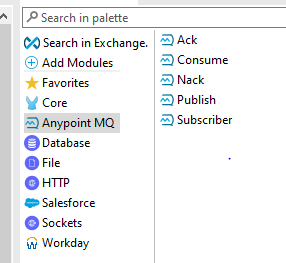
**AnyPointMq-Entreprise-Version-Usage-Document**

By Default in trial Accounts we don’t have this Anypoint-Mq message Queing Facility in the Platfrom-Account.

The Organisation Account who have the Platinum Liscence may have this service.



We don’t see the components by default, we need to import the AnypointMq Module from the Exchange to our Mule-Pallete.

The “publish” component is used to publish the message to the queue.

The Mandatory Configuration for the AnyPointMq connector is

\*The queue Destination URL:

\*The Client ID:

\*The Client Secret:

Once these three are configured, Now we can be able to connect with the Destination queue

For publishing the message.

\*The Mandatory Connector Configuration for all the Mq-operations :

\*The queue Destination URL:

\*The Client ID:

\*The Client Secret:

The “Subscribe” component is used to consume the Published message from the queue, and this connector is the Source, which initiaites the MuleEvent but it is not a Processor.

There are three Acknowledgement modes for this Component:

1. Automatic
2. Immediate
3. Manual

1.Automatic: If this mode was selected while consuming the message once the message is passed through All the Processors in the flow and gets consumed the message in the queue will gets deleted. Else if one of the processor gets failed during the message consumption the message will be sent back again to the queue. This is how the Automatic mode works.

2.Immediate: In this Mode, once the message gets consumed by at least one processor in the flow then the message gets Deleted from the Queue even the rest of the processors in the respective flow gets Failure. Which means the immediate mode wont bother wether the message was successfully consumed or not by the rest of the processors in the flow.

3.Manualmode: If the mode was selected to Manual until the message was acknowledged manually by the “ACK” component the message will stay in the queue like that itself. Even the message gets failure also the message will stay in the queue like that itself. If we make use of “NACK” component then the message won’t Acknowledge and then it will be sent back to queue. In this case even if the message gets consumed successfully by all the processors In the flow and if we use “NACK” component then it will get Un-Acknowledged and message will be sent back to the queue again.

The “Consume” component is a processor component and it can be used in the processor place alone but not in the Source connector place. We have the same Acknowledgement modes, as mentioned above. Wherever in the flow we can place the Consume component and listen to the destination queue for the data Consumption.