redhat.

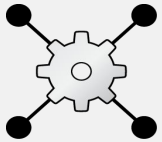
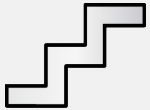
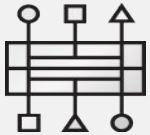
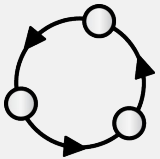
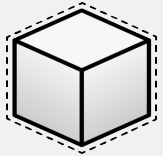
RED HAT JBOSS FUSE 6.3 UPDATE

Fuse Integration Services for OpenShift

2017 February

CHANGING LANDSCAPE FOR INTEGRATION

Enterprise IT is undergoing fundamental change. To remain competitive, businesses need an integration platform capable of supporting current ***and*** next generation architectures.

Service Endpoints	Architecture	Development Process	Deployment	Infrastructure
				
Webservices	Monolith	Waterfall	Server/VM	Data Center
				
APIs	Microservices	CI/CD	Container	Cloud

WHAT IS FUSE INTEGRATION SERVICES?

Fuse Integration Services (FIS) is a distribution of JBoss Fuse that provides tooling and runtime support for creating containerized integration services on OpenShift, including

- Docker-formatted container images
- Tooling to create, develop and build containerized Fuse applications
- Self-service deployment templates for common integration scenarios
- Native integration with Kubernetes for service discovery, clustering, and configuration management
- All based on the core technologies available in JBoss Fuse

LIGHTWEIGHT INTEGRATION RUNTIME

Docker-formatted container images for Karaf and Spring Boot provide the foundation for a built-for-purpose containerized integration runtime.

SPRING BOOT RUNTIME

Application Code and Configuration

Application-Specific Fuse Dependencies

Spring Boot

`fis-java-openshift`

- Convention over configuration, bean-driven container
- Supports Spring and Java DSL for Apache Camel
- Autowired configuration
- Based on Camel 2.18, ideal for development of lightweight integration microservices

KARAF RUNTIME

Application Code and Configuration

Application-Specific Fuse Dependencies

Apache Karaf

`fis-karaf-openshift`

- Karaf-based OSGi container
- Supports Blueprint for Apache Camel
- Version-aligned to Fuse 6.3 to ease transition from standalone/Fabric-based Fuse deployments to OpenShift

PATTERN BASED INTEGRATION

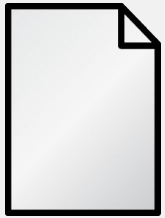
Based on Apache Camel, a powerful pattern-based integration engine with a comprehensive set of connectors and data formats to tackle any integration problem.

ENTERPRISE INTEGRATION PATTERNS

Build integrations using enterprise best practices.

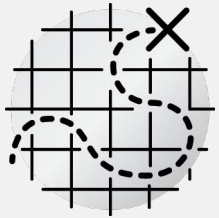
150+ COMPONENTS

Batch, messaging, web services, cloud, APIs, and more ...



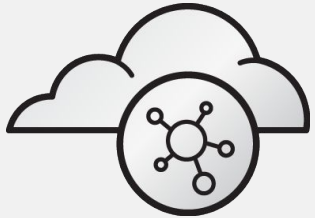
BUILT-IN DATA TRANSFORMATION

JSON, XML, HL7, YAML, SOAP, Java, CSV, Custom



INTUITIVE FRAMEWORK AND TOOLING

Build integrations quickly and easily without lock-in

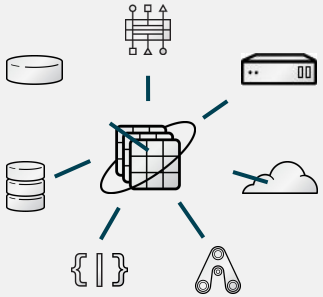


NATIVE REST SUPPORT

Create, connect, and compose APIs with ease.

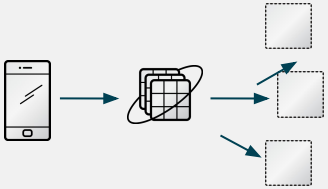
MULTIPLE INTEGRATION STYLES

A single technology stack to satisfy traditional and next generation integration requirements.



