On-Demand Component Deployment in the UPnP Device Architecture

Didier DONSEZ
University Joseph Fourier
IMAG - LSR –ADELE Team
Grenoble, France

didier.donsez@imag.fr, didier.donsez@ieee.org

Outline

- Our context : the connected SOHO
- Motivations
- Our component model
- Trading algorithm
- Examples
- Implementation details
- Conclusion & Perspectives

Our Context: The connected SOHO

Nodes in a SOHO network

Device

- provide services
  - set/get state variables
  - execute actions
- notify state changing

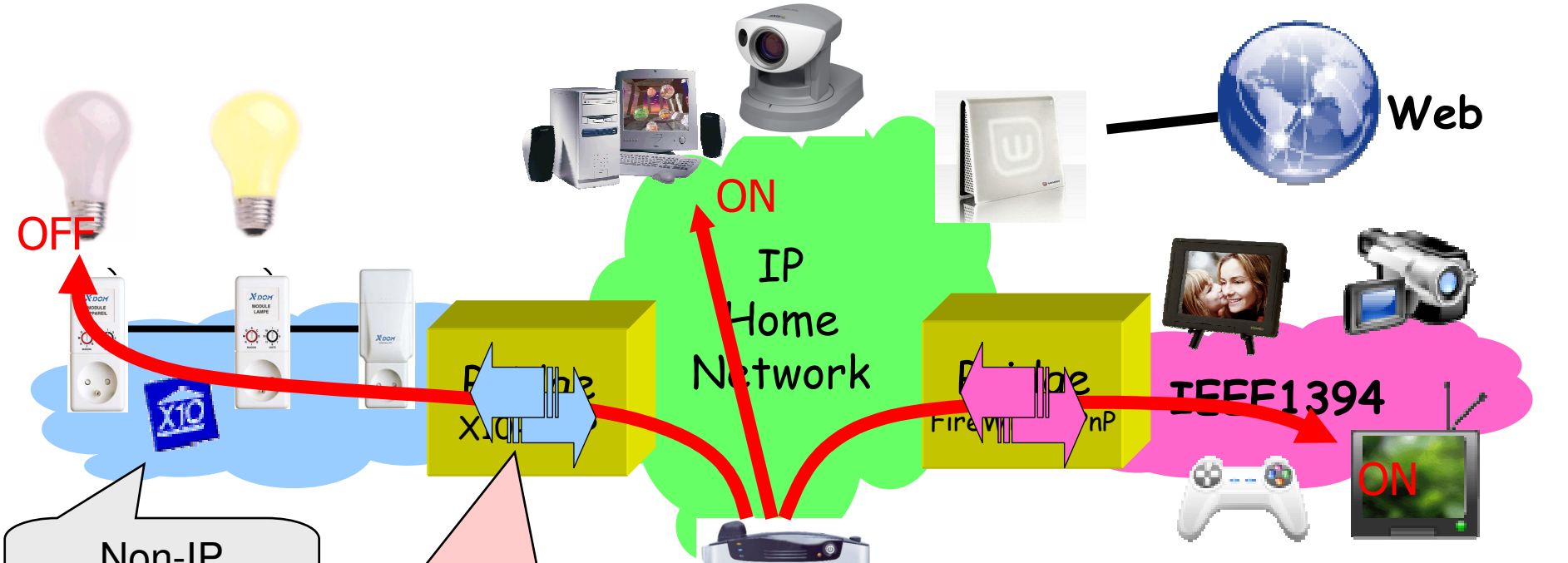
Control point

- invoke actions
- react to notified state changes

Motivations #1

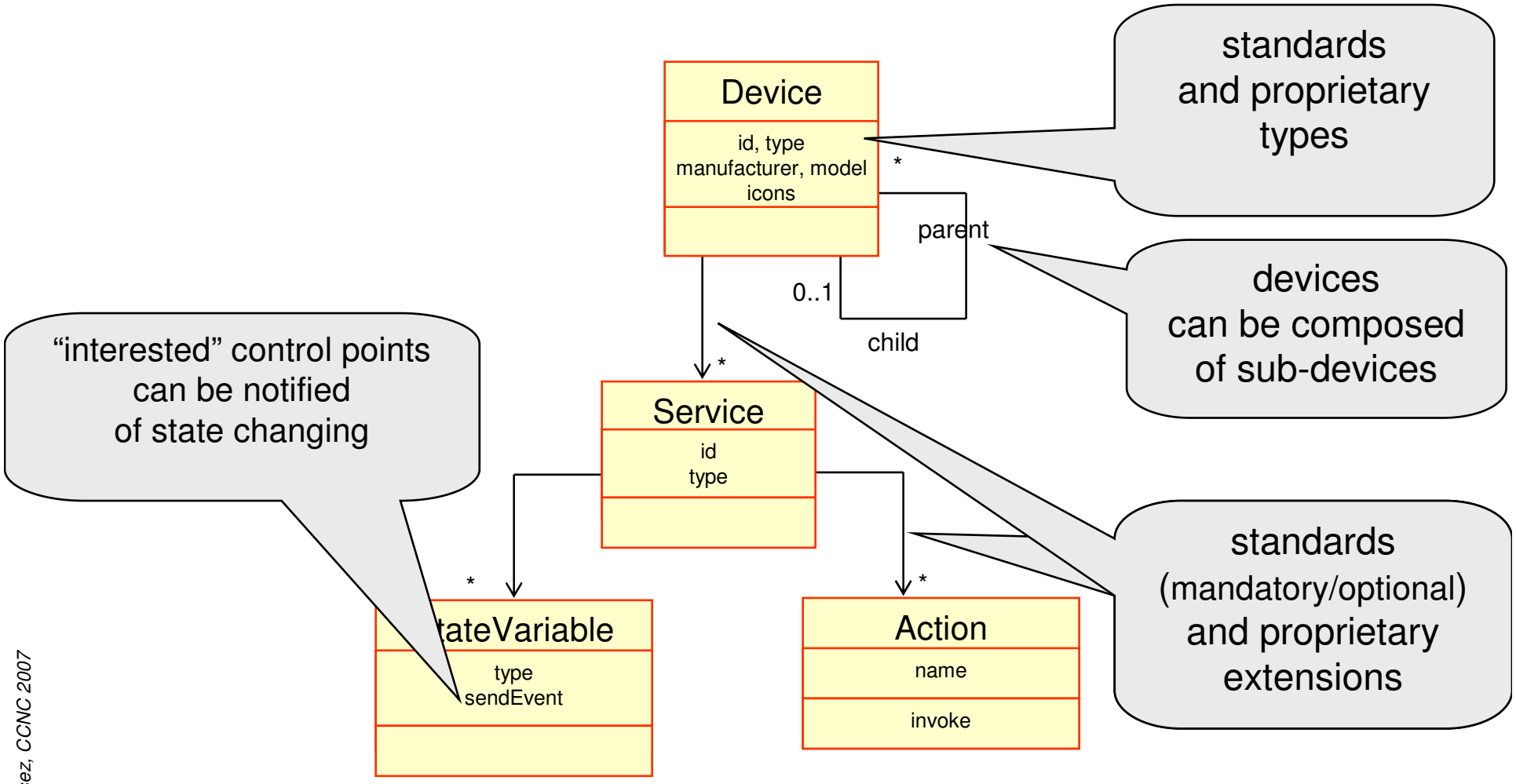
- *Small Office Home Office* (SOHO)
  - Proliferation of networked (home) appliances
  - Proliferation of their controllers
- "One for all" remote controller(s)

Motivation #2
A SOHO network of networks



Same Problem:
Find and install
the program
on this bridge
if it exists !

