

# Camel 2.13.0 Release

## Camel 2.13.0 release

### New and Noteworthy

Welcome to the 2.13.0 release which approx 480 issues resolved (new features, improvements and bug fixes such as...)

- Using `<camelContext>` in Spring will now shutdown `<camelContext>` eager by default. This ensure a cleaner shutdown of Camel, as dependent `<bean>` are not shutdown at this moment. The `<bean>` will then be shutdown after `<camelContext>`. There is a new attribute `shutdownEager` on `<camelContext>` to turn this off, and use the old behavior.
- The MBean names registered by [Camel JMX](#) no longer include the hostname in the `context` part, eg before `context=myHost/myCamelId` and now after `context=myCamelId`. Having the hostname in the MBean name does not bring much value, and in fact makes things more complicated as the mbean name changes depending on the host running Camel.
- [MBean naming](#) in OSGi cleaned up to use simpler naming with `symbolicName`. Before we could have MBean names with duplicate bundle ids such as `context=114-114-camel-6`, which now is using the symbolic name instead, `context=MyApplication`.
- When using multiple OSGi Blueprint `<cm:property-placeholder>`'s then Camel now favors using non-default placeholders, or the last property-placeholder defined in the Blueprint XML file. This allows for example to define default properties in one placeholder, and override these values in other placeholders.
- [FTP](#) consumer allow to download a single named file without using the FTP LIST command. This allows to download a known file from a FTP server even when the user account does not have permission to do FTP LIST command.
- [FTP](#) consumer allow to ignore file not found or insufficient file permission errors.
- [Data Format](#) using marshal now leverages [Stream caching](#) out of the box if enabled, which allows to marshal big streams and spool to disk, instead of being pure in-memory based.
- Improved using [Bean](#) when the bean is looked up in the [Registry](#), when using concurrent processing in the route.
- Added `cache` option to `beanRef` and `<bean>` in the DSL. This avoids looking up the [Bean](#) from the [Registry](#) on each usage; this can safely be done for singleton beans.
- Configuring [Data Formats](#) in XML attributes now supports reference lookup using the `#` syntax, eg `<jaxb xmlStreamWriterWrapper="#myWriterWrapper" . . .>`
- [JDBC](#) component now also support `outputType` to specify the expected output as either a List or single Object. As well allow to map to a bean using a `BeanRowMapper` to control the mapping of ROW names to bean properties.
- Both [Quartz](#) as well as [Quartz2](#) based [ScheduledRoutePolicy](#) has been improved to better support cluster setups (e.g. to not schedule jobs being already scheduled through another node inside a given cluster).
- Reduced the work the [Aggregate](#) EIP does while holding a lock during aggregation, which can lead to improved performance in some use-cases.
- [JndiRegistry](#) now implements all the find methods.
- [VM](#) component now supports `multipleConsumers=true` across deployment units.
- Added `@PreConsumed` to [JPA](#) consumer.
- Added `CamelFileName` header support to the [HDFS](#) producer
- Like as `JpaConsumer` now also `JpaProducer` of the [JPA](#) component supports the `CamelEntityManager` header.
- [Restlet](#) consumer now supports returning custom headers as HTTP headers from the Camel [Message](#).
- [Spring Java Config](#) `CamelConfiguration` now automatically detects all `RouteBuilder` instances registered in the Spring context if `CamelConfiguration#routes` method is not overridden.
- Added support for transferring custom headers with [RabbitMQ](#) component.
- [PGPDataFormat](#) enables stream cache during unmarshaling.
- `FileIdempotentRepository` now creates necessary parent directories together with the new repository file.
- [Properties](#) component and [property placeholder](#) will now ignore non existing JVM and environment options as well if `ignoreMissingLocation` is set to `true`.
- Improved [Stream](#) consumer to enrich the [Message](#) with headers about current index / completion flag.
- Added `@BeanInject` to inject beans (obtained from the [Registry](#)) in your beans such as `RouteBuilder` classes.
- Polished the [Karaf](#) commands to output tables sorted and auto calculated width, so the data is formatted nicely.
- Introduced `UnitOfWorkFactory` to allow end users to use custom `UnitOfWork` implementations created by the factory.
- Doing request/reply over [JMS](#) allows per individual message `requestTimeout` value by specifying the timeout value in the `"CamelJmsRequestTimeout"` header.
- [Quartz](#) and [Quartz2](#) components now enlist the quartz scheduler by default in JMX, unless configured to be disabled.
- [Twitter](#) Component supports to set proxy.
- [Netty HTTP](#) consumer now remembers the raw (not decoded) value of URI query in the `"CamelHttpRawQuery"` header.
- [Bean Validator](#) component can be now configured to use custom `ValidationProviderResolver` ([CAMEL-7162](#))
- `TypeConverterRegistry` exposes more information in JMX and allows to browse all the registered type converters.

