



Red Hat build of Apache Camel 4.4

Release Notes for Red Hat build of Apache Camel for Spring Boot

What's new in Red Hat build of Apache Camel

Red Hat build of Apache Camel 4.4 Release Notes for Red Hat build of Apache Camel for Spring Boot

What's new in Red Hat build of Apache Camel

Legal Notice

Copyright © 2024 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 build of Apache Camel product and provides the latest details on what's new in this release.

Table of Contents

CHAPTER 1. RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 4.4 RELEASE NOTES	3
1.1. FEATURES IN RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT	3
1.2. SUPPORTED PLATFORMS, CONFIGURATIONS, DATABASES, AND EXTENSIONS FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT	3
1.3. THE JAVAX TO JAKARTA PACKAGE NAMESPACE CHANGE	3
1.3.1. Migration tools	3
1.4. IMPORTANT NOTES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT	4
1.4.1. Support for IBM Power and IBM Z	4
1.4.2. Support for EIP circuit breaker	4
1.4.3. Support for Stateful transactions	4
1.5. FIXED ISSUES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT	4
1.5.1. Red Hat build of Apache Camel for Spring Boot version 4.4 Enhancements	4
1.5.2. Red Hat build of Apache Camel for Spring Boot version 4.4 fixed issues	5
1.6. KNOWN ISSUES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT	7
1.6.1. Red Hat build of Apache Camel for Spring Boot version 4.4 known issues	7
CHAPTER 2. ADDITIONAL RESOURCES	9

CHAPTER 1. RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 4.4 RELEASE NOTES

1.1. FEATURES IN RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT

Red Hat build of Apache Camel for Spring Boot introduces Camel support for Spring Boot which provides auto-configuration of Camel, and starters for many Camel components. The opinionated auto-configuration of the Camel context auto-detects Camel routes available in the Spring context and registers key Camel utilities (like producer template, consumer template and the type converter) as beans.

1.2. SUPPORTED PLATFORMS, CONFIGURATIONS, DATABASES, AND EXTENSIONS FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT

- For information about supported platforms, configurations, and databases in Red Hat build of Apache Camel for Spring Boot, see the [Supported Configuration](#) page on the Customer Portal (login required).
- For a list of Red Hat Red Hat build of Apache Camel for Spring Boot extensions, see the [Red Hat build of Apache Camel for Spring Boot Reference](#) (login required).

1.3. THE JAVAX TO JAKARTA PACKAGE NAMESPACE CHANGE

The Java EE move to the Eclipse Foundation and the establishment of Jakarta EE, since Jakarta EE 9, packages used for all EE APIs have changed to **jakarta.***

Code snippets in documentation have been updated to use the **jakarta.*** namespace, but you of course need to take care and review your own applications.



NOTE

This change does not affect javax packages that are part of Java SE.

When migrating applications to EE 10, you need to:

- Update any import statements or other source code uses of EE API classes from the **javax** package to **jakarta**.
- Change any EE-specified system properties or other configuration properties whose names begin with **javax.** to begin with **jakarta..**
- Use the **META-INF/services/jakarta.[rest_of_name]** name format to identify implementation classes in your applications that use the implement EE interfaces or abstract classes bootstrapped with the **java.util.ServiceLoader** mechanism.

1.3.1. Migration tools

- Source code migration: [How to use Red Hat Migration Toolkit for Auto-Migration of an Application to the Jakarta EE 10 Namespace](#)

- Bytecode transforms: For cases where source code migration is not an option, the open source [Eclipse Transformer](#)

Additional resources

- Background: [Update on Jakarta EE Rights to Java Trademarks](#)
- Red Hat Customer Portal: [Red Hat JBoss EAP Application Migration from Jakarta EE 8 to EE 10](#)
- Jakarta EE: [Javax to Jakarta Namespace Ecosystem Progress](#)

1.4. IMPORTANT NOTES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT

1.4.1. Support for IBM Power and IBM Z

Red Hat build of Camel Spring Boot is now supported on IBM Power and IBM Z.

1.4.2. Support for EIP circuit breaker

The Circuit Breaker EIP for Camel Spring Boot supports Resilience4j configuration. This configuration provides integration with Resilience4j to be used as Circuit Breaker in Camel routes.

1.4.3. Support for Stateful transactions

The Red Hat build of Camel Example Spring Boot provides a [Camel Spring Boot JTA quickstart](#). This quickstart demonstrates how to run a Camel Service on Spring Boot that supports JTA transactions on two external transactional resources: a database (MySQL) and a message broker (Artemis). These external resources are provided by OpenShift which must be started before running this quickstart.

1.5. FIXED ISSUES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT

The following sections list the issues that have been fixed in Red Hat build of Apache Camel for Spring Boot.