Case study
the UPnP Device metamodel

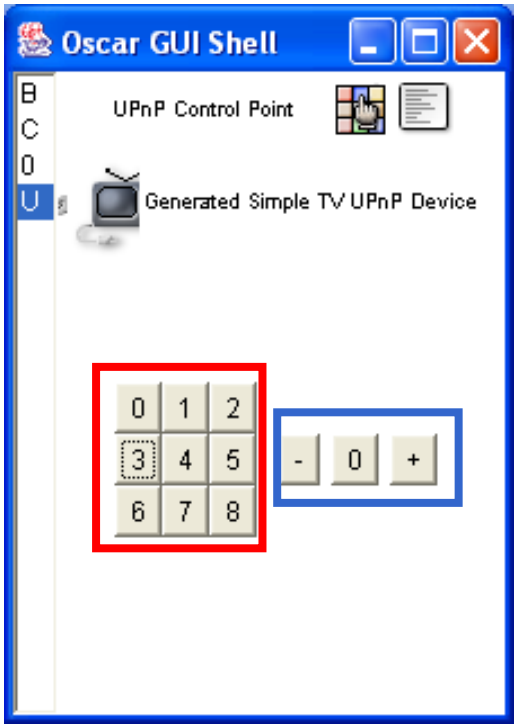


Our proposition
A component model for SOHO CP and bridge

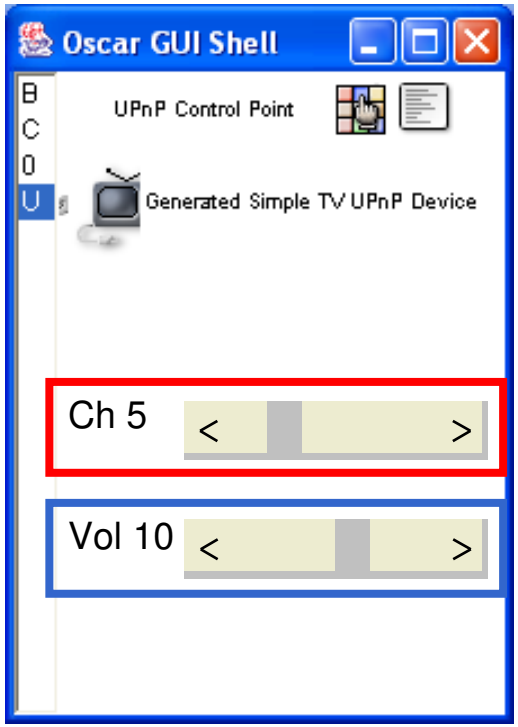
- Isomorphic with the UPnP device metamodel
- Component types
  - *Controllet* for control points
  - *Bridglet* for bridges
- Common features
  - Hierarchical composition
  - Dedicated or generic
  - Deployment on device detection
  - Component trading according to
    device properties (brand, model, type …)
    and execution environment constraints (GUI Tk …)

