

# Camel 2.17.0 Release

## Camel 2.17.0 Release



### New and Noteworthy

Welcome to the 2.17.0 release which resolved approx. 450 issues (new features, improvements and bug fixes).

- The component documentation generated from the source code has been double checked to be up to date and include all the options the endpoints supports.
- Reworked the Apache Karaf feature to not install camel-spring by default. The problem is camel-spring is using spring-dm which is only supports Spring 3.x, and therefore Spring 3.x was installed and used. All the Camel features that uses Spring JARs are now using Spring 4.x.
- [Camel commands](#) for [Spring Boot](#)
- Many improvements to Camel [Kura](#)
- Added transacted option to [SQL Component](#) when used as a consumer in a transacted route.
- Added support for UPDATE operation to [ElasticSearch](#).
- Allow to reuse existing configured Elasticsearch Client on the [ElasticSearch](#) component, instead of creating a client per endpoint.
- The `include` and `exclude` options on [File2](#) and [FTP](#) endpoints is now case in-sensitive out of the box.
- Resource based component such as [XSLT](#), [Velocity](#) etc can load the resource file from the [Registry](#) by using `ref: as` prefix.
- Upgraded camel-amqp to the latest `qpidd-jms-client` (also AMQP < 1.0 is not supported anymore).
- Many improvements to Camel [AMQP](#) component.
- The [Metrics Component](#) allows to capture [Message History](#) performance statistics with the `MetricsMessageHistoryFactory`
- Reduced the number of mbeans enlisted in the services tree, to only include mbeans that has value to be managed.
- The [Elasticsearch Component](#) now supports MultiGet operation
- The [Throttler](#) has been improved to be more performant and use a rolling window for time periods which gives a better flow.
- The `setHeader` and `setExchangeProperty` allows to use a dynamic header key using the [Simple](#) language if the name of the key is a [Simple](#) language expression.
- Add `collate` function to [Simple](#) language to make it easier to split a message body into sub lists of a specified size. The function is similar to the `collate` function from Groovy.
- The [Camel Run Maven Goal](#) is able to auto detect if its a OSGi Blueprint or CDI project so end users no longer have to explicit configure this on the plugin.
- [Camel-Elasticsearch](#) now supports Multiget, Multisearch and exists operation
- [Camel-Git](#) now supports Cherry-pick operation
- Stopping the `Main` class from JMX will now trigger shutdown of the `Main` class/JVM also, as it does when hitting `ctrl + c`.
- Added option to `skipFirst` to the [Tokenizer](#) language to make it easy to skip the very first element, when for example splitting a CSV file using the [Splitter](#) EIP.
- The [Rest DSL](#) now supports default values for query parameters
- The [Rest DSL](#) now only binds from POJO to JSON/XML if the content-type is for json or xml. This allows you to specify a custom content-type and omit the binding, such as when having binary payloads.
- Exchange and Message only output id in their `toString` method to avoid outputting any message details such as sensitive details from message bodies.
- Camel [Error Handler](#) no longer log message body/header details when logging the [Message History](#). This avoids logging any sensitive details from message bodies.
- Camel [Exception Clause](#) and [Error Handler](#) now supports using a custom `Processor` to be invoked right after an exception was thrown using the new `onExceptionOccurred` option.
- [RabbitMQ](#) consumer more resilient to auto re-connect in case of connection failures
- The [JSON](#) `camel-jackson` can be used as a type converter to convert json to/from POJO and `String/byte[]` types, if enabled.
- The [SQL Component](#) now supports loading the SQL queries from external files on the classpath or file system, where you can format the queries using newlines and indenting.
- The [Mail](#) consumer allows to use an idempotent repository which allows to cluster consuming from the same mailbox, and let the repository coordinate whether a mail message is valid for the consumer to process.
- Added support for completion interval on the [SJMS Batch](#) component, which allows to trigger batch completion in a fixed scheduled interval.
- The [HTTP](#) and [HTTP4](#) producers now support the `disableStreamCache` option to allow to use the raw response stream as the message body instead of wrapping using [Stream caching](#)
- The [Camel CDI](#) component has been improved to better fit into the CDI programming model, cover a larger set of containers, and provide these new features:
  - The Camel events from the `org.apache.camel.management.event` package (like `CamelContextStartedEvent`) can be observed as CDI events

