

Camel 2.2.0 Release

Camel 2.2.0 release

New and Noteworthy

Welcome to the 2.2.0 release which approx 180 issues resolved (new features, improvements and bug fixes such as...)

- [Routing Slip](#) now breaks when an exception occurred
- [MINA](#) now supports easier configuration of multiple custom codecs, which is needed when working with SSL.
- Improved [TransactionErrorHandler](#) two fold by reducing complexity of internal logic in Camel when working with transactions. And added better support for using multiple transactions in multiple routes. See more at section *Using multiple routes with different propagation behaviors* at [Transactional Client](#).
- [Recipient List](#) now supports `parallelProcessing`, `stopOnException` and references for `AggregationStrategy` and thread pools. This applies to the `@RecipientList` annotation as well.
- Cleanup of spring jar dependencies.
- [File Language](#) is now merged directly into [Simple](#) language which among others makes [Content Based Router](#) in Spring XML using file language syntax easier.
- [HTTP](#) is now using CamelContext Properties instead of System Properties to set the http proxy configuration.
- Fixed [Camel Proxy](#) throwing exception if `null` was returned.
- Improved error handling with [Recipient List](#) and [Multicast](#) which now performs redelivery etc. on that *output* node instead of starting again from the scratch.
- Improved the [ETL Example](#) and [BAM Example](#) so they can be deployed into ServiceMix4(Karaf).
- [SEDA](#) now supports `multipleConsumers` which allows you to use it for pub-sub messaging where each consumer concurrently will receive a copy of the [Exchange](#). This is done with the help from [Multicast](#) EIP.
- [DefaultErrorHandler](#) will now log a log line at `ERROR` level in case an Exception was thrown which could not be redelivered either. This aid to see that an Exception occurred instead of relying on the client logging it. You can customize this behavior on the [Error Handler](#).
- Fixed `groupedExchange` not working properly on [Aggregator](#).
- [Graceful Shutdown](#) which ensures Camel now shutdown in a more reliable and graceful manner. This also works with in memory components such as [SEDA](#).
- Added new [Log](#) in the DSL for dynamic human readable log messages.
- Camel now supports Spring 3.0 as well (will still use Camel 2.5.6 as default). However camel-spring-javaconfig now requires Spring 3.0 as Spring have native support for java configuration.
- Added methods for getting header and property (Exchange / Message API) which uses a default parameter.
- Added `OsgiClassResolver` which supports to load the class or resources within OSGi platform.
- Added `fireNow` option to [Quartz](#) component to fire the trigger when the route is started.
- [File](#) will delete orphaned lock files on startup. This allows it to selfheal in cases when Camel was terminated/killed abruptly.
- Fixed issue with using Spring XML for routing could in some cases *loose* route scoped configured `<onException>`, `<onCompletion/>` when using in combination with `<transacted/>`.
- `<proxy>`, `<export>`, `<endpoint>`, `<template>`, `<consumerTemplate>` can be defined outside of `<camelContext>`.
- [JAXB](#) will filter the nonXml characters when it unmarshals the inputstream or marshal the java object.
- Slightly reworked how correlation id is determined in [JMS](#) when using InOut messaging. See more at *important changes to consider when upgrading*.
- [EventNotifier](#) improved to allow ignoring events based on categories. Easier configuration to only notify those events you care about.
- Added `PublishEventNotifier` to send the events to any kind of Camel [Endpoint](#).
- [JDBC](#) added option `useJDBC4ColumnNameAndLabelSemantics` to let you choose semantics to use in case you have issue with your JDBC driver when using SQL SELECT with aliases.
- [Cache](#) can now store message bodies which are `Serializable` out of the box and not forcing to be converted into a byte stream as it otherwise would.
- Fixed an issue with running the [Examples](#) using ANT.
- `ManagementStrategy` now supports using multiple `EventNotifier` instead of just one.
- Introduced [NotifyBuilder](#) which helps during testing. For example when testing routes without mocks etc. It will help to notify when a certain condition has occurred such as X messages has is done etc.
- [FTP](#) now allows using starting directory with a dot in the name. It used to invalidate those to easy transition from Camel 1.x to 2.x as 1.x did also support using a file as starting (which was not good practice).
- [File](#) producer now supports option to keep last modified timestamp in case a message was received from some file source. You can also set the timestamp it should use in the header `Exchange.FILE_LAST_MODIFIED`.

