**Custom Monitoring for Mule Runtime.**

Monitoring mule runtimes without installing the Anypoint monitoring agent. This is helpful when there are large number of mule runtime targets to be monitored.

**Background**

In cases where you need to monitor large number of mule runtimes and you do not want to or cannot install the Anypoint Monitoring agent on each runtime. This demo was created monitor mule runtime running across more than 300 locations for a big grocery store chain. They want to run mule in each location, and would like to get the basic monitoring info from each mule. They wanted to know if the mule runtime is running or not in each location, then time it came online and time it went down or become unavailable.

**Method**

There are two parts to this monitoring application.

**Server**

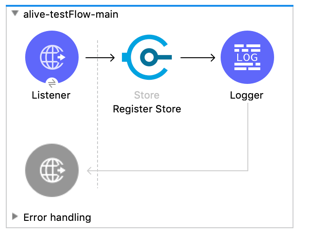
A mule application running on cloud hub which acts as a server, and it is listing on HTTP port. The server gets messages from the clients and it logs the message with the time stamp when the message was received. There is an additional scheduled process running on the server, which reads the object store, find all the entries in object store and compare it against the time stamp value in object store to the current time stamp. If the difference is more than 30 sec then it marks that the store is down. If it keeps receiving the messages every 30 second then it would indicate that the store is up and actively sending alive message to the server.

**Client**

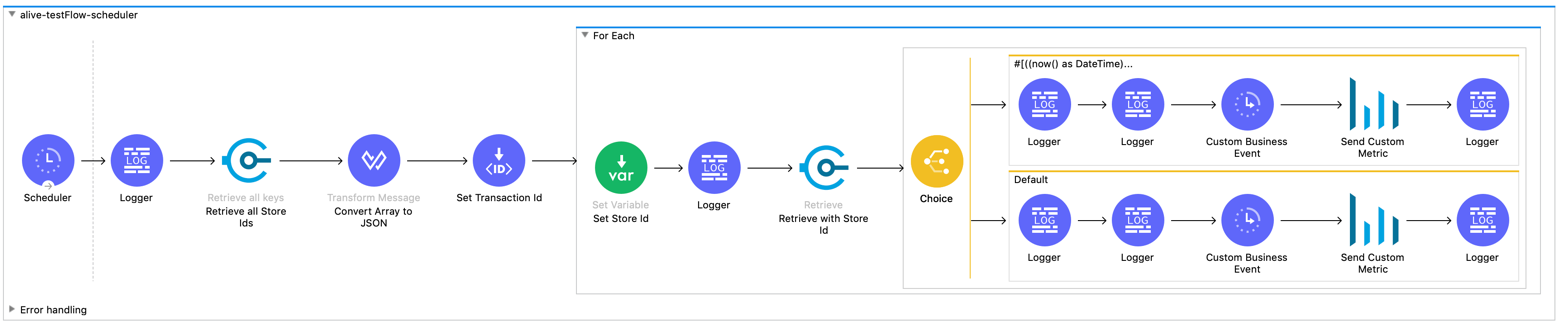
Each mule running at the store acts as a client. There is a small mule application running on each store, which parodically sends message (every 30 sec) to the server along with the store identifier.

**Server Flow**

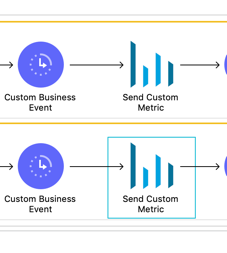
**Listener Flow:** This is a HTTP listener, which receives messages from the client and add the messages in object store using the client identifier as key.



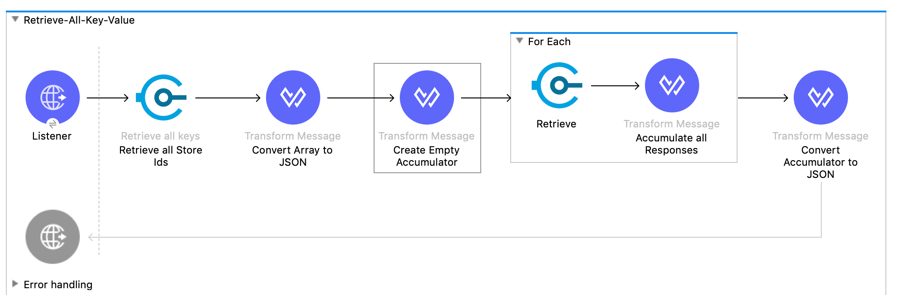
**Scheduled Flow:** This is a scheduled flow which runs every 30 seconds and reads the object store entries. It then checks the time stamp for each entry, if the difference between last entry and current time stamp is > 30 seconds then it marks the server as down.

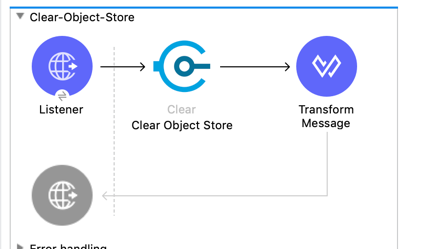


This flow is capturing customer Business Events to generate insights and is also sending customer metrics to Anypoint monitoring for creating custom dashboard.



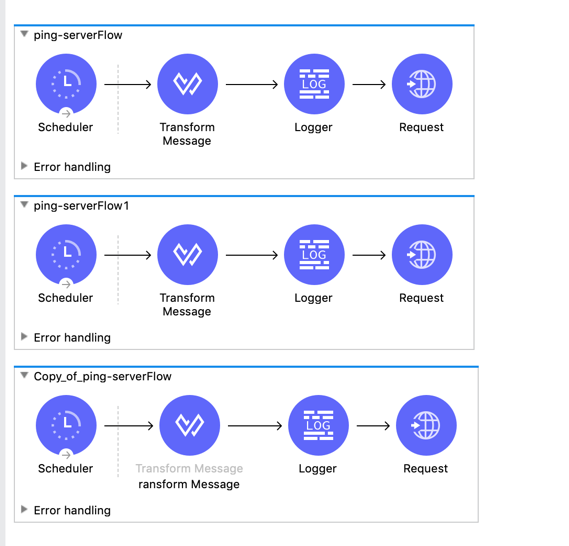
In additional to above flows there are additional flows for retrieving all key-values and cleaning and the object store.





**Client Side App**

This app is to simulates 3 clients sending message to the server every 30 seconds

›

**Custom Monitoring Dashboard**



**Configuration for Custom Dashboard**