- The new CDI event Camel endpoint enable CDI events to be seamlessly consumed from (respectively produced by) Camel consumers (respectively Camel producers)
- CDI beans annotated with the `@Converter` annotation are automatically registered as type converters
- The CDI Camel contexts can be properly customised programmatically in bean constructor or `@PostConstruct` lifecycle callback
- Camel routes configured in `RouteBuilder` beans are added before the corresponding Camel contexts get started
- CDI Camel contexts are now properly adapted when deployed in OSGi containers and are registered as OSGi services
- Proper support of multiple Camel contexts across all the features
- A new [Camel CDI Test](#) module is available to ease testing of Camel CDI applications
- The [XSLT](#) component now supports Saxon's [integrated extension functions](#).
- Added option `reuseChannel` to [Netty4](#) which allows to reuse existing Channel when using a netty producer to call a server multiple times during routing an Exchange.
- Idempotent Repository now supports optional serialized headers
- [Rest DSL](#) adjusted to swagger spec 2.0 and [Swagger Java](#) as well.
- [Rest DSL](#) allows to specify response headers as well.
- [Rest DSL](#) is exposing the REST services using all local IP address (eg 0.0.0.0) by default.
- [Camel-quartz2](#) now supports customCalendar to avoid specific range of dates (Holidays for example) in Scheduler and Triggers.
- Improved [Bean Language](#) to be able to invoke static methods on pure static classes.
- When [Bean Language](#) failed to invoke a method from an OGNL method chain (eg invoke the following 3 methods "doFoo.doSomething.getBar") and a method returned null which prevent further method invocation as that would cause a NullPointerException. Now you have a more detailed exception message to better explain this.
- [JAXB](#) data format and type converter now supports converting from XML to POJO for classes that are using ObjectFactory classes instead of XmlRootElement annotations.
- The [Kafka](#) component has been migrated to use the Java Kafka client instead of Scala. As such there may be migration efforts or code changes that can affect users upgrading.
- The [Loop](#) EIP now allows to run in a while loop mode, so it loops until the predicate returns false.
- Improved [Rest DSL](#) when CORS enabled to process and return the configured CORS headers in the rest-dsl in all the supported [Rest DSL](#) components.
- The `options` verb in the [Rest DSL](#) has been deprecated and are not in use if CORS is enabled.
- The [LogEIP](#) allows to use a global configured log name, instead of using route id as the logger name
- A new `camel-spring-boot-starter` module that is a Spring-Boot starter module like any other starter modules.
- Allow to run Camel [Spring Boot](#) applications and keep the JVM running in a easy way by setting `camel.springboot.main-run-controller = true` option.
- The [SQL Component](#) supports using SQL IN queries in the producer where the number of values in the IN clause is dynamic from the current exchange.
- The [Swagger Java](#) module now supports YAML as response format in addition to the existing JSON format.

