



Red Hat Integration 2020-Q3

Release Notes for Red Hat Integration 2020-Q3

What's new in Red Hat Integration

Red Hat Integration 2020-Q3 Release Notes for Red Hat Integration 2020-Q3

What's new in Red Hat Integration

Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

Describes the Red Hat Integration platform and provides the latest details on what's new in this release.

Table of Contents

CHAPTER 1. RED HAT INTEGRATION	3
CHAPTER 2. NEW FEATURES IN THIS RELEASE	4
2.1. KEY FEATURES	4
2.2. NEW COMPONENT FEATURES	4
CHAPTER 3. REMOVED FEATURES IN THIS RELEASE	5
3.1. DATA VIRTUALIZATION	5
CHAPTER 4. DEBEZIUM RELEASE NOTES	6
4.1. DEBEZIUM DATABASE CONNECTORS	6
4.2. SUPPORTED DATABASE VERSIONS FOR DEBEZIUM	6
4.3. DEBEZIUM INSTALLATION OPTIONS	7
4.4. NEW DEBEZIUM FEATURES	7
CHAPTER 5. SERVICE REGISTRY RELEASE NOTES	9
5.1. SERVICE REGISTRY INSTALLATION OPTIONS	9
5.2. SERVICE REGISTRY NEW FEATURES	9
Updated features in Service Registry 1.0.1	9
New features in Service Registry 1.0	10
5.3. SERVICE REGISTRY KNOWN ISSUES	11
CHAPTER 6. RED HAT INTEGRATION OPERATORS	12
6.1. 3SCALE OPERATOR	12
6.2. AMQ OPERATORS	12
6.3. FUSE OPERATORS	12
6.4. SERVICE REGISTRY OPERATOR	12
Additional resources	12

CHAPTER 1. RED HAT INTEGRATION

Red Hat Integration is a comprehensive set of integration and event processing technologies for creating, extending, and deploying container-based integration services across hybrid and multicloud environments. Red Hat Integration provides an agile, distributed, and API-centric solution that organizations can use to connect and share data between applications and systems required in a digital world.

Red Hat Integration includes the following capabilities:

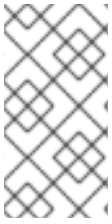
- API connectivity
- Data transformation
- Service composition and orchestration
- Real-time messaging
- Cross-datacenter message streaming
- API management

Additional resources

- [Understanding enterprise integration](#)

CHAPTER 2. NEW FEATURES IN THIS RELEASE

This section provides a summary of the key new features in Red Hat Integration 2020-Q3 and provides links to more details on new features available in different components.



NOTE

These release notes include details on components updated in Red Hat Integration 2020-Q3 only. For details on the latest versions of other components, such as Camel K, Camel Kafka Connector, and Data Virtualization, see [Red Hat Integration Release Notes for 2020-Q2](#).

2.1. KEY FEATURES

Data integration

- Change data capture and real-time events with [Debezium 1.2](#)

Disconnected installation

- Installation in restricted networks with no Internet access with:
 - [Red Hat Fuse on OpenShift 7.7](#)
 - [Red Hat 3scale API Management 2.9 On-Premises](#)

2.2. NEW COMPONENT FEATURES

For more details on what's new in Red Hat Integration 2020-Q3 components:

- [Red Hat 3scale API Management 2.9 On-Premises Release Notes](#)
- [Red Hat 3scale API Management SaaS Release Notes](#)
- [Red Hat AMQ 7.7 Product Documentation](#)
- [Red Hat Fuse 7.7 Release Notes](#)

CHAPTER 3. REMOVED FEATURES IN THIS RELEASE

This section describes the features that are removed in Red Hat Integration 2020-Q3 and provides links to more detailed information.

3.1. DATA VIRTUALIZATION

The Data Virtualization Technology Preview component of Red Hat Integration based on the Teiid project is removed in this release. The following capabilities are removed:

- Data Virtualization Operator released in Red Hat Integration 2020-Q2
- Data Virtualization user interface released in Fuse Online 7.7

Additional resources

- [Data Virtualization Technology Preview in Release Notes for Red Hat Integration 2020-Q2](#)
- [Connecting to virtual databases in Fuse Online](#)
- <http://teiid.io/>

CHAPTER 4. DEBEZIUM RELEASE NOTES

Red Hat Integration 2020-Q3 includes a General Availability release of Debezium on OpenShift based on the [Debezium](#) open source project. Debezium is a distributed change data capture platform that tracks database operations and streams data change events. Debezium is built on Apache Kafka and is deployed and integrated with AMQ Streams.