- [Bean Validator](#) component is now shipped with the optional OSGi-friendly [HibernateValidationProviderResolver](#) ([CAMEL-7168](#))
- [AMQP](#) has been updated to QPid 0.24 and has been tested using AMQP 1.0.
- Allow to turn off useOriginalMessage (camelContext.setAllowUseOriginalMessage) which avoids a defensive copy of the incoming original message, which can improve performance; in situations where access to the original message is not needed.
- [Bean Validator](#) component now automatically uses OSGi-friendly [ValidationProviderResolver](#) if deployed in the OSGi environment ([CAMEL-7212](#))
- [Threads DSL](#) thread pool options and [executorServiceRef](#) option are now mutually exclusive. ([CAMEL-7250](#))
- [Shiro Security](#) now supports access control by roles.

## Fixed Issues

- Fixed an [ArrayIndexOutOfBoundsException](#) with [Message History](#) when using [SEDA](#)
- Fixed [requestTimeout](#) on [Netty](#) not triggering when we have received message.
- Fixed [Parameter Binding Annotations](#) on boolean types to evaluate as [Predicate](#) instead of [Expression](#)
- Fixed using [File](#) consumer with `delete=true&readLock=fileLock` not being able to delete the file on Windows.
- Fixed [Throttler](#) to honor time slots after period expires (eg so it works consistently and as expected).
- Fixed getting [JMSXUserID](#) property when consuming from [ActiveMQ](#)
- Fixed [interceptFrom](#) to support property placeholders
- Fixed a race condition in initializing [SSLContext](#) in [Netty](#) and [Netty HTTP](#)
- Fixed using [Recipient List](#), [Routing Slip](#) calling another route which is configured with `NoErrorHandler`, and an exception occurred in that route, would be propagated back as not-exhausted, allow the caller route to have its error handler react on the exception.
- Fixed [Quartz](#) and exception was thrown when scheduling a job, would affect during shutdown, assuming the job was still in progress, and not shutdown the Quartz scheduler.
- Fixed so you can configure [Stomp](#) endpoints using [URIs](#)
- Fixed memory leak when using [Language](#) component with `camel-script` languages and having `contentCache=false`
- Fixed [Error Handler](#) may log at `WARN` level "Cannot determine current route from Exchange" when using [Splitter](#)
- Fixed `camel-fop` to work in Apache [Karaf](#) and [ServiceMix](#)
- Fixed [HDFS](#) producer to use the configured [UuidGenerator](#) when generating split file names to avoid filename collisions
- Fixed [JpaProducer](#) and [JpaConsumer](#) of the [JPA](#) component to not share/reuse the same [EntityManager](#) object which could cause problems if this would occur in the context of multiple/different threads.
- Fixed [HTTP4](#) to support multiple custom component names in use, each configured using different [SSLContext](#).
- Fixed [Content Based Router](#) using method call's as [Predicates](#) if threw an exception, the [Content Based Router](#) will call next predicate before triggering [Error Handler](#).
- Fixed [Netty HTTP](#) producer with query parameters may use wrong parameters from a previous call.
- Fixed [Netty](#) producer may cause [NullPointerException](#) in `messageReceived` in client handler, if a [Channel](#) was destroyed from the producer pool.
- Fixed [File](#) and [FTP](#) consumer(s) when `idempotent=true` could not detect changed file(s) as a new file, but would regard the file as idempotent.
- Fixed [Stream](#) consumer reading a file using `groupLines` would not read last group of message if there was not enough lines to hit the threshold.
- Fixed deleting the *done file* if [File](#) or [FTP](#) consumer rollback and the `moveFailed` option is in use.
- Fixed [avro](#) data format in OSGi not being able to load schema classes.
- Fixed [JMX](#) when using custom beans in routes with `@ManagedResource` to expose [JMX](#) attributes/operations to not include Camel standard attributes/operations from its processors, which would be shown as unavailable in [JMX](#) consoles.
- Fixed [Quartz2](#) simple trigger repeat count default value not working causing the trigger to only fire once, instead of forever.
- Fixed [JMS](#) with a `JMSReplyTo` header using a topic, was mistakenly used as a queue instead.
- Fixed [CSV](#) to not modify the properties of the strategy constants declared by `org.apache.commons.csv.CSVStrategy` which would cause side effects when multiple `CsvDataFormat` objects are **concurrently** in use.
- Fixed `camel-script` may return result from previous evaluation instead of result from current exchange.
- Fixed [ThreadsProcessor](#) to resolve [RejectedPolicy](#) from the referenced [ThreadPoolProfile](#) ([CAMEL-7240](#))
- Fixed [XmlJson](#) `elementName` and `arrayName` properties when setting them from the `xmljson(Map)` DSL.