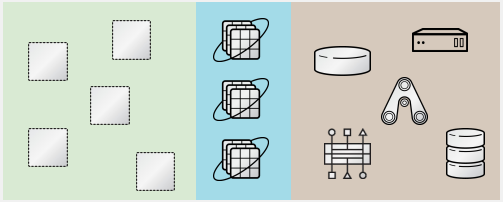
TRADITIONAL INTEGRATION

Pattern-oriented integration for on-premise and cloud-based resources.



INTEGRATION MICROSERVICES

Create and compose microservices using API and event-driven interactions.



TRANSITIONAL INTEGRATION

Blend greenfield and brownfield to deliver next generation services.

TRUSTED ENTERPRISE PLATFORM

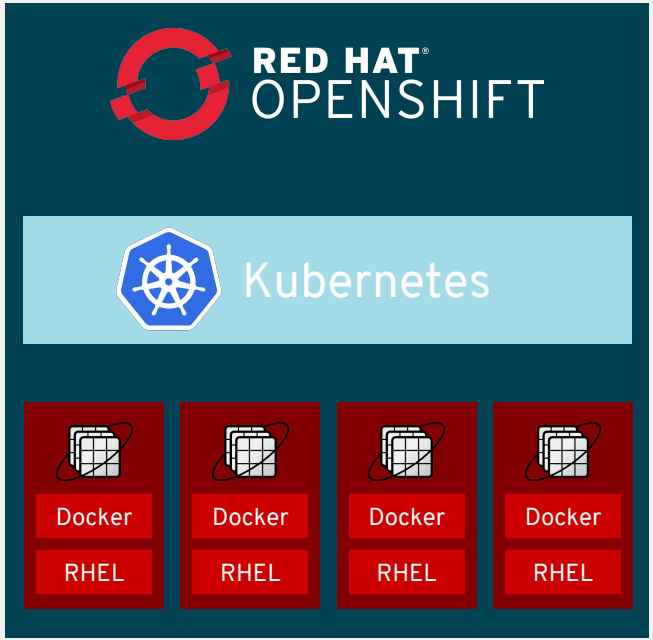
Build on a foundation of best-of-breed container technologies.

ENTERPRISE CONTAINERS

Protect services with network and failure isolation
Deploy independently for agility and flexibility
Scale on command and on demand

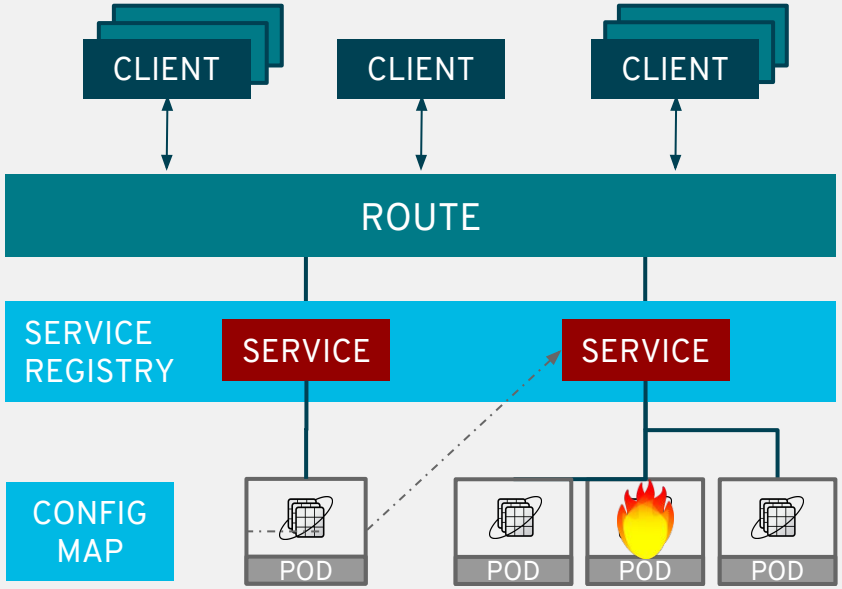
ADVANCED CONTAINER ORCHESTRATION

Allows for dynamic service discovery and invocation
Enables service resiliency and availability
Provides dynamic configuration to immutable services
Supports stateful and stateless services



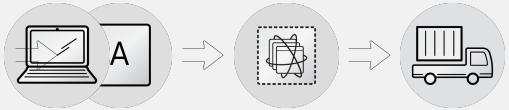
INTEGRATION AT SCALE

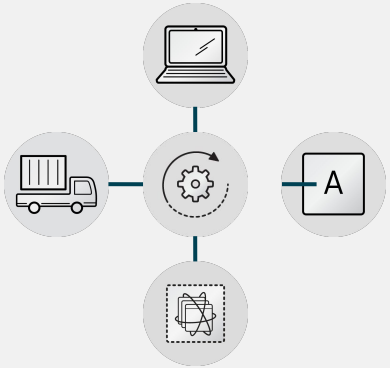
Bring agility and flexibility to integration solutions with decoupled configuration, automatic service lookup,health monitoring and ensures service resiliency.



