
JBoss®
Fuse


Apache
Camel

Apache Camel & Fuse Interview Questions

Q: What is Apache Camel ?

A: In an enterprise, a number of systems of different types exist. Some of these may be legacy systems while some may be new. These systems often interact with each other,and need to be integrated.

- This interaction or integration is not easy as the implementations of the systems, their message formats may differ.
- One way to achieve this is to implement code which bridges these differences. However this will be point to point integration. If tomorrow again if there is change in a system the other might also have to be changed which is not good. Instead of this point to point integration which causes tight coupling we can implement an additional layer to mediate the differences between the systems.
- This results in loose coupling and not affect much our existing systems.
- Apache Camel is a rule-based routing and mediation engine that provides a Java object-based implementation of the Enterprise Integration Patterns using an API (or declarative Java Domain Specific Language) to configure routing and mediation rules.

Q: What are routes in Apache Camel?

A:The core functionality of Apache Camel is its routing engine. It allocates messages based on the related routes. A route contains flow and integration logic. It is implemented using EIPs and a specific DSL.

Q: What is an exchange in Apache camel?

A: The message to be routed in Camel route is present in the Exchange. It is the message holder. Apache camel uses Message Exchange Patterns(MEP). Apache camel exchange can hold any kind of message. It supports a variety of formats like xml, JSON etc.

Q: What are endpoints in apache camel?

A: Camel supports the Message Endpoint pattern using the Endpoint interface. Endpoints are usually created by a Component and Endpoints are usually referred to in the DSL via their URIs.

Q:What is Apache Karaf?

A: Apache Karaf is an OSGi based runtime, it is where our Application bundles run. Fuse uses Apache Karaf as its runtime in which bundles run and collaborate to provide business functionality.

Q) How do you define routes in camel?

A) Routes can written using one of the DSL languages. The popular DSLs used are Java and Spring XML. Below is a snippet of defining the routes in Spring and Java DSL.

Spring:

```
<camel:route>

<camel:from uri="...."></camel:from>

<camel:to uri="..."></camel:to>

</camel:route>
```

Java:

```
public class MyRouteBuilder extends org.apache.camel.builder.RouteBuilder

{

public void configure() {



from("...")

.to("...");

}

}
```

Q) How is data passed between the routes?

A) The data is passed in the routes as an object of the class org.apache.camel.Exchange.

Q) What is a producer and a consumer endpoint?

A) A camel route is similar to a channel though which data flows. Producer and consumer are two endpoints that are present at each end of the channel.
A consumer endpoint is the starting point of the route. A definition of a camel route starts by writing a camel consumer endpoint.
A producer endpoint appears (not always) at the end of the route. It consumes the data that is passed through the route

Q) How do I debug my route?

If you've created a route and its not doing what you think it is you could try using one of these features from version 1.4 onwards

•Tracer to trace in commons-logging / log4j each step that Camel takes

•Debugger to let you set breakpoints at points in the route and examine historic message exchanges

•Debug from your unit test if you use the Camel camel-test component