Fixed these issues

- The [Swagger Java](#) now parses nested types in the POJO model that has been annotated with the swagger api annotations to use in the schema api model
- Fixed [Rest DSL](#) with `apiContextPath` fail to start if there are 2 or more rest's in use.
- Fixed [Rest DSL](#) with [Swagger Java](#) to output the swagger API with property placeholder resolved to actual used values.
- [Paho](#) component name is not limited to 4 characters anymore.
- Fixed [Spring Boot](#) not starting Camel routes if running in Spring Cloud.
- Fixed an issue with [Swagger Java](#) using api-docs could lead to api-doc route being added multiple times
- Fixed a few things missing in the generated swagger model when using [Swagger Java](#)
- Fixed using `statement.xxx` options on the [JDBC](#) consumer would only be used in first poll.
- Fixed [HTTP](#) and [HTTP4](#) to keep trailing slash if provided in uri when calling remote HTTP service.
- Fixed [OnCompletion](#) to keep any caught exception stored as property on the `Exchange` which allows to access that information to know if there was an exception during routing.
- Fixed an issue with [Bean](#) component or [Simple](#) language with OGNL method call, would pick method with `java.lang.Object` type over a better suited method, when the method is overloaded.
- Fixed `camel-blueprint` to wait for the bundles with the component and data-formats the [Rest DSL](#) has been configured to use, are available before starting Camel.
- Fixed an issue with [Error Handler](#) would log any handled(true) exceptions.
- Fixed a memory leak if adding/removing routes using [Idempotent Consumer](#) where the repository would not be removed from the JMX registry when routes is removed.
- Fixed issue with [Spring Boot](#) would eager shutdown some Camel resources like type converters, which spring-boot should let Camel handle the lifecycle of these resources.
- Fixed [CXFRS](#) using simple binding mode would use the content-length header from the incoming message in the response, instead of calculating the length based on the output message body.
- Fixed an issue with [Simple](#) if starting the expression with `file:` could fail with an error.
- Fixed [Spring Boot](#) may report spring type converter errors if [Stream caching](#) is enabled.
- Fixed [Idempotent Consumer](#) would not propagate exception thrown from `IdempotentRepository` to Camel's [Error Handler](#) such as `onException`

## New Enterprise Integration Patterns

### New Components

- `camel-aws` - AWS kinesis streams component added
- `camel-aws` - DynamoDB streams component added
- `camel-braintree` - for interacting with Braintree Payments.
- `camel-test-cdi` - ease testing of Camel CDI applications
- `camel-etcd` - Integrating Camel with [Etcd](#) key value store
- `camel-kubernetes` - Integrates Camel with Kubernetes
- `camel-ignite` - for working with Apache Ignite.
- `camel-ironmq` - messaging with the [IronMQ](#) cloud messaging.

- [camel-jcache](#) - support JCache / JSR107 caching.
- [camel-mlp](#) - for working with the HL7 MLLP protocol. This is an alternative to [HL7](#) component.
- [camel-nats](#) - for interacting with Nats messaging platform
- [camel-spark](#) - bridges Apache Spark computations with Camel endpoints
- [camel-sql](#) - Now supports calling stored procedures using the new [sql-stored component](#).

## New DSL

## New Annotations

## New Data Formats

- [MIME-Multipart](#) - mime based data format.
- [LZF Data Format](#) - LZF based data format.
- [YAML Data Format](#) - YAML based data format.
- [Hessian](#) - Caucho Hessian based data format.

## New Languages

## New Examples

- [camel-example-cdi-metrics](#) - illustrates the integration between Camel, Dropwizard Metrics and CDI.
- [camel-example-cdi-properties](#) - illustrates the integration between Camel, DeltaSpike and CDI for configuration properties.
- [camel-example-cdi-osgi](#) - a CDI application using the SJMS component that can be executed inside an OSGi container using PAX CDI.
- [camel-example-cdi-rest-servlet](#) - illustrates the Camel REST DSL being used in a Web application that uses CDI as dependency injection framework.
- [camel-example-cdi-test](#) - demonstrates the testing features that are provided as part of the integration between Camel and CDI.
- [camel-example-spring-boot-metrics](#) - showing a Camel spring-boot application that report metrics to Graphite. Requires Graphite running on your network.
- [camel-example-widget-gadget-cdi](#) - The Widget and Gadget use-case from the EIP book implemented in Java with CDI dependency Injection.
- [camel-example-widget-gadget-java](#) - The Widget and Gadget use-case from the EIP book implemented in plain old Java Main without any kind of *application server*.
- [camel-example-widget-gadget-xml](#) - The Widget and Gadget use-case from the EIP book implemented in Spring XML without any Java code or any kind of *application server*.

