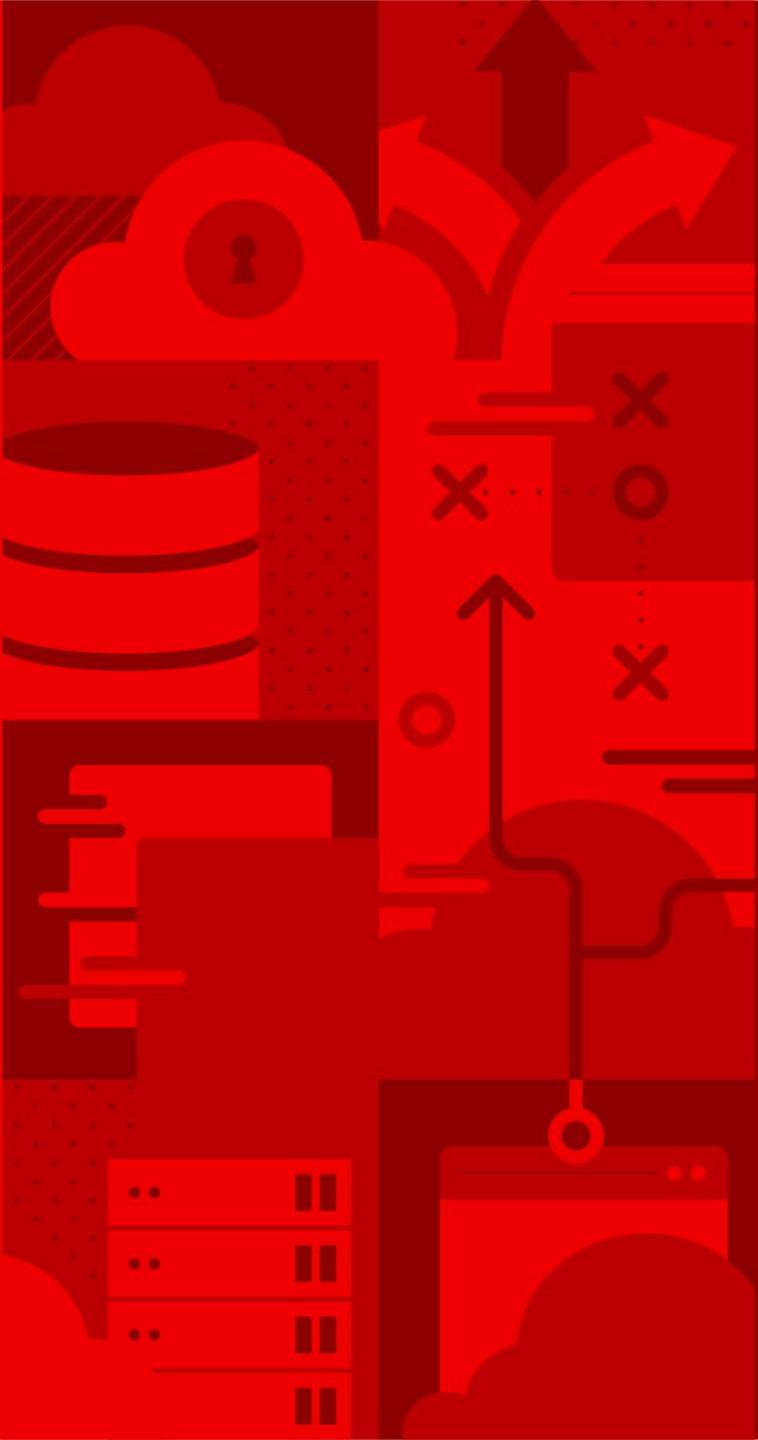
Competitor  
Platform

Apache Kafka +  
Kubernetes  
Operator +  
Kafka Connectors +

**KSQL + Replicator  
+ Control Center**

Red Hat  
Integration

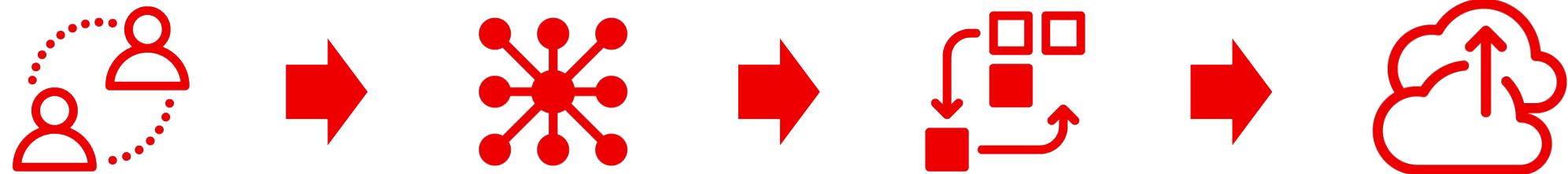
AMQ Streams +  
OpenShift  
Operators + Service  
Registry + Debezium  
Connectors + Camel  
Kafka Connectors +  
AMQ + Fuse  
Connectors + API  
Management +  
Single Sign On +  
Quarkus +  
Serverless (Knative)



