Smart Applications for the Maintenance of Large Buildings: How to Achieve Ontology-based Interoperability at the Information Level

Alfredo D’Elia
Riccione – June 22, 2010

Smart Environment Objectives

• Connect physical world with information world

• Cross-industry interoperability

• Maintain value of existing legacy

• Innovative user interaction with embedded devices
Ontologies

- Built on existing standards
- Specify information semantics
- Machine interpretable
- Support reasoning
- Extendible

Once a certain domain has been specified using an ontology, software agents based on the same ontology are interoperable and so may communicate and cooperate.
Generic Approach to Interoperability

- Scenario analysis and application requirements
- Ontology as a formal representation of the detected concepts and properties
- HW/SW architecture
- Implementation and testing

Maintenance Domain

- Sensor networks provide data about environment
- Rules and normatives specify limits over which fault are detected
- Faults are signaled and communicated to maintenance companies and operators
- Maintenance interventions are scheduled and performed
- Office tenants are informed of faults and intervention which happens in their office to have a less intrusive intervention
- Maintenance operators are guided to fault location and signal start and end of operation
ARTEMIS JU SP3 / 100017: Smart Objects For Intelligent Applications

Building Maintenance Scenario

WSN

Contextualized Observations

Normatives Business rules Aggregators

Contextualized Observations

Contextualized Faults and corrective interventions

Knowledge Base

corrective interventions details

corrective interventions details

Maintenance company and operator

Sensor Network adaptation

• Dolce: high abstraction level entities
• Sensor Ontology: entities related to sensors and data

Characteristic ≡ Dolce:Quality
MeasureRegion ≡ Dolce:region
DataValue ≡ Dolce:Quale
Faults and Corrective interventions

Fault definition from EN 13306:
state of an item (e.g., an A/C unit) or a building element (e.g., the HVAC system) characterized by its inability to perform a required function”

Maintenance: software infrastructure
Building monitoring GUI

<table>
<thead>
<tr>
<th>Place</th>
<th>Fault</th>
<th>Urgency</th>
<th>Parameter</th>
<th>Value</th>
<th>Measurement unit</th>
<th>Detection time</th>
<th>start time</th>
<th>stop time</th>
<th>Operator</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Room2</td>
<td>anomalous-water-presence</td>
<td>high</td>
<td>Water presence</td>
<td>YES</td>
<td>YES/NO</td>
<td>2010-06-18 11:44:22</td>
<td>2010-05-20 14:30:10</td>
<td>Maria Bianchi</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Room1</td>
<td>Humidity-out-of-range</td>
<td>medium</td>
<td>Relative humidity</td>
<td>70</td>
<td>%</td>
<td>2010-05-20 22:12:55</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Real time indoor space parameters monitor

<table>
<thead>
<tr>
<th>Place</th>
<th>Temperature (°C)</th>
<th>Relative humidity (%)</th>
<th>Water on the floor (YES/NO)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Room1</td>
<td>18.5</td>
<td>70</td>
<td>NO</td>
</tr>
<tr>
<td>Room2</td>
<td>27.5</td>
<td>63</td>
<td>YES</td>
</tr>
</tbody>
</table>

Operator GUI

- Acceptation by only one of the pool of appropriately skilled operators
- Synchronization of multiple acceptations
- Fault info to help the operator
- Buttons to interact with the SIB
- Notification of all processes interested in the modified fault instances => Tenant GUI
Conclusions and future works

• Evaluation of ontology based approach in a real scenario

• Work with domain experts for conceptualization of the domain

• HW/SW architecture to support the desired behaviour

• Software implementation on multiple platforms

• Intrinsic extendibility and interoperability