Red Hat’s Integration Roadmap

Jack Britton
Product Manager - Messaging

Keith Babo
Product Manager - Integration
THE WORLD IS CHANGING

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>90%</td>
<td>Big data: of all data created in last 2 years</td>
</tr>
<tr>
<td>80%</td>
<td>Mobile: of internet users use smartphones</td>
</tr>
<tr>
<td>81%</td>
<td>Social: of customers rely on social sites for purchasing advice</td>
</tr>
<tr>
<td>62%</td>
<td>Cloud: of total workload is running in the cloud</td>
</tr>
<tr>
<td>50 Billion</td>
<td>IoT: devices connected to internet by 2020</td>
</tr>
</tbody>
</table>
INTEGRATION CONTINUES TO EVOLVE

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Shared Mainframe Assets</strong></td>
<td>ISAM VSAM IDMS</td>
<td><strong>Point to point</strong></td>
<td>Proprietary TCP Sockets, FTP</td>
<td><strong>Hub and Spoke</strong></td>
<td>Integration Brokers Messaging, CORBA</td>
</tr>
<tr>
<td><strong>Orchestration</strong></td>
<td>SOA</td>
<td><strong>Web Services</strong></td>
<td>XML</td>
<td><strong>Open Source</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Service Mediation</strong></td>
<td>REST</td>
<td>JSON</td>
<td><strong>APIs</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Problems</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data Consistency</strong></td>
<td></td>
<td>Data Consistency Consolidated Reporting</td>
<td></td>
<td>All of those, <strong>plus</strong> Multi-step Processes Partner Managed Processes Composite Applications Mobile Applications</td>
<td>All of those, <strong>plus</strong> Migration to the Cloud APIs as a Business AI “Integration”</td>
</tr>
<tr>
<td><strong>Consolidated Reporting</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Electronic Ordering</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Case Management</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Straight Through Processing</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

All of those, **plus** Multi-step Processes Partner Managed Processes Composite Applications Mobile Applications
Enterprise IT is undergoing fundamental change. To remain competitive, businesses need an integration platform capable of supporting current and next generation architectures.
INTEGRATION IS EVERYWHERE

Integration is becoming pervasive within the enterprise with new actors involved (including non-technical)

Integration now means not just “internal” connections but also “externally” (with customers and partners)

Integration is now central to business initiatives. Connecting it to revenue related projects
THREE PILLARS OF AGILE INTEGRATION

Key foundational capabilities needed by today’s enterprises

- **Distributed Integration**
  - Lightweight, Pattern-Based, Event-Oriented, Connectivity
  - Flexibility

- **Containers**
  - Container-based scaling, encapsulation and systems management
  - Scalability

- **APIs**
  - Well-defined, reusable, managed endpoints
  - Re-usability
TECHNOLOGY OF THE THREE PILLARS

**Distributed integration**
Lightweight, Pattern-Based, Event-Oriented, Connectivity

**Containers**
Container-based scaling, encapsulation and systems management

**APIs**
Well-defined, reusable, managed endpoints

RED HAT® JBOSS® FUSE
RED HAT® JBOSS® A-MQ

RED HAT® OPENSHIFT
3scale by RED HAT®
MESSAGING ROADMAP
Agenda

- AMQ Overview
  - Broker consolidation
  - Protocols and clients
  - AMQ 7 Broker features
  - AMQ Interconnect features
- Messaging as a Service introduction
- Roadmap timeline
What is AMQ 7?

- A suite of technologies to support a full range of common messaging patterns
  - Store and forward style queuing
  - Publish and subscribe
  - Direct, anycast, multicast, and request reply
  - Wide-area messaging networks
  - Elastic-scale cloud messaging: messaging as a service

- Open Standards support: AMQP 1.0 and MQTT

- Polyglot: Java/JMS, C++, .NET, Python, JavaScript (incl. Node.js)
Red Hat JBoss AMQ 7

Flexible, standards-based messaging for the enterprise, cloud and Internet of Things

**Broker**
- New broker core with modern async architecture
- Improved performance and scalability

**Interconnect**
- Messaging Router
- High-performance direct messaging
- Distributed messaging backbone

**Clients**
- New set of AMQP 1.0 clients
- Backward compatibility with A-MQ 6.x and HornetQ

Standard protocols
Common tooling
AMQ 7 Broker
Broker Consolidation
Protocols and clients

AMQP Clients
JMS, C++, Python, JavaScript/Node.js, .NET, ...

MQTT Clients
Community

STOMP Clients
Community

AMQP 6.x Clients
JMS, C++ (CMS), .NET (NMS)

HornetQ / Artemis JMS Client

AMQP
MQTT
STOMP
OpenWire
Artemis “Core”

A-MQ 7 Broker
## AMQ 7 Broker vs. EAP Broker

<table>
<thead>
<tr>
<th>Feature</th>
<th>A-MQ 7</th>
<th>EAP 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>High performance broker based on ActiveMQ Artemis</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>JMS 2.0 client library using “Core” protocol</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AMQP 1.0 protocol support</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>MQTT protocol support</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>OpenWire protocol support</td>
<td>✔</td>
<td></td>
</tr>
</tbody>
</table>
AMQ 7.0 Broker

New Features

- Faster throughput (completely non-blocking)
- JMS 2.0 support
- Shared-nothing HA
- Certified in AWS, GPE, and Azure (post GA)

Backward Compatibility

- Backward compatible with A-MQ 6.3
  - Openwire and AMQP
- Backward Compatible with HornetQ
  - Hornet Core clients
- Backward Compatible with MRG 3.2
  - AMQP 1.0 clients
AMQ 7 Interconnect
What is AMQ Interconnect?