# Enterprise Integration

# Evolution of Integration

## Architecture



### Point to Point

Direct connection between systems, application both internally and with external services

### Enterprise Service Bus

Placing a centralized bus that integrate between loosely coupled services.

### Microservices

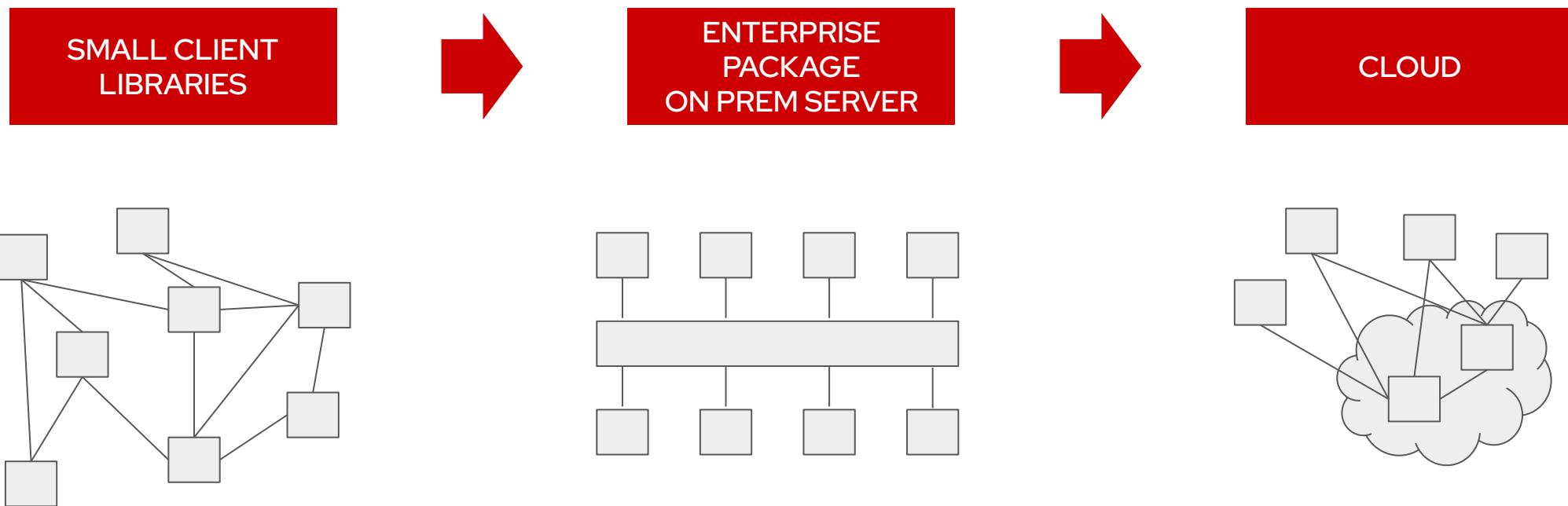
Fine grained distributed services, allowing faster turnover rate, more agile and flexible deployment model.

### Serverless

Scale down to zero. Optimize Resource Usage. Avoid random, arbitrary workload prediction

