**Spring Integration**

**Step 1:** Generate Classes from XSD’s.

1. Create JAX-B binding xml for each message.





1. Create xjc entry in builcallbacks.xml



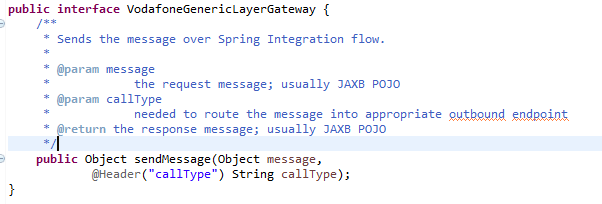
1. Do “ant all” it will create java classes.

**Step 2:** Create DTO’s for request and response for local coding purpose.

**Step 3:**  Create an **extn-genericlayaer-spring.xml**



1. Create a gateway interface, which is needed to interact with other systems.



1. Create a gate way request and response channel to interact with other systems.

<int:channel id=*"vodafoneGenericLayerGatewayReplyChannel"* />

<int:channel id=*"vodafoneGenericLayerGatewayChannel"* />

<int:gateway

id=*"vodafoneGenericLayerGateway"*

service-interface=*"packageName.VodafoneGenericLayerGateway"*

default-request-channel=*"vodafoneGenericLayerGatewayChannel"*

default-reply-channel=*"vodafoneGenericLayerGatewayReplyChannel"*

/>

1. Create Unique code for each service in vodafonecore-items.xml

<enumtype code=*"InterfaceCallTypeEnum"* generate=*"true"* dynamic=*"false"*>

<description>Collects all interface types. It's static as on the code level we need to know about all interfaces during the development (to create all mappings, transformations, etc.).</description>

<value code=*"IF0201"* />

</enumtype>

1. Create header-value-router to routes the message into appropriate outbound endpoint by it's value from InterfaceCallTypeEnum.

NOTE: the same behavior could be achieved by using int:payload-type-router, however then we would need to provide the full Java package and class name which might change over a time when InterfaceCallTypeEnum should be mapped once.

<int:header-value-router

header-name=*"callType"*

input-channel=*"vodafoneGenericLayerGatewayChannel"*

resolution-required=*"true"*>

<int:mapping

value=*"IF0201"*

channel=*"vodafoneGenericLayerCreditVettingChannel"* />

</int:header-value-router>

1. There are 3 steps that are important to create a specific logger:

1. namespaceUri: this namespace must be present in the incoming message

2. class="hu.vodafone.genericintegration.genericlayer.logging.VodafoneClientInterceptor

3. Configure the log4j file in the project.properties in extension vodafonegenericintegration for the loggerName hu.vodafone.genericintegration.messagetracing.CreditVettingInboundLoggingInterceptor.

The reason for this configuration is that the requirement for every module which needs to be logged, must be logged into a separate file. This file is configured in the project.properties.

The **<sws:payloadRoot>** is used because of the option to specify which messages are logged based on the namespace, which can be directed to the specific log-files

<sws:interceptors>

<sws:payloadRoot namespaceUri=*"http://xmlns.vodafone.hu/BusinessMessage/VettingInterface/VettingService/V1.1"*>

<bean class=*"hu.vodafone.genericintegration.genericlayer.logging.VodafoneClientInterceptor"*>

<property name=*"loggerName"* value=*"hu.vodafone.genericintegration.messagetracing.CreditVettingInboundLoggingInterceptor"*/>

</bean>

</sws:payloadRoot>

**NOTE:** Here what is value?

**NOTE:** Name space URI can be get it from WSDL and the client interceptor is

****

1. If External services has basic authentication then we have to send username password while calling the url.





1. We have to create on xml for each and every service (one WSDL on xml file). One WSDL can have multiple operations (sub services)



1. We have to make the above file entry in **extn-genericlayaer-spring.xml**

<import resource=*"classpath:/vodafonegenericintegration/spring/vodafonegenericintegration-genericlayer-creditvetting-spring.xml"* />

1. We have to create request and response converters.
2. Map the converts in extn-services-spring.xml



1. Import above 2 spring files in main extn-spring.xml



1. Configure service id and mock response in project.properties



vodafonegenericintegration.header.serviceid.IF0197=http://xmlns.vodafone.hu/BusinessService/VettingInterface/VettingService/doCreditVettingV1/V1

vodafonegenericintegration.creditvetting.uri=${integration.environment.host}:${integration.environment.port}/vodafonemocks/soap

1. Create response.xml in abc-mocks extention and make an entry in mocks-project.properties

****