Dedicated vs Generic components

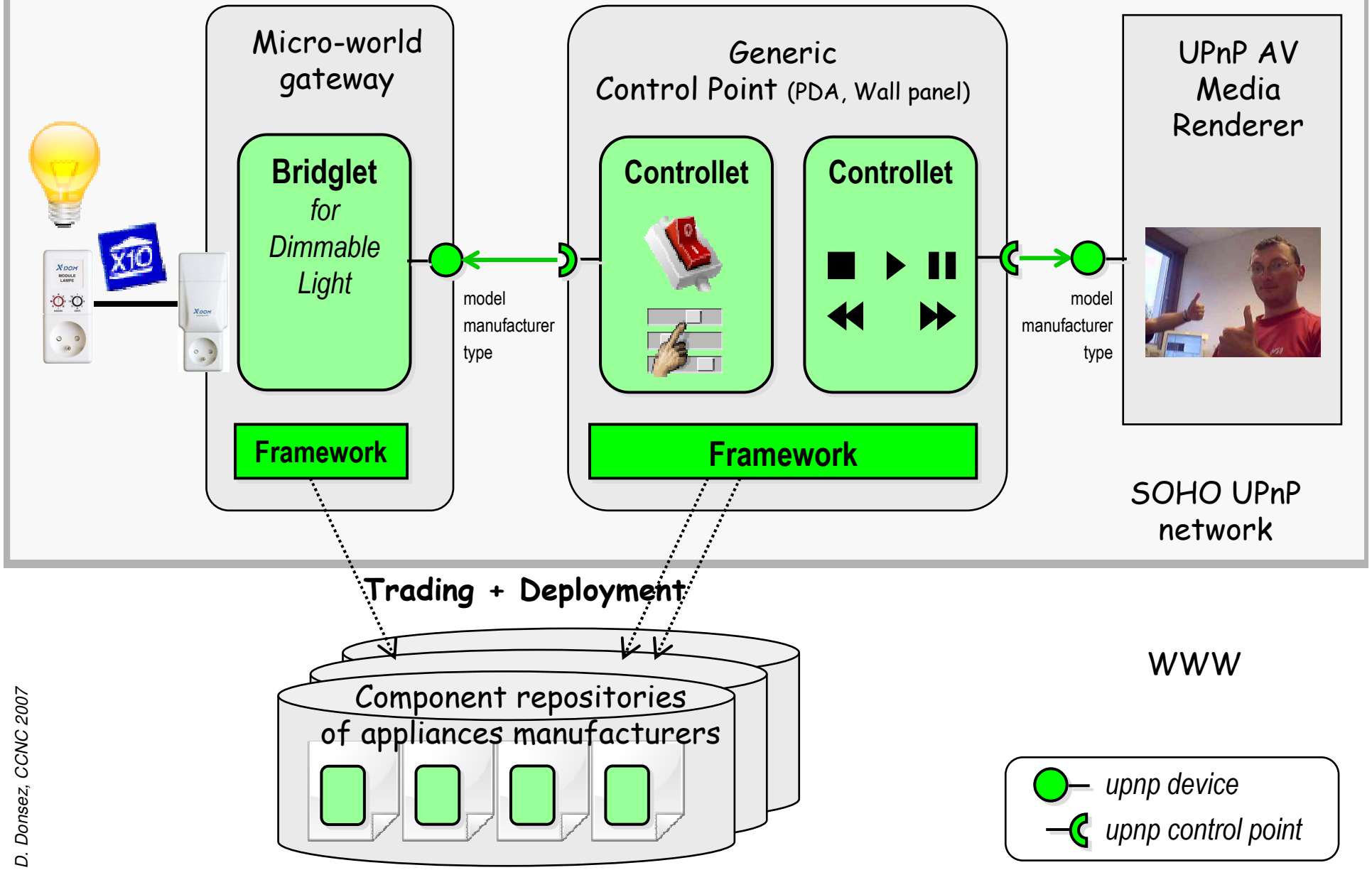
- Dedicated
  - to a device id
  - to a device model
  - to a device type
  - to a service type



- Generic
  - defined for a datatype
    - min/max, allowed value, …



Trading and deployment



Trading algorithm

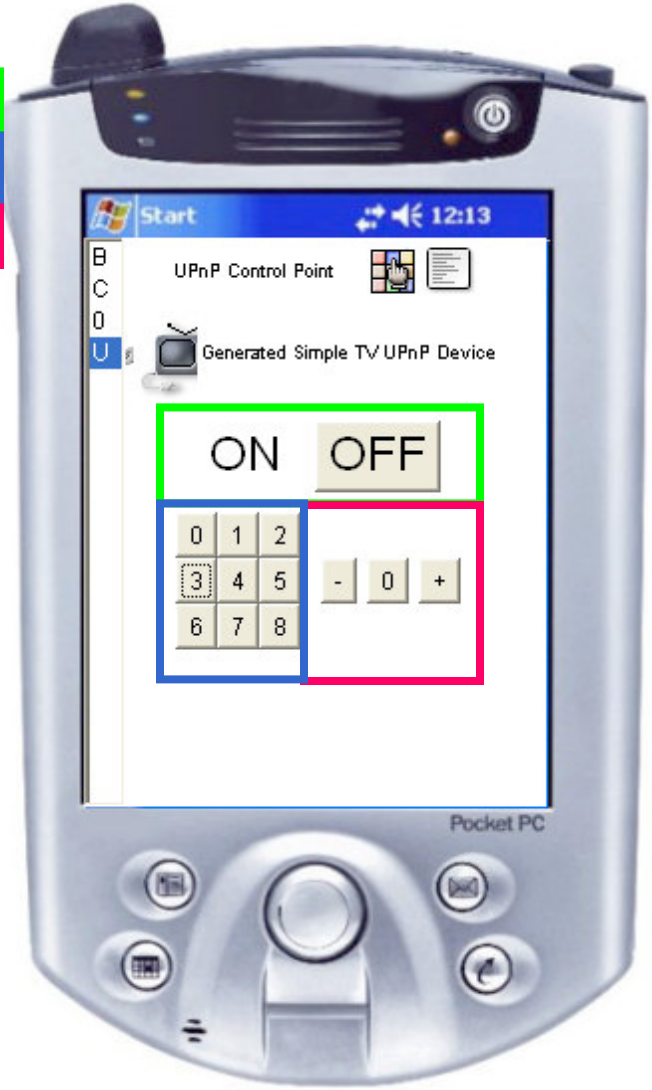
- Trig on device detection
- Lookup for a component representing a device
  - Refinement [id] → [manufacturer+model] → [type]
  - Lookup for sub-components of embedded devices
- Else for each service of the device
  - Lookup for a component representing a service
    - Refinement [id] → [type]
- If not found
  use a generic component using datatypes

