

Trav Feller Lab 7

- Can a sensor be in more than one collection based on the code you wrote? Why or why not? What are the implications of a sensor being in more than one collection?
 - Yes. In my code, I check all of my subscriptions and send an event to each subscription. Because I cannot have multiple subscriptions to the same pico, each sensor pico will be called exactly once for each correlationId and therefore added to the reports for that correlationId. If a sensor is in more than one collection, the
- How could you ensure that only certain picos can raise an event that causes a temperature report to be generated?
 - In my rule that notified each subscription picos, I first validated that only subscription picos whose role was “sensor” had events sent to them. This way, no other roles are notified and only temperature reports from sensor picos are added to the reports.
- How do the debug logs show that your scatter-gather system worked?
 - In the image below, you can see the logs from my sensor pico (first) and then the sensor manager pico (second). In the first image you can see that the sensor pico selects the rule to generate a report and send it up to the manager pico. In the second image you can see that the temp:get_temps event is called and loops 3 times, one for each of my sensor picos that are subscribed with the sensor role. Above that, you can see that the “sensor:cur_temp” event is raised 3 separate times, one for each time that the sensor picos returned their respective reports.

```
2022-03-17T01:09:37.698Z - EVENT cl0u9vwx901zeuixxcpyqeu77 sensor:get_temp {"id":7}
01:09:37.698Z [DEBUG] txnQueued EVENT cl0u9vwx901zeuixxcpyqeu77 sensor:get_temp {"id":7}
01:09:37.700Z [DEBUG] txnStart {}
01:09:37.700Z [DEBUG] event added to schedule {"eci":"cl0u9vwx901zeuixxcpyqeu77","domain":"sensor","name":"get_temp","data":{"attrs":{"id":7}},"
01:09:37.702Z [DEBUG] rule selected sensor_profile : temp_getter
01:09:37.732Z [DEBUG] fired
01:09:37.732Z [DEBUG] txnDone {}

2022-03-17T01:09:37.732Z - EVENT cl0u9vwx601zcuixx5tr56vp4 sensor:cur_temp
{"id":7,"curTemp":74.6,"sensorName":11,"rx":"cl0u9vwx301z7uixx775v8kg2"}
2022-03-17T01:09:37.725Z - EVENT cl0u9vy3m020muixx9h9o57vm sensor:cur_temp
{"id":7,"curTemp":73.5,"sensorName":13,"rx":"cl0u9vy2m020huixx09pwfezm"}
2022-03-17T01:09:37.713Z - EVENT cl0u9vxp01zzuixx4qer11xa sensor:cur_temp
{"id":7,"curTemp":79.7,"sensorName":12,"rx":"cl0u9vxjl01zuuixx8xrve8it"}
2022-03-17T01:09:37.680Z - EVENT cl0r5h93100q788t678z56bg3 temp:get_temps {}
01:09:37.680Z [DEBUG] txnQueued EVENT cl0r5h93100q788t678z56bg3 temp:get_temps {}
01:09:37.682Z [DEBUG] txnStart {}
01:09:37.682Z [DEBUG] event added to schedule {"eci":"cl0r5h93100q788t678z56bg3","domain":"temp","name":"get_temps","data":{"attrs":{"_headers":
01:09:37.690Z [DEBUG] rule selected sensor_manager : get_temps
01:09:37.692Z [DEBUG] fired
01:09:37.695Z [DEBUG] event added to schedule {"eci":"cl0r5h93100q788t678z56bg3","domain":"tempe","name":"get_temps_helper","data":{"attrs":{"id
01:09:37.697Z [DEBUG] rule selected sensor_manager : get_temps_help
01:09:37.698Z [DEBUG] fired
01:09:37.698Z [DEBUG] rule selected sensor_manager : get_temps_help
01:09:37.699Z [DEBUG] fired
01:09:37.699Z [DEBUG] rule selected sensor_manager : get_temps_help
01:09:37.699Z [DEBUG] fired
01:09:37.700Z [DEBUG] txnDone {}
```

- How can you know a report is done and all the sensors that are going to respond have reported?
 - In my case, I initialized my report before notifying all of my sensor picos. In the report, I had values for “num sensors notified” and “num sensors replied” (not exact naming). I initialized the “num replied” to 0 and every time a sensor pico responded, I incremented that number by 1. Rules do not have race conditions so we have no issues with this method. I am able to tell when the sensors have all returned their reports when the num replied is equal to the num notified.
- Given your answer above, how would you recover if the number of responding sensors is less than the total number of sensors?
 - In my setup, I do not have a blocking function or anything that would break the reports because it just keeps track of an integer. If I were to utilize these reports elsewhere that depended on the reports being fully generated, I would write a rule that checked each report correlationId before it returned the report and modified the report as done or otherwise marked before returning the report. For example, I could include a value for “num not replied”. I did not see a reason why I should block or wait for each of the sensors to reply because I am not running any rules that are dependent on the reports being finished.