## New Enterprise Integration Patterns

### New Components

- `camel-hdfs2` - integration with HDFS using Hadoop 2.x client
- `camel-infinispan` - to interact with [Infinispan](#) distributed data grid / cache.
- `camel-jgroups` - provides exchange of messages between Camel infrastructure and [JGroups](#) clusters.
- `camel-kafka` - integration with Apache Kafka
- `camel-optaplanner` - to use [OptaPlanner](#) for problem solving plans.
- `camel-splunk` - enables you to publish and search for events in Splunk
- `camel-swf` - for managing workflows running on Amazon's [Simple Workflow Service](#)

### New Camel Maven Archetypes

- `camel-archetype-cxf-code-first-blueprint`
- `camel-archetype-cxf-contract-first-blueprint`

### New DSL

### New Annotations

- `@BeanInject`

### New Data Formats

## New Languages

- [JXPath](#) - To perform [Expression](#) and [Predicate](#) on json payloads.

## New Examples

## New Tutorials

## New Tooling

## API changes

- The interface of `org.apache.camel.component.http4.HttpClientConfigurer`'s method `configureHttpClient(HttpClient client)` was changed to `configureHttpClient(HttpClientBuilder clientBuilder)`
- Added `getRegistry(T)` to `CamelContext`

## Known Issues

- Spring 4.x is not yet official supported.
- Cannot build the source code using Java 8.

## Dependency Upgrades

You can see any dependency changes using [Camel Dependencies Version Differences web tool](#) created by Tracy Snell from the Camel Team.

The below list is a best effort, and we may miss some updates, its better to use the [Camel Dependencies Version Differences web tool](#) which uses the released projects for comparison.

