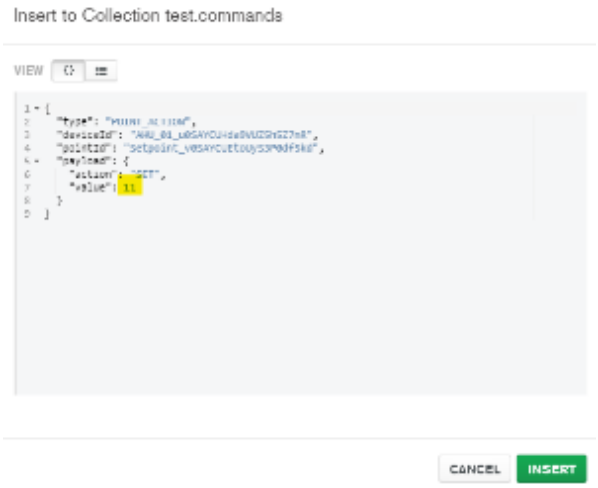
Step 5 Send commands from MongoDb to Niagara

MongoDb Connector supports broker to niagara communication via mqtt messages to control points remotely.

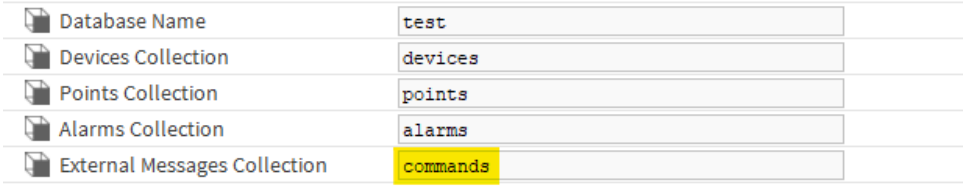
> You can only send commands from a cluster which is replica set, otherwise the application director will display error messages similar to this one
> com.mongodb.MongoCommandException: Command failed with error 40573 (Location40573): 'The $changeStream stage is only supported on replica sets' on server localhost:27017. The full response is {"ok": 0.0, "errmsg": "The $changeStream stage is only supported on replica sets", "code": 40573, "codeName": "Location40573"}

Send a point action command

1. Go to the Mongo Compass application then add a new document to the commands collection.



You can change the command collection name in the connector configuration.



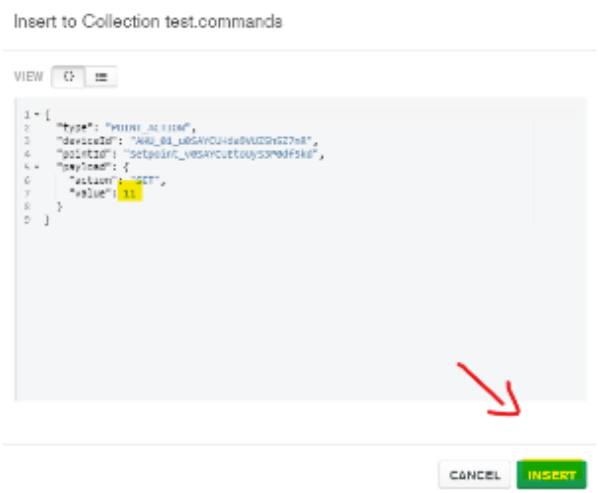
2. By default we use this message template for **POINT_ACTION** command. by you can use any format that meat your needs. check the connector advanced settings.
    a. This is the default command template.

```
{
        "type": "POINT_ACTION",
        "deviceId": "AHU_01_u0SAYCUHda9VUZ5h5Z7nR",
        "pointId": "Setpoint_v0SAYCUEtoUyS3P0df5kd",
        "payload": {
                "action": "SET",
                "value": 11
        }
}
```