Trading algorithm : Example

urn:schemas-upnp-org:service:**SwitchPower**:1

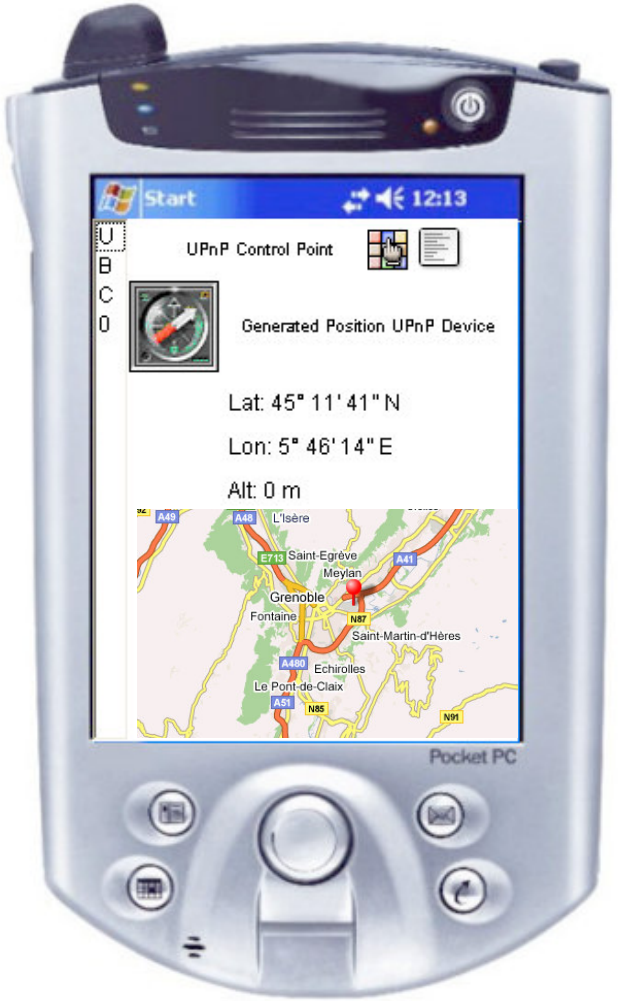
urn:*schemas-adele-imag-fr*:service:**ChannelSelector**:1

urn:*schemas-adele-imag-fr*:service:**VolumeSelector**:1



A generic control point for PDA and controllets

More controllets



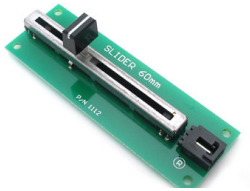
Implementation details

- Controllets
  - Java-based (J2ME/CDC/PBP)
  - Prototype limited to the AWT for the moment (eSWT, MIDP, …)
- OSGi technology
  - UPnP Base Driver
  - Dynamic Deployment (install, activate, update, uninstall)
- Deployment with OBR v1
  + extra metadata
    describing the [id],[model],[type] supported by the controllet/brigdlet
- Extra : S*tubs* and Skeleton generator
  - Several real devices
    - WebCam USB, GPS receiver, Thermocron iButton, RFID reader, Nabaztag, *X10 switch* …

Conclusion and Perspectives

- Emerging problem
  - MicroSoft Vista Sideshow
  - DPWS (Device Profile for Web Services)

- Extensions
  - Context-aware deployment
  - User-preferences aware deployment
  - Stateful components (user session history)
  - Model-Driven Engineering (MDE)



- Physical control point association
  - *phydgetlet* for *phydget* (dimmer, …) [Greenberg2001&2002]

Q & A

Visit us
at the
demonstration session