## New Tutorials

## API breaking

- As part of the [Camel CDI](#) component refactoring:
  - The `@ContextName` qualifier does not have a default empty value anymore as it is irrelevant
  - The `CdiPropertiesComponent` class has been removed, the standard `PropertiesComponent` can be used instead

## Known Issues

- [camel-guice](#) cannot install in Karaf/ServiceMix

## Dependency upgrades

- Apache Cassandra from 2.1.8 to 2.2.2
- Apache Deltaspike from 1.5.1 to 1.5.2
- Apache Gora from 0.6 to 0.6.1
- Apache Jackrabbit from 2.11.0 to 2.11.3
- Apache Kafka from 0.8.2.2 to 0.9.0.0
- Apache Lucene from 4.10.4 to 5.2.1
- Apache Mina from 2.0.9 to 2.0.10
- Apache OpenEjb from 4.7.2 to 4.7.3
- Apache Solr from 4.10.4 to 5.2.1
- Aws SDK Java to version 1.10.37
- Atmosphere from 2.4.0 to 2.4.1.4
- Cassandra driver from 2.1.8 to 2.1.9
- Cassandra-unit from 2.1.9.2 to 2.2.2.1
- Cglib from 3.1 to 3.2.0
- Classmate from 1.3.0 to 1.3.1
- Chunk from 3.0.1 to 3.1.2
- Drools from 6.2.0.Final to 6.3.0.Final
- EHCache from 2.10.0 to 2.10.1
- Elasticsearch from 1.7.3 to 2.0.0
- Embedded MongoDB from 1.50.0 to 1.50.2
- Facebook4j from 2.3.1 to 2.4.2
- GAE from 1.9.26 to 1.9.32
- Google Gson from 2.4 to 2.6.1

- Hazelcast from 3.5.2 to 3.6
- Infinispan from 8.0.1.Final to 8.1.2.Final
- Jackson from 2.6.3 to 2.7.1
- Jboss Xnio from 3.3.1.Final to 3.3.4.Final
- Jboss Weld from 2.3.0.Final to 2.3.3.Final
- Jbpm from 6.2.0.Final to 6.3.0.Final
- Jgit from 4.1.0.201509280440-r to 4.1.1.201511131810-r
- Joda-time from 2.8.2 to 2.9.1
- Json-path from 2.0.0 to 2.1.0
- Jt400 from 8.6 to 8.7
- Kie from 6.2.0.Final to 6.3.0.Final
- Maven Exec plugin from 1.2.1 to 1.4.0
- MongoDB Java driver from 3.0.4 to 3.2.0
- Mvel from 2.2.6.Final to 2.2.7.Final
- Ognl from 3.1 to 3.1.1
- Olingo from 2.0.4 to 2.0.5
- Optaplanner from 6.2.0.Final to 6.3.0.Final
- Pax-exam from 4.6.0 to 4.8.0
- Spring 4.1.x to 4.2.x
- Spring-Boot 1.2.x to 1.3.x
- Spring Integration 2.x to 4.x
- slf4j-api 1.6.x to 1.7.x
- Undertow from 1.3.3.Final to 1.3.7.Final
- Vertx from 3.1.0 to 3.2.0
- Zookeeper from 3.4.6 to 3.4.7

