HEADS
Heterogenous and Distributed
Services for the Future Computing Continuum

The Room X3C – ThingML Sessions
Dynamic processes

Motivation and overview

- In The Room X1-X3B we have static configurations
  - no processes are dynamically created (or destroyed)
- Static systems are less ready for changes of the environment
- In X1 to X3B we still use long arrays to hold information about thermometers in the PSM
  - We do not know how many thermometers are sufficient to prepare for
- Could we make ThermometerSet a set of processes?

HEADS
Heterogenous and Distributed
Services for the Future Computing Continuum

The Sessions of ThingML

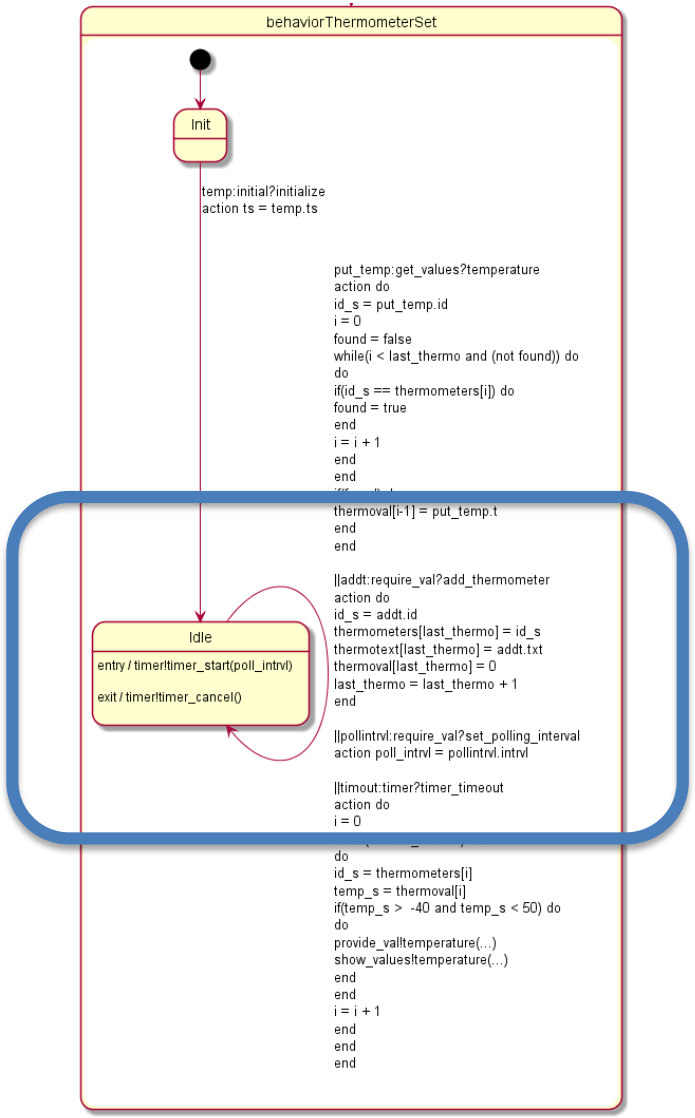
A Session in ThingML

- ThingML has one way to define a set of things relating to the same thing definition
- A set of similar things is made by:
  - **fork** instruction to generate **sessions**
  - cloning the forking object
  - multicasting messages that are received by the set to all its members
- One *core* object is defined differently from the sessions