Debezium captures row-level changes to database tables and passes corresponding change event records to AMQ Streams. Applications can read these *change event streams* and access the change events in the order in which they occurred.

The following topics provide release details:

- [Section 4.1, "Debezium database connectors"](#)
- [Section 4.2, "Supported database versions for Debezium"](#)
- [Section 4.3, "Debezium installation options"](#)
- [Section 4.4, "New Debezium features"](#)

4.1. DEBEZIUM DATABASE CONNECTORS

Debezium provides connectors based on Kafka Connect for the following common databases:

- MySQL
- PostgreSQL
- MongoDB
- SQL Server
- Db2 - this is a Technology Preview release.

4.2. SUPPORTED DATABASE VERSIONS FOR DEBEZIUM

When trying out the database connectors, the following database versions are supported for this release:

Database	Versions
MySQL	5.7, 8.0
PostgreSQL *	10, 11, 12
MongoDB	3.6, 4.0, 4.2 **
SQL Server	2017, 2019
Db2	11.5.0.0

*For PostgreSQL deployments, you use the **pgoutput** logical decoding output plug-in, which is the default for PostgreSQL versions 10 and later.

**When using the Debezium connector for MongoDB, there is a limitation if you are using MongoDB 4.2. The limitation is that you cannot use the connector's transaction metadata feature. This limitation is expected to be removed in a future release.

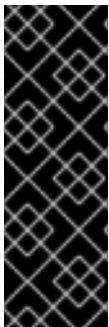
Additional resources

- [Getting Started with Debezium](#)
- [Debezium User Guide](#)

4.3. DEBEZIUM INSTALLATION OPTIONS

You can install Debezium with AMQ Streams on OpenShift or RHEL:

- [Installing Debezium on OpenShift - General Availability](#)
- [Installing Debezium on RHEL - Technology Preview](#)



IMPORTANT

Technology Preview features are not supported with Red Hat production service-level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend implementing any Technology Preview features in production environments. This Technology Preview feature provides early access to upcoming product innovations, enabling you to test functionality and provide feedback during the development process. For more information about support scope, see [Technology Preview Features Support Scope](#).

4.4. NEW DEBEZIUM FEATURES

This release provides the following new Debezium features:

- [Schema change topic](#) - The SQL Server connector now emits events that capture schema updates to a schema change topic.

This release provides the following Technology Preview features:

- [Debezium Db2 connector](#).
- [Content-based router](#) single message transformation (SMT) for re-routing data change event records to topics based on event content.
- [Filter](#) SMT that evaluates an expression for each change event and drops or emits the event based on the evaluation result.
- [CloudEvents converter](#) for emitting change event records that conform to the CloudEvents specification. Avro encoding type is now supported for the CloudEvents envelope structure.
- [Avro serialization](#) - You can configure Debezium connectors to use Avro to serialize message keys and values.

- **Outbox event router** SMT that supports the outbox pattern for safely and reliably exchanging data between multiple (micro) services.

CHAPTER 5. SERVICE REGISTRY RELEASE NOTES

Red Hat Integration - Service Registry 1.0.1 is provided as a micro release in Red Hat Integration 2020-Q3. Service Registry is a datastore for standard event schemas and API designs that is based on the [Apicurio Registry](#) open source community project.

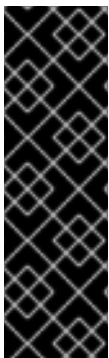
You can use Service Registry to manage and share the structure of your data using a REST interface. For example, client applications can dynamically push or pull the latest updates to or from Service Registry without needing to redeploy. You can also use Service Registry to create optional rules to govern how registry content evolves over time. For example, this includes rules for content type validation or backwards and forwards compatibility of schema or API versions.

5.1. SERVICE REGISTRY INSTALLATION OPTIONS

You can install Service Registry with the following storage options:

Table 5.1. Service Registry storage options

Storage option	Release
Kafka Streams-based storage in AMQ Streams 1.5	General Availability
Cache-based storage in embedded Infinispan 10	Technology Preview only
Java Persistence API-based storage in PostgreSQL 12 database	Technology Preview only



IMPORTANT

Technology Preview features are not supported with Red Hat production service-level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend implementing any Technology Preview features in production environments.