- ActiveMQ 5.8.0 to 5.9.0
- AHC 1.7.20 to 1.8.3
- APNS 0.1.6 to 0.2.3
- Atomikos 3.8.0 to 3.9.2
- AWS-SDK 1.5.1 to 1.6.3
- Avro 1.7.3 to 1.7.5
- BeanIO 2.0.6 to 2.0.7
- Classmate from 0.8.0 to 1.0.0
- Commons HttpClient 4.2.5 to 4.3.3
- Commons Httpcore 4.2.4 to 4.3.2
- CXF 2.7.6 to 2.7.10
- Ehcache 2.7.2 to 2.8.1
- Elasticsearch 0.20.6 to 1.0.0
- FOP 1.0 to 1.1
- Guava 14.0.1 to 16.0.1
- Hadoop 1.2.0 to 1.2.1. Hadoop 2.3.0 supported by camel-hdfs2 component.
- Hazelcast 2.6 to 3.0.2
- Hibernate Validator 5.0.1.Final to 5.0.3.Final
- ICAL4j 1.0.4 to 1.0.5.2
- Jackson 2.2.2 to 2.3.2
- Jclouds 1.6.2-incubating to 1.7.0
- Jettison 1.3.4 to 1.3.5
- Jetty 7.6.9 to 8.1.12
- JLine 0.9.94 to 2.11
- Joda time 2.1 to 2.3
- JRuby 1.7.4 to 1.7.5
- JSCH 0.1.49 to 0.1.50
- LevelDb JNI 1.7 to 1.8.
- Lucene 3.6.0 to 4.6.1
- MongoDB Java Driver 2.11.2 to 2.11.4
- Mustache 0.8.12 to 0.8.13
- MVEL 2.1.6.Final to 2.1.7.Final
- MyBatis 3.2.2 to 3.2.5
- Netty3 3.8.0.Final to 3.9.0.Final
- OGNL 3.0.6 to 3.0.8
- Pax Logging 1.6.10 to 1.7.1
- Protobuf 2.3 to 2.5
- Qpid 0.20 to 0.26
- Quartz 2.2.0 to 2.2.1
- RabbitMQ amqp Java Client 3.1.3 to 3.2.2.
- Restlet 2.0.15 to 2.1.7
- RxJava 0.11.1 to 0.17.1
- Saxon 9.5.0.2 to 9.5.1-4
- Scala 2.10.2 to 2.10.3

- Servlet API 2.5 to 3.0
- Shiro to 1.2.3.
- Slf4j 1.7.5 to 1.7.6
- Snappy 1.0.4.1 to 1.1.0.1
- SNMP4J 2.2.2 to 2.2.3
- SolrJ 3.6.2 to 4.6.1
- Spring Batch 2.2.1.RELEASE to 2.2.2.RELEASE
- Spring Integration 2.2.4.RELEASE to 2.2.6.RELEASE
- Spring Redis 1.0.4.RELEASE to 1.1.1.RELEASE
- Spring WS 2.1.3.RELEASE to 2.1.4.RELEASE
- SSHD 0.8.0 to 0.10.1
- StompJMS 1.17 to 1.19
- TestNG 6.8.5 to 6.8.7
- Twitter4j 3.0.3 to 3.0.5
- Weld 1.1.5.Final to 1.1.18.Final
- XBean Spring 3.14 to 3.16
- XmlSec 1.5.5 to 1.5.6
- XStream 1.4.4 to 1.4.7

## Internal changes

- Add `RouteAware` API to inject `Route` for `Service`'s in routes, for example the `Consumer`'s of the routes.
- `IOHelper.getCharsetName` will now lookup key `Exchange.CHARSET_NAME` in header first and then fallback to property.
- Camel Spring no longer automatic registers "spring-event://default" endpoint, which wasn't in use anyway.
- [BAM](#) component now uses raw `EntityManager` JPA API instead of deprecated `Spring JpaTemplate`