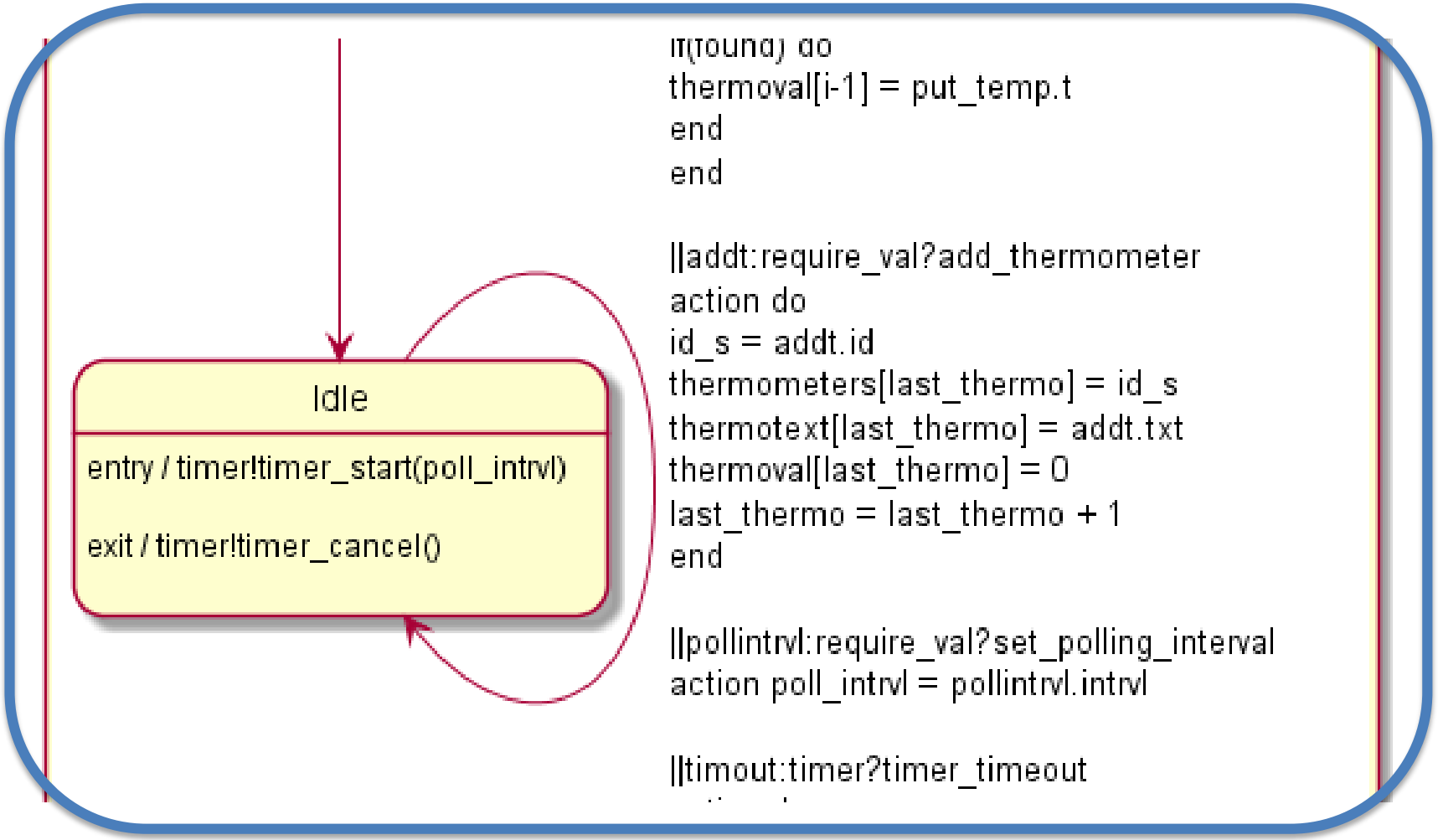
HEADS
Heterogenous and Distributed
Services for the Future Computing Continuum