## Important changes to consider when upgrading

- **This is the last release that will support Java 1.7. Releases after this is expected to require Java 1.8+.**
- When installing the camel feature in Apache Karaf, then camel-spring is not longer installed by default. You need to install camel-spring feature if using spring-dm on Karaf.
- Reworked the Apache Karaf feature to not install camel-spring by default. The problem is camel-spring is using spring-dm which is only supports Spring 3.x, and therefore Spring 3.x was installed and used. All the Camel features that uses Spring JARs are now using Spring 4.x.
- Removed camel-docker from karaf features as it does not work in OSGi
- Some changes in [Rest DSL](#) to adjust naming and types to the Swagger Spec 2.0
- Any custom component that supports suspension in doSuspend/doResume should implement the new `Suspendable` marker interface, so Camel knows there is custom logic for suspension in the component.
- Exchange and Message only output id in their toString method to avoid outputting any message details such as sensitive details from message bodies.
- Removed camel-hbase as Karaf feature as it did not really work well in OSGi
- Upgraded camel-hbase to Hadoop 2.x and HBase 1.1.x
- camel-infinispan requires Java 8.
- camel-mustache requires Java 8.
- Support for Spring 4.0.x is deprecated. Support for Spring 3.x is being **removed** in the next release.
- Support for Karaf 2.x and 3.x is deprecated. And removed in next release, where Karaf 4.x onwards is only supported.
- camel-jetty8 is deprecated and being removed in next release.
- Moved some Camel tooling related dependencies (such as maven/plexus) from the Camel Parent BOM to the tooling BOM (to have them separated).
- camel-amqp do not support 0.9 anymore.
- camel-spring-integration feature has been removed from the Camel karaf.
- The [Mail](#) component now requires to configure to, cc, and bcc using lower case keys, eg to=foo@bar.com, instead of `To=foo@bar.com` as previously.
- The [File](#) consumer no longer probe the file content by default. See the option `probeContentType` for more details.
- If using [Bean](#) or [Class](#) component and specifying additional parameters in the endpoint uri to configure on the bean, then these options should now be prefixed with bean., eg `foo=123` is now `bean.foo=123`.
- The [Twitter](#) delay option is changed from seconds to milli seconds by default, eg 10 should be 10000 to indicate 10 seconds. This is aligned how other components with delay option behaves.
- The options `attributeNames` and `messageAttributeNames` on [AWS-SQS](#) is changed to a string type where you can separate multiple values using comma. Before the type was a Collection which was much harder to configure in the Camel uris.
- [Rest DSL](#) is exposing the REST services using all local IP address (eg 0.0.0.0) by default, instead of the local IP address of the host.
- The [hbase](#) component now require row mapping from the endpoint uri to be prefixed with `row.` as prefix.  
Before: `family=info&qualifier=firstName&family2=birthdate&qualifier2=year`. After: `row.family=info&row.qualifier=firstName&row.family2=birthdate&row.qualifier2=year`.
- As part of the [Camel CDI](#) component refactoring, [DeltaSpike](#) is not used anymore for the sourcing of the configuration properties. This new version of the component is agnostic to any configuration sourcing mechanism and delegates that concern to the application so that it can declare a custom `PropertiesComponent` bean whose sourcing is tailored to its need. [DeltaSpike](#) can still be used by the application by declaring a `PropertiesComponent` bean configured with a `PropertiesParser` relying on [DeltaSpike](#). See the `camel-example-cdi-properties` example for more details.
- The [Kafka](#) component has been migrated to use the Java Kafka client instead of Scala. As such there may be migration efforts or code changes that can affect users upgrading.
- Improved [Rest DSL](#) when CORS enabled to process and return the configured CORS headers in the rest-dsl in all the supported [Rest DSL](#) components.
- The `options` verb in the [Rest DSL](#) has been deprecated and are not in use if CORS is enabled.
- camel-gae is deprecated and will be removed from Camel 2.18 onwards.

## Getting the Distributions

### Binary Distributions

Description	Download Link	PGP Signature file of download
Windows Distribution	<a href="#">apache-camel-2.17.0.zip</a>	<a href="#">apache-camel-2.17.0.zip.asc</a>
Unix/Linux/Cygwin Distribution	<a href="#">apache-camel-2.17.0.tar.gz</a>	<a href="#">apache-camel-2.17.0.tar.gz.asc</a>

The above URLs use redirection

 The above URLs use the Apache Mirror system to redirect you to a suitable mirror for your download. Some users have experienced issues with some versions of browsers (e.g. some Safari browsers). If the download doesn't seem to work for you from the above URL then try using [FireFox](#)

### Source Distributions

Description	Download Link	PGP Signature file of download
Source for Windows	<a href="#">apache-camel-2.17.0-src.zip</a>	<a href="#">apache-camel-2.17.0-src.zip.asc</a>

## Getting the Binaries using Maven 2

To use this release in your maven project, the proper dependency configuration that you should use in your [Maven POM](#) is:

```
<dependency>
  <groupId>org.apache.camel</groupId>
  <artifactId>camel-core</artifactId>
  <version>2.17.0</version>
</dependency>
```

## SVN Tag Checkout

```
svn co http://svn.apache.org/repos/asf/camel/tags/camel-2.17.0
```

## Changelog

For a more detailed view of new features and bug fixes, see the:

- [Release notes for 2.17.0](#)