More services – more processes

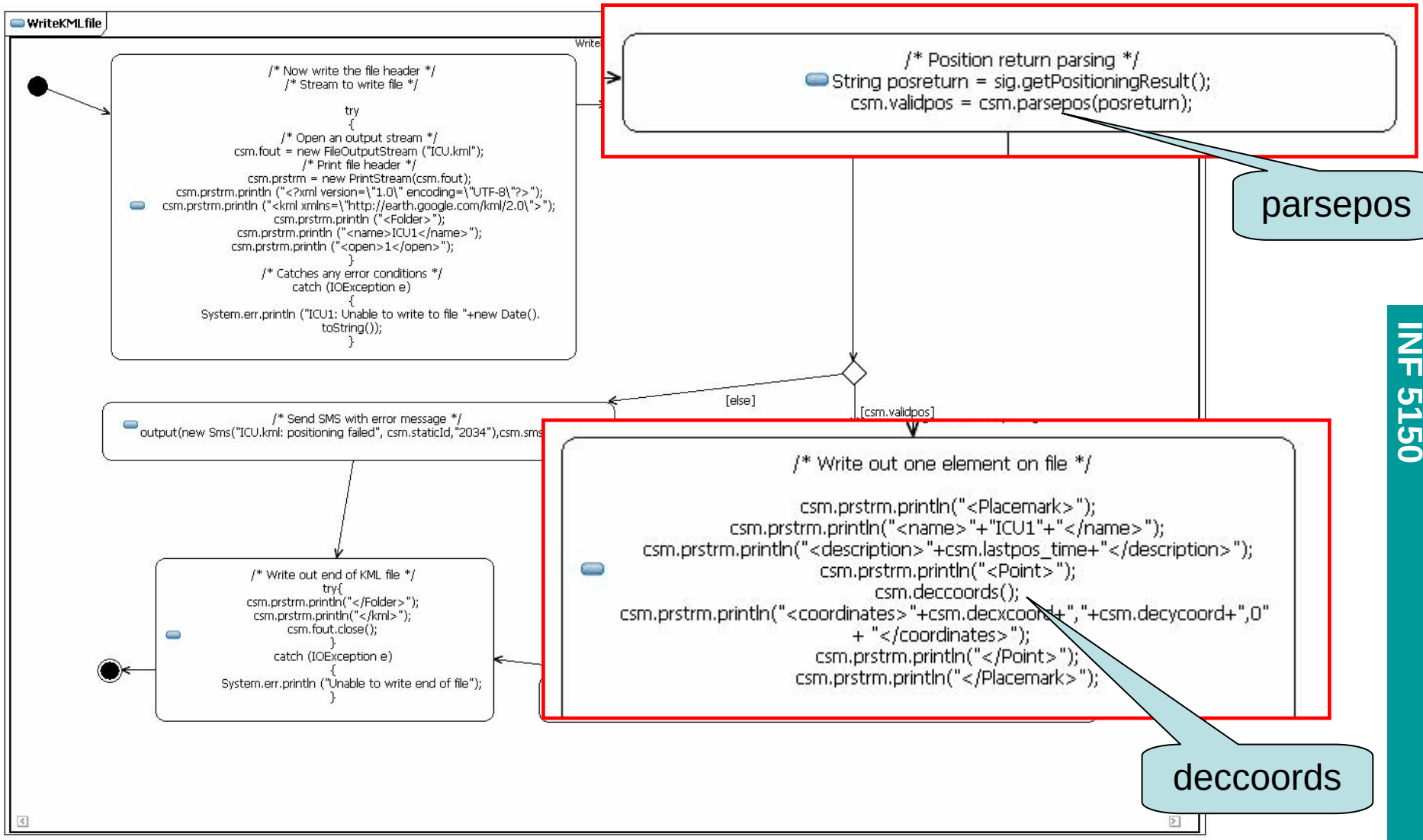
.... not quite the same thing!

Version 071019

About operations and methods

In order to keep the low-level java code away from the beautiful symbols of our UML models, we may want to separate some of the nitty, gritty details in out in chunks

We will introduce operations/methods



INF 5150

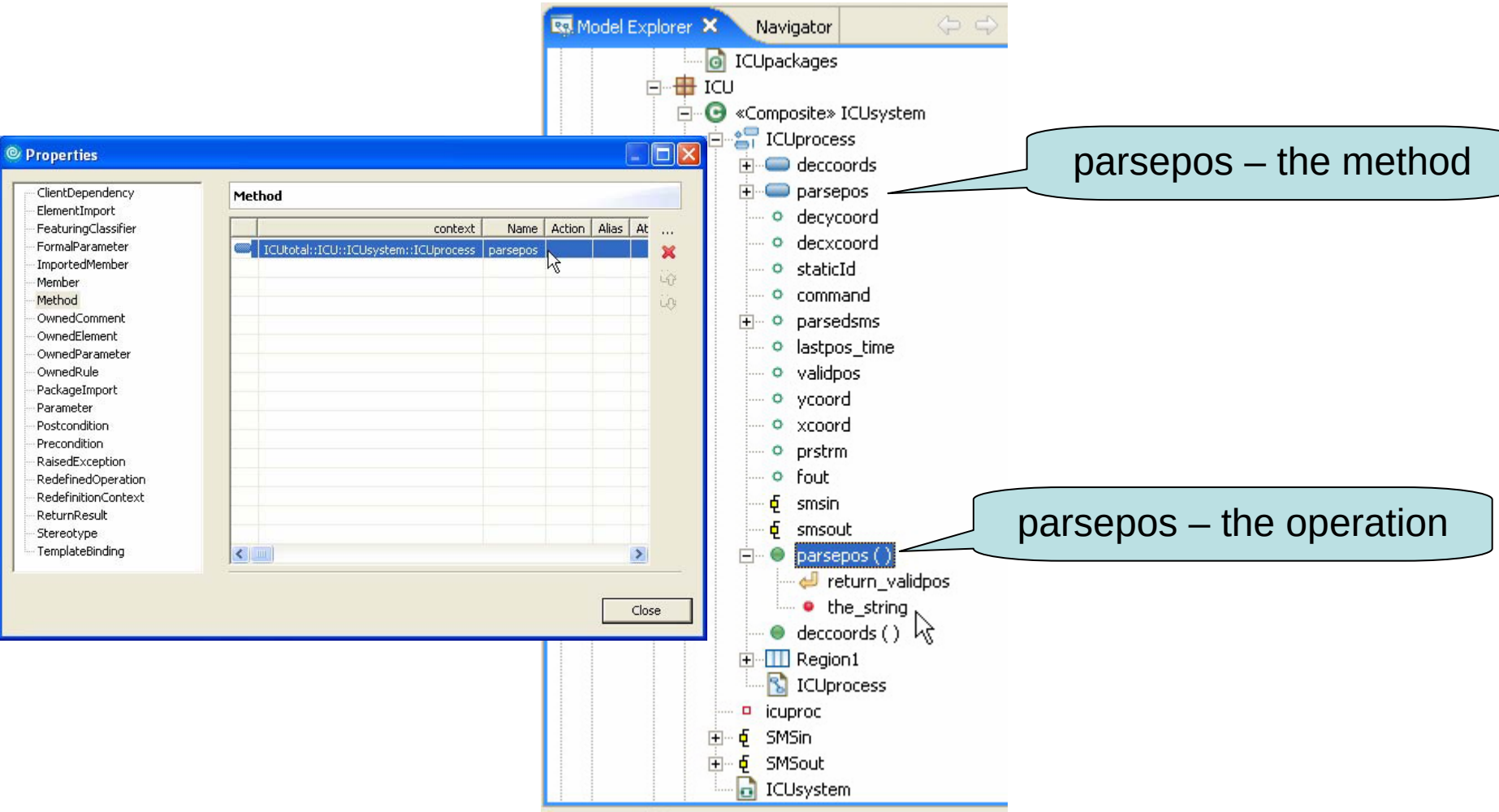
RSM coding rules for state machines (1)

- Trigger of transitions
  - Name of the transition
  - or Generate a SignalTrigger by rightclicking on transition
- Effect of transition
  - Name of effect
  - or Use one Action within an Activity diagram (forget flow lines etc.) created when doubleclicking the effect icon.
- Inside the effect
  - JavaFrame statements
  - or Branch by using Choice points
    - outgoing transitions from a choice point should have a guard (predicate condition for this piece of the transition)

RSM coding rules for state machines (2)

