

# Coloured Petri Nets in UML-Based SW Development – Designing Middleware for Pervasive Healthcare

Jens Bæk Jørgensen

Centre for Pervasive Computing  
Department of Computer Science  
University of Aarhus

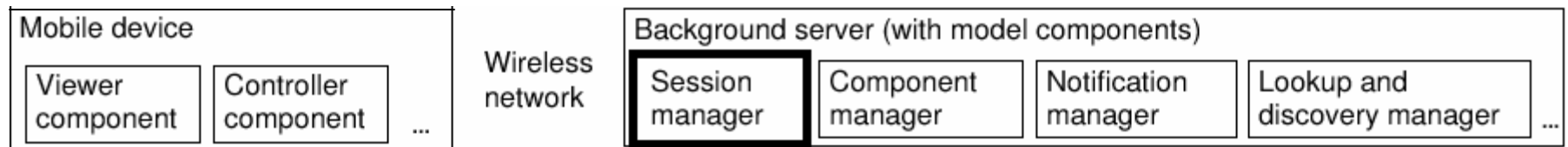
# Challenging UML

- UML is:
  - The “lingua-franca” of the software industry.
  - Well-supported by tools, literature, and commercial interests.
  - Good to describe structural aspects.
  - Rather limited to describe behaviour.
- Our purpose:
  - Pinpoint weaknesses in UML.
  - Propose CPN as a remedy.

# Healthcare IT

- Core hospital IT system: The electronic patient record (EPR).
- Access only from stationary PC's a problem.
- Pervasive healthcare: Computers “everywhere”, pervasive computing.
- Pervasive healthcare example: Medicine administration.