CONTAINER-NATIVE TOOLCHAIN

Comprehensive tooling across all stages of delivery provides out-of-the-box support for continuous delivery with the flexibility to integrate with existing CI/CD solutions.





SUPPORT AT EACH STAGE OF DELIVERY

Develop, build, containerize, deploy

INCREASED AGILITY THROUGH CONTINUOUS DELIVERY

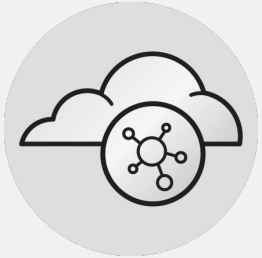
Automated delivery pipelines

HYBRID INTEGRATION PLATFORM

Single platform and toolchain across cloud environments provides consistency and flexibility for current and future deployment plans.

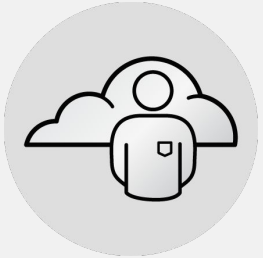
PRIVATE CLOUD

Deploy on-premise



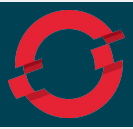
PUBLIC CLOUD

Deploy on public cloud provider



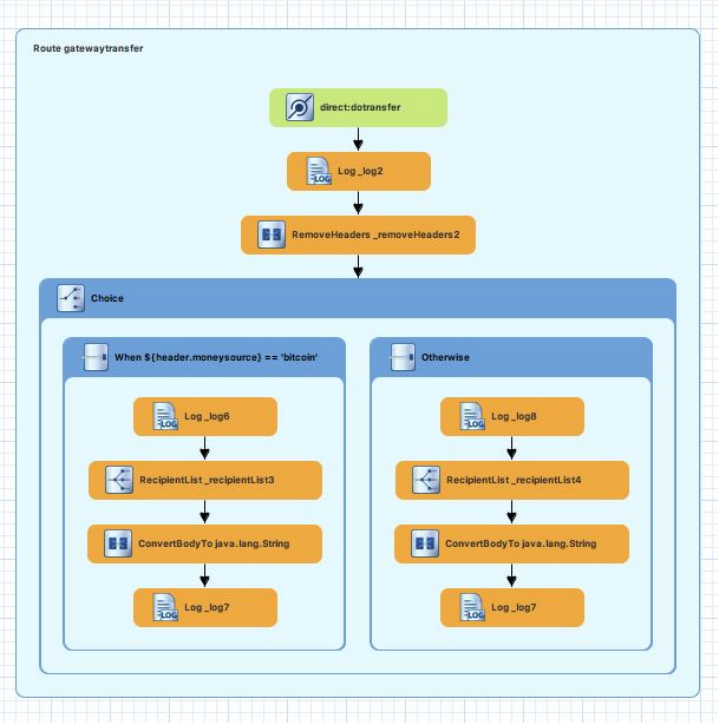
MANAGED CLOUD

Deployed and managed by Red Hat



RED HAT® OPENSHIFT

PATTERN-BASED FOR BUILDING AND COMPOSING MICROSERVICES

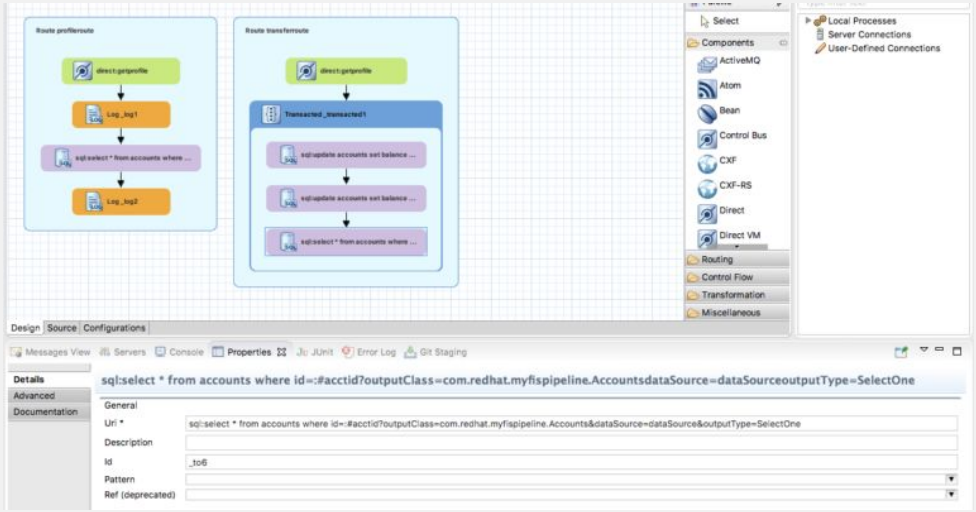


- The new FIS 2.0 comes with Apache Camel 2.18, with 150+ built-in components and data transformation, it fits perfectly with the microservice principle of building *smart* endpoints. Developers can simply configure connectors to various systems and services.

- Enterprise Integration Patterns – are a new, best practice in the concept of agile integration; developers can compose microservices with ease (*Simple pipeline*), and simply reuse the pattern without reinventing each time.

DEVELOPER EXPERIENCE WITH FUSE TOOLING