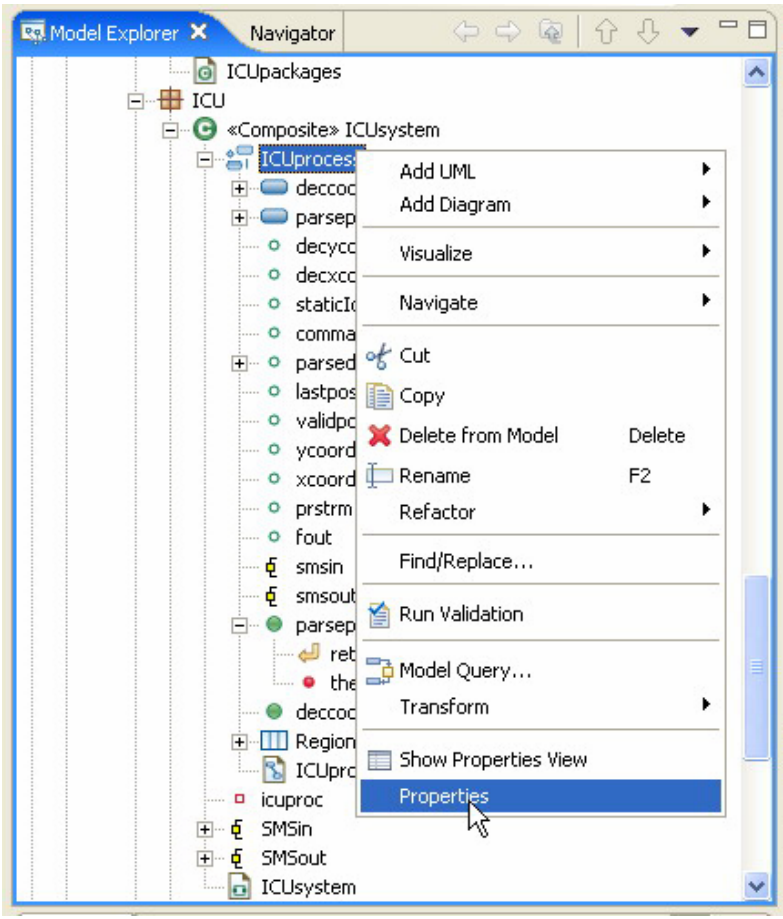
- output (Signal, Port, csm)
  - sends a signal through the local port.
  - typically the signal is like ”new S(parm1, parm2)”
  - typically the port is like ”csm.toSomewhere”
  - ”csm” is like a keyword meaning ”current state machine”
- To read from the consumed signal, use ”sig”
  - sig has been cast to the right type (normally)
  - Example: ”sig.parm1” when sig is consumed as object of class S
- UML defer
  - to add a **deferrable trigger**, make sure the trigger to be deferred has a signaltrigger element in the state machine
  - right click the state > Properties > DeferrableTrigger and add the appropriate signaltrigger.
  - But you will not see the **defer** in the diagram – only in the model

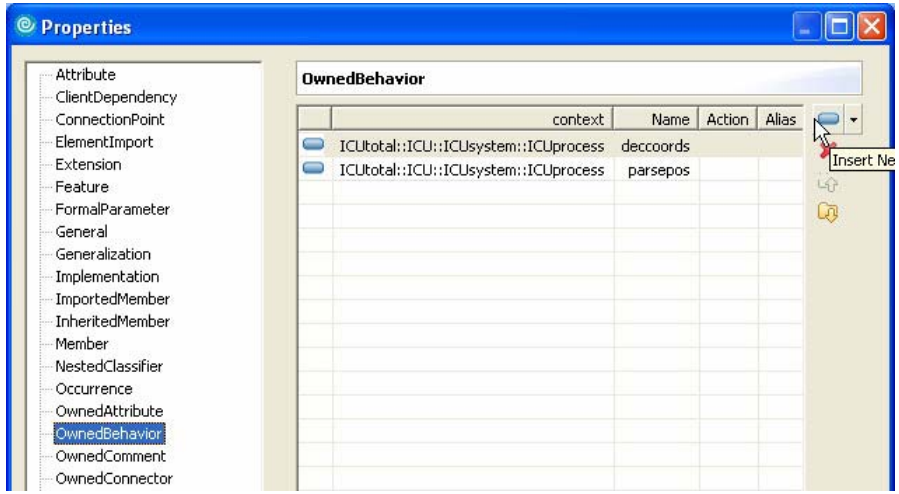
UML distinguish between operation and method