The ThermometerSet

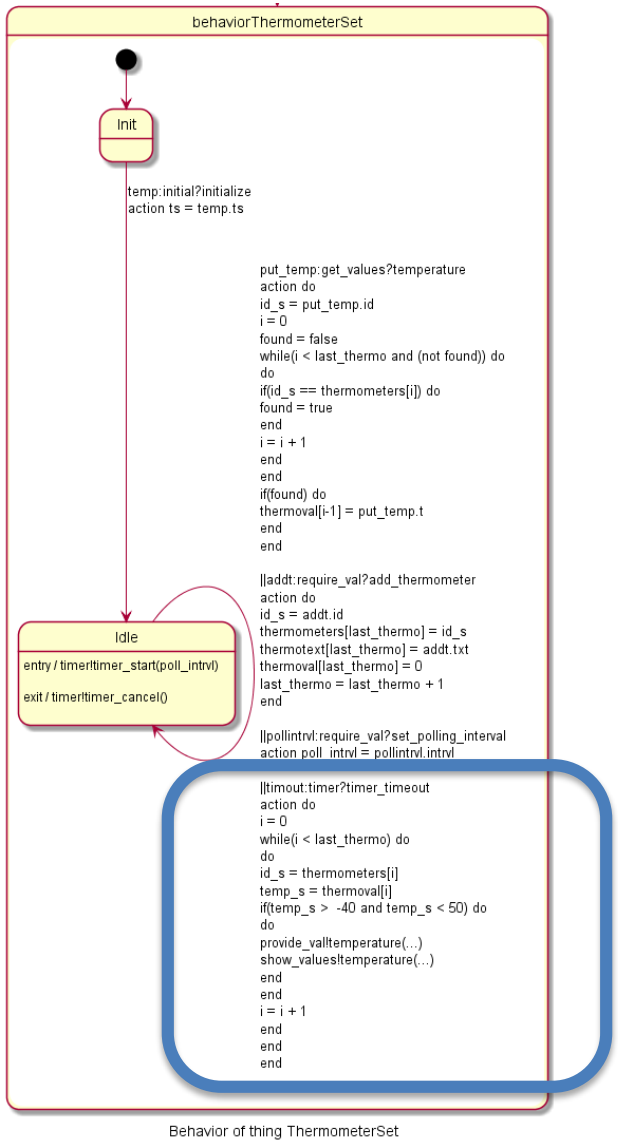
X3B excerpts on adding thermometer



Behavior of thing ThermometerSet



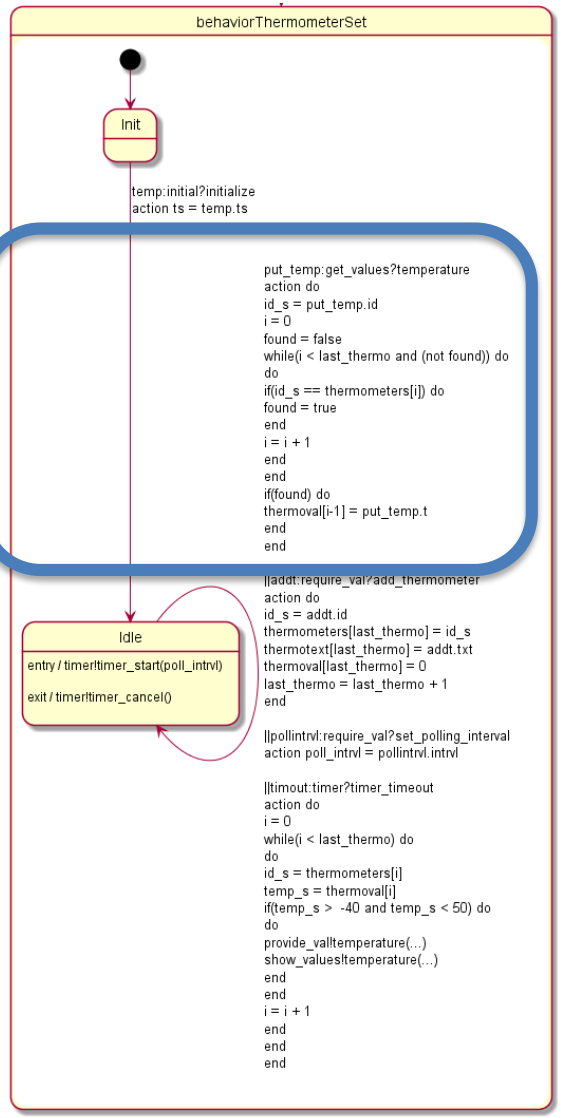
X3B excerpts on sending temp to PIM



Behavior of thing ThermometerSet

```
||timout:timer?timer_timeout
action do
i = 0
while(i < last_thermo) do
do
id_s = thermometers[i]
temp_s = thermoval[i]
if(temp_s > -40 and temp_s < 50) do
do
provide_val!temperature(...)
show_values!temperature(...)
end
end
i = i + 1
end
end
end
```