- From getting started, to real world production deployments, Fuse provides a comprehensive set of tools to help developers through the complete application life cycle.
- Developers can choose between traditional Java programing styles or leverage drag and drop features from tooling.
- Debugging and unit testing can also be done in the IDE with testing suite libraries.
- Maven is included for dependency and build management.
- For getting started, Fuse also provide a set of quick-start examples that simplify the learning curve, but are also great for experienced developers wanting to rapidly prototype new projects.



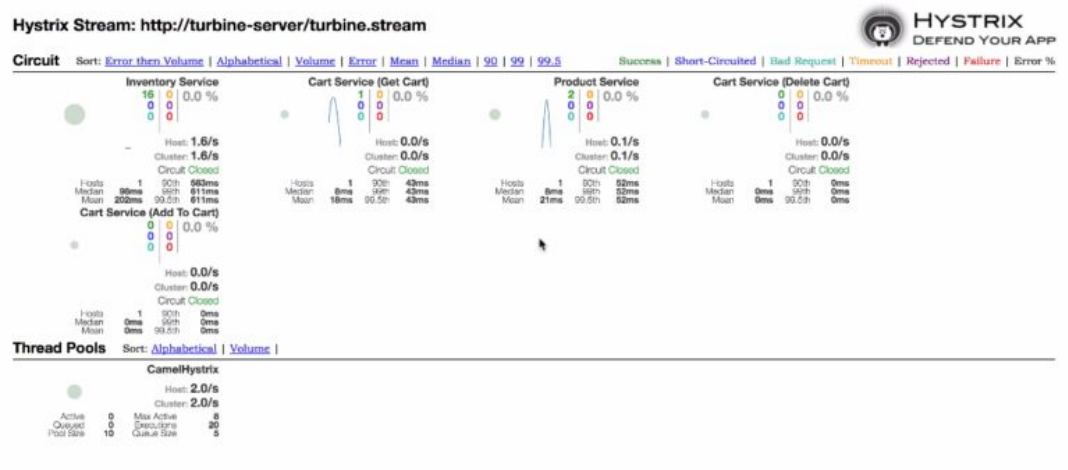
CONTAINERIZED APPLICATIONS

- FIS 2.0 offers a repeatable and declarative environment, allowing developers to quickly package an integration application into a container, simplifying the use of the same image in development, QA, and production environments.
- Fuse has pre-defined a base image for the docker-like container, allowing developers to use it as a base for application logic after which it will generate images using the tooling provided.
- Governance teams may also layer on top to provide the firm's unique environment integrations and configurations

SUPPORT SPRINGBOOT AND KARAF RUNTIMES

- FIS now officially supports Spring Boot, a widely adopted env for microservices.
- Spring Boot's "autowire" capability, and ability to create lightweight stand-alone applications has made it a natural fit as a microservice runtime.
- Karaf as an OSGi runtime is also supported for existing Fuse developers.