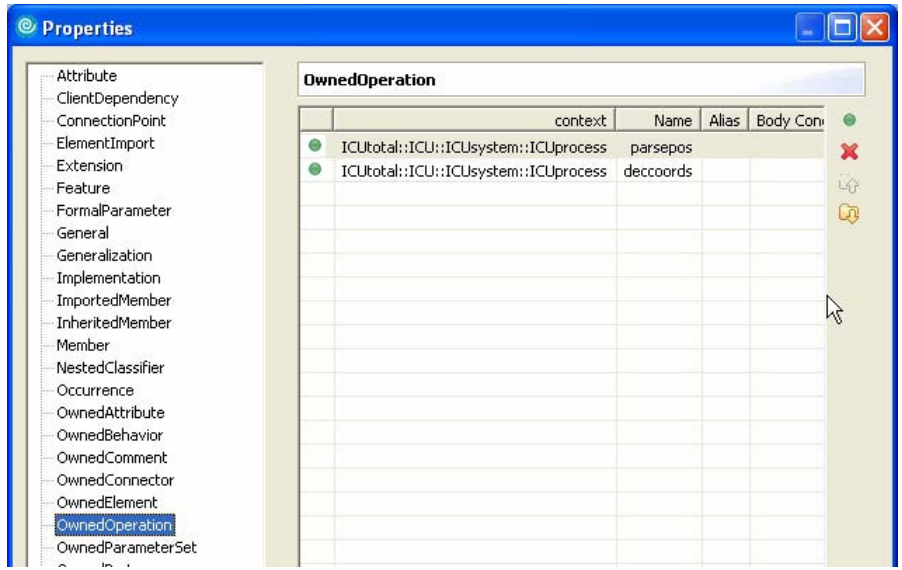


INF 5150

How to make operations and activities

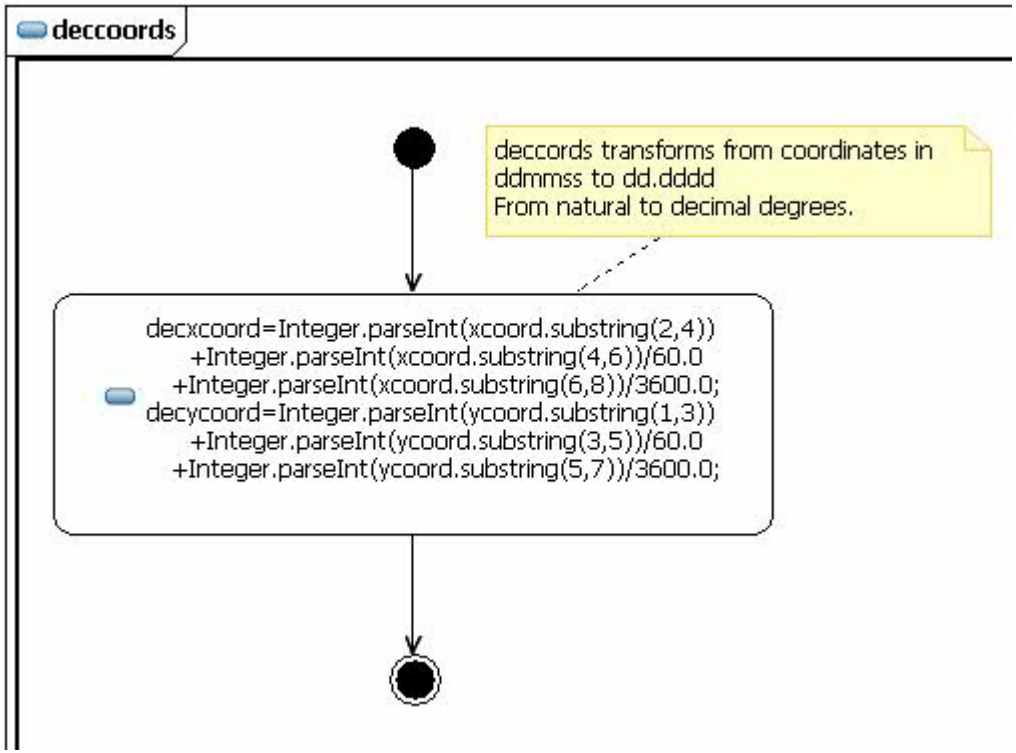






INF 5150

The method definition is an action diagram



More than one service

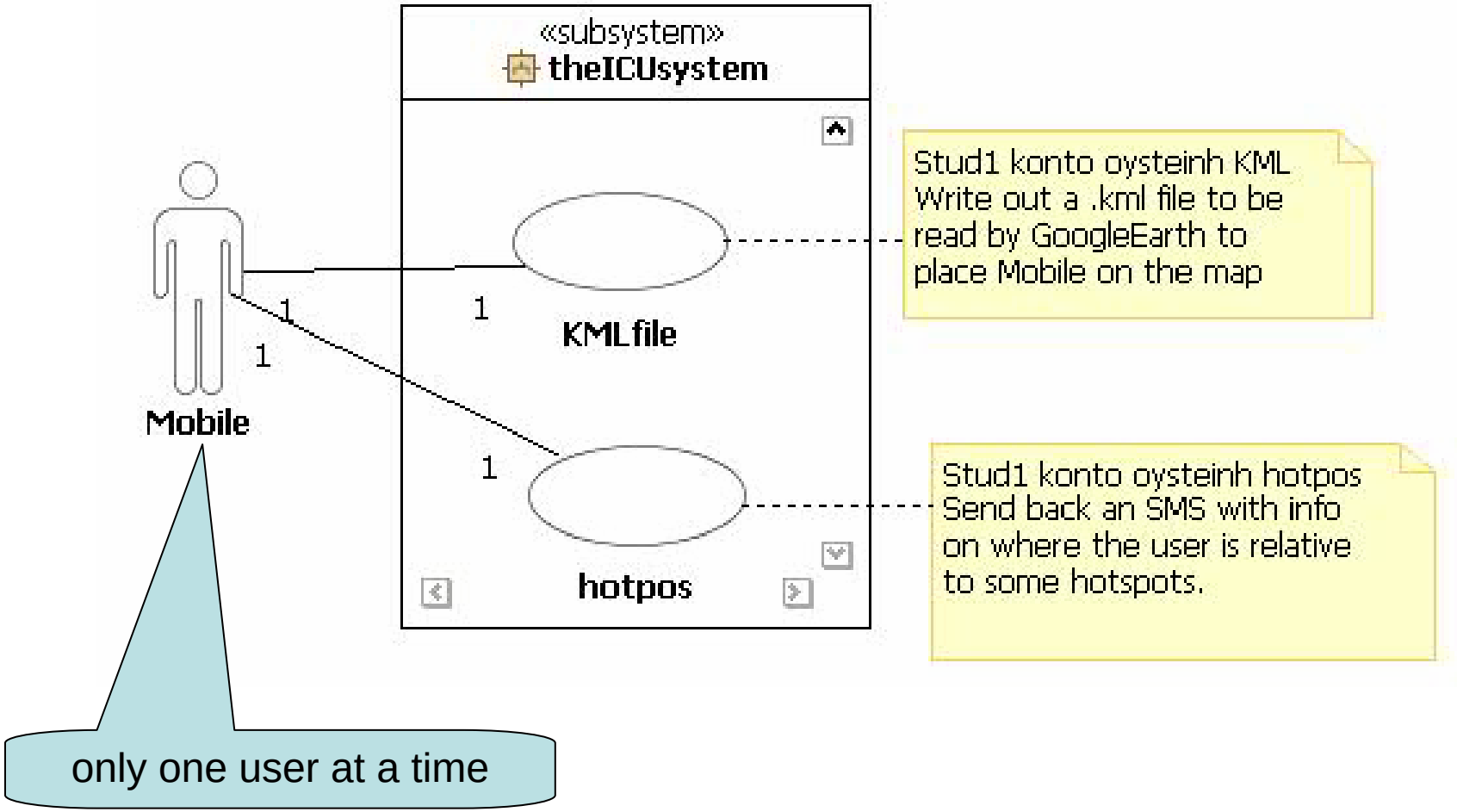
We introduce our second service

hotpos

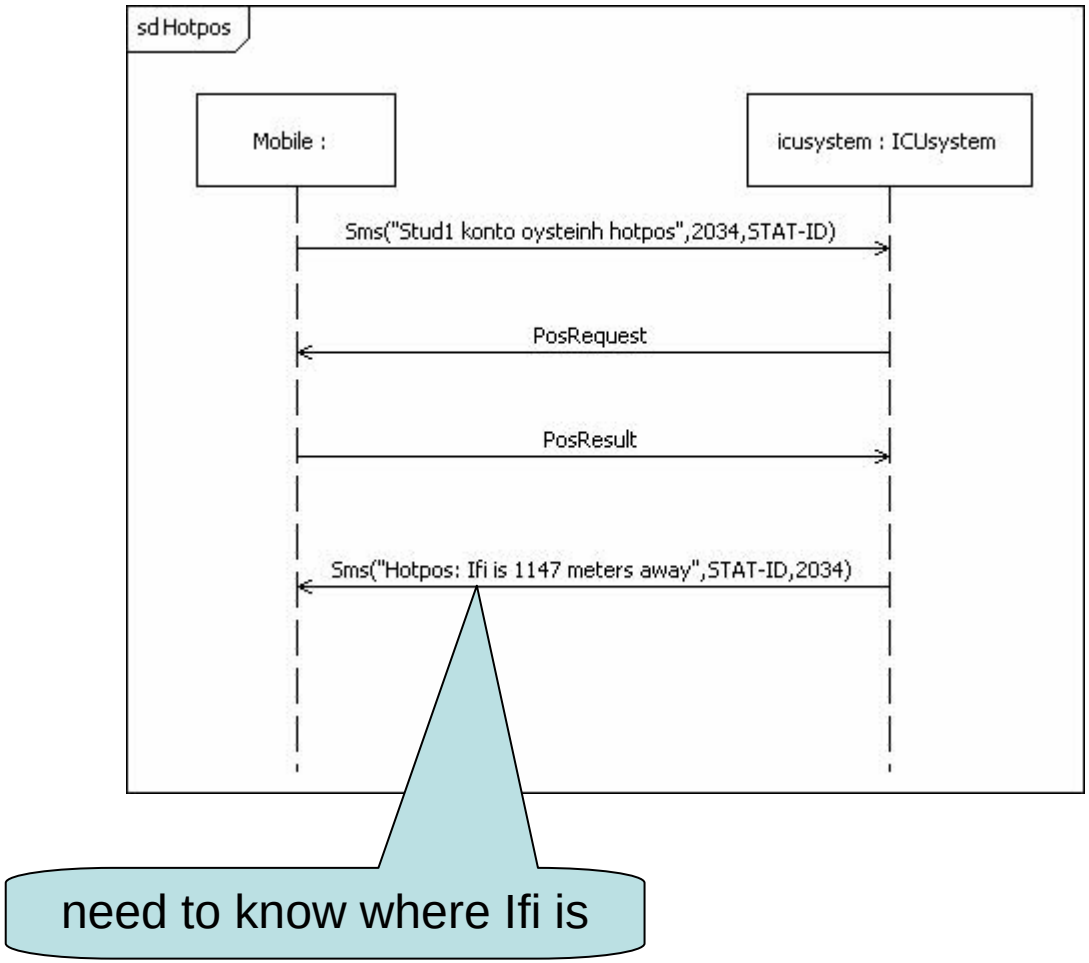
where it may be useful to apply methods

ie. define concepts only one place

