**The mechanism to derfine and execute the Route DSL and execute in Camel**

Define Java class extending RouteBuilder,override configure method and define java DSL Route definitions inside this method,Instantioate the Camel-Spring Main class to load this derfualt Java DSL by adding RouteBuider via addRoute method and run.

Anotrher way to load route definitions from the default xml on default path and executes it

\* The default Camel sets the \*single\* configuration to

private String applicationContextUri = "META-INF/spring/\*.xml";

To load multipolke XML Route DSL files :

Main.setApplicationContextUri("META-INF/spring/camel-context-1.xml;META-INF/spring/camel-context-2.xml");

To load multiple camel contexts from spring xml definitions

AbstractApplicationContext context = new ClassPathXmlApplicationContext("classpath:META-INF/spring/camel-context-1.xml", "classpath:META-INF/spring/camel-context-2.xml");

and then wire it using:

Main.setApplicationContext(context);

Another possible approach would be to use spring import feature , that's import camel-context-2.xml into camel-context-1.xml,

and then just try to load camel-context-1.xml using Main.setApplicationContextUri().

Working with Maven projects

Maven default-project-directory-layout the directory "src/main/resources" is \*already\* on the classpath\*,

so that you can omit that prefix while providing the URIs.