

# Karaf

## Karaf Support

Apache Camel is designed to work nicely inside [Apache Karaf](#) OSGi container.

It includes:

- Camel features descriptor allowing to easily and quickly install Camel in Karaf.
- Karaf commands allowing you to view, start, stop, get info, about the Camel contexts and routes running in the Karaf instance.

It is recommended to use at least Apache Karaf 4.x with current Camel versions.

## Install Camel in Karaf

Assuming that you have a running Karaf instance, you can register the Camel features descriptor:

```
karaf@root> feature:repo-add camel 2.18.4
```

Now, we have all Camel features available:

```
karaf@root> features:list|grep -i camel
[uninstalled] [2.8.0      ] camel                repo-0
[uninstalled] [2.8.0      ] camel-core          repo-0
[uninstalled] [2.8.0      ] camel-spring         repo-0
[uninstalled] [2.8.0      ] camel-blueprint      repo-0
[uninstalled] [2.8.0      ] camel-test           repo-0
[uninstalled] [2.8.0      ] camel-cxf            repo-0
...
```

It is a good start to at least install camel-blueprint

```
karaf@root> features:install camel-blueprint
```

You might need other features depending on what components you use in your routes. So for example if you use `stream:out` then you need the `camel-stream` feature.

```
karaf@root> features:install camel-stream
```

## Karaf commands

The camel commands below help managing your integrations. In many cases the commands support tab completion for context and route names.

Command	Description
<b>camel:context-list</b>	Lists the camel contexts available in the current Karaf instance
<b>camel:context-info</b>	Displays detail information about a given Camel context
<b>camel:context-start</b>	Starts the given Camel context
<b>camel:context-stop</b>	Stops the given Camel context (it becomes unavailable and it can't be started afterwards)
<b>camel:route-list</b>	Displays the list of Camel routes available in the current Karaf instance
<b>camel:route-info</b>	Provides detail information about a Camel route
<b>camel:route-show</b>	Renders the route in XML
<b>camel:route-start</b>	Starts the given route. From Camel <b>2.10</b> onwards you can use * as wildcard to match multiple routes.
<b>camel:route-stop</b>	Stops the given route. From Camel <b>2.10</b> onwards you can use * as wildcard to match multiple routes.

<b>camel:route-suspend</b>	Suspends the given route. From Camel <b>2.10</b> onwards you can use * as wildcard to match multiple routes.
<b>camel:route-resume</b>	Resumes the given route. From Camel <b>2.10</b> onwards you can use * as wildcard to match multiple routes.
<b>camel:endpoint-list</b>	Lists endpoints from all camel contexts available in the current Karaf instance
<b>camel:route-profile</b>	To profile route(s)
<b>camel:route-reset-stats</b>	To reset performance stats on the given route(s)
<b>camel:context-suspend</b>	Suspends the given Camel context
<b>camel:context-resume</b>	Resumes the given Camel context

## camel:route-list

```
karaf@root> camel:route-list
[routel
    ]
```

You can also filter the routes by CamelContext:

```
karaf@root> camel:route-list 66-camel-3
[routel
    ]
```

*Tip: use the TAB key to completion on the CamelContext ID.*

## camel:context-info

Displays detailed information about a given CamelContext:

add the **\*--verbose\*** option (following the context name) to also list the endpoints

```
karaf@root> camel:context-info 66-camel-3
Camel Context 66-camel-3
  Name: 66-camel-3
  Version: 2.8.0
  Status: Started
  Uptime: 1 minute

Advanced
  Auto Startup: true
  Starting Routes: false
  Suspended: false
  Tracing: false

Properties

Components
  timer
  properties
  log

Endpoints
  timer://test
  log://test

Routes
  routel

Used Languages
```

You can see the current Camel version used by the CamelContext, some context attributes, the components involved in the context, and the endpoints defined.

*Tip: use TAB key for completion on the CamelContext name.*

## camel:route-info

The **\*camel:route-info\*** command provides detail information about a Camel route:

```
karaf@root> camel:route-info routel
Camel Route routel
  Camel Context: 66-camel-3

Properties
  id = routel
  parent = 2e7aaccl

Statistics
  Exchanges Total: 98
  Exchanges Completed: 98
  Exchanges Failed: 0
  Min Processing Time: 1ms
  Max Processing Time: 2ms
  Mean Processing Time: 1ms
  Total Processing Time: 134ms
  Last Processing Time: 1ms
  First Exchange Date: 2011-06-29 07:21:57
  Last Exchange Completed Date: 2011-06-29 07:23:34

Definition
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<route id="routel" xmlns="http://camel.apache.org/schema/spring">
  <from uri="timer:test"/>
  <to uri="log:test" id="tol"/>
</route>
```

You can see some statistics (the number of processed exchanges, the processing time, etc) and a XML rendering of your route (whatever DSL used to define the route).

*Tip: use TAB key for completion on the route name.*

## camel:route-show

The **\*camel:route-show\*** command renders the route in XML. It's independent from the DSL used to define the route:

```
karaf@root> camel:route-show routel
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<route id="routel" xmlns="http://camel.apache.org/schema/spring">
  <from uri="timer:test"/>
  <to uri="log:test" id="tol"/>
</route>
```

*Tip: use TAB key for completion on the route name.*

## camel:route-suspend

The **\*camel:route-suspend\*** command suspends a Camel route:

```
karaf@root> camel:route-suspend routel
```

*Tip: use TAB key for completion on the route name.*

## camel:route-resume

The **\*camel:route-resume\*** command resume a Camel route:

```
karaf@root> camel:route-resume routel
```

*Tip: use TAB key for completion on the route name.*

## **camel:endpoint-list**

The **\*camel:endpoint-list\*** command displays the list of the endpoints available in all camel contexts of the current Karaf instance:

```
karaf@root> camel:endpoint-list
camel-id      uri              Status
[test         ] [timer://test   ] [Started      ]
[test         ] [direct://A      ] [Started      ]
```

It displays the context name/ID (used in others commands), the URI of the endpoint and the current status (started/stopped).