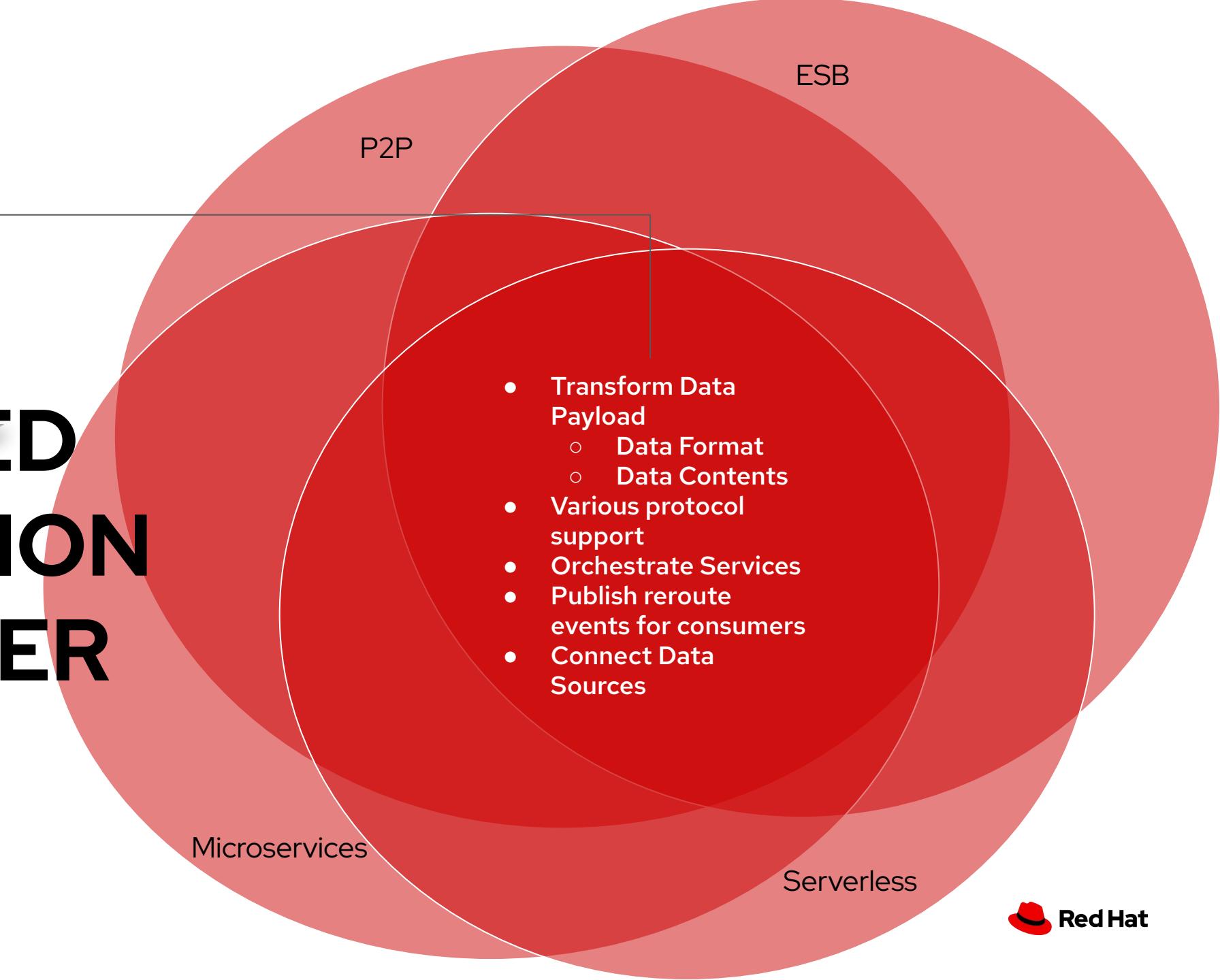
# Evolution of Integration

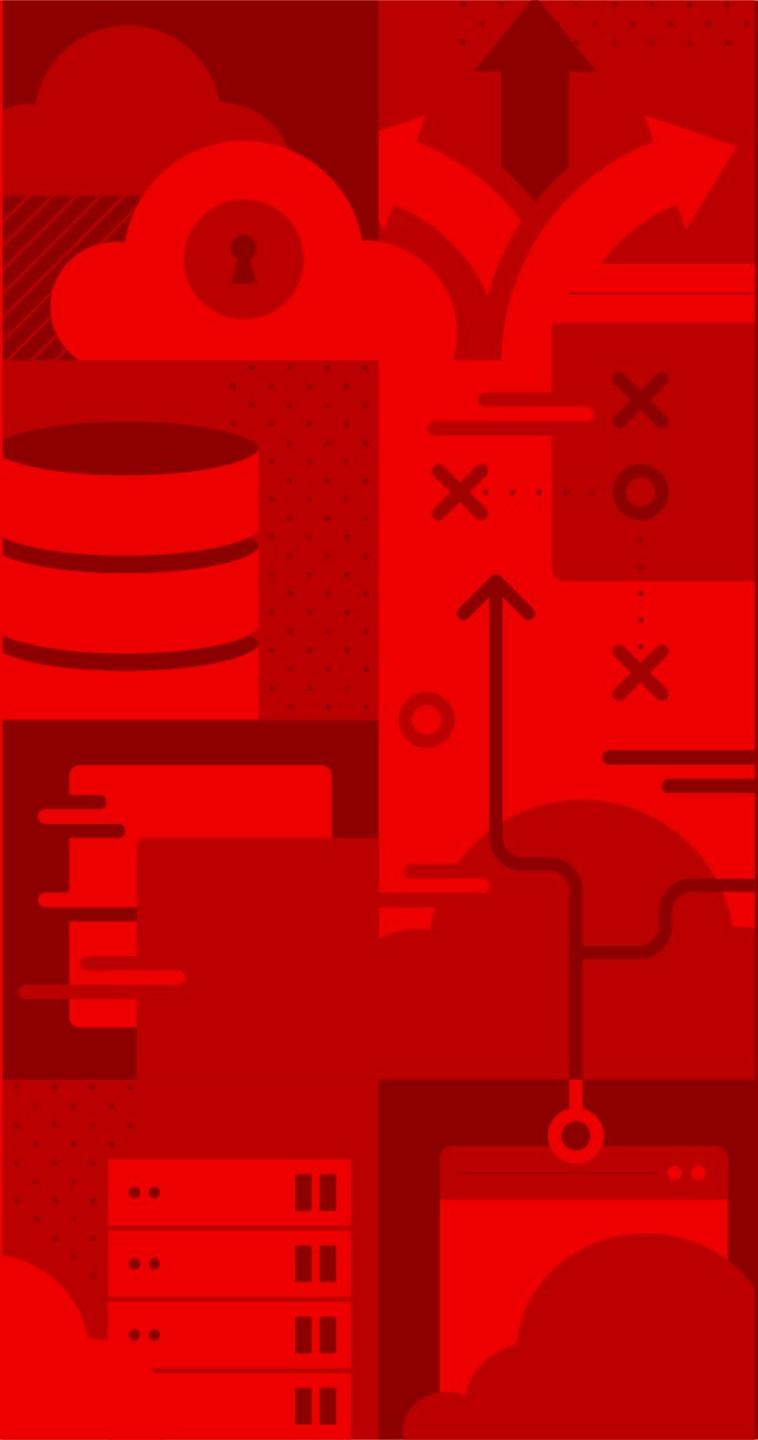
Application Delivery





**YOU NEED  
INTEGRATION  
NO MATTER  
WHAT!**





# Serverless Integration

## Serverless Market Trends

*"Use Serverless To optimize The Benefits of The cloud"*<sup>2</sup>

40%

of enterprises adopted Serverless technologies or practices with expected growth coming in the next 12 to 18 months.<sup>1</sup>



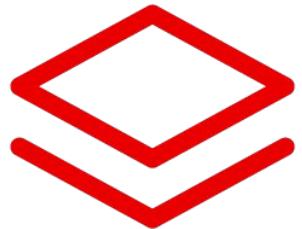
**Vendor lock-in** is the second biggest concern when adopting Serverless technologies.<sup>1</sup>

60%

of the serverless practitioners reported "*reduction of operational costs*" with the second biggest benefit being "*scale with demand automatically*"