1.5.1. Red Hat build of Apache Camel for Spring Boot version 4.4 Enhancements

Table 1.1. Red Hat build of Apache Camel for Spring Boot version 4.4 Enhancements

Issue	Description
CSB-470	Support Hawtio console for Camel for Spring Boot
CSB-1246	camel-olingo4 support
CSB-1693	Adding a Kafka Batch Consumer
CSB-2460	[RFE] Support component camel-smb

Issue	Description
CSB-2479	Enhancing XML IO DSL to support beans like in YAML DSL
CSB-2649	Camel for Spring Boot support for IBM Z/P
CSB-2841	Provide support to configure algorithm for camel-ssh component
CSB-2968	Add support for camel-flink
CSB-2973	Add Azure SAS support for azure blob storage
CSB-3025	Create and support a new Camel CICS component
CSB-3061	Support component camel-splunk
CSB-3236	Offline Maven Builder Script
CSB-3244	Support component camel-jasypt
CSB-3357	Support component camel-kudu
CSB-3331	Support cxf-integration-tracing-opentelemetry
CSB-3371	Support component camel-groovy
CSB-3462	BeanIO support
CSB-4117	camel-cics - support connectivity via channels

1.5.2. Red Hat build of Apache Camel for Spring Boot version 4.4 fixed issues

Table 1.2. Red Hat build of Apache Camel for Spring Boot version 4.4 resolved issues

Issue	Description
CSB-1913	CVE-2023-35116 jackson-databind: denial of service via cyclic dependencies
CSB-2007	CVE-2023-2976 guava: insecure temporary directory creation
CSB-2041	AWS SQS component, OCP probes cause POD error
CSB-2139	[Micrometer Observability] Unable to see trace id and span id in MDC
CSB-2644	Please provide examples that show Camel AMQP/JMS used with a connection pool

Issue	Description
CSB-2846	CVE-2023-5632 mosquito: Possible Denial of Service due to excessive CPE consumption
CSB-3042	[camel-mail] java.lang.ClassNotFoundException: org.eclipse.angus.mail.imap.IMAPStore
CSB-3294	Dependency convergence error for org.ow2.asm:asm when using CXF and JSON Path
CSB-3298	Dependency convergence error for org.bouncycastle:bcprov-jdk18on:jar:1.72
CSB-3302	Add support for findAndModify Operation
CSB-3316	CVE-2023-51074 json-path: stack-based buffer overflow in Criteria.parse method
CSB-3331	Support cxf-integration-tracing-opentelemetry
CSB-3438	CVE-2024-21733 tomcat: Leaking of unrelated request bodies in default error page
CSB-3454	camel-bean - Allow to configure bean introspection cache on component
CSB-3601	Dependency convergence errors when using cxf-rt-rs-service-description-openapi-v3:4.0.2.fuse-redhat-00046 and camel-openapi-java-starter:4.0.0.redhat-00039
CSB-3713	CVE-2023-45860 Hazelcast: Permission checking in CSV File Source connector
CSB-3716	AMQP publisher application is losing messages with local JMS transaction enabled
CSB-3722	CVE-2024-26308 commons-compress: OutOfMemoryError unpacking broken Pack200 file
CSB-3725	commons-compress: Denial of service caused by an infinite loop for a corrupted DUMP file [rhint-camel-spring-boot-4]
CSB-3731	restConfiguration section is ignored when using XML DSL IO
CSB-3765	Issue while marshalling/unmarshalling XML to JSON.
CSB-3837	CVE-2023-5685 xnio: StackOverflowException when the chain of notifier states becomes problematically big

Issue	Description
CSB-3851	onExceptionHandler does not set content in the body response when used with servlet/platform-http
CSB-3884	[Camel-sap] Unable to connect to SAP server through CSB configuration properties
CSB-3892	camel-file - Can ant filter be optimized when using min/max depth with orphan marker file check
CSB-3916	NPE occurs If user uses OpenTelemetryTracingStrategy and opentelemetry.exclude-patterns to exclude "direct"
CSB-3922	OpenTelemetryTracingStrategy separates a trace into 2 branches with opentelemetry.exclude-patterns "process*" or "bean"
CSB-3925	Request to offer connection pooling in camel-cics
CSB-4022	Put a max default configurable limit on the Jose P2C parameter & Only explicitly return the stylesheet in WadlGenerator and not other URLs
CSB-4092	Type Conversion Error from byte[] to Long in Camel 4 from Kafka Topic for JMS* headers
CSB-4095	camel-salesforce - startup error
CSB-4102	CVE-2024-22262 springframework: URL Parsing with Host Validation

1.6. KNOWN ISSUES FOR RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT

The following sections list known issues for Red Hat build of Apache Camel for Spring Boot.

1.6.1. Red Hat build of Apache Camel for Spring Boot version 4.4 known issues

[CSB-4318](#) Fail to deploy on OCP using Openshift Maven Plugin if `spring.boot.actuator.autoconfigure` is not in the dependencies

Jkube maven plugin uses the following condition to check if the application exposes health endpoint (using `SpringBootHealthCheckEnricher`). Both classes are in the classpath:

- `org.springframework.boot.actuate.health.HealthIndicator`
- `org.springframework.web.context.support.GenericWebApplicationContext`

However, the `/actuator/health` will be not exposed without the configuration of the actuator. This creates discordance between the readiness/liveness probes configured by JKube (they both uses the above endpoint) and what the application is exposing.

This misconfiguration causes a failing deployment config on OpenShift Container Platform since the generated pod will never be in Ready status since the probe's call for an endpoint is not configured. So in order to make the application work on OpenShift Container Platform, which is deployed using JKube (openshift-maven-plugin), it is necessary to have both web and actuator autoconfiguration in the dependencies.

Following example shows how to configure web and actuator autoconfiguration.

Example

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

Update the archetype as shown below. The applications built from the following archetype will be deployed correctly using JKube.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
  <exclusions>
    <exclusion>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-tomcat</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-undertow</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

This issue affects the custom applications with missing one of the above dependencies.

CHAPTER 2. ADDITIONAL RESOURCES

- [Supported Configurations](#)
- [Getting Started with Red Hat build of Apache Camel for Spring Boot](#)
- [Migrating to Red Hat build of Apache Camel for Spring Boot](#)
- [Red Hat build of Apache Camel for Spring Boot Reference](#)