This Technology Preview feature provides early access to upcoming product innovations, enabling you to test functionality and provide feedback during the development process. For more information about support scope, see [Technology Preview Features Support Scope](#).

5.2. SERVICE REGISTRY NEW FEATURES

Updated features in Service Registry 1.0.1

The Service Registry 1.0.1 micro release updates the Operator metadata that is used to install Service Registry 1.0 from the OpenShift OperatorHub user interface. The Service Registry product images have not changed and remain at version 1.0.

The new Operator metadata includes a new Operator bundle format for release packaging, which is designed for use with OpenShift Container Platform 4.5 or 4.6. For more details, see the [Bundle Format in the OpenShift documentation](#).

**NOTE**

You can upgrade to Service Registry 1.0.1 if you are using OpenShift 4.5. However, you must upgrade to Service Registry 1.0.1 before upgrading to OpenShift 4.6. The new Operator bundle format is required for use with OpenShift 4.6.

New features in Service Registry 1.0

Service Registry provides the following new features in the 1.0 General Availability release:

Service Registry OpenShift Operator

The Red Hat Integration - Service Registry Operator is available from the OpenShift OperatorHub for installation and upgrade. This replaces the OpenShift template previously available in Technology Preview. Example custom resource definitions are also provided for storage in AMQ Streams, embedded Infinispan, or PostgreSQL database.

Service Registry web console

- Search and browse for artifacts and versions (based on new Search REST API)
- View details about an artifact, and view generated documentation (OpenAPI only)
- Perform artifact actions:
 - Upload to registry
 - Edit metadata
 - Configure rules
 - Download locally
- Configure global rules

Additional storage options (Technology Preview only)

- Java Persistence API (JPA) - PostgreSQL 12.x database
- Embedded Infinispan cache - Infinispan 10.0.1.Final (Community build)

Additional artifact types

- WSDL - Web Services Description Language
- XSD - XML Schema Definition
- XML - Extensible Markup Language

Improved rules for registry content

- Improved compatibility rule for JSON Schema artifacts

Improved artifact metadata

- Automatic extraction of metadata defined in artifact content, for example, artifact name and description
- Labels for searching and browsing

Quarkus version

- Red Hat build of Quarkus 1.3.2.Final

User documentation

- [Getting Started with Service Registry](#)
- [Registry REST API documentation](#)
- [Creating Service Registry client applications and serializers/deserializers with AMQ Streams](#)
- [Deploying a Debezium connector and Apache Avro serialization with Service Registry](#)

5.3. SERVICE REGISTRY KNOWN ISSUES

The following known issues apply to Service Registry 1.0:

Operator-32 - Operator should support SCRAM authorization without TLS, not only SCRAM+TLS

The Service Registry Operator should support Salted Challenge Response Authentication Mechanism (SCRAM) authorization without Transport Layer Security (TLS), not only SCRAM+TLS.

Operator-41 - Example CRD should not be empty::

Example **ApicurioRegistry** custom resource definition should not be empty.

Operator-42 - Auto-generation of OpenShift route may use wrong base host value

The auto-generation of the Service Registry OpenShift route may use a wrong base host value if there are multiple **routerCanonicalHostname** values.

Operator-45 - Operator may not delete all resources

The Service Registry Operator may not delete all resources when deleting **ApicurioRegistry** custom resource definition.

CHAPTER 6. RED HAT INTEGRATION OPERATORS

Red Hat Integration provides Operators to enable you to automate the deployment of Red Hat Integration components on OpenShift. This section provides links to detailed information on how to use Operators for components in this release.

6.1. 3SCALE OPERATOR

- [3scale Operator](#)

6.2. AMQ OPERATORS

- [AMQ Broker Operator](#)
- [AMQ Interconnect Operator](#)
- [AMQ Streams Cluster Operator](#)
- [AMQ Online Operator](#)

6.3. FUSE OPERATORS

- [Fuse on OpenShift - Samples Operator](#)
- [Fuse on OpenShift - Fuse Console Operator](#)
- [Fuse on OpenShift - API Designer Operator](#)
- [Fuse Online Operator](#)

6.4. SERVICE REGISTRY OPERATOR

- [Service Registry Operator](#)

Additional resources

- [Understanding Operators in the OpenShift documentation](#)
- [OpenShift tech topic on Operators](#)