Mule Aggregator

- Ihab.yousif@gmail.com

Mule Aggregator Types

1 - Collection-aggregator-router

- Converts the aggregated message parts to collection (java.util.List) in the returned payload.
- <collection-aggregator-router timeout="1000000"
- failOnTimeout="false" />

Mule Aggregator Types

- 2- message-chunking-aggregator-router
- Converts the aggregated message parts to byte array in the returned payload.
- <message-chunking-aggregator-router timeout="1000000" failOnTimeout="false"/>

Mule Aggregator Types

- 3- custom-correlation-aggregator-router
- Give you the control to handle the return formate for the aggregated parts.
- <custom-correlation-aggregator-router class="com.test.tcp.CustomAggregator"></custom-correlation-aggregator-router>

Aggregator required parameters

- The aggregator require three parameters to be exists in the mule message to work

  1- MULE_CORRELATION_SEQUENCE

- To re-organize the parts

  2- MULE_CORRELATION_GROUP_SIZE

- The number of message parts

  3- MULE_CORRELATION_ID

- The shared id between all the related parts

Aggregator required parameters

- The issue during transfere the message on TCP endpoint is that only the payload move through the TCP connector, then the aggregator fail to re collect the parts together.
- The message parts generator must include the required attributes, String type used throgh the TCP communication so the only way is to put convention for including the required parameters with the message part, then extract back after the TCP inbound endpoint

Aggregator required parameters

- Sample for TCP message format
- 1_4_6c9d5dc3-147f-11e1-9cd3-75f5162d916c_ItemOne
- Which means
- MULE_CORRELATION_SEQUENCE=1, MULE_CORRELATION_GROUP_SIZE=4, MULE_CORRELATION_ID=6c9d5dc3-147f-11e1-9cd3-75f5162d916c, payload=ItemOne

Sample Code

- The sample code divided into three parts, every part with separated mule configuration and classes for simplicity, the three parts cover the three aggregation types over socket endpoint.
-

Questions?