- [Bean Language](#) using `bean` or `method` to invoke a [Bean](#) now supports using an existing bean instance. This is convenient for testing where you can quickly use an instance you have prepared in the unit test.
- [camel-jboss](#) for running Camel inside JBoss Application Server which uses its specialized classloading which prevents typically frameworks to not work out of the box inside JBoss.
- [Camel Extra](#) projects migrated to use Camel 2.1.0 release.
- [JAXB](#) now supports partial marshalling/unmarshalling.
- [Bean](#) parameter binding using Camel annotations now works with CGLIB enhanced classes.
- [CXF](#) supports to configure the endpoint with CXF style configuration.
- [DefaultErrorHandler](#) will now log stacktrace when an [Exchange](#) is exhausted (give up). Added new `logRetryStackTrace` option to set whether or not stacktraces are logged for failed redelivery attempts. Most people don't like the logs spammed if you retry 5 times etc.
- [Bean](#) component now honors void method invocations and accepts methods returning `null` as a valid response.
- [Error Handler](#) now catches and handles all exceptions (`Throwable`). This ensures fatal errors such as `NoSuchMethodException` etc. will now be handled by Camel and logged etc.
- [cxfbean](#) now propagates `CONTENT_TYPE` for other camel component to use.

New Enterprise Integration Patterns

New Components

- [camel-ftp](#) now supports `FTPS` (FTP over SSL/TLS)
- [camel-smpp](#)
- [camel-lucene](#)
- [camel-hdfs](#)
- [camel-jboss](#) for running Camel inside JBoss out of the box

New DSL

- `markRollbackOnlyLast`
- `log`

New Annotations

New Data Formats

- [camel-protobuf](#)
- [camel-dozer](#)

New Languages

New Examples

- [Cafe Example](#)

API breaking

DefaultComponent

The methods `DefaultComponent.resolveAndRemoveReferenceParameter` and `DefaultComponent.resolveAndRemoveReferenceParameterList` have been changed to make the leading `#` sign for bean references optional and to throw an `IllegalArgumentException` if a referenced bean is not in the registry. For details see [CAMEL-2248](#).

Exchange

The property `Exchange.FILTERED` have been removed as part of [CAMEL-2288](#).

Added two new methods for `getProperty` to accept a `defaultValue` as parameter to be used when the property does not exist.

Message

Added two new methods for `getHeader` to accept a `defaultValue` as parameter to be used when the header does not exist.

ExceptionHandler

The class `org.apache.camel.spi.ExceptionHandler` now have two additional methods which accept more details when an `Exception` occurred such as a message and the current `Exchange`.

@EndpointInject

The `name` attribute in `@EndpointInject` has been renamed to `ref` to be consistent with the other annotations. It's a reference for an `Endpoint` to be looked up in the [Registry](#).

BatchConsumer

The `BatchConsumer` have a new method `isBatchAllowed` which should be used during processing the batch, to break out the batch loop in case its not allowed anymore. Also `BatchConsumer` should consider implement `ShutdownAware` to support fine grained control during shutdown. See for example the `GenericFileConsumer` for details.

GenericFileProcessStrategy

The `GenericFileProcessStrategy` has a new method `prepareOnStartup` which is used to *prepare* the `File` component on startup, such as deleting orphaned lock files. The same applies for the `GenericFileExclusiveReadLockStrategy` interface in case you have developed a custom lock strategy.

ManagementStrategy

The `ManagementStrategy` now supports multiple `EventNotifier` and thus have its setter/getter methods adjusted to a `List<EventNotifier>` and there is a `addEventNotifier` method as well to add a single `EventNotifier`.