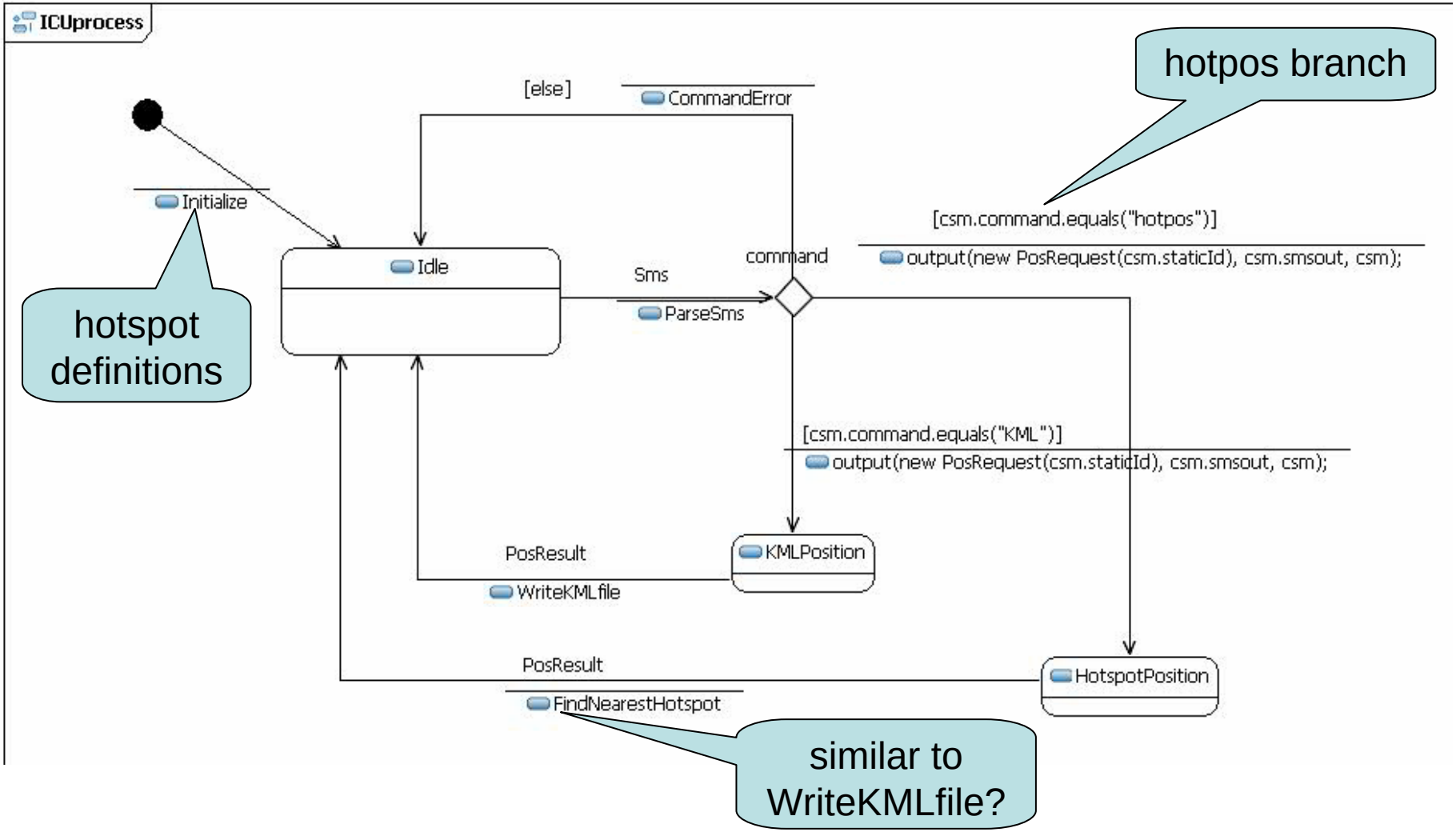
Adding a new service



Hotpos described by a sequence diagram



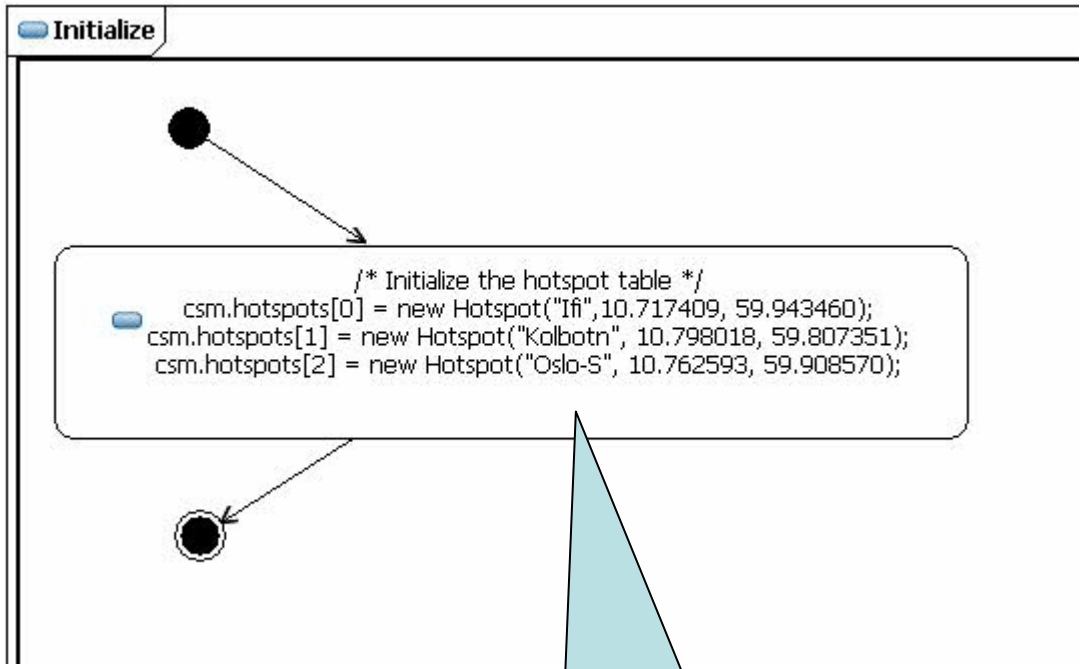
The modified ICUprocess



Buzzz 1: Why limiting to one user?

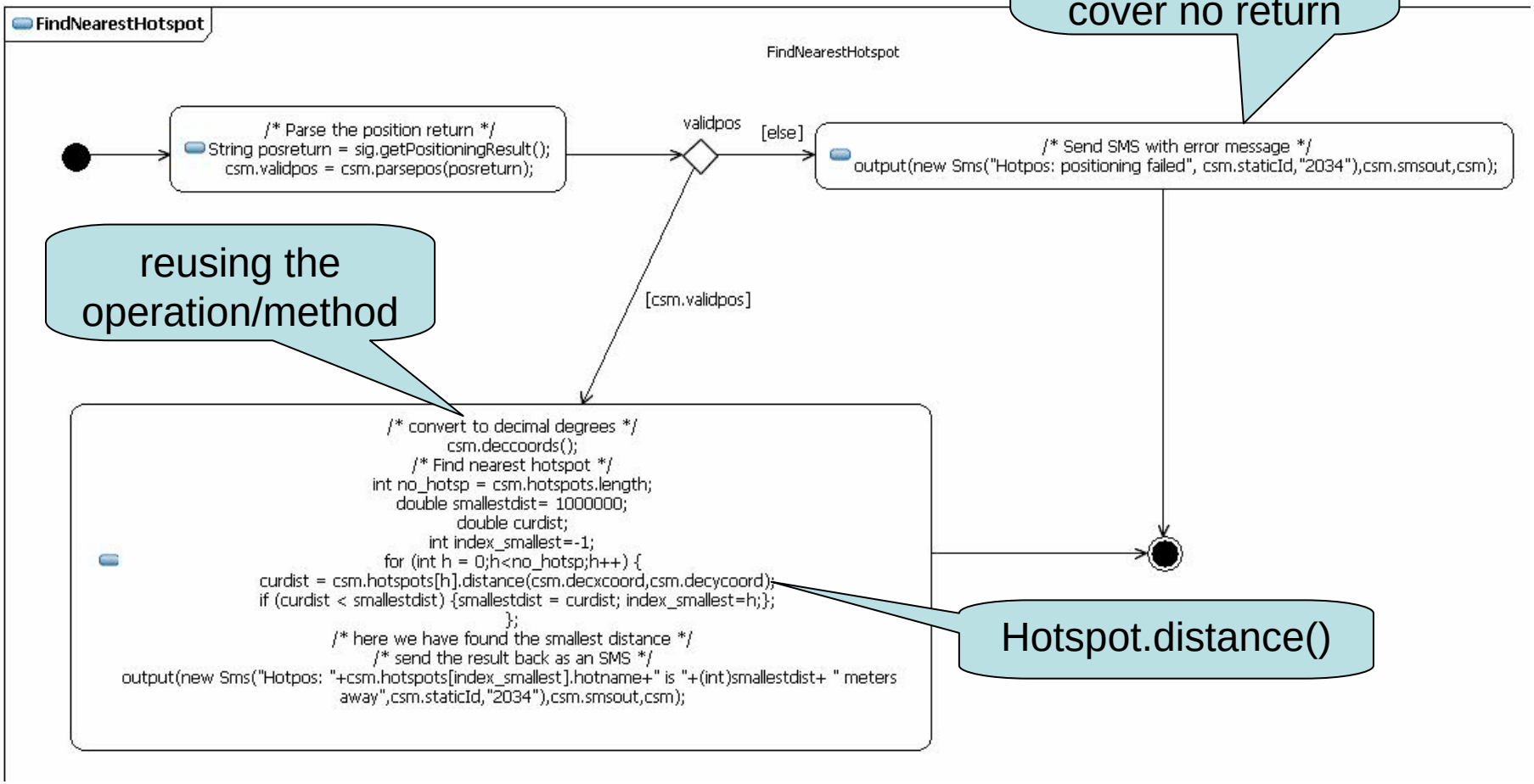
- Make up pairs with one person just beside you
- Discuss for 3 minutes why we have restricted the system to consider only one user at the time