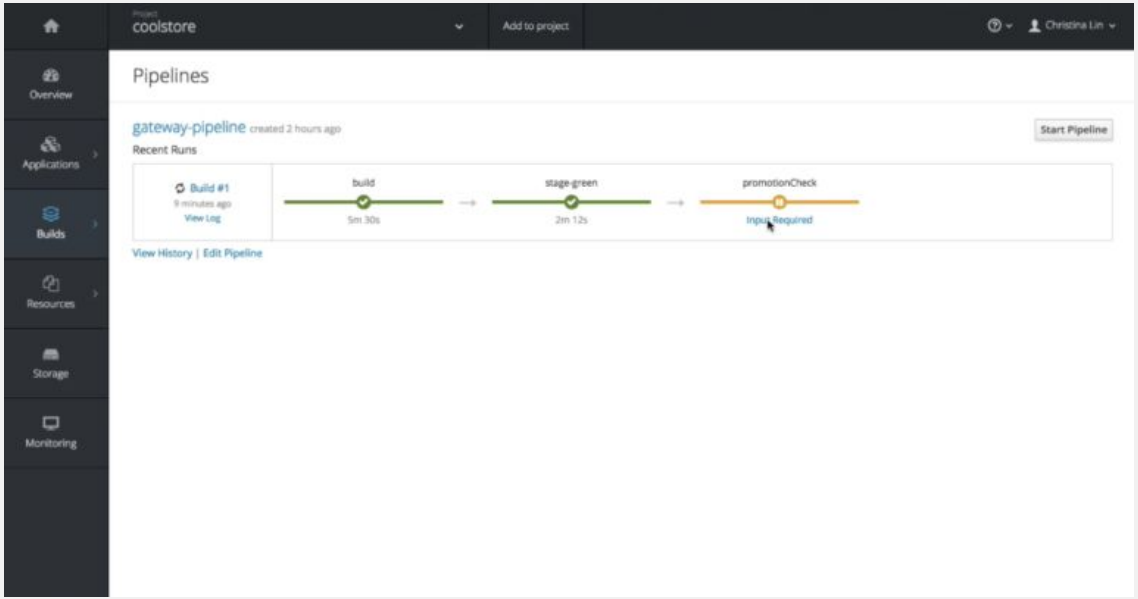
API SUPPORT AND SERVICE RESILIENCY



- With REST DSL, developers can now define a REST endpoint within minutes and
- automatically export the API documentation (Swagger).
- When connecting APIs, it is important to make sure to maintain service resiliency.

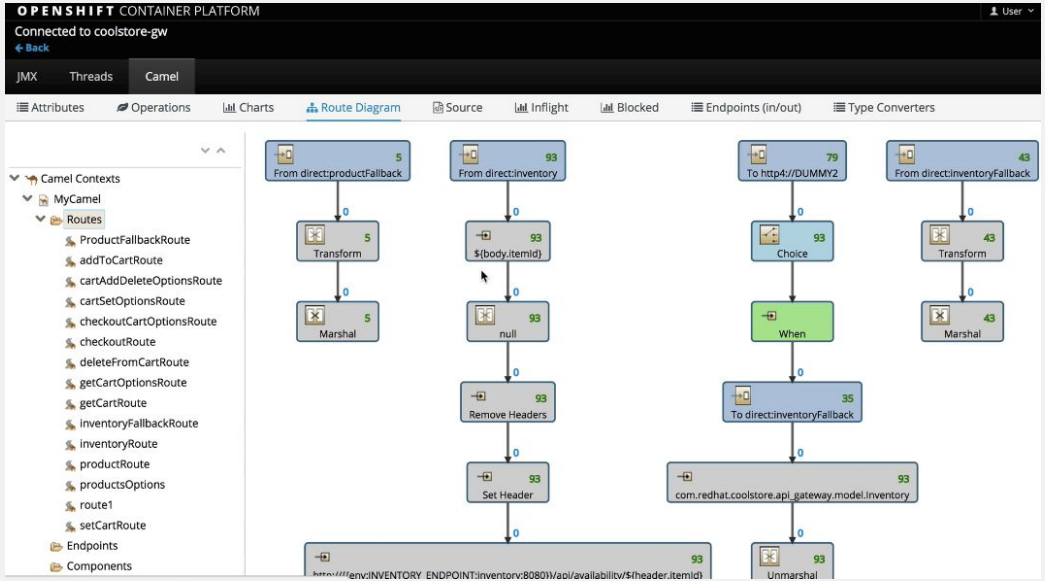
- Fuse Integration Service adopts Kubernetes as it's orchestration layer for containers, which will detect any failure of the service and recover by spinning up another running instance.
- By supporting Hystrix in Camel, Fuse makes sure the the failure is isolated without affecting other instances.

