How to Discover Sensors in the Sensor Web?

Simon Jirka
(52° North/Westfälische Wilhelms-Universität Münster)
Overview

• What is Sensor Discovery?
• Challenges of Sensor Discovery
• Sensor Discovery Architecture
• SensorML Discovery Profile and ebRIM mapping
• Implementations
  – Sensor Instance Registry
  – Catalogues
  – Sensor Observable Registry
• Outlook and Conclusion
What is Sensor Discovery?

- Two types of sensor discovery
  - Sensor instance discovery
  - Sensor service discovery
- Sensor instance discovery $\rightarrow$ finding specific physical sensing devices
- Sensor service discovery $\rightarrow$ finding SWE services that encapsulate certain sensors or sensor data
Challenges of Sensor Discovery

- Specific metadata formats → i.e. SensorML
  - How to extract the relevant information from a SensorML document?
  - What must be contained in a SensorML document? → Profiles
  - How to map from SWE encodings to catalogue information models?
  - How to interact with the different SWE service interfaces?
Challenges of Sensor Discovery

- Dynamic structure of sensor networks
  - How to handle continuously changing sensor metadata (e.g. mobile sensors)?
  - How to deal with sensors that are available through different SWE services? (potentially time dependent)
  - How to handle time dependent data availability?
Challenges of Sensor Discovery

- **Sensor Status**
  - How to integrate/use additional sensor status information (e.g. battery level)?

- **Semantics**
  - How to describe what a sensor measures?
  - How to use semantics for improving interoperable search mechanisms?
Challenges of Sensor Discovery

- Search Interface
  - How to design an interface for a sensor catalogue/registry?
  - How to align sensor discovery with the OGC Catalogue?
Architecture of the Discovery Framework
SensorML Discovery Profile

- Need for a common metadata encoding for sensor metadata
- SensorML is the relevant OGC standard for describing sensors
- Due to the generic character of SensorML a profile is needed that defines
  - a minimum set of metadata
  - a structure how to provide the minimum set of metadata
- Formal definition using Schematron
SensorML Discovery Profile
SensorML-ebRIM Mapping

• SensorML is not supported as a data model for OGC Catalogues
• Approach: Provide an according Catalogue extension
• Mapping of SensorML to the ebRIM Catalogue Information Model
• Definition of object types, associations, attributes
• OGC Discussion Paper
Sensor Instance Registry (SIR)

- Sensor Instance Registry (SIR)
  - Concept created within the EU funded FP6 project OSIRIS
  - Continued work within GENESIS
- Functionality
  - Managing sensor networks
  - Supervising the status of sensors
  - Discovering sensors and SWE services
SIR Sensor Instance Registry
Test Client - Version 0.1

sensor search: SearchSensor DescribeSensor
metadata handling: HarvestService InsertSensorInfo DeleteSensorInfo UpdateSensorDescription
status handling: GetSensorStatus InsertSensorStatus SubscribeSensorStatus Renew [...] Subscription Cancel [...] Subscription
catalog connection: ConnectToCatalog DisconnectFromCatalog
other: GetCapabilities Textbox-based Test Client SensorML to ebRIN Transformation Catalog Push Information

Harvest Service Request

Service URL: http://www.uni-muenster.de/mySOS
Service Type: SOS

Build request

Send request
SWE Catalogue

• The SIR possesses all information necessary for sensor/SWE discovery
• However, the amount of information within the SIR is too much to be published via a Catalogue
• Idea: Aggregate and generalize the information contained in the SIR and feed it into an OGC Catalogue
SWE Catalogue

• Lower level registry
  – Harvesting of sensor metadata
  – Management of sensor status data
• Metadata within the lower level registry too detailed for
  Catalogues → aggregation
• Conversion of sensor metadata to ebRIM → XSLT
• Lower level registry pushes (aggregated and) converted
  metadata into the Catalogue
• Catalogue based on the Buddata ebXML Registry/Repository
**SIR** Sensor Instance Registry  
**Test Client - Version 0.1**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor search</td>
<td>SearchSensor, DescribeSensor</td>
</tr>
<tr>
<td>Metadata handling</td>
<td>HarvestService, InsertSensorInfo, DeleteSensorInfo, UpdateSensorDescription</td>
</tr>
<tr>
<td>Status handling</td>
<td>GetSensorStatus, InsertSensorStatus, SubscribeSensorStatus, Renew/暂停Subscription, Cancel/取消Subscription</td>
</tr>
<tr>
<td>Catalog connection</td>
<td>ConnectToCatalog, DisconnectFromCatalog</td>
</tr>
<tr>
<td>Other</td>
<td>GetCapabilities, Textbox-based TestClient, SensorML to ebXML Transformation, Catalog Push Information</td>
</tr>
</tbody>
</table>

**ConnectToCatalogRequest**

Catalog URL: http://www.uni-muenster.de/myCatalogue  
Push Interval: 240 (seconds, '0' for single catalog connection)

```xml
<sir:CatalogURL>http://www.uni-muenster.de/myCatalogue</sir:CatalogURL>
</sir:ConnectToCatalogRequest>
```
Catalogue Link
Sensor Observable Registry (SOR)

• Need for handling semantics in the SWE context
  – Specify the phenomena that are observed by a sensor
  – Handling phenomenon definitions

• Two requirements:
  – Access the descriptions of phenomena identified by certain URNs
  – Enhancing the sensor discovery process by exploring and investigating the semantics of observed phenomena
Get Matching Definitions Request - Mozilla Firefox

SOR Sensor Observable Registry
Test Client - Version 0.3

Get Matching Definitions Request:

Input URI:

Matching Type: SUPER_TYPE
Search Depth: 1

Build request

Send request

FOSS4G 2010
Sensor Observable Registry (SOR)

User

Sensor Registry/Catalogue

SearchSensors(searchParameters)

GetMatchingDefinitions(URN, MatchingType, Depth)

Matching Definition URNs()

Sensor Records()
Outlook and Conclusion

• Prototypes available as Open Source Software
  – 52° North Sensor Instance Registry
  – 52° North Sensor Observable Registry
  – Buddata ebXML Registry/Repository

• Ongoing specification process
  – SensorML Discovery Profile
  – SensorML-ebRIM Mapping

• Work will be continued: EO2HEAVEN

• Closing one of the last gaps for fully integrating SWE into SDIs
Thank you for your attention!

More information:

http://sensorweb.uni-muenster.de
http://52north.org/swe
jirka@52north.org