Mule 4 - Mule Server Notifications

Praneeth Venishetty

Context

Performance - A Non Functional Requirement

- Key factor for efficient and optimal API implementation
- NFR Testing Phase
- Defining Response Time SLAs for an API
- Response Time :
    - Base metric to define performance

MuleSoft

Mule Server Notifications

Mechanism of accessing the mule server to get notified with access changes that occur on Mule Server. These Notifications are disabled by default.

- Message Processor Notifications:
  - A notification is sent before and after a message processor is invoked

Use Case

As a developer I want to debug across all the Mule flows in my application and log the response times of each and every message processor.

Functional Scenario:

- Response time of an API largely depend on external API invocations and Message Processors present in Mule Flows.
- External APIS' response time cannot be controlled by a developer and so for any performance improvements to be done on the API, optimisations had to be around message processors.

Implementation Steps

- Create a Mule Project
- Create a Class - *MuleComponentProcessListener* in src/main/java folder.
    - This class implements MessageProcessorNotificationListener Interface where we will be using onNotification() method to log and notify.

- Create beans.xml file in src/main/resources folder
- Create a bean as below in beans.xml specifying the class created in step 2 for the type of notification we want to receive.

```
<bean name="messageProcessorNotifications"
                class="message_processor_notifications.MuleComponentProcessListener" />
```

- Add beans.xml reference to Spring Configuration as below

```
<spring:config name="springConfig"
        doc:name="Spring Config" files="beans.xml" />
```

MuleSoft

Implementation Steps

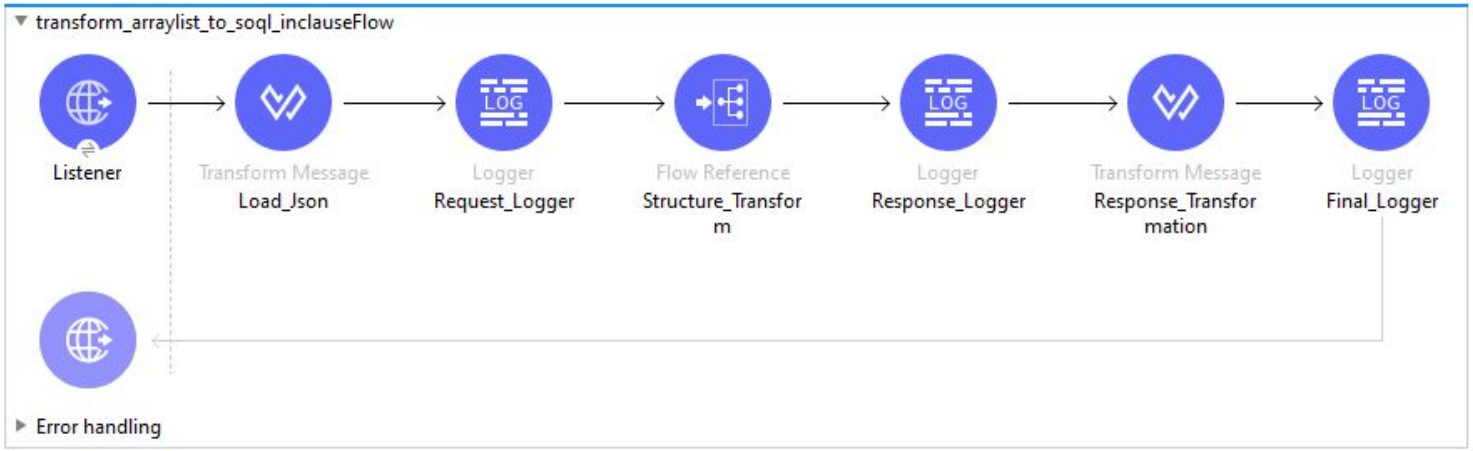
- Then we have to specify notifications we want to receive using <notification> element and have to register the listener class bean using <notification-listener> as below:
    - *Note: Listener Class is the Spring bean which we created in beans.xml at step 4*

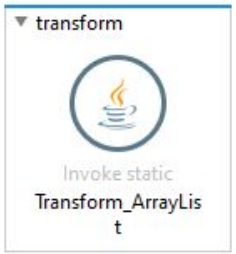
```
<notifications>
        <notification event="MESSAGE-PROCESSOR" />
        <notification-listener
                ref="messageProcessorNotifications" />
</notifications>
```

- Update pom.xml with required dependencies and shared libraries respective to mule-spring modules.
    - *Please refer to the pom file in the codebase in GitHub link shared at the end.*
- Now build the mule flow with required message processors.

MuleSoft

Mule flow





MuleSoft

Working

- In the above mule flow, as soon as the request is received by the Http Listener, as in when the message is passed through the message processors like
    - Transform Message
    - Logger
    - Invoke-Static
    - Flow-Reference

- Mule Server Notification class log the information which is set in MessageProcessorNotificationListener.onNotification() method

Deploy and Run

- Deploy the Mule Application
- Invoke the Http Listener
- We will then see the information related to each message processor in the runtime console.
- Format of the message processor information logged:

    flow = <Flow Name>

    module in flow = <Module # in the flow>

    module name = <Name of the Module>

    line in file = <Line Number in XML>

    component took: <Response Time in Milli Seconds>

MuleSoft

Console Output

```
WARN  2020-09-22 18:59:30,734 [[MuleRuntime].uber.02: [message_processor_notifications].transform_arraylist_to_soql_inclauseFlow.BLOCKING @4df57dc5] [process
message_processor_notifications.MuleComponentProcessListener:
flow = transform
kind processors
module in flow = 0
module name = java:invoke-static
line in file = Optional[79]
component took: 71ms response time.
WARN  2020-09-22 18:59:30,734 [[MuleRuntime].uber.02: [message_processor_notifications].transform_arraylist_to_soql_inclauseFlow.BLOCKING @4df57dc5] [process
message_processor_notifications.MuleComponentProcessListener:
flow = transform_arraylist_to_soql_inclauseFlow
kind processors
module in flow = 2
module name = flow-ref
line in file = Optional[56]
component took: 71ms response time.
INFO  2020-09-22 18:59:30,735 [[MuleRuntime].uber.02: [message_processor_notifications].transform_arraylist_to_soql_inclauseFlow.BLOCKING @4df57dc5] [process
org.mule.runtime.core.internal.processor.LoggerMessageProcessor: ('0012w00000M8mUSAAZ','0012w00000M8mUKAAU','0012w00000M8mUAEEU')
WARN  2020-09-22 18:59:30,735 [[MuleRuntime].uber.02: [message_processor_notifications].transform_arraylist_to_soql_inclauseFlow.BLOCKING @4df57dc5] [process
message_processor_notifications.MuleComponentProcessListener:
flow = transform_arraylist_to_soql_inclauseFlow
kind processors
module in flow = 3
module name = logger
line in file = Optional[58]
component took: 1ms response time.
WARN  2020-09-22 18:59:30,758 [[MuleRuntime].uber.01: [message_processor_notifications].transform_arraylist_to_soql_inclauseFlow.CPU_INTENSIVE @4c5c5d38] [pr
message_processor_notifications.MuleComponentProcessListener:
flow = transform_arraylist_to_soql_inclauseFlow
kind processors
module in flow = 4
module name = ee:transform
line in file = Optional[60]
component took: 23ms response time.
```

MuleSoft

Additional Information

Codebase Location :

https://github.com/praneethveni/mule-artifacts/tree/master/message_processor_notifications

MuleSoft