Camel 2.19.0 Release

Camel 2.19.0 Release

New and Noteworthy

Welcome to the 2.19.0 release which resolved over 670 issues including new features, improvements and bug fixes.

- Introduced Camel Connector's which is a simplified version of a Camel component that has been pre-configured for a specific use-case.
- Upgraded to Spring Boot 1.5.x.
- The Camel Maven Plugin now provides the `camel:validate` goal to parse your Java and XML source code for any Camel routes and report invalid Camel endpoint uri and simple expression errors. You can run this at code time (not runtime).
- Camel `Main` and Spring Boot and Camel Maven Plugin can now auto terminate the JVM after Camel has been running for maximum duration of seconds, processed messages or been idle for a period.
- Camel source code can build with Java 9 in preparation for official support for Java 9 in a future release
- All the Camel Spring Boot starter components now has more of the components default values included in their metadata which allows tooling to display such information
- Deprecated more components and camel-core APIs that will be dropped in Camel 3.0 or sometime in the future
- Introduced `ReloadStrategy` as SPI which allows custom providers to implement logic for triggering live reloads of Camel routes.
- The Camel Maven Plugin now allows to live reload route changes from XML files when running Camel. This can also be enabled from the `Main` class.
- Introduced a new `camel-catalog-rest` artifact which is a tiny standalone REST API of the CamelCatalog using JAX-RS and Swagger Annotations.
- Added `camel-catalog-rest-app` as a standalone application which used Apache CXF with Jetty to host the Catalog REST API with embedded Swagger UI
- Returning `null` from Bean should work similar to how `setBody` and `transform` works when they set a `null` body.
- The Camel Spring Boot starter components now have their auto configuration depends on `org.apache.camel.springboot.CamelAutoConfiguration` which makes it easier writing unit tests where you can exclude `org.apache.camel.springboot.CamelAutoConfiguration` to turn off Camel Spring Boot auto configuration completely.
- Camel now supports OWASP dependency check maven plugin
- NATS component now supports TLS and explicit flushing (with timeout) of the connection
- Metrics component now supports Gauge type
- File consumer now supports idempotent-changed and idempotent-rename read lock strategies for clustering.
- Camel Catalog now supports custom runtime providers that only includes the supported Camel components, languages and data formats running in that container. For example for Karaf or Spring Boot in the camel-catalog-provider-karaf and camel-catalog-provider-springboot.
- The bean component will when calling a method that returned an instance of `Callable` now call that callable to obtain the chained result. This allows to call Groovy functions/closures etc.
- Failover Load Balancer with inheritErrorHandler=false, now allows Camel's Error Handler to react after the load balancer is exhausted.
- Salesforce component now supports limits, recent items, approvals and composite API
- Dumping Camel routes as XML now includes custom namespaces which are at xpath expressions etc. Likewise updating Camel routes from XML can now include namespaces which will be associated on xpath expressions.
- Added `RouteIdFactory` which can auto assign route ids based on the consumer endpoints to use more sensible names, instead of route1, route2, etc.
- Add `skip` function to Simple language
- Upgraded to Jetty 9.3 (Jetty 9.2 still supported for Karaf 4.0.x users)
- `RouteBuilder` auto-configuration can now be disabled from Camel CDI configuration
- Camel contexts automatic start can now be disabled from Camel CDI configuration
- Camel CDI now provides support for `TransactionErrorHandler` and `TransactionPolicy` via JTA
- Asynchronous support for CXF JAX-RS producers has been added
- The JSonPath language now supports Map and List types and POJOs as well. For POJOs you need to have Jackson on the classpath.
- Data Format which marshal to JSon or XML now set the content-type header to application/json or application/xml appropriately.
- The Kafka component can now store offset state offline (stateRepository) to preserve this information and be able to resume from previous offset.
- The Kafka component has been improved to be easier to configure and use. Notice there is a backwards incompatible change so users need to migrate.
- A topic based idempotent repository that is Kafka based for the Idempotent Consumer EIP
- The Kafka component will automatic type convert the message body to the type specified by the configured serializer (is string by default) when sending to kafka. You can also now configure key and partitionKey in the endpoint uri, instead of having to specify as headers.