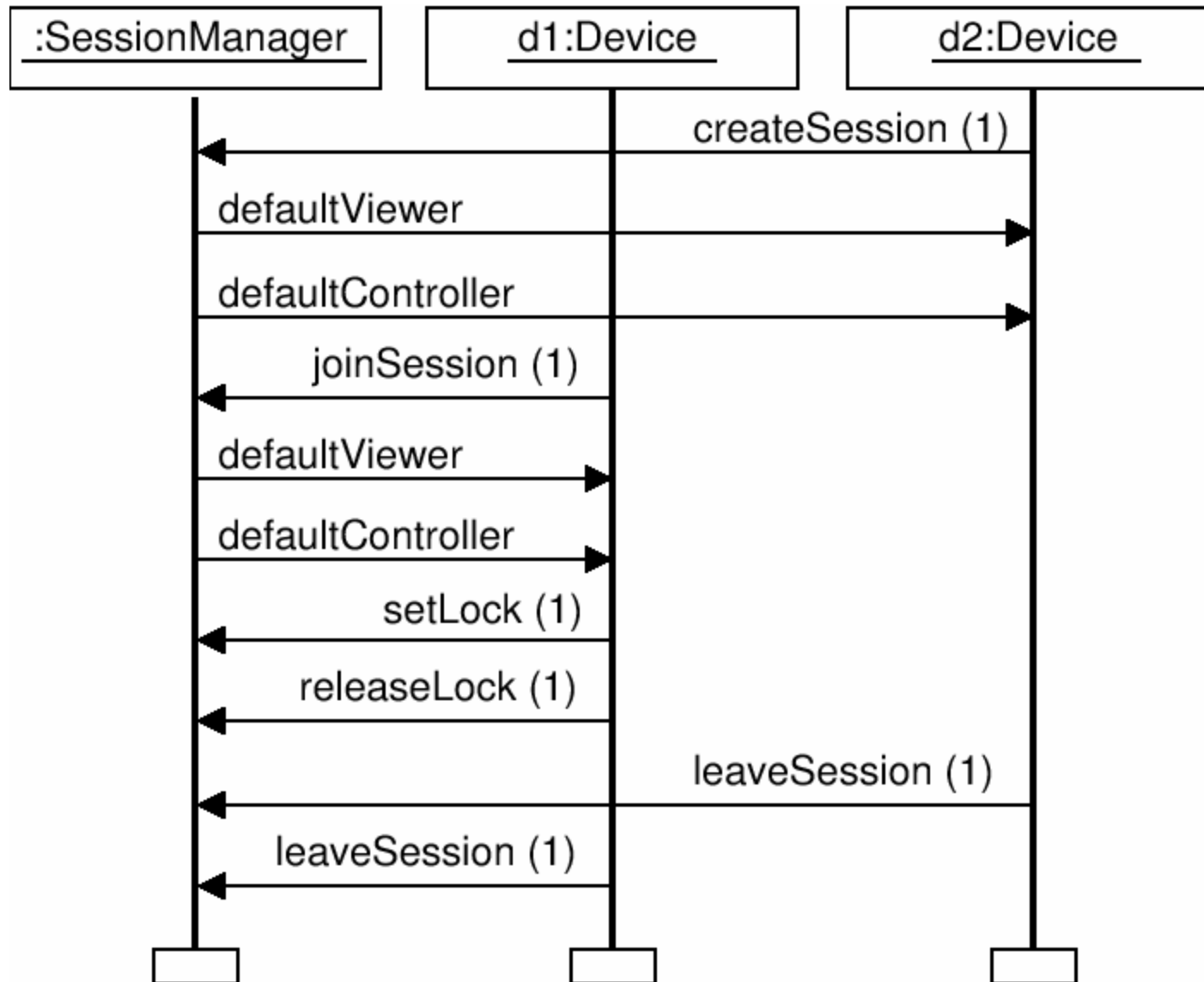
# The Pervasive Healthcare Middleware



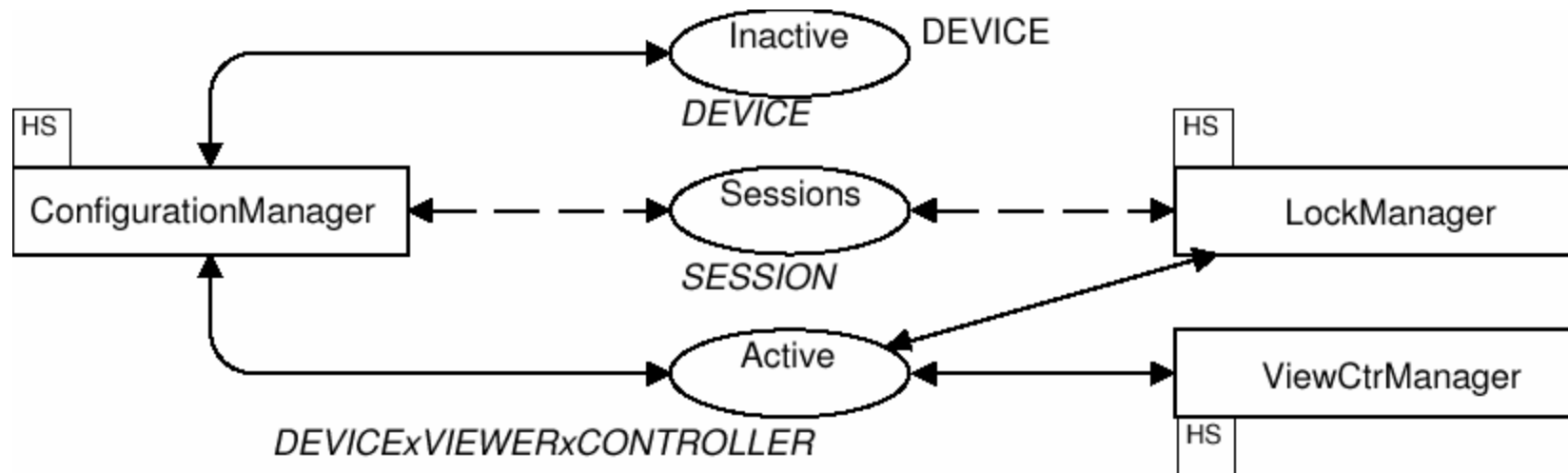
# The Session Manager

- Session: A number of devices joined together, sharing data, and communicating.
- Session management involves:
  - Configuration management.
  - Lock management.
  - Viewer/controller management.