receiving simulated temperature



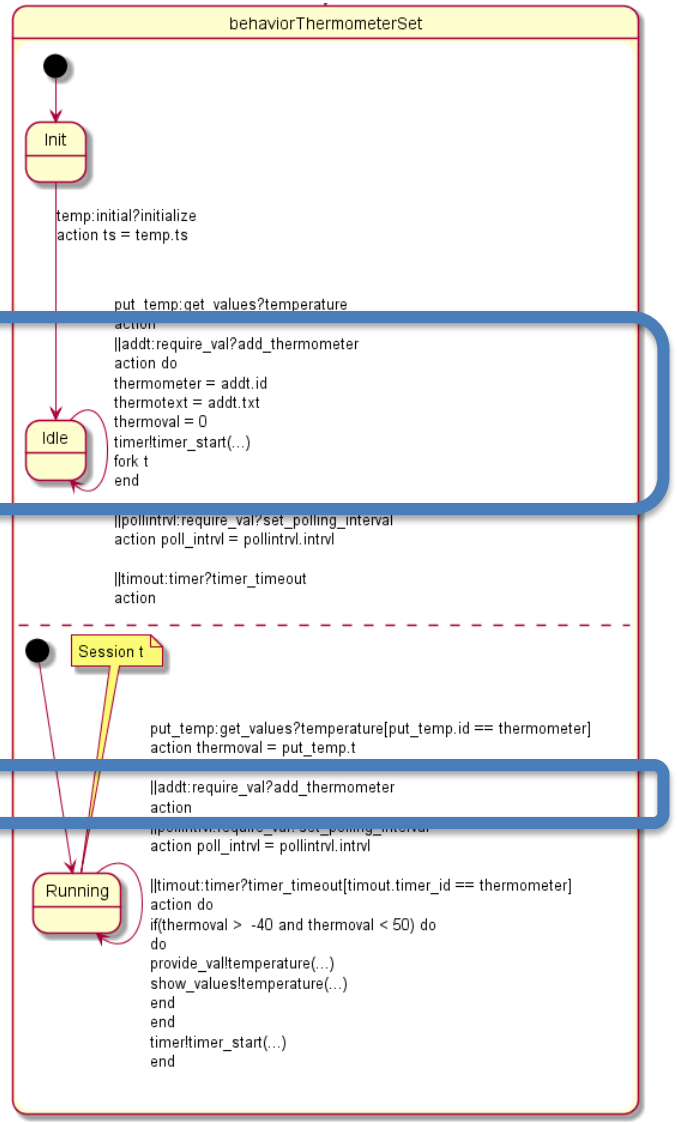
Behavior of thing ThermometerSet

```
put_temp:get_values?temperature
action do
id_s = put_temp.id
i = 0
found = false
while(i < last_thermo and (not found)) do
do
if(id_s == thermometers[i]) do
found = true
end
i = i + 1
end
end
if(found) do
thermoval[i-1] = put_temp.t
end
end
```