Hardcoding the hotspots



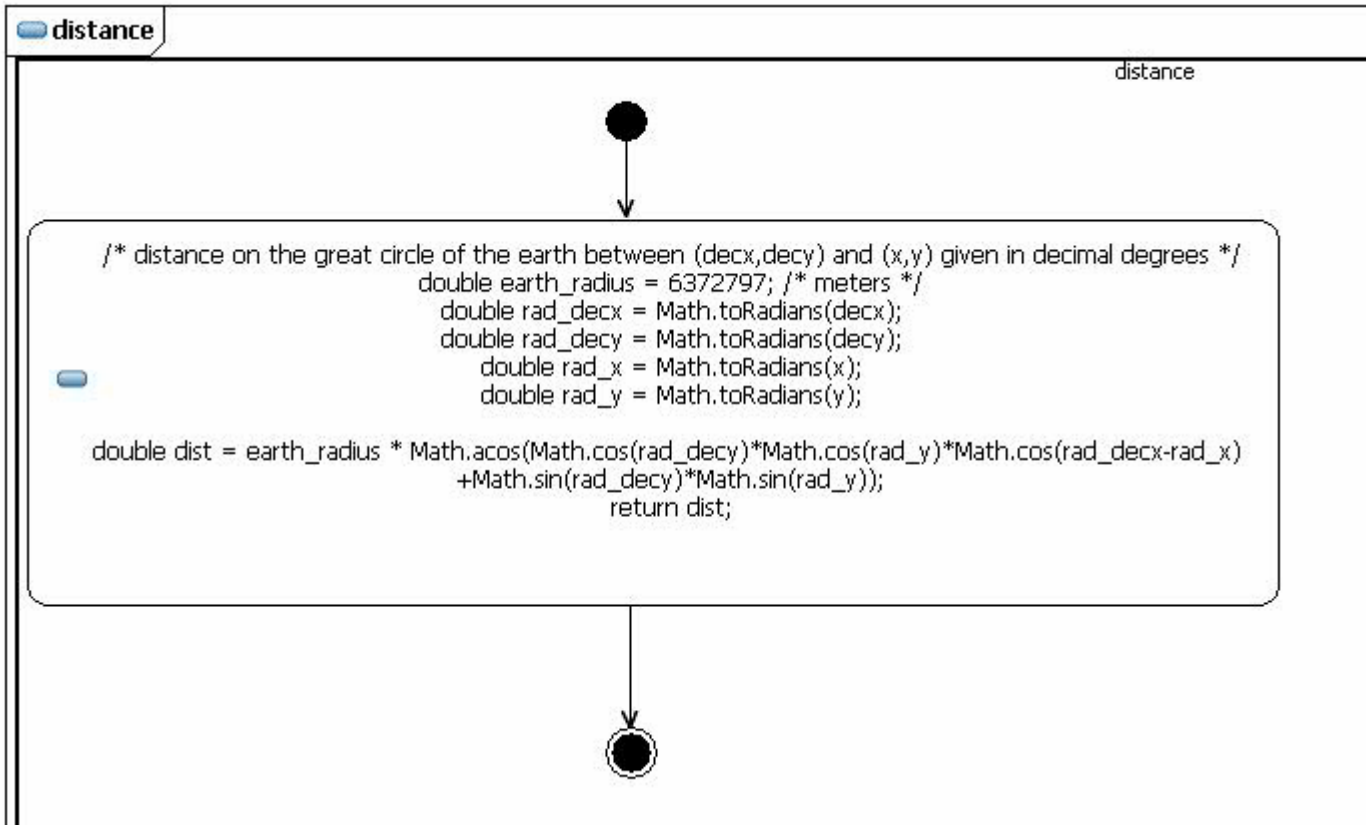
Feel free to add your own hotspots. Remember to change the size of the array

FindNearestHotspot



INF 5150

Hotspot.distance()

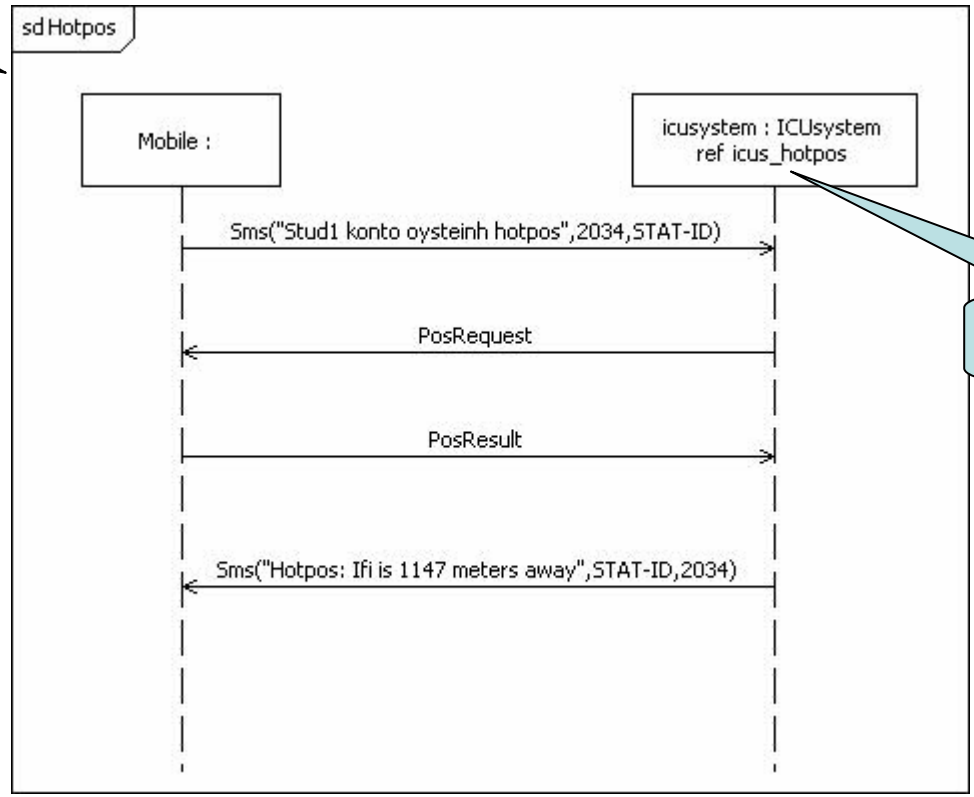


Separation of concerns

- We want to separate different concerns of the ICU system through using separate state machines that communicate
- The architecture of the ICUSystem will evolve
- One process controls
  - the handling of SMSes
  - and the production of the KML file
- One process controls the handling of the data
  - which are still going to be hardcoded (for now)
- These processes communicate with signals that we define ourselves

Hotpos service – as seen from the context

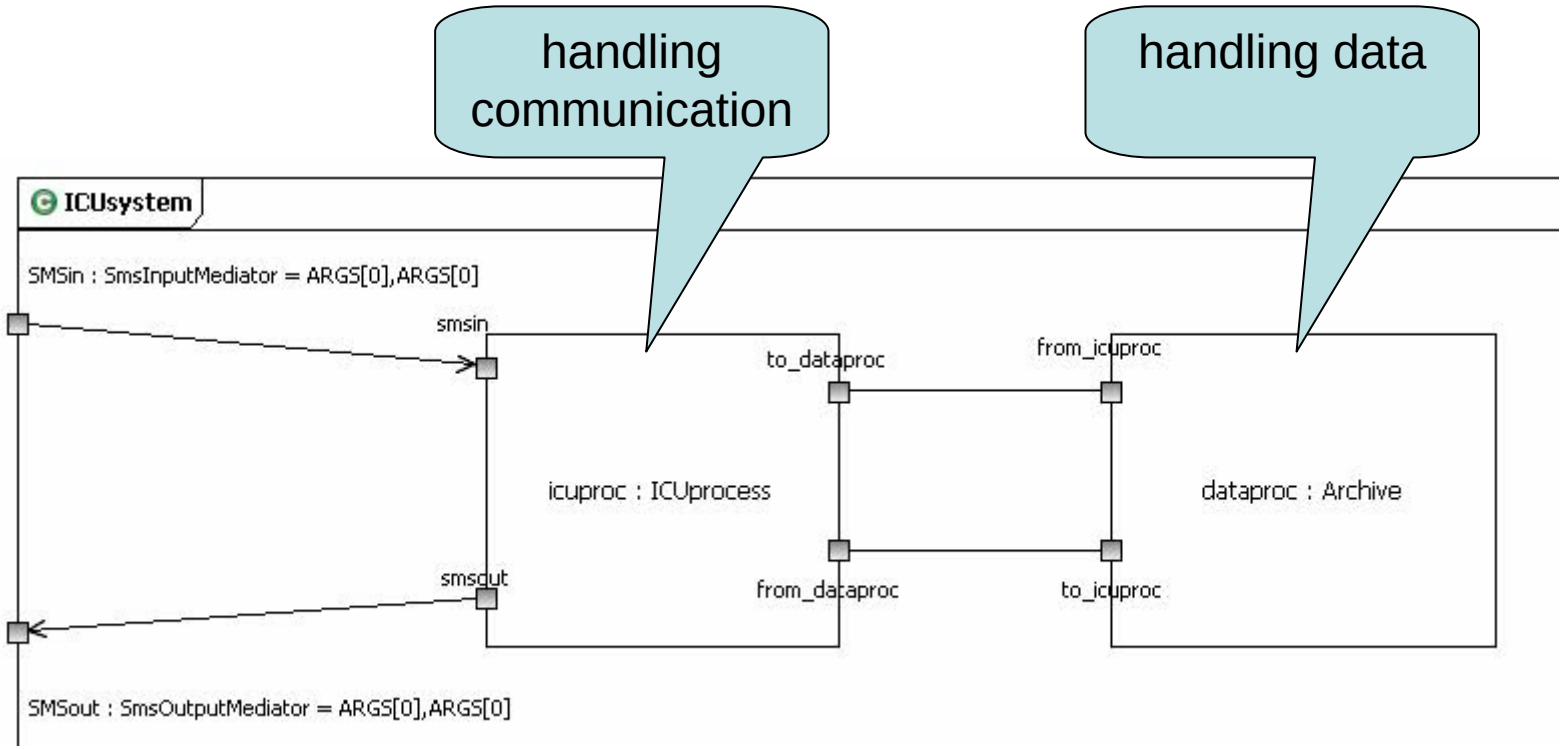
very similar to ICU2!