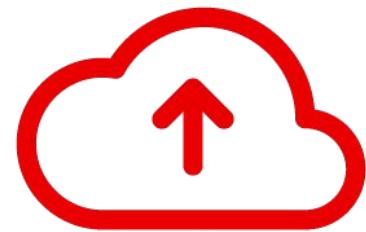
# Serverless

*"The future of cloud computing"*



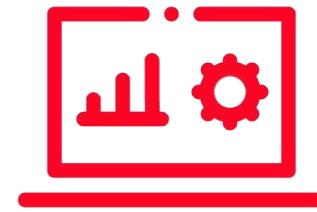
**Beyond PaaS**

Focus on Code



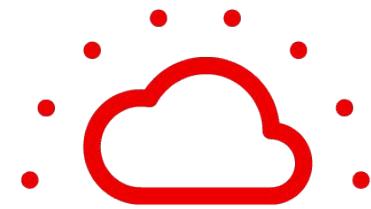
**Scale by Demand**

Scale to Zero



**Optimize Resource Usage**

Balancing workload



**Event Driven**

Near real-time  
latency



# Serverless Benefits



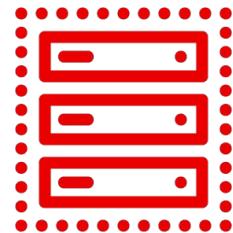
Avoid random,  
arbitrary prediction



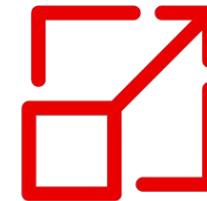
Lower operational  
cost



Faster time to  
market



Reduced packaging  
and deployment  