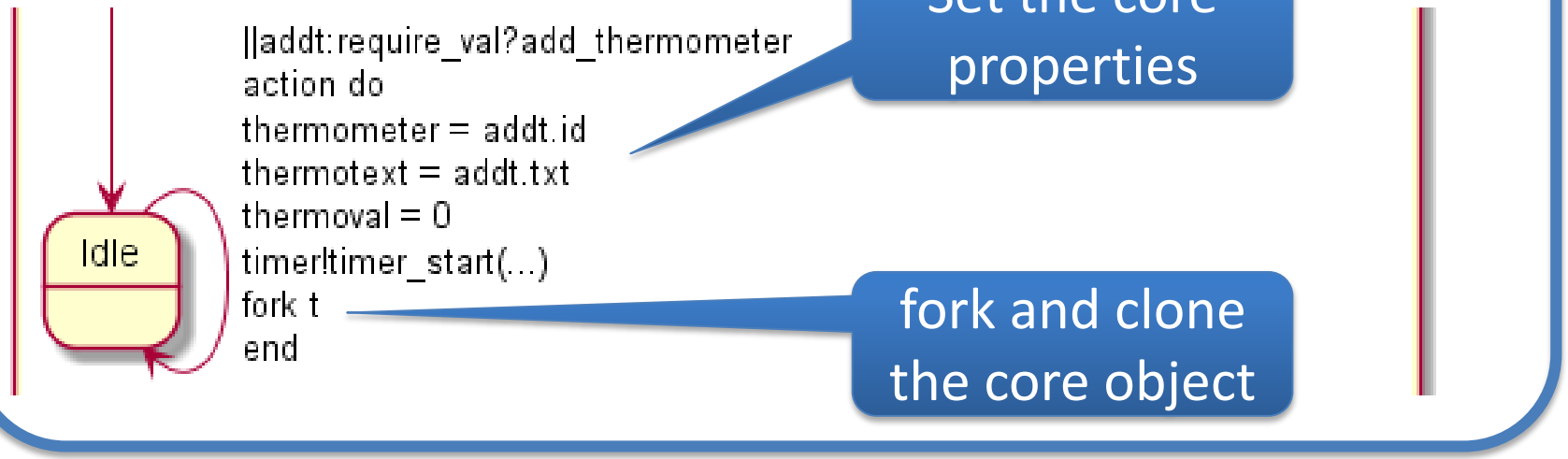
In X3C we change to sessions

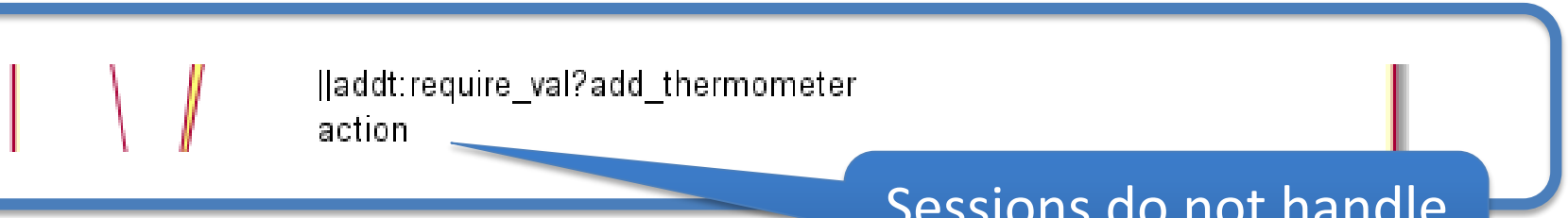
- One session per added thermometer
- One core to handle forks