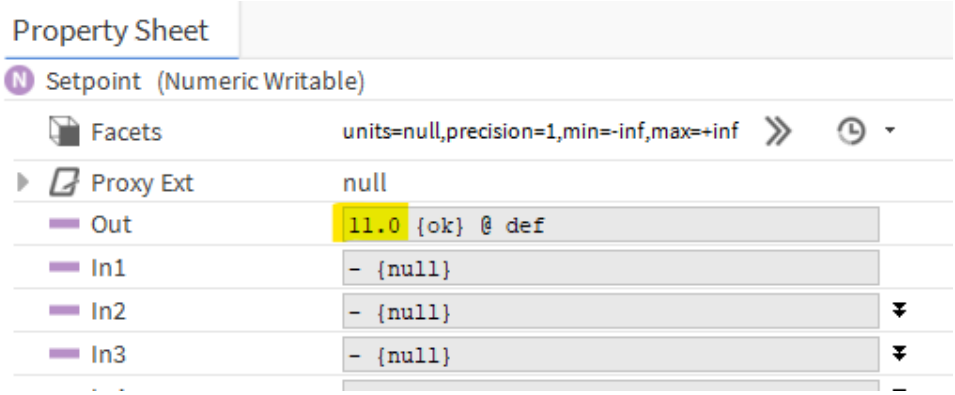
    b. This is the default connector point action command configuration.

Commands Policy	Single Point Command
Message Type	{json('type')}
Command Set Object	{json('').escape}
Command Device Id	{json('deviceId')}
Command Point Id	{json('pointId')}
Command Action	{json('payload.action')}
Command Value	{json('payload.value')}
Command Duration	{json('payload.duration')}

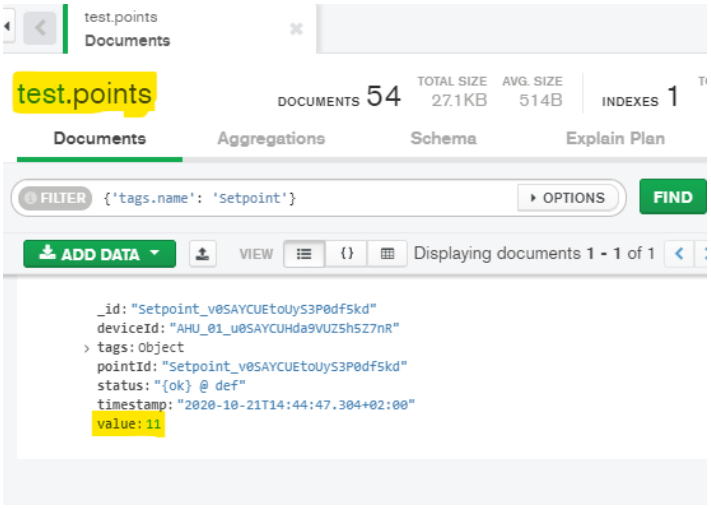
3. Hit Insert.



4. On the niagara you should see the point value applied successfully.



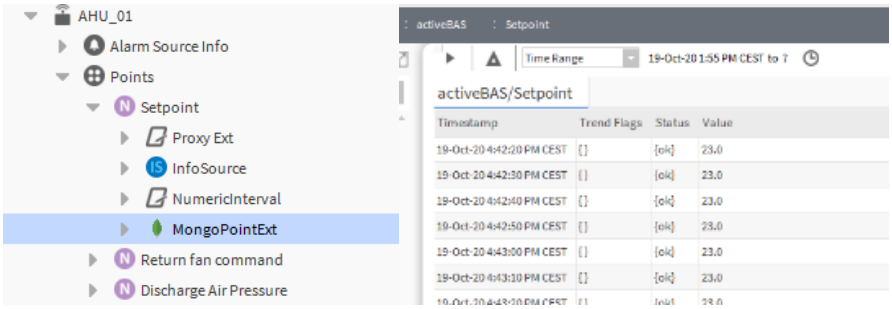
5. And the new value sent to the broker.