complexity

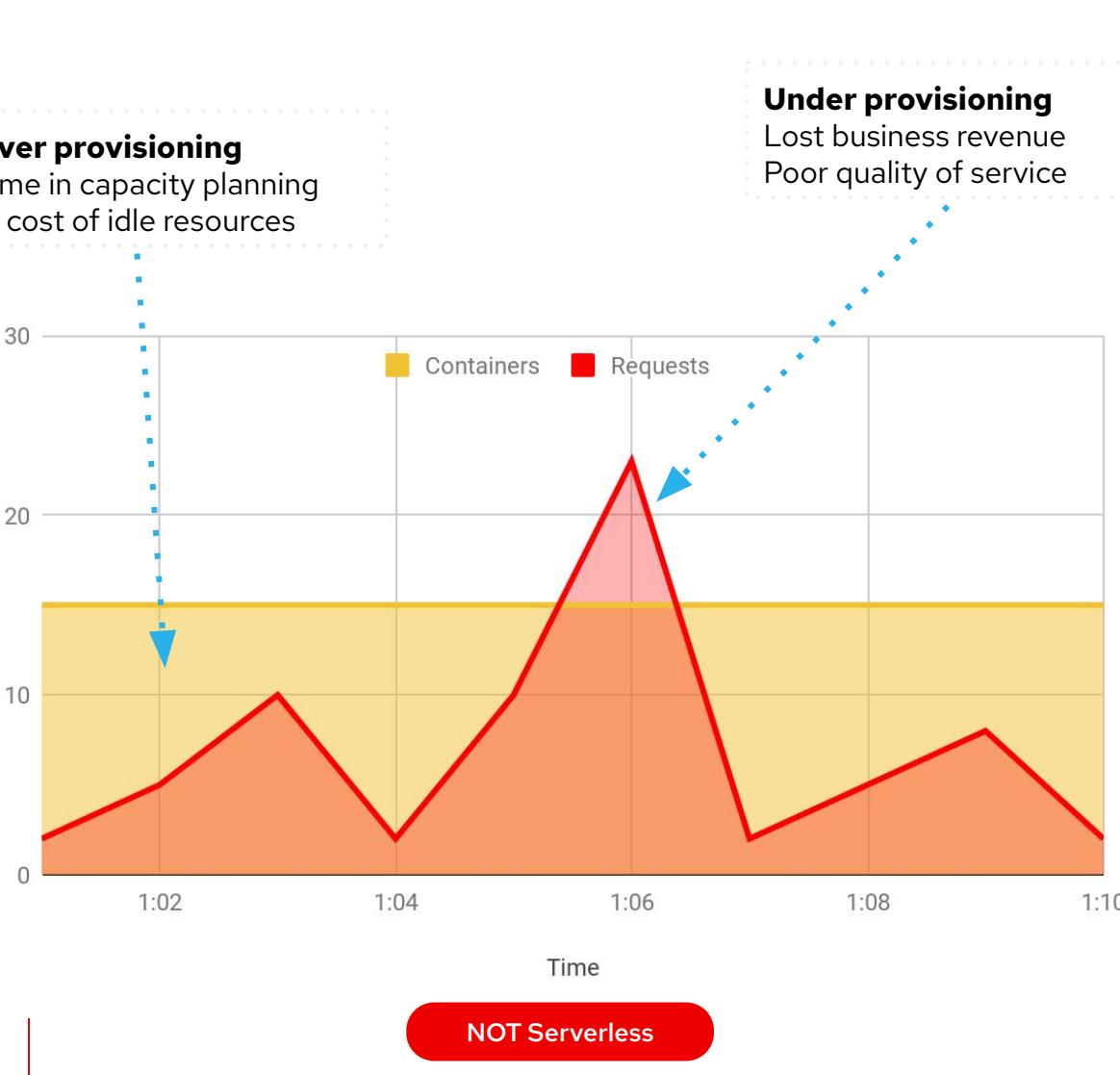


Flexible Scalability  
on-demand

# Serverless Operational Benefits

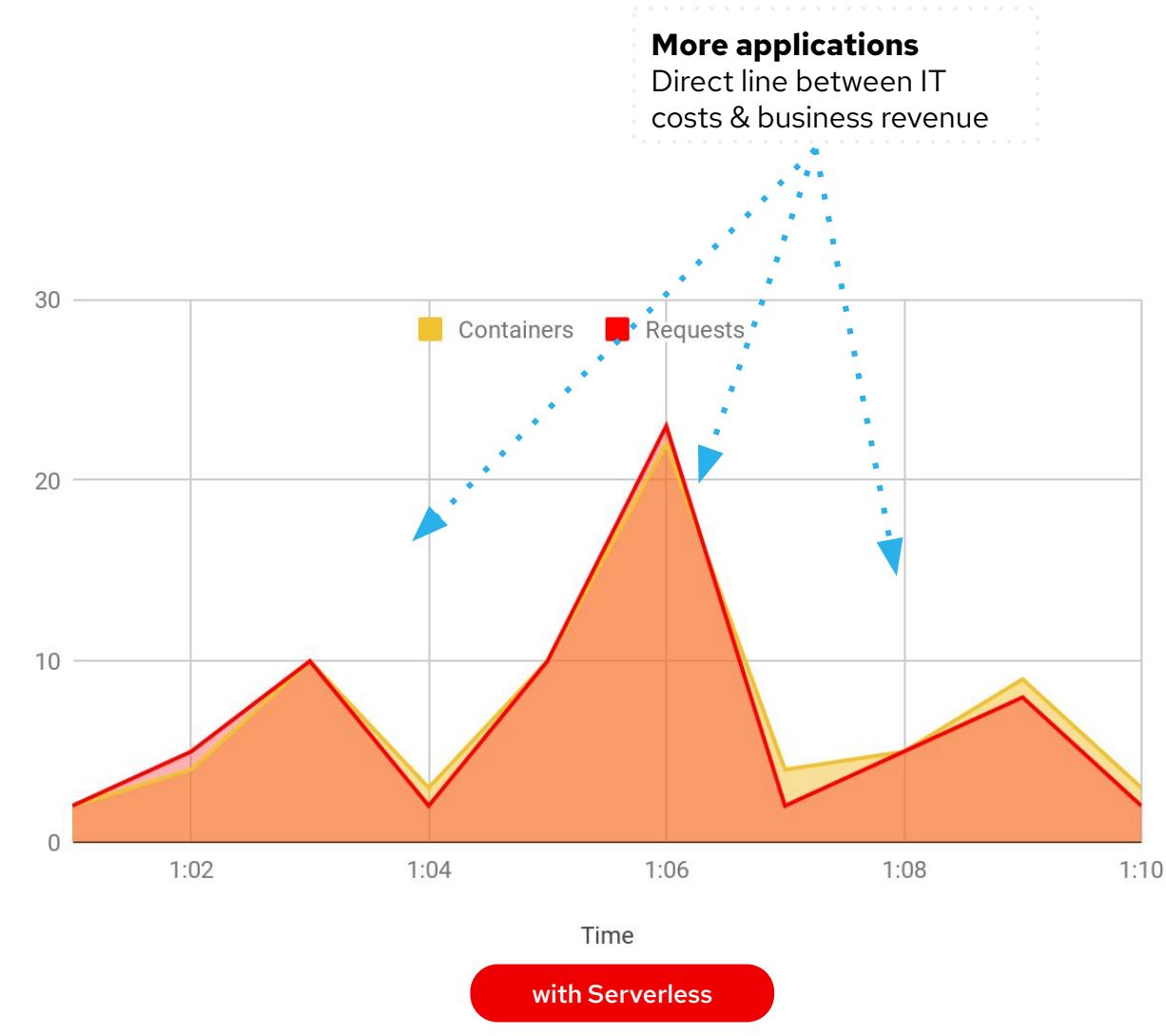
## Over provisioning

Time in capacity planning  
IT cost of idle resources

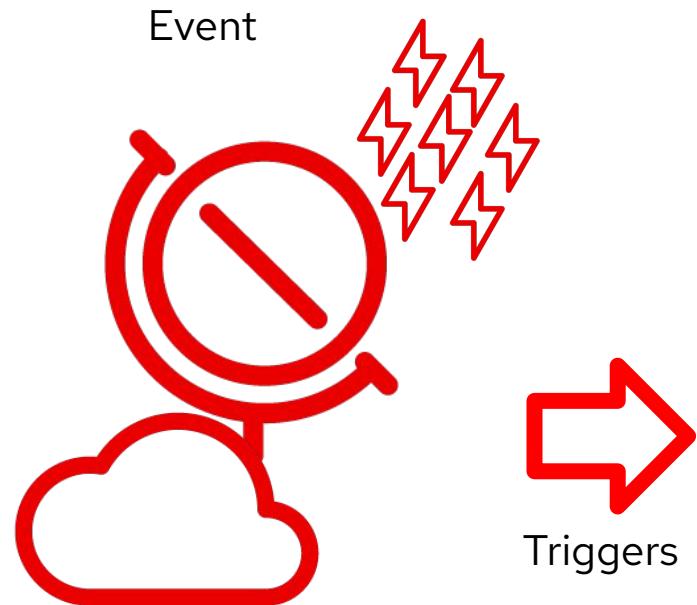


## More applications

Direct line between IT costs & business revenue



# Serverless Function/Application Behaviour



Serverless  
Application/Function

- Fast boot up time
- Stateless
- Natively consuming cloud events
- Code on the fly, no more packaging/build and auto deployment