X3C excerpts on adding thermometer

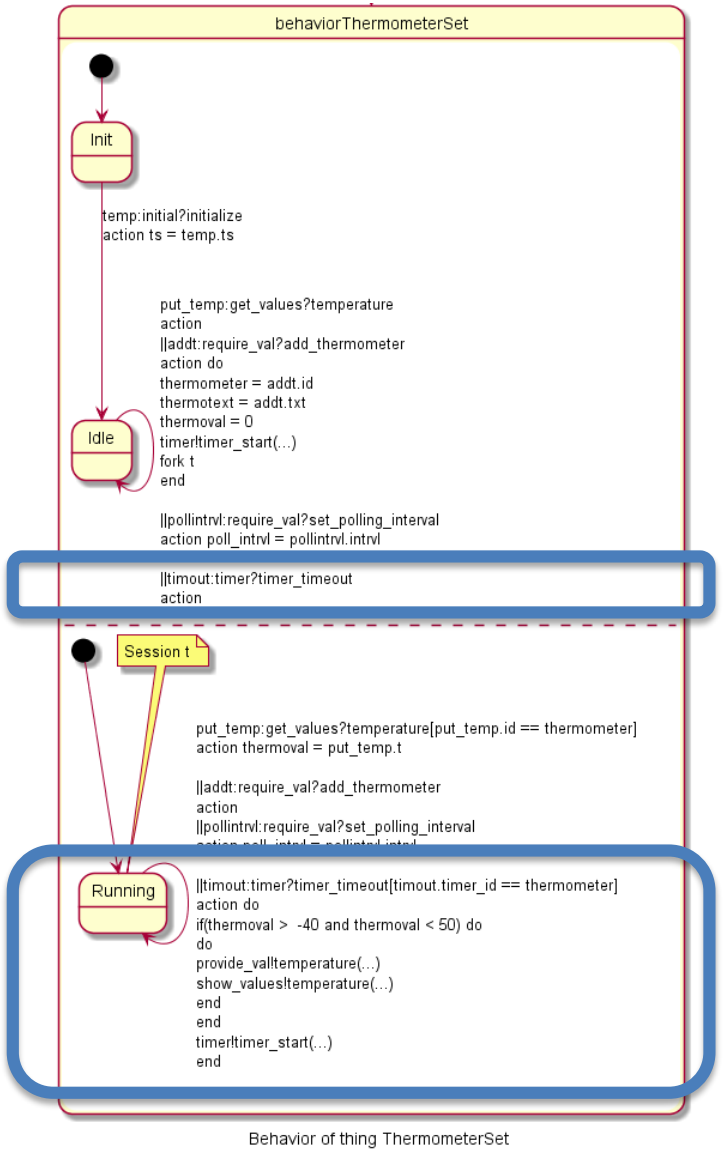


Behavior of thing ThermometerSet





X3C excerpts on sending temp to PIM



Behavior of thing ThermometerSet

```
||timout:timer?timer_timeout
action
```

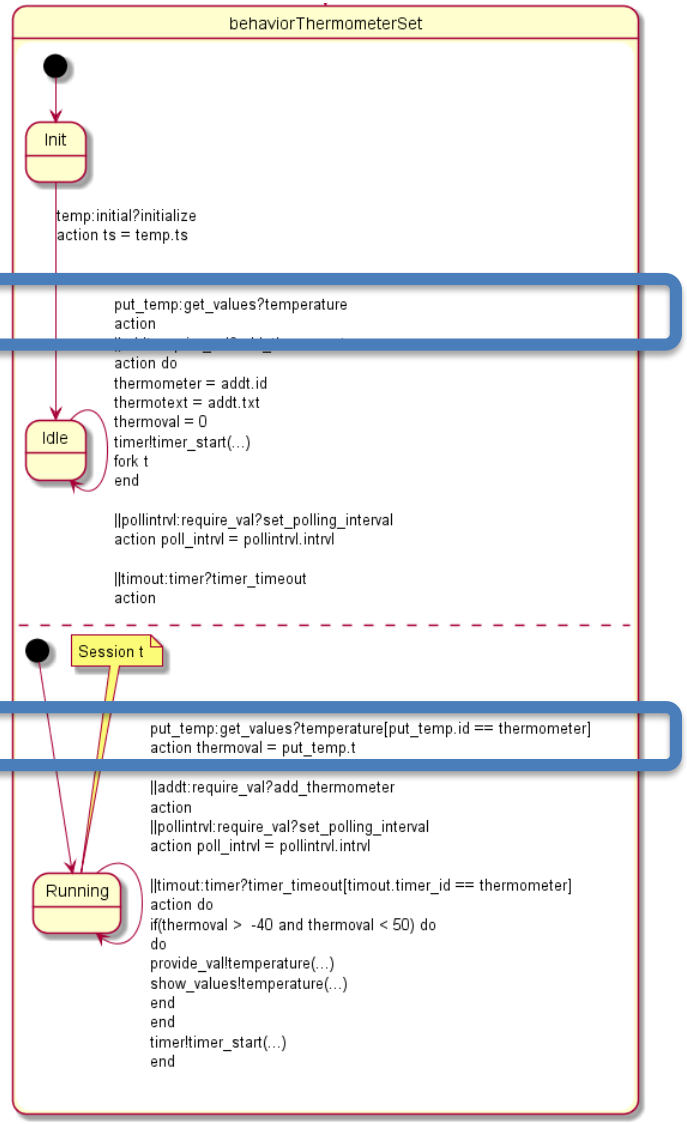
Core does not handle sending temperature, but the timeout is multicast to all sessions

Running

```
||timout:timer?timer_timeout[timout.timer_id == thermometer]
action do
if(thermoval > -40 and thermoval < 50) do
do
provide_val!temperature(...)
show_values!temperature(...)
end
end
timer!timer_start(...)
end
```

Each session sends its temperature

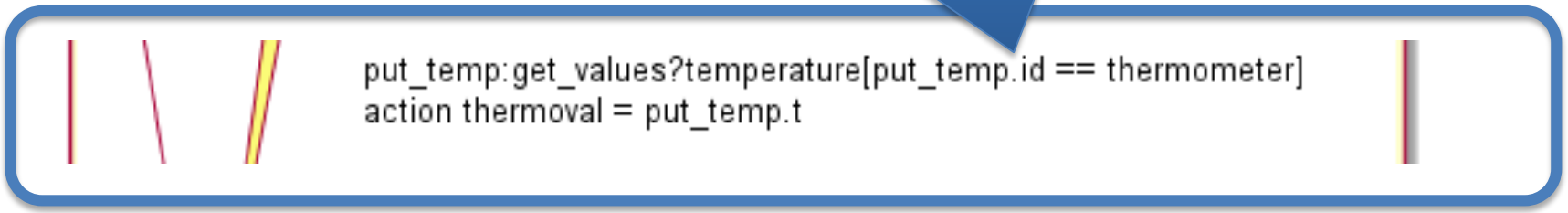
receiving simulated temperature



Behavior of thing ThermometerSet

Core does not handle simulated temperature, but the temperature is multicast to all sessions