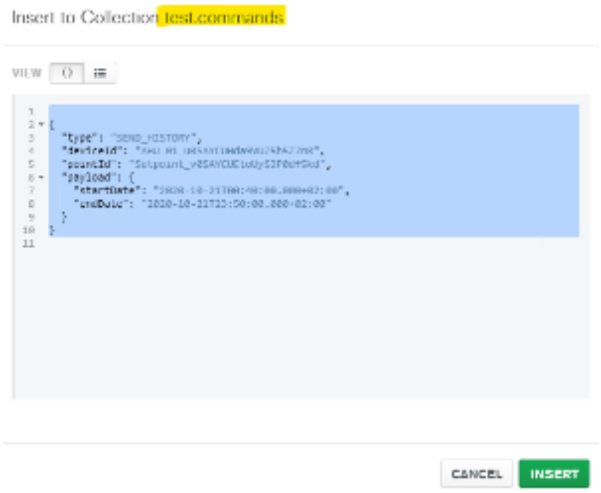
Send a send history command

You can pull historical value for any point that has a history associated by sending a **SEND_HISTORY** command.

1. Add a history. extension to the point.



2. Go to the mqtt application and send the command to the device.

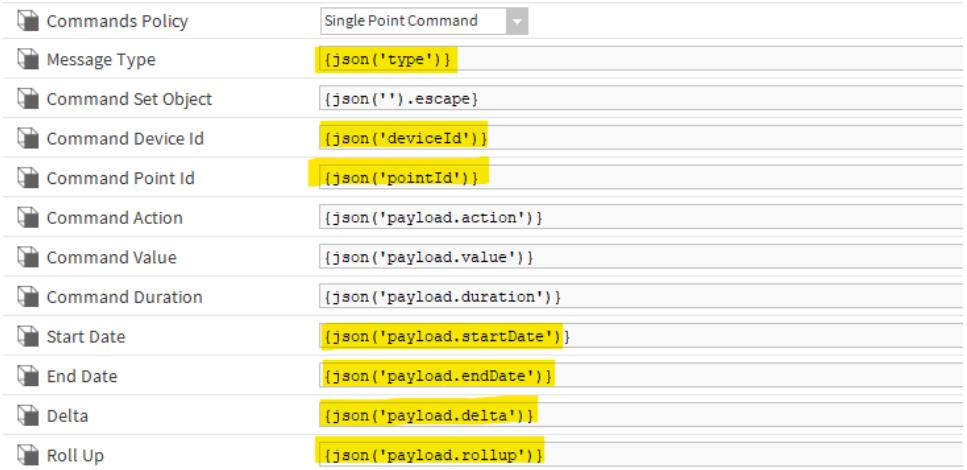


3. By default we use this message template for send history command. by you can use any format that meat your needs. check the connector advanced settings.
    a. This is the default command template.