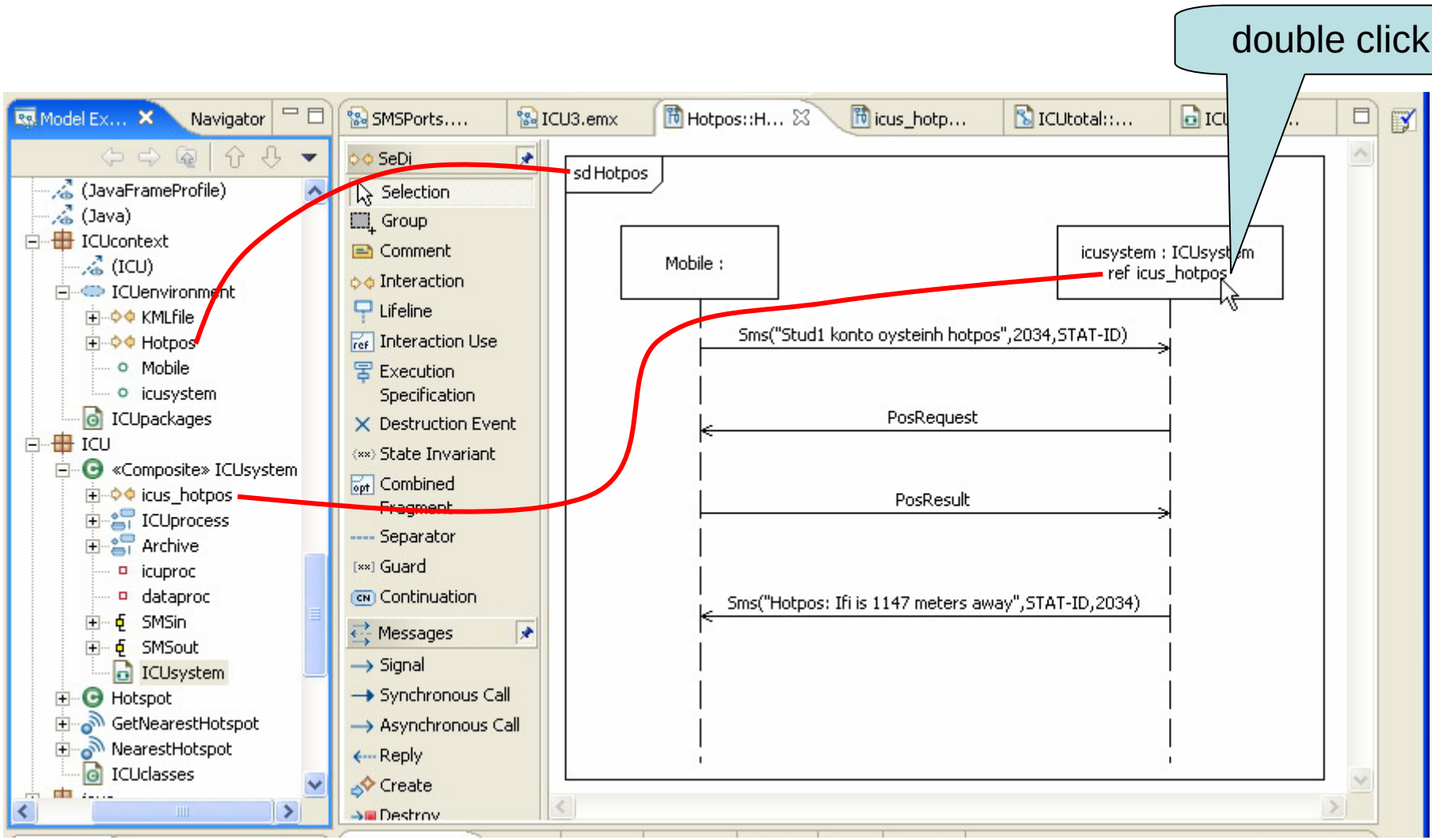
what is this?