Each session checks whether the simulated temperature is its temperature



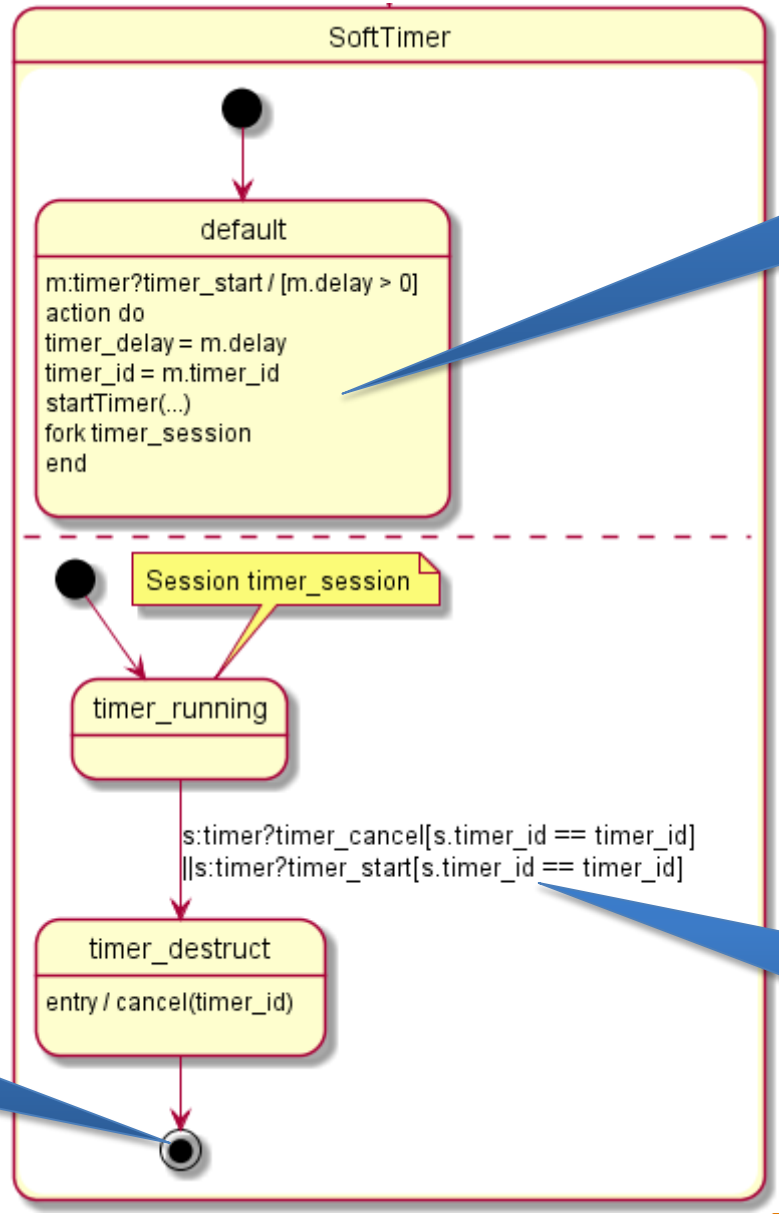
HEADS
Heterogenous and Distributed
Services for the Future Computing Continuum

The TimerSet

Motivation for TimerSet with dynamic timers

- We have shown how each thermometer could be made a dynamic session
  - but we were able to provide the same functionality by using a few arrays that we assumed were longer than the maximum number of thermometers
- But what if each thermometer should have its own timer?
  - For the simulation this would prevent all temperatures to come almost at the same time, and we could have different polling intervals for different thermometers
  - Timers cannot be put in arrays

The TimeSet with Sessions



Every timer_start will create a fresh timer with the given timer_id

Sessions are destroyed when they reach the final state

Every existing timer session with the timer_id of a timer_start will be canceled

Consortium

http://heads-project.eu