## Changes that may affect end users

- Components depending on Servlet API has been upgrade from Servlet API 2.5 to 3.0
- Jetty upgraded from 7.6.x to 8.1.x
- [Hazelcast Component](#) uses `Object` instead of `String` as the key.
- [Hazelcast Component](#) atomic number producer was using incorrect name ([CAMEL-6833](#)). Name will be different with this release.
- [Hazelcast Component](#) instance consumer was incorrectly writing headers to the "out" message as opposed to the "in" message. Headers are now written to the "in" message.
- [Hazelcast Component](#) map/multimap consumer were sending header value "evnict" when an item was evicted. This has been corrected and the value "evicted" is used now.
- The enum value `NON_BLOCKING` has been removed in `camel-apns` as its no longer support in APNS itself.
- `DefaultTimeoutMap` must call `start` to initialize the map before use.
- `@ExcludeRoutes` test annotation now accepts only classes implementing `RoutesBuilder` interface.
- The MBean names registered by [Camel JMX](#) no longer include the hostname in the `context` part, eg before `context=myHost/myCamelId` and now after `context=myCamelId`. Having the hostname in the MBean name does not bring much value, and in fact makes things more complicated as the mbean name changes depending on the host running Camel. There is an option `includeHostName` that can be configure as `true` to preserve the old behavior.
- [MBean naming](#) in OSGi cleaned up to use simpler naming with `symbolicName`. Before we could have MBean names with duplicate bundle ids such as `context=114-114-camel-6`, which now is using the symbolic name instead, `context=MyApplication`.
- Added `beforeAddRoutes` and `afterAddRoutes` to `org.apache.camel.component.servlet.listener.CamelContextLifecycle` in [ServletListener Component](#)
- The [Language](#) component now no longer caches the compiled script as that could cause side-effects. You can set `cacheScript=true` to enable the previous behavior if your script is safe to do so.
- The default value of `urlDecodeHeaders` option on [Netty HTTP](#) has been changed from `true` to `false`.
- When using `<jmxAgent>` in `spring/blueprint` then the `createConnector` attribute has changed default value from `true` to `false` (which is also the default in `camel-core` without `spring/blueprint`)
- [camel-cache](#) no longer includes `ehcache.xml` out of the box. Instead the default configuration from `ehcache` is in use, if end user has not explicit configured a configuration file to be used.
- `camel-cdi` upgraded from `deltaspike 0.3` to `0.5` which may affect upgrades.
- Java DSL when using `onException.backOffMultiplier` or `onException.collisionAvoidancePercent/collisionAvoidanceFactor` will automatic enable `backoff/collision avoidance`. Before you had to call `useExponentialBackOff/useCollisionAvoidance` as well. This is now aligned how `ErrorHandler` does the same.
- changed default value of [NoLocal](#) from `true` to `false` in `camel-sjms` when creating consumer.

## Important changes to consider when upgrading

- Spring 3.1.x is no longer supported.
- Using `<camelContext>` in Spring will now shutdown `<camelContext>` eager by default. This ensure a cleaner shutdown of Camel, as dependent `<bean>` are not shutdown at this moment. The `<bean>` will then be shutdown after `<camelContext>`. There is a new attribute `shutdownEager` on `<camelContext>` to turn this off, and use the old behavior.
- If using `groupedExchanges` option on [Aggregate](#) EIP then the [Exchanges](#) is now stored on the message body as well. The old way of storing as a property is considered deprecated. See more details at [Aggregate](#).
- JMX Load statistics has been disabled by default. You can enable this by setting `loadStatisticsEnabled=true` in `<jmxAgent>`. See more details at [Camel JMX](#).
- Java 6 support is being deprecated and dropped from next release onwards.

## Getting the Distributions

## Binary Distributions

Description	Download Link	PGP Signature file of download
Windows Distribution	<a href="#">apache-camel-2.13.0.zip</a>	<a href="#">apache-camel-2.13.0.zip.asc</a>
Unix/Linux/Cygwin Distribution	<a href="#">apache-camel-2.13.0.tar.gz</a>	<a href="#">apache-camel-2.13.0.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-2.13.0-src.zip</a>	<a href="#">apache-camel-2.13.0-src.zip.asc</a>
Source for Unix/Linux/Cygwin	<a href="#">apache-camel-2.13.0-src.tar.gz</a>	<a href="#">apache-camel-2.13.0-src.tar.gz.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>2.13.0</version>
</dependency>
```

## Git Tag Checkout

```
git clone https://git-wip-us.apache.org/repos/asf/camel.git
git checkout camel-2.13.0
```

## Changelog

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

- [release notes for 2.13.0](#)