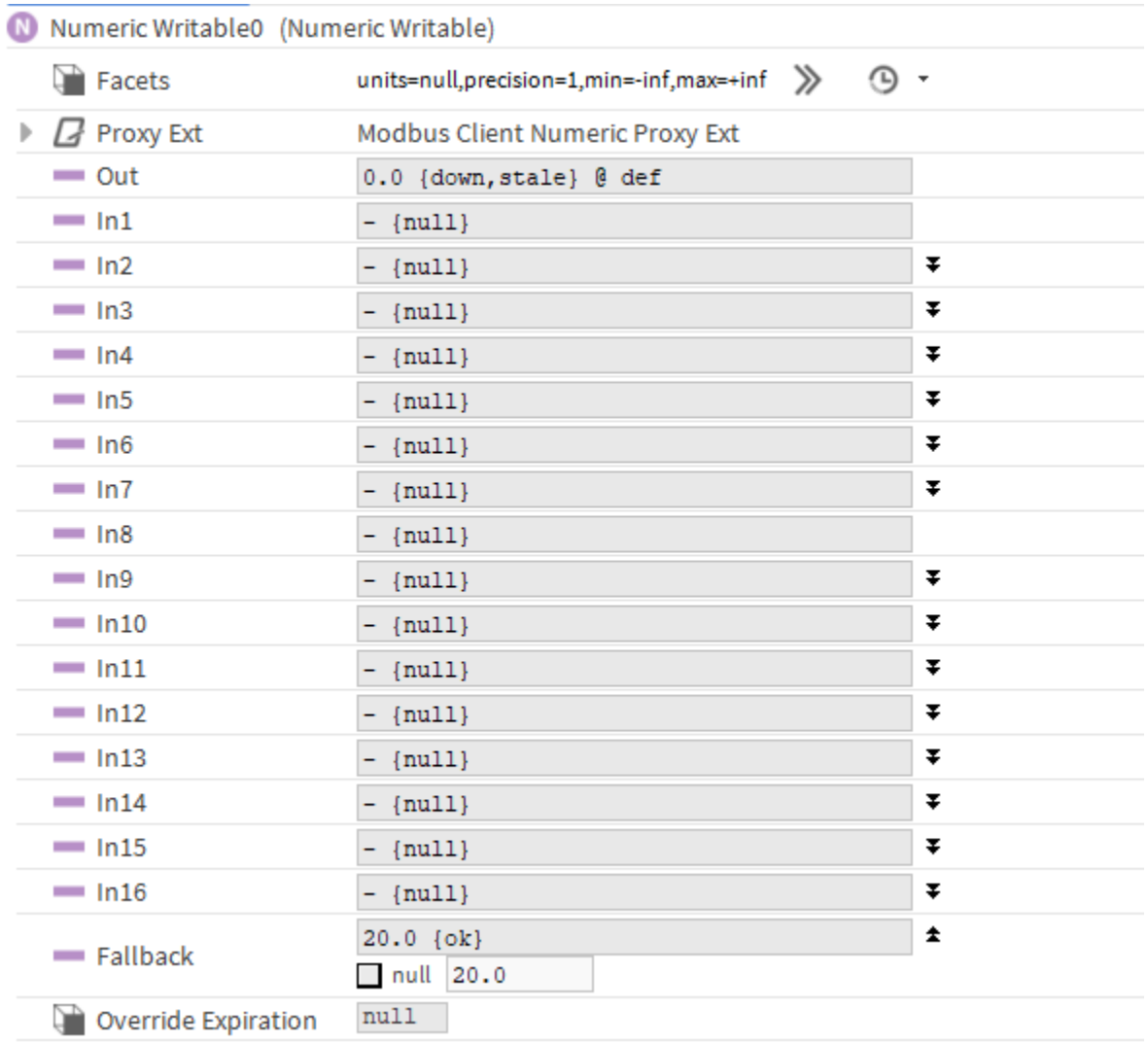
Inbound commands

The same extension (DeviceExt and PointExt) can be used to receive commands from a third party.

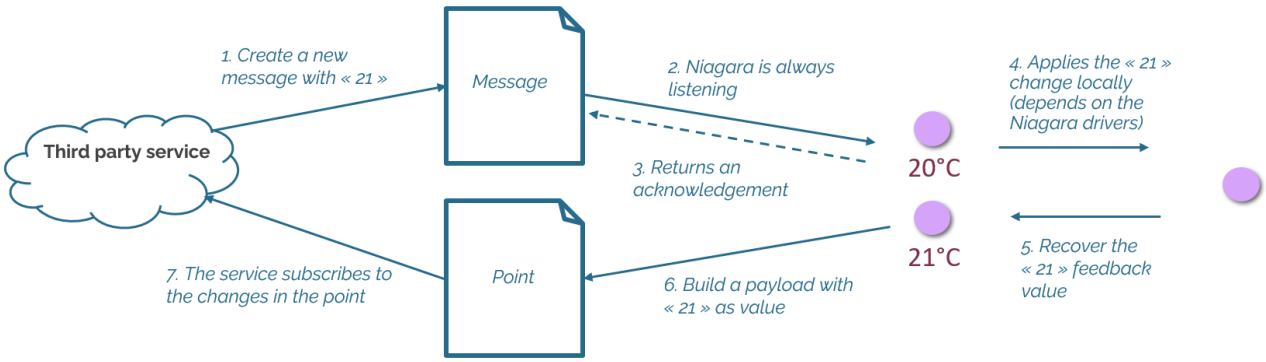
The goal is to modify the value of a point using Niagara actions (set, override...). These commands are not applied directly to the document containing the value or the topic representing the point by modifying the "value" field. Indeed a command in Niagara is applied through an action that Niagara will try to push on the associated network and then read value in return to update the out slot.

In the example below, Niagara is trying to push the value through the proxy extension to the Modbus network. The out slot value will be updated once the drivers read the value of the point from the device



Inbound commands with connectors work the same. A "demand" for change is sent by the third party, an action is called (set, override..) and once the out value is updated, the extension will push back the updated value to the third party, so the third party can be certain the value was pushed on the field. Each connector implements its own way of creating this change request message but the concept is common.

The acknowledgment mechanism indicates that the message has been received in Niagara but does not confirm whether the change has been correctly applied or not.



Several options:

- Commands payload can be personalized. See Outbound Data
- Multiple commands can be sent from a single message
- It is possible to restraint a point to accept a command (an option in the extension)
- Use the vanilla Niagara actions to keep a native behavior

Payload default format

The control format of a point must follow a default template in which it is necessary to indicate at least :

- The id of its associated device (the container)
- The id of the point,
- The type of request: POINT_ACTION in the case of an command
- The value to be modified in a payload object
- The action determines the writing level. The SET action corresponds to level 17 (the lowest), the "OVERRIDE" action to level 8 and the "EMERGENCY_OVERRIDE" action to level 1. The AUTO and EMERGENCY AUTO actions correspond to a return to a lower level. The SET action is recommended.

```
{
   "deviceId":"< DeviceId>"
   "pointId": "< PointId>",
   "type": "POINT_ACTION",
   "payload":{
      "action": "< PointAction>", // SET, AUTO, OVERRIDE ....
      "value": <Value>,
      "duration": <duration in seconds>.
   }
}
```

The format of the message can be customized under certain conditions

Batch commands

MQTT-based connectors allow multiple commands to be sent in a single message. This allows, for example, to turn on the light in several rooms at the same time. This requires a special command sending format in which the point id is specified.

```
"points": {
   "$(pointId)": {
     "present_value": $(pointValue)
   }
```