- The Kafka consumer will now auto commit on stop to ensure the broker has the latest offset commit. The option `autoCommitOnStop` can be configured to be sync,async or none.
- Added easy predicate parser to JSonPath to more easily define simple predicates without using the more complex jsonpath notation with all the symbols to just say price < 50
- The Box component has been migrated to use the Box v2 Java API as the old v1 API is going to be shutdown from summer 2017
- Examples overview now generate from the source code to ensure its up to date at https://github.com/apache/camel/tree/master/examples#examples
- Added declarative Transformer and Validator which performs transformation/validation according to the data type information declared on a route by inputType and/or outputType. There're a few examples demonstrates this feature: camel-example-transformer-blueprint, camel-example-transformer-cdi, camel-example-transformer-demo, and camel-example-validator-spring-boot
- Added query support for JPA Producer

Fixed these issues

- Fixed starting Camel on Oracle JDK 1.8.0_19 or lower, which would throw an UnsupportedOperationException
- Fixed running `mvn camel:run` when using OSGi Blueprint
- Fixed Hystrix EIP to also execute fallback if execution was rejected or short-circuited or other reasons from Hystrix.
- Fixed Hystrix EIP race condition when timeout was hit and fallback is executed could let to Camel Exchange having wrong caused exception.
- Fixed adding new routes to running CamelContext and if the new routes would fail to startup, then before these routes would "hang around". Now only succesful started routes are added.
- Adding or removing routes that starts from Undertow no longer restart the entire HTTP server
- VM endpoint should prepare exchange with the CamelContext from the consumer and not from cached endpoint which can be different
- Fixed a bug when using Rest DSL with SERVLET could cause a java.io.IOException: Stream closed exception when using Bean in the route.
- Fixed an issue when using `pipeline` in Java DSL not setting up the EIP correctly which could lead to runtime route not as intended.
- Fixed Dropbox to use Stream caching to avoid reading entire file into memory so Camel can process big files
- Fixed `toD` issue with splitting uris when RAW values had + sign
- Fixed adviceWith may behave differently when using multiple advices in the same order and you would advice on the same nodes.
- Fixed camel-zipkin to be able to startup and work with Camel XML
- Fixed FTP2 readLock=changed not working (when fastFileExists=false) if no sub folder was specified as starting directory.
- Fixed Simple language when using indexing with a nested function
- Fixed issue with `@Consume` not having `CamelContext` injected and its lifecycle managed by `CamelContext`
- Fixed Netty double buffer release leak in Netty4 and Netty4 HTTP

New Enterprise Integration Patterns

New Components

- camel-azure - to integrate with Microsoft Azure Cloud services
- camel-bonita - allow you to communicate with a remote Bonita engine.
- camel-couchbase - The couchbase component allows you to treat CouchBase instances as a producer or consumer of messages.
- camel-digitalocean - The digitalocean component allows you to interact with the Digitalocean cloud
- camel-drill - The drill component gives you the ability to querying to Apache Drill Cluster
- camel-elasticsearch5 - to communicate with Elasticsearch Server using 5.x APIs
- camel-google-pubsub - allow you communicate with Google Cloud Pub/Sub
- camel-grpc - The gRPC component is using for calling remote procedures via HTTP/2
- camel-milo - An OPC UA client and server component based on Eclipse Milo
- camel-mongodb3 - Evolution of the component based on v3 driver. Camel-Mongodb still exists allowing compatibility with the v2 driver.
- camel-olingo4 - Communicates with OData 4.0 services using Apache Olingo OData API.
- camel-openstack - allow you to communicate with an Openstack infrastructure
- camel-opentracing - Distributed tracing using OpenTracing
- camel-pubnub - To send and receive messages to PubNub data stream network for connected devices.
- camel-rest-swagger - for accessing REST resources using Swagger specification as configuration
- camel-tika - integrates with Apache Tika to extract content and metadata from thousands of file types
- camel-sjms2 - JMS 2.0 api compatible component of the SJMS component.
- camel-spring-cloud - integrates with Spring Cloud
- camel-spring-cloud-netflix - integrates with Spring Cloud Netflix
- camel-zendesk - interacts with Zendesk server with using Zendesk Java Client

