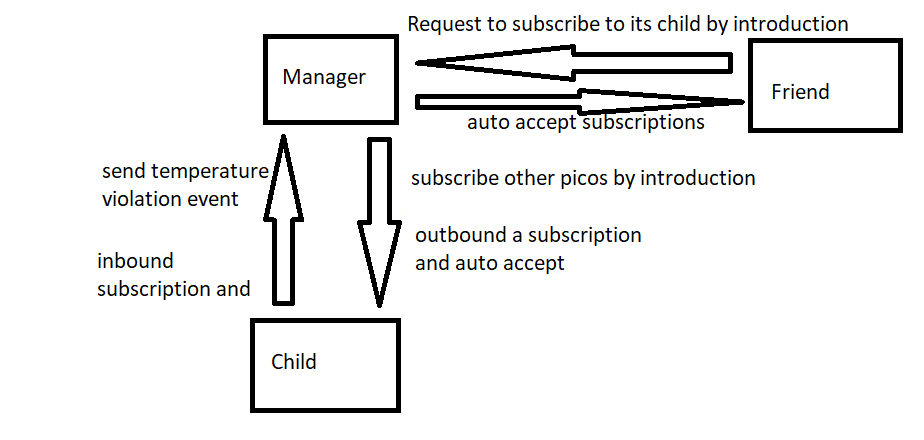
Manage\_sensors.krl: https://github.com/marceloarchiza/cs462lab7/blob/master/manage\_sensors.krl

Wovyn\_base.krl: https://github.com/marceloarchiza/cs462lab7/blob/master/wovyn\_base.krl

Lesson\_keys.krl: https://github.com/marceloarchiza/cs462lab7/blob/master/lesson\_keys.krl

**Diagram**

s

1. Why might an auto-approval rule for subscriptions be considered insecure?

Anyone that tries to subscribe to it, the rule would automatically accept, this can be dangerous because someone with bad intentions could just subscribe to the pico and get access to what he/she is not supposed to.

1. Can you put a sensor pico in more than one sensor management pico (i.e. can it have subscriptions to more than one sensor management pico)?

A pico can have multiple managers, but for this lab we did not add more. But it is possible to create more managers to the same pico child.

1. Imagine I have sensor types besides temperature sensors (e.g. pressure, humidity, air quality, etc.). How would you properly manage collections of sensors that include heterogeneous sensor types?

I would have a sensor manager that would have a subscription to all sensor types. This manager would have access to all children and it would be able to manage them.

1. Describe how you'd use the techniques from this lesson to create collections of temperature sensors in particular rooms or areas of a building. For example, I would still have the sensor management pico, but might have collections for each floor in a building.

I would have a main sensor manager, and its child would be sensor manager for each floor. Each floor sensor manager would have a pico that represent each room on the building.

1. Can a sensor pico belong to more than one collection? After the modifications of this lab, if a sensor belonged to more than one collection and had a threshold violation, what would happen?

Yes, it can. But there might be some bugs, for example, they way I coded, the child pico is only getting the eci of the first manager subscribed to it. So if a threshold violation happened, only the first manager would be notified and the others would not.

1. When you moved threshold violation notifications from the sensor to the management ruleset, did you add the rules to an existing ruleset or create a new one? Why?

I added to an existing rule, the manage\_sensors. I do believe it had too much code on it and it would be better to actually create a new rule for it, but it worked out fine and I ended up leaving all inside of manage\_sensors ruleset.

1. When you moved threshold violation notifications from the sensor to the management ruleset, did you add only one rule or more than one rule to achieve this end? Which rules did you add and why (i.e. justify the architectural decisions did you made)?

I added only one rule and I added the lesson\_keys to the manage\_sensors as well. I just added a “sensor/notification” rule that sends a sms using “twlio\_v2” ruleset. The reason why I added only one is because it looked simple and easier to understand.