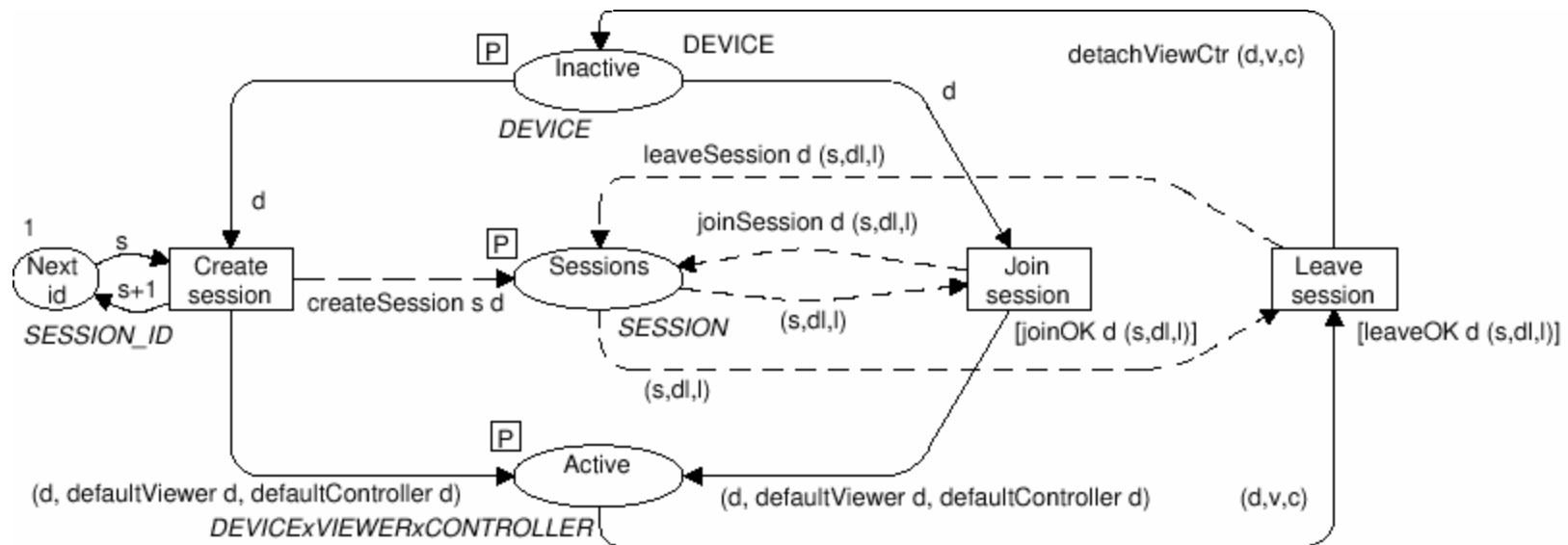
# Session Manager Behaviour



# Session Manager CPN Model – Top-level



# Session Manager CPN Model – Configuration Management



```
(* Configuration management functions *)
fun createSession: SESSION_ID -> DEVICE -> SESSION
fun joinOK: DEVICE -> SESSION -> bool
fun joinSession: DEVICE -> SESSION -> SESSION
fun leaveOK: DEVICE -> SESSION -> bool
fun leaveSession: DEVICE -> SESSION -> SESSION
fun defaultViewer: DEVICE -> VIEWER
fun defaultController: DEVICE -> CONTROLLER
fun detachViewCtr: DEVICE * VIEWER * CONTROLLER -> DEVICE
```

# UML – The Unified Modeling Language

- Booch, Jacobson, Rumbaugh; OMG (1997).
- Nine kinds of diagrams, e.g.:
  - Class diagrams.
  - Sequence diagrams.
  - Activity diagrams.
  - State machines.
- Tools, e.g., from Rational and I-Logix.

# Session Management UML Class Diagram



# Problems in UML Behavioural Modelling (Solved with CPN)

- Executable models:
  - UML lacks a formal execution semantics.
- Modelling of dependencies:
  - Dependencies between session manager main functional areas difficult and awkward to model in UML (as communicating state machines).
- Modelling of bookkeeping:
  - Session management bookkeeping records cannot be properly modelled in UML; state concept insufficient.

# UML and CPN in SW Industry

- UML:
  - In wide-spread use.
  - Static and dynamic properties.
  - Good for traditional administrative systems.
  - Automatic code generation from class diagrams (and more).
  - No formal execution semantics.
- CPN:
  - In limited use.
  - Dynamic properties.
  - Perhaps not needed for traditional administrative systems.
  - Automatic code generation difficult.
  - Well-founded formal execution semantics.

# Conclusions

- Much research has focused on building object-oriented concepts into Petri nets.
- For wide-spread use in sw industry, do the opposite:
  - The structure (the class diagram) comes first.
  - Use CPN to describe behaviour and interaction of objects from complex classes.
- Consider CPN a supplement to UML.