INF 5150

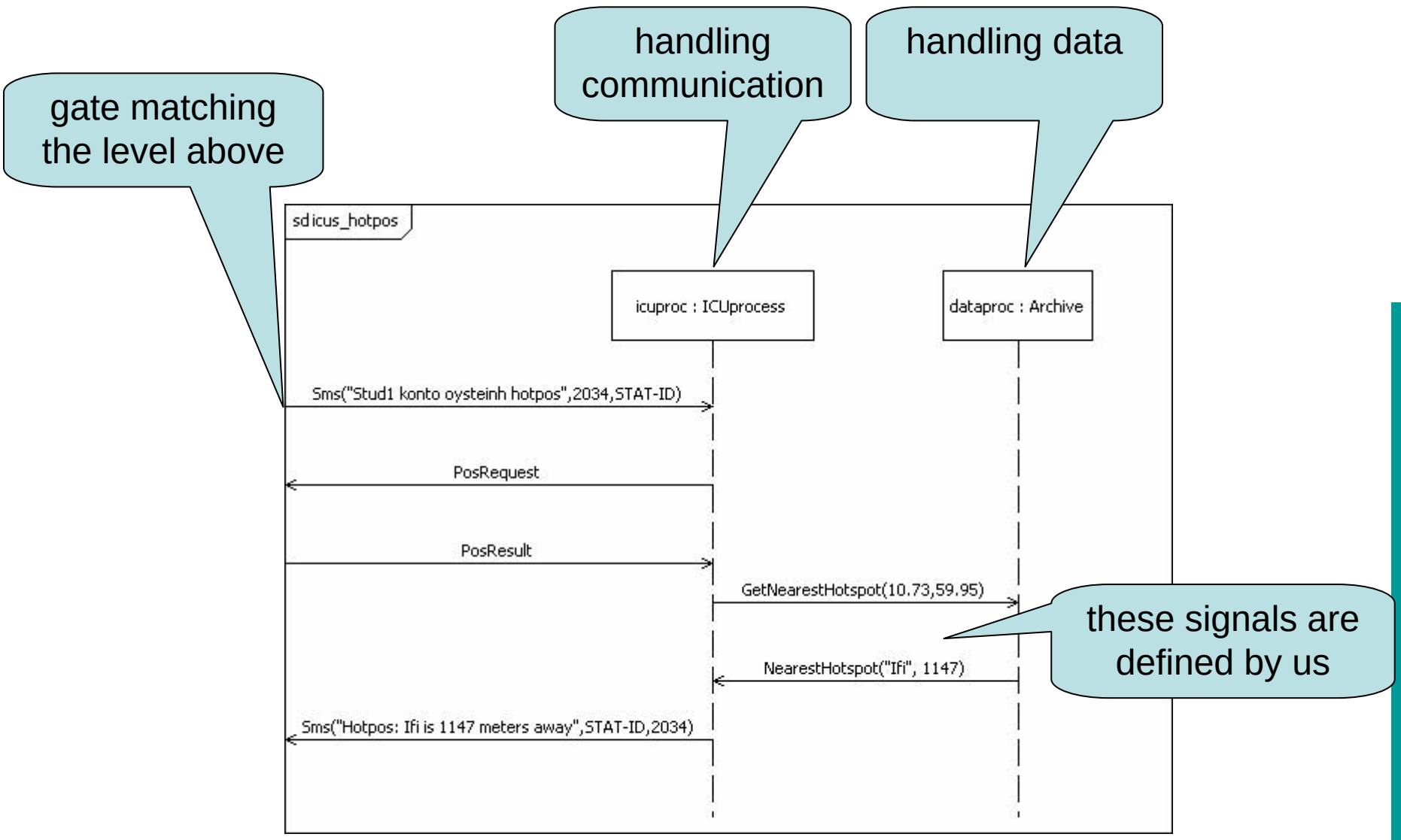
Inside the ICUsystem



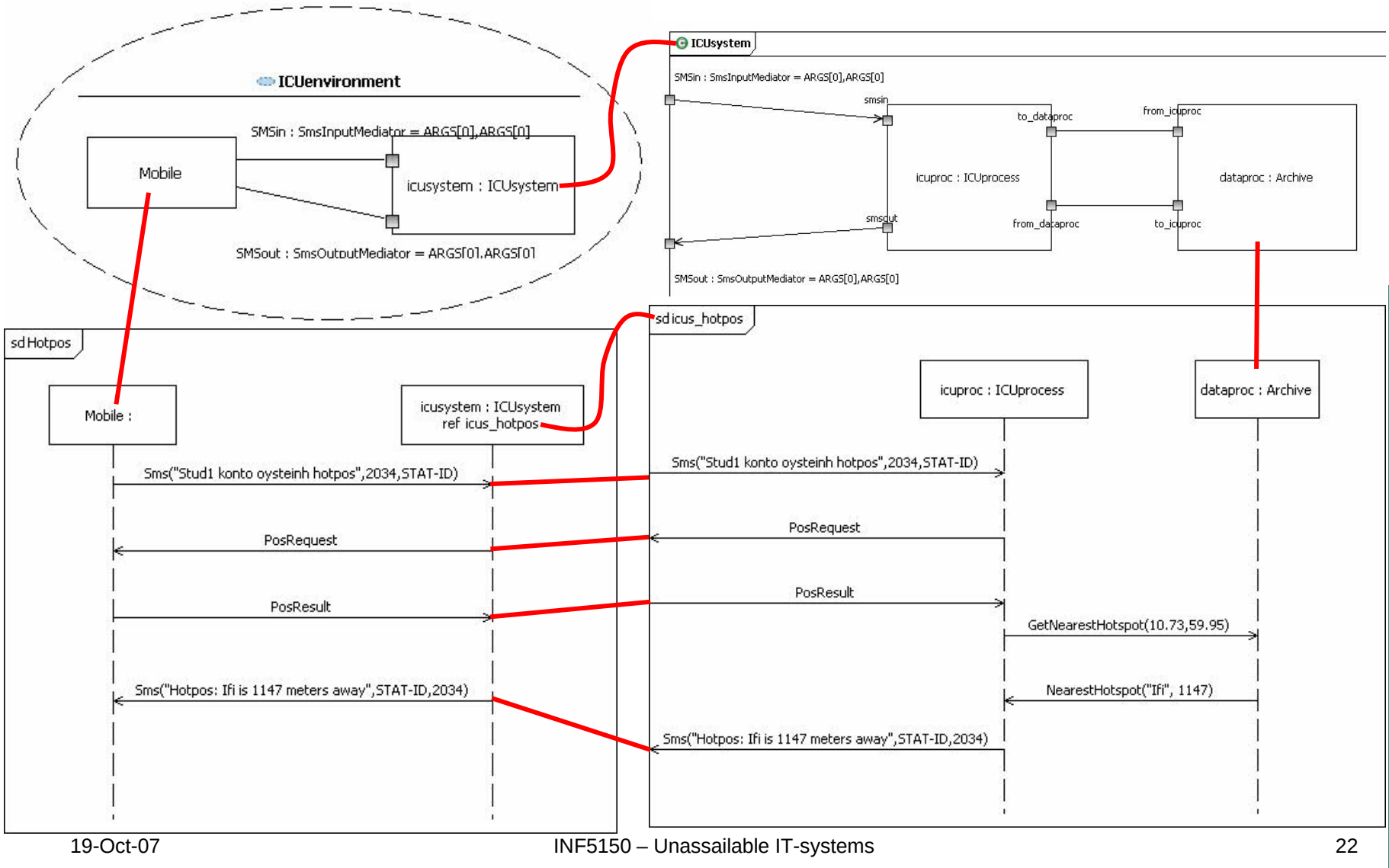
Decomposing the ICUsystem



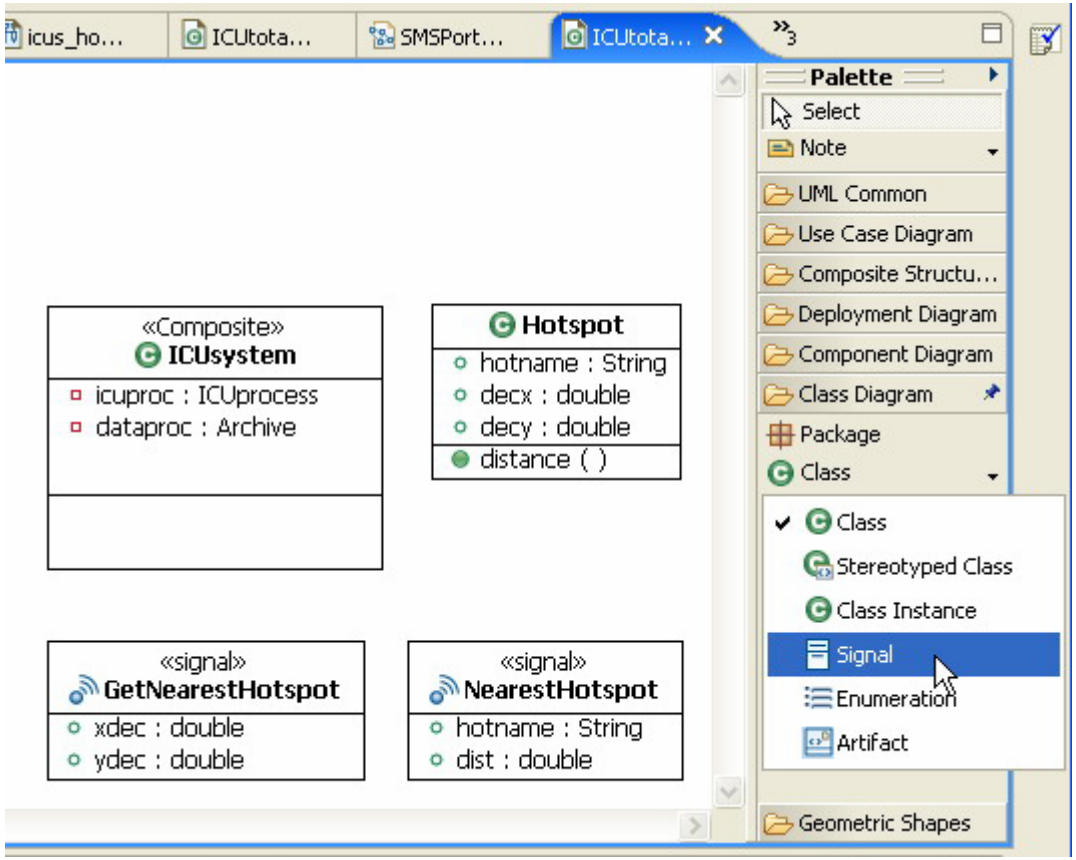
The behavior inside ICUsystem



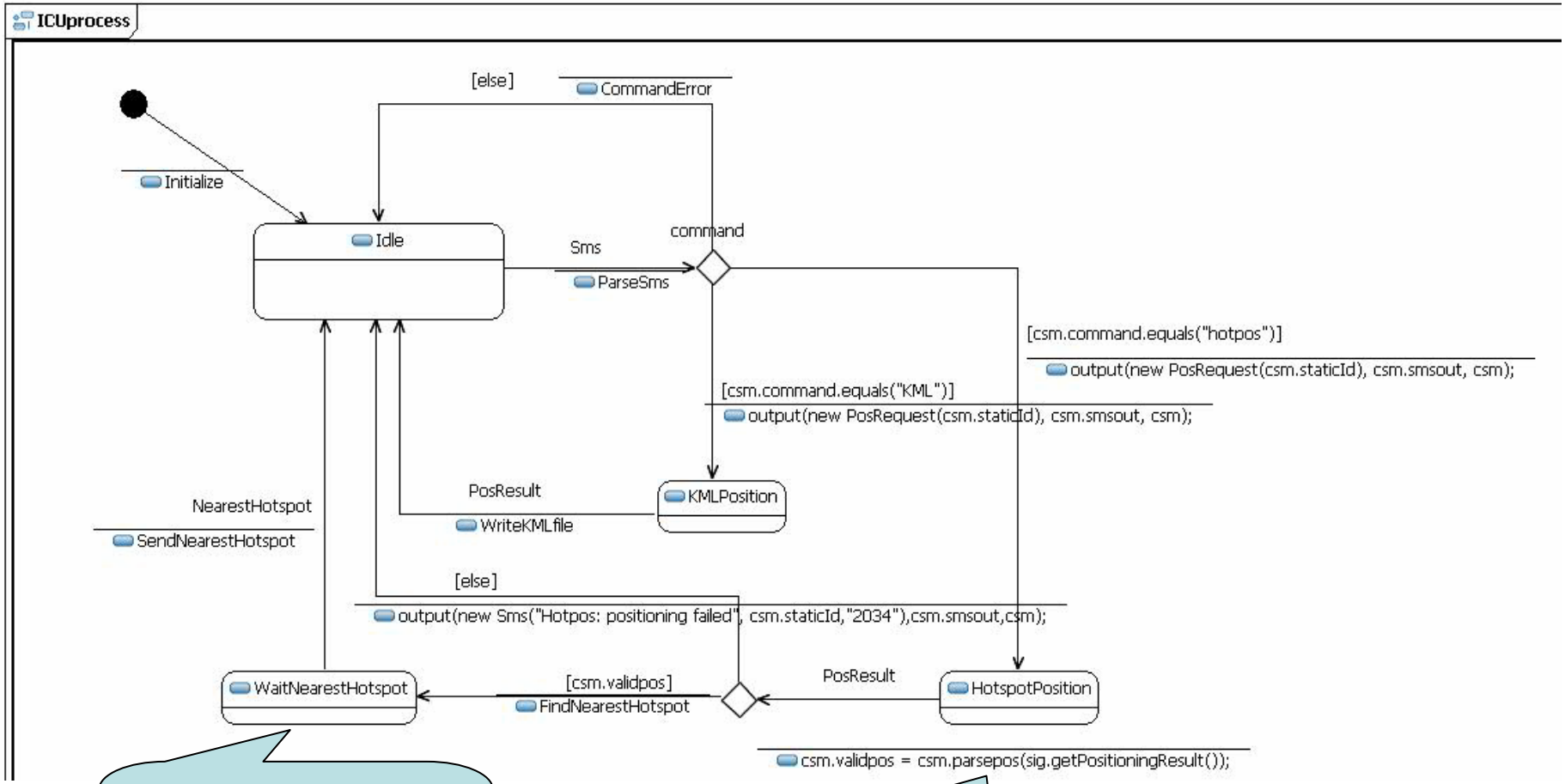
The essence of decomposition



The classes and signals



ICUprocess revisited (when intro Archive)