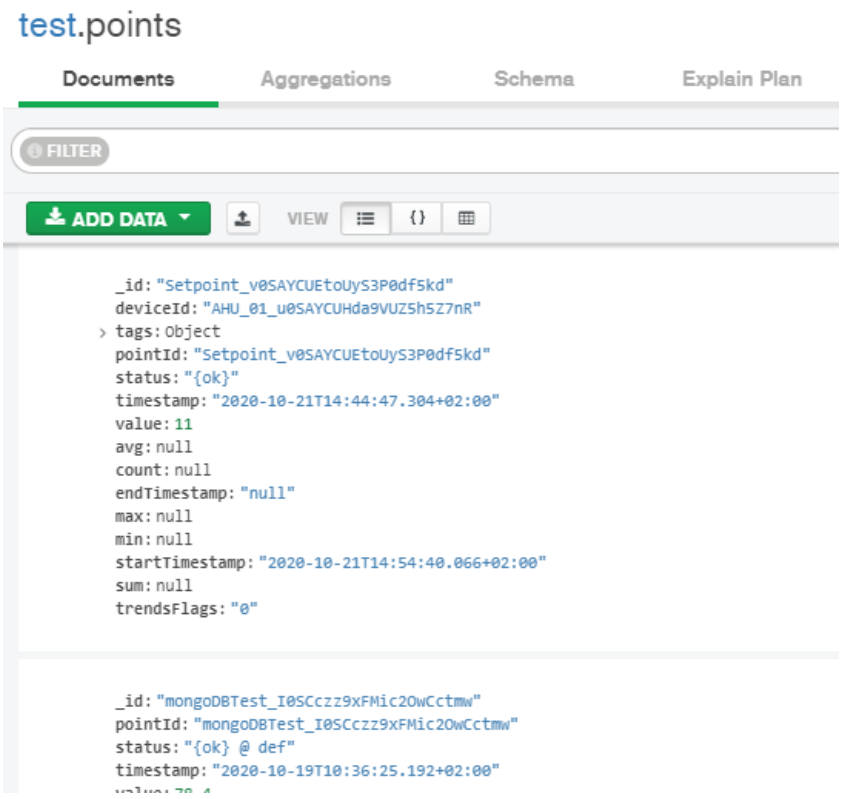
```
{
        "type": "SEND_HISTORY",
        "deviceId": "AHU_01_u0SAYCUHda9VUZ5h5Z7nR",
```

```
        "pointId": "Setpoint_v0SAYCUEtoUyS3P0df5kd",
        "payload": {
                "startDate": "2020-10-21T00:40:00.000+02:00",
                "endDate": "2020-10-21T23:50:00.000+02:00"
        }
}
```

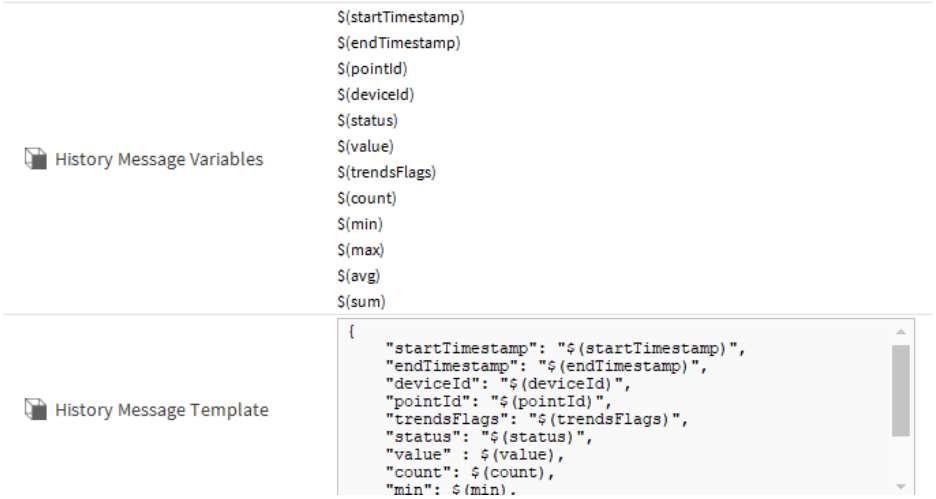
b. This is the default connector send history command configuration.



4. And you should see the messages being sent.



To change the message format check the connector advanced setting then history message template



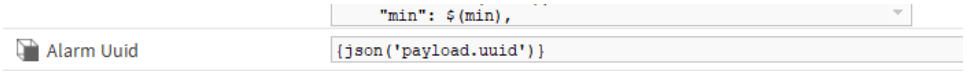
Send ack alarm command

You can ack alarms by sending an **ACK_ALARM** command to any alarm recipient device.

