Towards Real World Object Orientation

Paul Holleis  
paul@hcilab.org

Albrecht Schmidt  
albrecht@hcilab.org

Research Group Embedded Interaction  
University of Munich, Germany
“Definitions”

- **Real World Objects**
  All sorts of tangible, not virtual devices
  Display, PDA, TV, Phone, Ball

- **Object Orientation**
  See Software Engineering
  Objects as instances of abstractions
  (classes, interfaces)
  Properties and Capabilities
Problem Statement

• Many devices / technologies exist (Particles, Smart-Its, PocketPC, Symbian)

• Different ways of programming for each (high level, low level, hard level)

• Different communication as well

• Hardly any way of combining several of them (neatly)
Example Problem

- Knob based on Pin&Play (N. Villar et al.)
- Java programmable cell phone with display

- Should be easy:
  Get informed on phone when knob changed
  - Technology is available
  - How to connect?
Proposed Solution

• Use existing programming language and environment

• Treat real world objects as common programming objects

• Provide the environment with capabilities to use such objects in an obvious way
Example Solution

Eclipse Plug-in
  – sense new devices
  – retrieve interface
  – create proxy classes
  – show and alter properties

```java
void main() {
    DISPLAY_1.setText(
        KNOB_1.getCurrentValue()
    );
}
```

```
<table>
<thead>
<tr>
<th>ID:</th>
<th>KNOB_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Value:</td>
<td>27</td>
</tr>
<tr>
<td>Step:</td>
<td>1</td>
</tr>
<tr>
<td>Minimum:</td>
<td>0</td>
</tr>
<tr>
<td>Maximum:</td>
<td>100</td>
</tr>
</tbody>
</table>
```
Some Open Questions?

- Categorisation of devices?
- Event based mechanism?
- Is there an object / class hierarchy / inheritance?
- What about abstraction, polymorphism?

- Deploy interfaces on central server / web page associated to each device?
- IDL, suitable interface description language?
- Interface documentation?