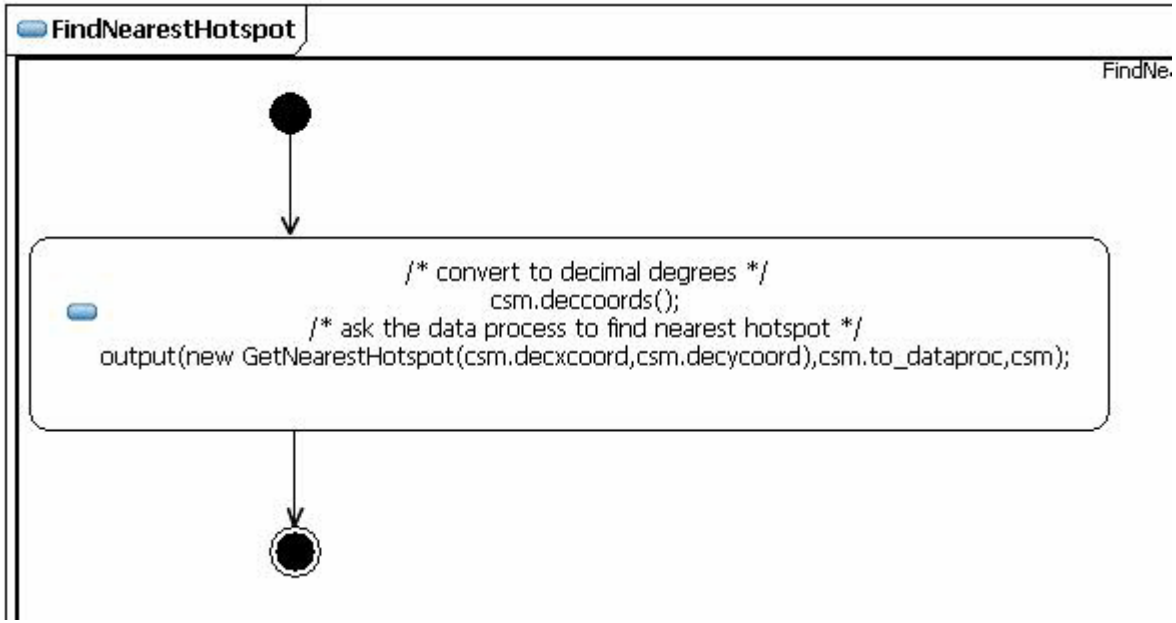


Here the data process does the calculations

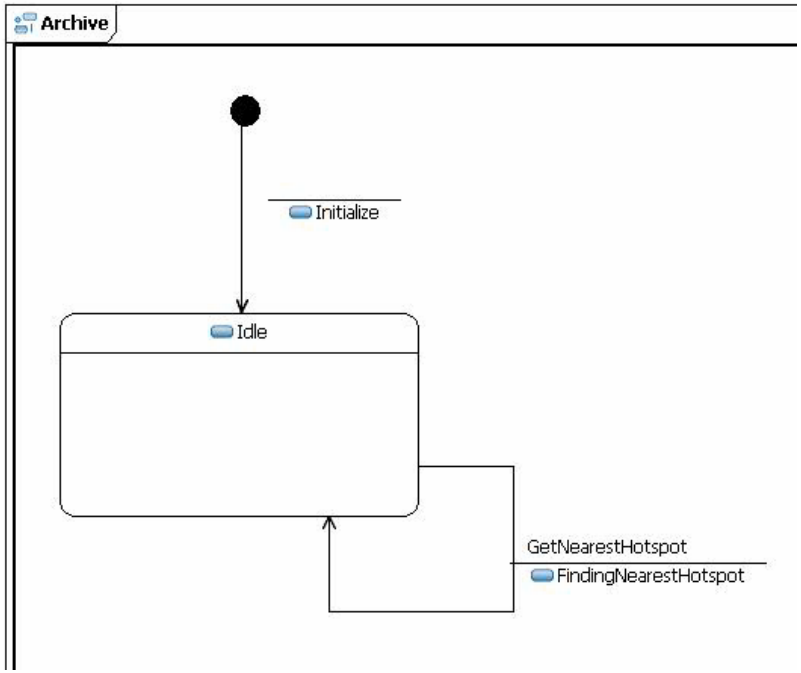
FindNearestHotspot has been split up

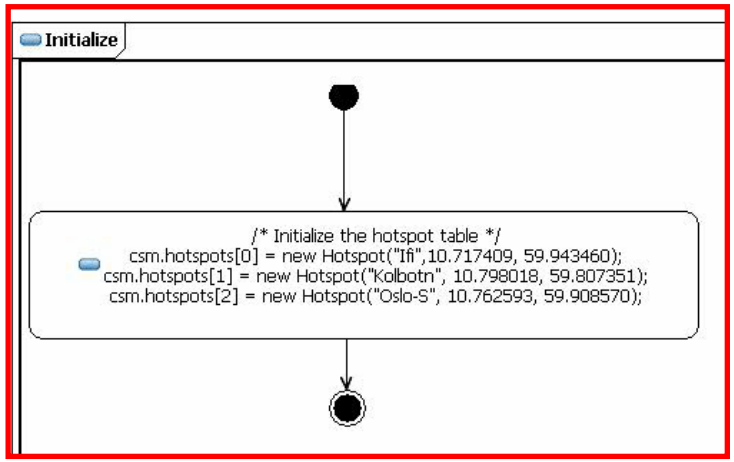
INF 5150

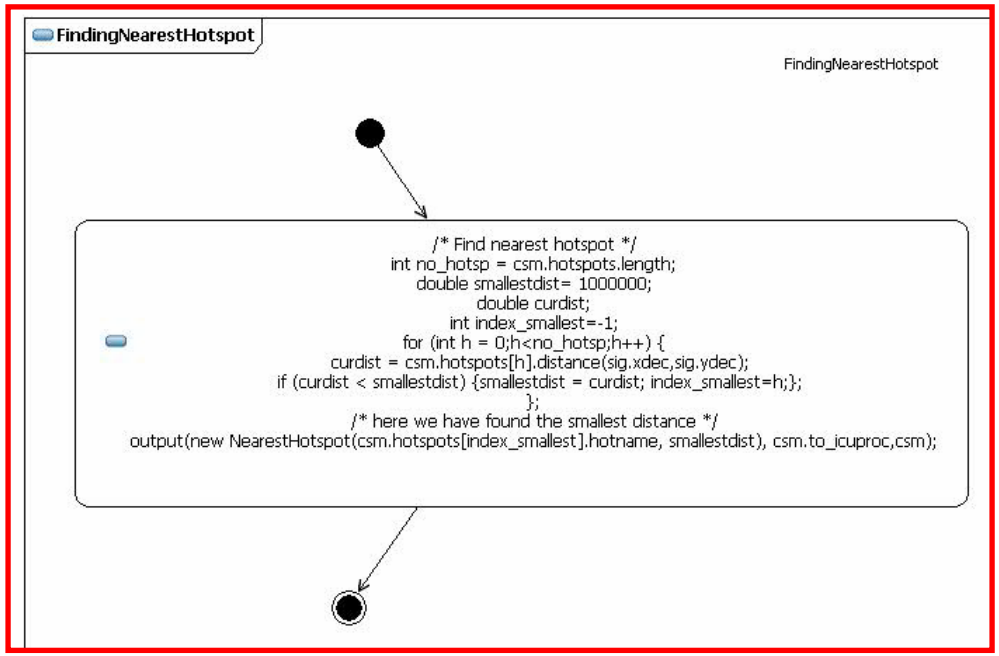
FindNearestHotspot has become pure sending



Archive – the data process







Buzzz 2: Why the Archive process?

- Pair up with another student
- Discuss 3 minutes what benefits there are with introducing the Archive process

Why the separate data process?

- Isolate the work on the (semi-)persistent data
  - we shall later show how the handling of data can change without changing its interfaces
- Provide a simple critical region
  - this will be clearer later when we interface to a database system that works concurrently with our system
- The Archive process and the ICUprocess can be designed by different persons

How to make the protocol with the Archive?

- Signals close to the application
  - this is what we have chosen
  - we want to branch on signals rather than on data
- Signals close to data
  - such as e.g. SQL
  - most important information will be in the parameters and branching will be on decision-nodes
- Do not worry about many signal types!