New DSL

New Annotations

New Data Format

New Languages

New Examples

- camel-example-reload - To demonstrate the live reload routes from XML files when running Camel.
- camel-example-rest-swagger - Demonstrates the usage of REST Swagger component as a generic Swagger client
- camel-example-opentracing - An example showing how to trace incoming and outgoing messages from Camel with OpenTracing

New Tutorials

API breaking

- The groovy DSL from camel-groovy has been moved into its own camel-groovy-dsl module. The camel-groovy now only contains the Camel Groovy Language
- Camel-spring-LDAP now uses java.util.function.BiFunction<L, Q, S> instead of org.apache.camel.component.springldap. LdapOperationsFunction<Q, S>
- The deprecated APIs from camel-spring-boot has been removed as part of upgrading and supporting Spring Boot 1.5.x
- The `getComponentDocumentation` method on `CamelContext` is deprecated and returns null. The embedded HTML documentation in all the Camel components has been removed as they are not in use/maintained, and the JSon schema is the actual information. Use the camel-catalog for component documentation where you can get all the documentation in both ascii doc and html format.
- camel-mongodb-gridf schema has been renamed from **gridfs** to **mongodb-gridfs** to avoid confusion.
- The commands-core has the Catalog commands removed
- The org.apache.camel.spring.boot.FatJarRouter has been removed, just use regular `RouteBuilder` classes in Spring Boot applications.
- The Kafka endpoint option `seekToBeginning=true` should be migrated to `seekTo=beginning`
- The Kafka endpoint option bridgeEndpoint has moved from endpoint to the KafkaConfiguration class so all options are together.
- The Kafka component has been improved to be easier to configure and use. Notice there is a backwards incompatible change so users need to migrate. The kafka uri is changed from kafka:brokers to kafka:topic. So you need to specify the topic name in the context-path and the brokers as parameters, eg before `kafka:myserver?topic=sometopic` is now `kafka:sometopic?brokers=myserver`
- The Infinispan uri syntax has changed from infinispan:hostName?options to infinispan:cacheName?options

Known Issues

Important changes to consider when upgrading

- camel-spring-dm has been disabled from the karaf features file so users cannot install it out of the box as it does not work properly. camel-spring-dm has been deprecated for a long time and users are encouraged to use osgi blueprint instead. The JAR is still shipped and can be installed manually but then you are on your own. The JAR will be removed completed in a future release.
- Groovy DSL and Scala DSL is deprecated and planned to be moved to Camel Extra and not distributed out of the box in the future.
- Camel now uses Karaf 4.x API and therefore not possible to run on older Karaf versions.
- `camel-blueprint` changed startup behavior to start on Blueprint.CREATED event which would be more `correct` way of startup instead of Blueprint.REGISTERED as before.
- camel-spring-boot now don't include prototype scoped beans when auto scanning for RouteBuilder instances, which is how camel-spring works. You can turn this back using the includeNonSingletons option.
- camel-spring-javaconfig removed from Karaf features as it was not really supported in OSGi/Karaf.
- camel spring-boot shell commands have been removed as spring-boot shell has been deprecated in spring-boot.
- camel-mongodb-gridf schema has been renamed from **gridfs** to **mongodb-gridfs** to avoid confusion.
- camel-box has been migrated to use box v2 api so there may be some migration needed as the old camel-box component was using box v1 api
- The JSon schema from camel-catalog have changed to use boolean, integer and numeric values when applicable instead of using string values for everything.
- The camel-catalog Karaf commands has been removed

Getting the Distributions

Binary Distributions

Description	Download Link	PGP Signature file of download
Windows Distribution	apache-camel-2.19.0.zip	apache-camel-2.19.0.zip.asc
Unix/Linux/Cygwin Distribution	apache-camel-2.19.0.tar.gz	apache-camel-2.19.0.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 (zip)	apache-camel-2.19.0-src.zip	apache-camel-2.19.0-src.zip.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>2.19.0</version>
</dependency>
```

Git Tag Checkout

```
git clone https://git-wip-us.apache.org/repos/asf/camel.git
cd camel
git checkout camel-2.19.0
```

Changelog

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

- Release notes for 2.19.0