# Types of Serverless functions



## Collectors

Source to Sink, header filtering

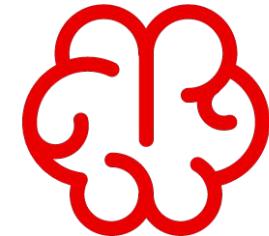
ie: Camel Source



## Connectors

Data format, aggregate, split,  
orchestrate events, content filters

ie: Sequence, Trigger, Camel  
Source



## Task

Work load, and process that you will  
normally do

ie: knative service



## DevOps and Serverless

"DevOps Made Easy"

Quick turnaround  
time for deployment  
and switch between  
versions

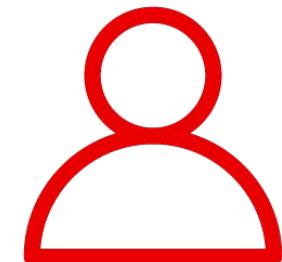
Centralize place to  
manage and apply  
scaling policy for  
serverless in platform

Setup event mesh  
with ease with various  
<sup>40</sup>implementations

Running revisions

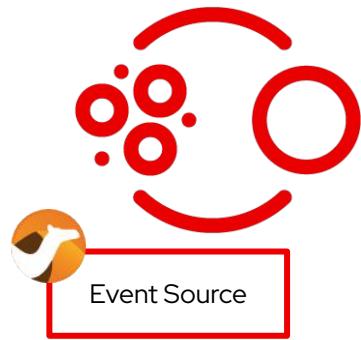
Scaling Policy

Event Mesh



DEVOPS

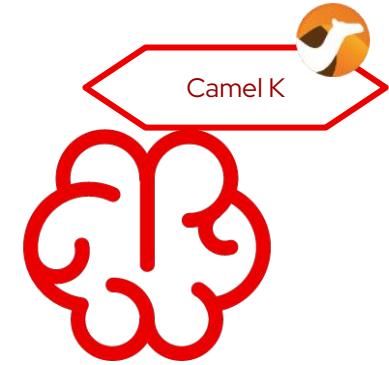
# Serverless Event Mesh



Receive or Polling events  
Wraps payload to Cloud Events

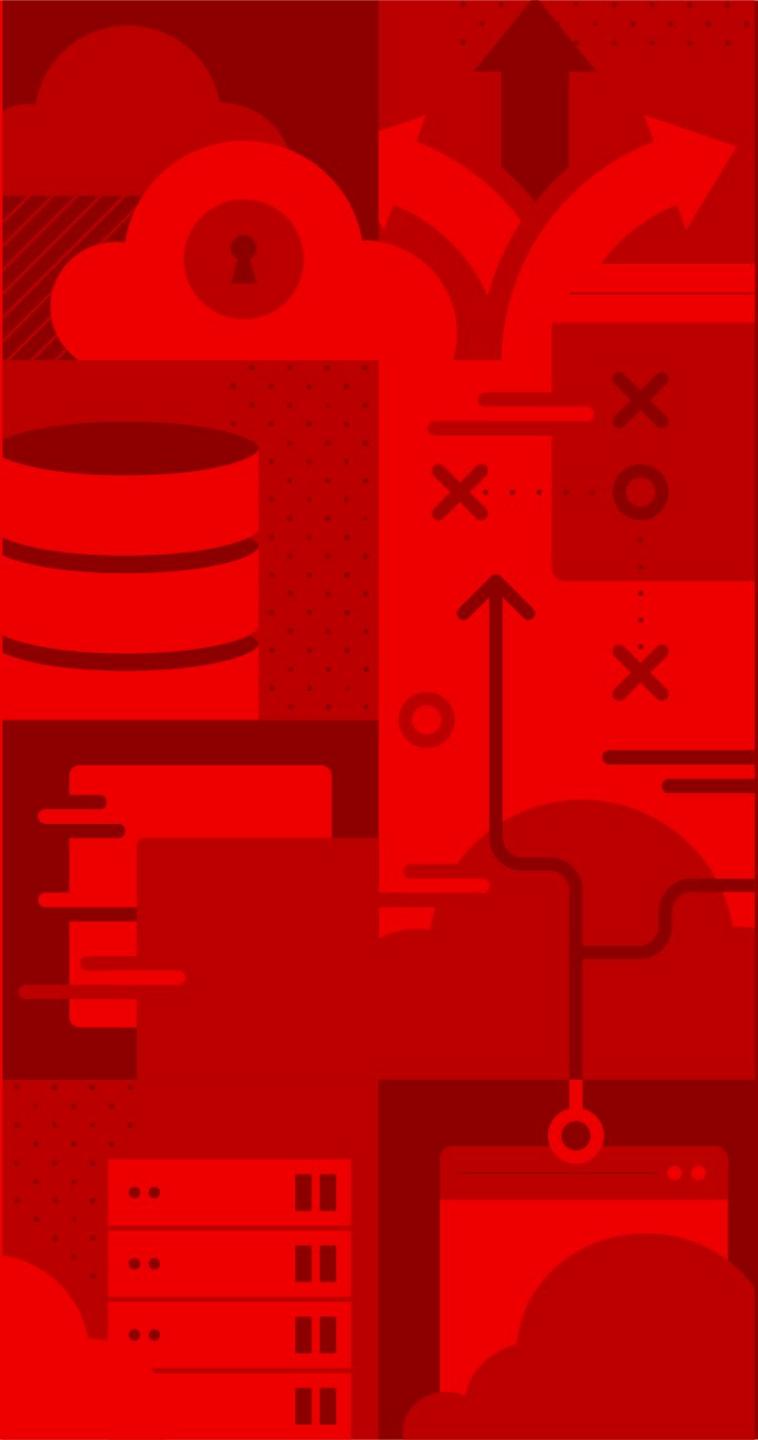


Event Mesh channels



Wraps payload to Cloud Events  
Data format  
Payload aggregate and split  
Orchestrate events  
EIP  
Push event to external





# Red Hat Integration Operator



Red Hat Integration Operator allows you to **choose** and **install** the Operators that manage your Red Hat Integration **components**.

# Red Hat Integration Operator availability

The screenshot shows the Red Hat Marketplace interface. At the top, there's a navigation bar with a search icon and a user profile icon. Below it, a large card for 'Red Hat Integration' by Red Hat is displayed. The card features a red hat icon, the product name, and a subtitle 'Connect application, data and streams across hybrid clouds'. It includes a 'Certified enterprise ready' badge and a 'About certification' link. Two prominent buttons at the bottom are 'Purchase' (in red) and 