COMPLETE CI/CD CYCLE



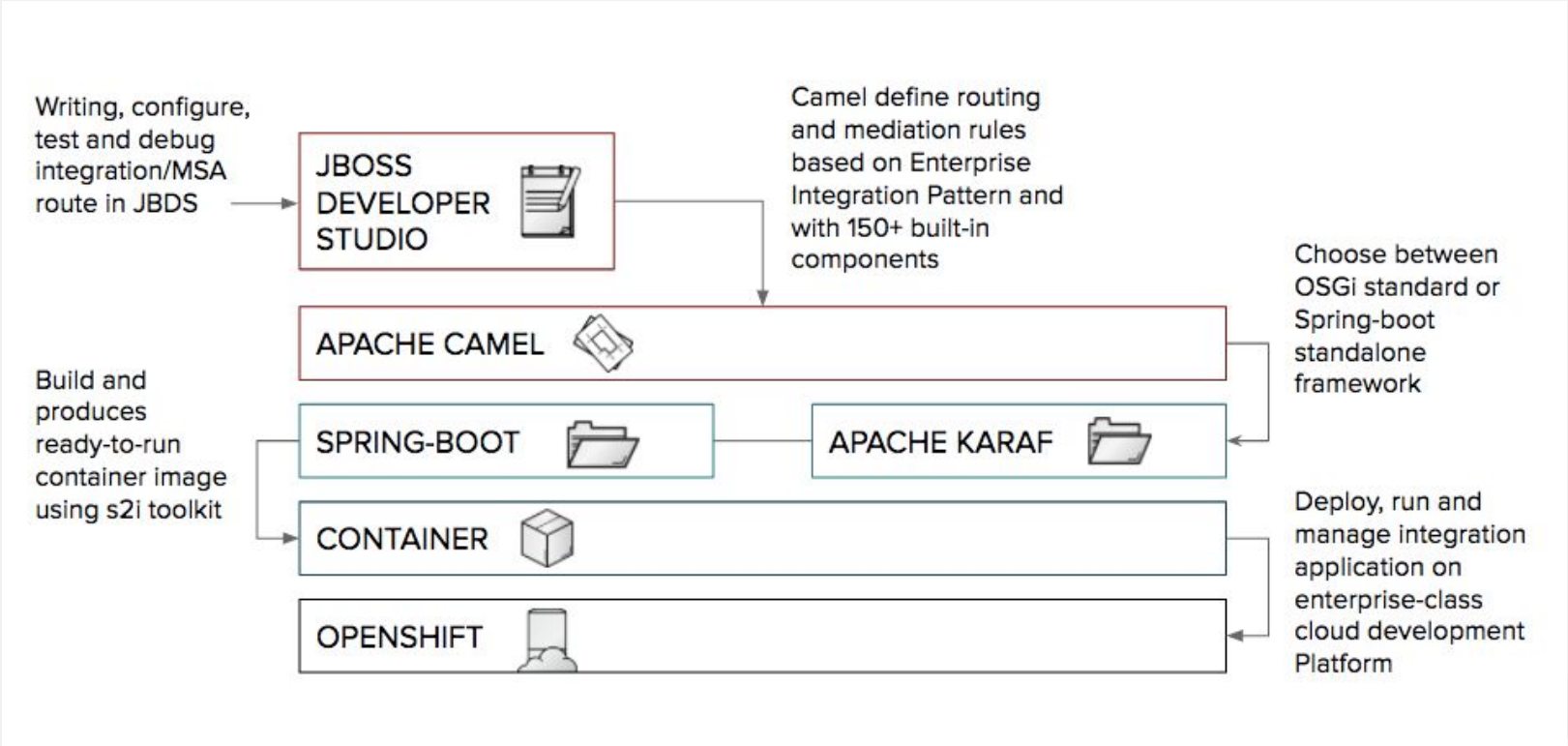
- FIS 2.0 provides a great user experience for continuous integration with features in the IDE, Maven, source control with git and other SCM applications, and the source-to-image plugin helps developers build images either to test locally or for deploying into an actual cloud platform.
- The out-of-the-box pipeline support helps developers create a complete cycle for continuous delivery.