- A message router for the AMQP 1.0 protocol
- Separates messaging routing from message persistence
- Written in C and embedded python
- Built on the proton-C “engine” API
- Asynch architecture
- Routing protocol similar to OSPF (link-state routing)
AMQ Interconnect

- Brokers are moved to the edge of the message bus (Interconnect layer)
- Endpoints make network connections to the Interconnect layer
At the AMQP level, endpoints may exchange messages with brokers and/or directly with other endpoints.
AMQ Interconnect - Wide Area Message Bus

Sea

Bos

Ord

Dfw

Atl

Lax
AMQ Interconnect - Wide Area Message Bus

Public Cloud

Private Network

Private Network
Messaging as a Service - Coming Soon
Messaging as a Service

AMQ 7 built for OpenShift

● Enables applications deployed to an openshift cluster to include a scalable messaging component within them
● Expose service deployed on an openshift cluster to external applications
● Provide a service for communication between separate applications/projects on an openshift cluster
● Provide message fabric between OpenShift clusters on-prem and in the cloud
2017

2ND QTR 2017
- A-MQ 7.0 Beta
  - Release 2
  - Public Beta

3RD QTR 2017
- A-MQ 7.0 GA
  - Broker
  - Replicated HA
  - Backward Compatibility
  - Interconnect Router
  - AMQP Clients

4TH QTR 2017
- MaaS Alpha
INTEGRATION ROADMAP
WHAT HAVE WE BEEN UP TO?

Fuse 6.3
FIS 2.0
Fuse 7.0
FUSE 6.3
FUSE 6.3 RELEASE GOALS

REFRESH
- Camel 2.17
- 23 new Camel components
- CXF 3.1
- Jetty 9

USABILITY
- Revamped EIP Editor
- New Getting Started Experience
- Maven Repository Unification

STABILITY
- 2500 issues fixed across all projects
- 1191 issues fixed in product stream
- 204 issues fixed in Fabric v1
REVAMPED EIP EDITOR
Fuse 6.3 will be the long-term support stream for 6.x

<table>
<thead>
<tr>
<th>Family</th>
<th>GA</th>
<th>End of Full Support</th>
<th>End of Maintenance Support</th>
<th>End of ELS-1</th>
<th>End of ELS-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.x</td>
<td>Apr 2013</td>
<td>Jul 2018</td>
<td>Jan 2022</td>
<td>Jan 2024</td>
<td>Jan 2025</td>
</tr>
</tbody>
</table>
Fuse Integration Services
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
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
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.
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
Fuse 7.0
FUSE 7.x
Three Fuse Product Initiatives

- Fuse Standalone
- Fuse Integration Services
- Fuse iPaaS
FUSE 7.x
One Comprehensive Product for Agile Integration
FUSE STANDALONE

- CORE FUSE
  - Camel
  - Narayana
  - Undertow
  - A-MQ
  - CXF
- EAP
- Spring Boot
- Karaf

Hawtio
DevStudio

JVM
OS
FUSE STANDALONE

- Camel 2.20
- Karaf 4
- Add Spring Boot
- EAP 7.x Support
- Narayana transaction manager
  - Geronimo removed
- Undertow web container
  - Jetty deprecated
- AMQ 6/7 certification
- Further component modularization for EAP 7
- SwitchYard removed
- API-focused integration
  - REST DSL editor
  - SOAP <-> REST
  - 3scale integration
FUSE INTEGRATION SERVICES

Provisioning
Clustering
Centralized Admin

CORE FUSE
Camel

Narayana
Undertow
A-MQ
CXF

EAP
Spring Boot
Karaf

Containers
OpenShift

Hawtio
DevStudio

#redhat #rhusummit
Fuse 6.x → Fuse 7.0

- Fuse 6.x
  - Fabric v1

- Fuse 6.3
  - Fabric v1

- Fuse 6.3
  - FIS

- Fuse 7.0
  - FIS

Stay where you are
Adopt FIS 2.0 when ready
Wait for Fuse 7.0
FABRIC v1 PROFILES

Apache ZooKeeper

Fabric8 V1 Configuration

Karaf git

Karaf git

Karaf git
FUSE 7 PROFILES

OpenShift CI/CD Pipeline

- Maven & git
  - Fabric8 Profiles Configuration
  - FIS PContainer
  - Admin Console Tooling
  - OpenShift
FUSE iPaaS

- No local install footprint
- 100% web-based interface
- Designed for everyday users
- Cloud, SaaS, and API connectivity
- Comprehensive coverage of integration lifecycle
- Build on a truly hybrid integration platform
Integration Made Easy

S101856 - Red Hat iPaaS—integration made easy

Keith Babo - Product Manager, Red Hat
Hiram Chirino - FL, Red Hat

Session schedule
Tuesday, May 2, 4:30 PM - 5:15 PM
– Room 151B
Fuse iPaaS Architecture

- iPaaS UI
- Fuse Integration Services
  - Apache Camel
  - Spring Boot
- iPaaS Infrastructure
- Containers
- OpenShift Online
Release Timeline

All roadmap dates are projections and subject to change.
Interested in this topic? Head over to the **User Experience Design** booth to learn more.

Partner Pavilion
Exhibit Hall A

Red Hat iPaaS Feedback
Dongni Wang
Questions?
THANK YOU

plus.google.com/+RedHat
linkedin.com/company/red-hat
youtube.com/user/RedHatVideos
facebook.com/redhatinc
twitter.com/RedHatNews
LEARN. NETWORK. EXPERIENCE OPEN SOURCE.