EventNotifier

The `EventNotifier` has new methods for filtering events based on categories. The `DefaultEventNotifier` class has been renamed to `LoggingEventNotifier` and a new `EventNotifierSupport` class is added which is meant for custom implementations to extend.

UnitOfWork

The `UnitOfWork` have adjusted its methods for setting `RouteContext` from a setter to a push and pop style.

Known Issues

See known issues from previous releases.

Using RecipientList and Mina

When using the [Recipient List](#) together with [MINA](#) endpoints Camel will over time hold on to memory which could lead to `OutOfMemoryErrors`. See more at: [CAMEL-2484](#)

Important changes to consider when upgrading

Features file moved

For Camel 2.2.0, the newly named `features.xml` file can be found in <http://repo2.maven.org/maven2/org/apache/camel/karaf/apache-camel/2.2.0/> You can use it in Karaf by doing a `features:addUrl mvn:org.apache.camel.karaf/apache-camel/2.2.0/xml/features`

Please don't install camel-osgi and camel-spring-osgi bundle in OSGi environment at the same time

We have an issue that installing `camel-osgi` and `camel-spring` in different order will cause the "http://camel.apache.org/schema/spring" namespace can't be handled rightly in OSGi environment. From Camel 2.2.0 we merged the `camel-osgi` and `camel-spring` bundles together, it's `camel-spring-osgi`.

In this way `camel-osgi` and `camel-spring-osgi` both have the `BundleActivator` which helps `camel-core` to load the component in OSGi container. If you install these two bundle at the same time will cause some conflict on the component finding.

You just need to install `camel-core`, `camel-spring-osgi` bundle to let the basic camel application work.

Spring jar dependencies have been cut down in camel-core.

The `pom.xml` file for `camel-core` have cut down number of spring jar dependencies to allow Camel to easily run without any Spring .jars at all. However this affects you if you use maven and use `camel-core` without `camel-spring` or `spring` at all. If you want to use JMX with Camel then you need to add the following 4 spring jars required: `spring-core.jar`, `spring-beans.jar`, `spring-context.jar`, `spring-aop.jar`. With maven you only need to depend on `spring-context` and `spring-aop` as they will bring in the other two jars.

See also [Camel JAR Dependencies](#) and [Camel JMX](#).

Using filter in Splitter

The `filter` no long influence the `split` and its `AggregationStrategy` as it was just too confusing why it did that. See [CAMEL-2288](#) for details.

camel-http and proxy

The `HTTP` component used to read Proxy settings directly from system properties. You should instead configure proxy wide information as properties to the `CamelContext` instead. See [HTTP](#) for details.

camel-jms

How correlationId is determined has been changed slightly. Now **JMS** will always use messageId if configured to do so using `useMessageIDAsCorrelationID==true`. Otherwise `JMSCorrelationID` is used but if that has not been set then Camel will fallback to use `messageId` instead. See the method `determineCorrelationId` in the class `org.apache.camel.component.jms.EndpointMessageListener` for actual implementation. This change will most likely not affect most users.

Getting the Distributions

Binary Distributions

Description	Download Link	PGP Signature file of download
Windows Distribution	apache-camel-2.2.0.zip	apache-camel-2.2.0.zip.asc
Unix/Linux/Cygwin Distribution	apache-camel-2.2.0.tar.gz	apache-camel-2.2.0.tar.gz.asc

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	apache-camel-2.2.0-src.zip	apache-camel-2.2.0-src.zip.asc
Source for Unix/Linux/Cygwin	apache-camel-2.2.0-src.tar.gz	apache-camel-2.2.0-src.tar.gz.asc

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>2.2.0</version>
</dependency>
```

SVN Tag Checkout

```
svn co http://svn.apache.org/repos/asf/camel/tags/camel-2.2.0
```

Changelog

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

- [JIRA Release notes for 2.2.0](#)