CONTAINER ORCHESTRATION



- As the number of microservices grow, developers needed a way to manage and orchestrate all containers and services.
- Kubernetes in FIS 2.0 will help to automatically discover services, load balance incoming requests, handle clustering and dynamically configure services when they come alive.
- Operations can scale up and down services on-demand and manage resource as needed.

CONTAINER ORCHESTRATION



Red Hat JBoss Fuse

A lightweight integration platform

JBoss Fuse

- Patterns based integrations
- Broad connectivity -> 200+ connectors included
- Real-time notification (includes JBoss A-MQ)
- API foundation – create and connect API's
- Lightweight, cloud-ready

JBoss Fuse Integration Services

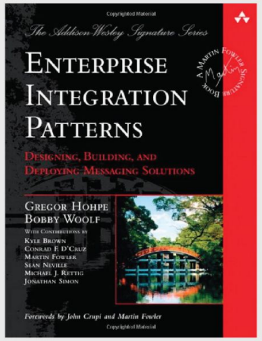
- Containerized integrations
- Rapid, adaptive & continuous integrations
- Patterns-based, microservices style integrations
- Path towards hybrid integrations

Connect apps, data, APIs, devices and IoT

Enterprise Integration Patterns

Apache Camel IS the de-facto integration standard

- Commonly used integration patterns to design, develop, accelerate and simplify integration solution development
- Camel-—-—most popular EIP implementation
- De-facto standard to build flexible and lightweight integrations
- Large, vibrant, growing ecosystem



redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos

Red Hat Integration Portfolio

INTEGRATION AND API PLATFORM

MODULAR

Supports multiple use cases, adaptable architecture

LIGHTWEIGHT

Flexible footprint, use only what is required

CONNECTIVITY

Connect traditional information sources, SaaS/PaaS, mobile, IoT

CLOUD ENABLED

Based on foundation of agility, flexibility, deployment choice

Red Hat Integration Portfolio

Connect APIs, applications and data across your entire enterprise

API Development

RED HAT® JBOSS® A-MQ — Lightweight multi-protocol messaging platform

RED HAT® JBOSS® FUSE — Lightweight integration platform (Includes A-MQ)

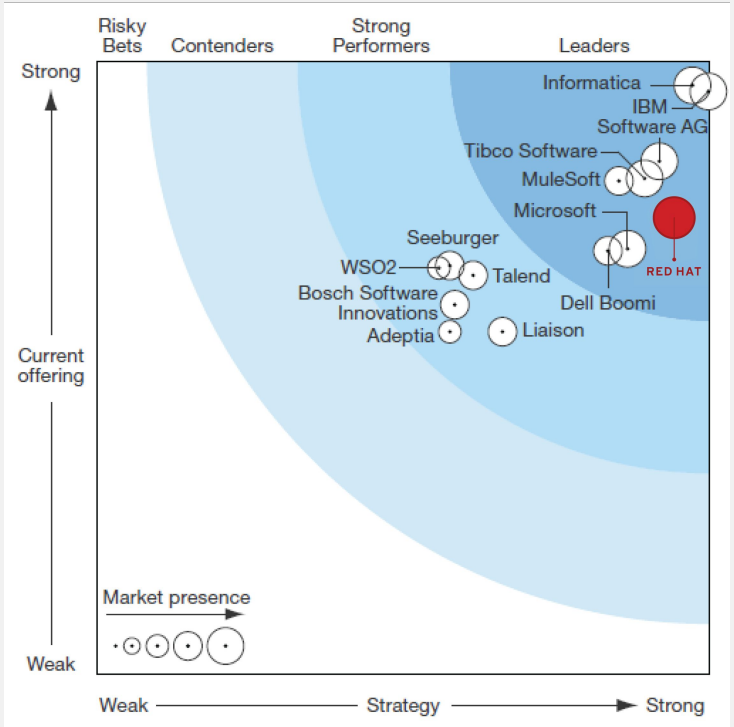
RED HAT® JBOSS® DATA VIRTUALIZATION — Lightweight data integration platform

API Management

3scale by Red Hat® — Full lifecycle API Management

FORRESTER INTEGRATION WAVE

Red Hat a LEADER in the Forrester Hybrid2 Integration Wave



Source: Forrester Hybrid2 Integration Wave

A PLATFORM THAT GROWS WITH YOUR BUSINESS