'Free trial'. Below these buttons, technical details are listed: 'Software version 2021-Q1', 'Runs on OpenShift 4.6.x+', 'Delivery method Operator', and a 'Rating' section showing 5 stars and 6 reviews. At the very bottom of the card, there's a brief description of the product's capabilities.

## Red Hat Marketplace

Go to the Red Hat Marketplace website, search for the RH Integration Operator, and install it in on-premise OCP cluster.

The screenshot shows the OperatorHub interface. At the top, there's a header with the Red Hat logo and a close button. Below the header, a card for 'Red Hat Integration' is shown. The card includes an 'Install' button, the 'Latest Version' (1.0.5), and a 'Capability Level' section with options like 'Basic Install' (selected), 'Seamless Upgrades', 'Full Lifecycle', 'Deep Insights', and 'Auto Pilot'. Below this, sections for 'Provider Type' (Red Hat), 'Provider' (Red Hat), and 'Repository' (<https://github.com/redhat-integration/integration-operator>) are displayed. At the bottom, there's a table for 'Installation options' with columns for 'Operator name', 'Update Channel', 'Default Installation mode', and 'Default namespace'.

## OperatorHub

Go to the OperatorHub page in my on-premise OCP or OSD console, search for the RH Integration Operator, and install it.

# Operators available for installation

- ▶ 3scale
- ▶ 3scale APIcast
- ▶ AMQ Broker
- ▶ AMQ Interconnect
- ▶ AMQ Streams
- ▶ API Designer
- ▶ Camel K
- ▶ Fuse Console
- ▶ Fuse Online
- ▶ Service Registry

# 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](https://www.linkedin.com/company/red-hat)

 [youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)

 [facebook.com/redhatinc](https://www.facebook.com/redhatinc)

 [twitter.com/RedHat](https://twitter.com/RedHat)