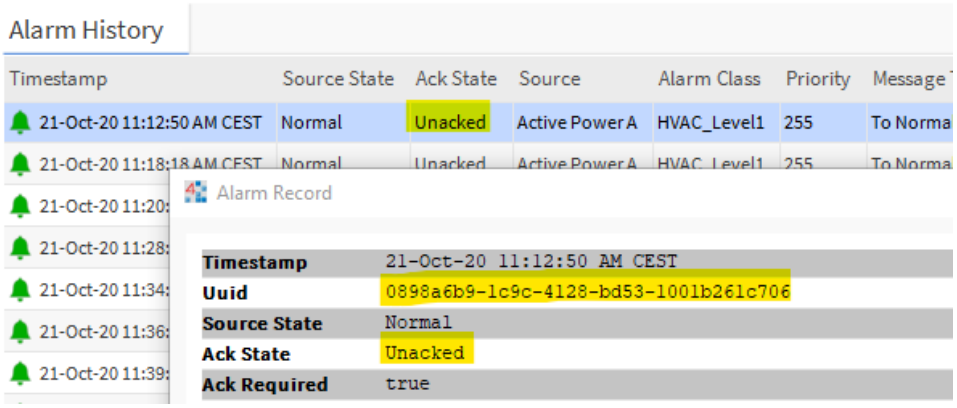
1. By default we use this message template for ack alarm command. by you can use any format that meat your needs. check the connector advanced settings.
    a. This is the default command template.

```
{
        "type": "ACK_ALARM",
        "deviceId": "alarms"
        "payload": {
                "uuid": "0898a6b9-1c9c-4128-bd53-1001b261c706"
        }
}
```

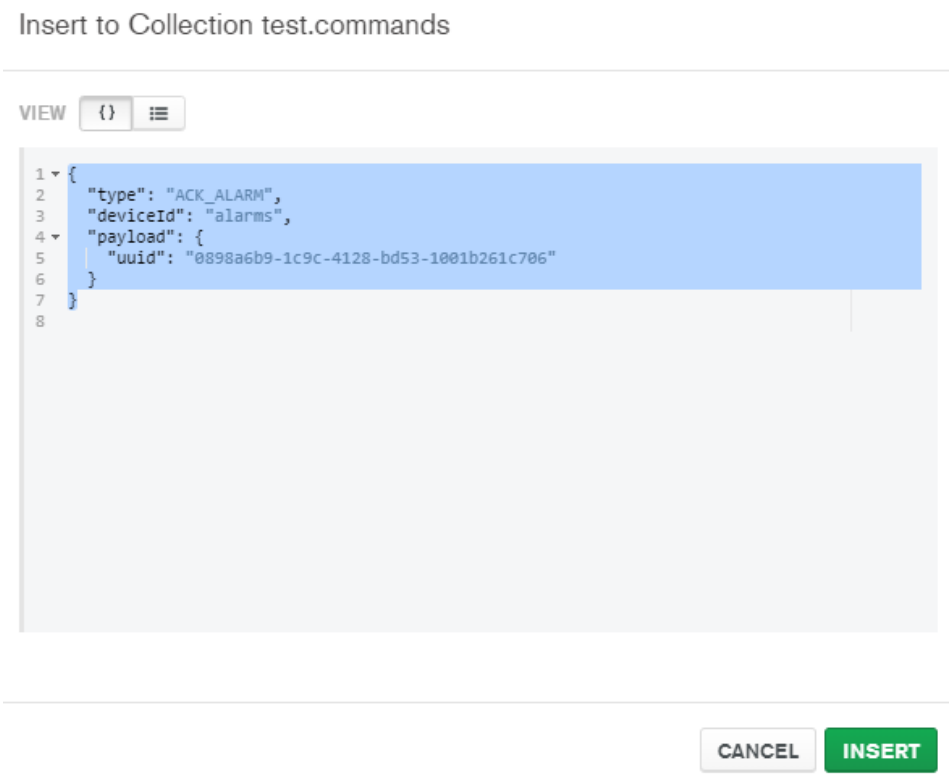
    b. This is the default connector send history command configuration.



2. Go the alarms console and pick an unacked alarm id.



3. Go to the alarm device and send the command.



4. Go back to the console and you should see that the alarm has been acked.