



**ifgi**  
Institut für Geoinformatik  
Universität Münster



# 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 → finding specific physical sensing devices
- Sensor service discovery → 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?

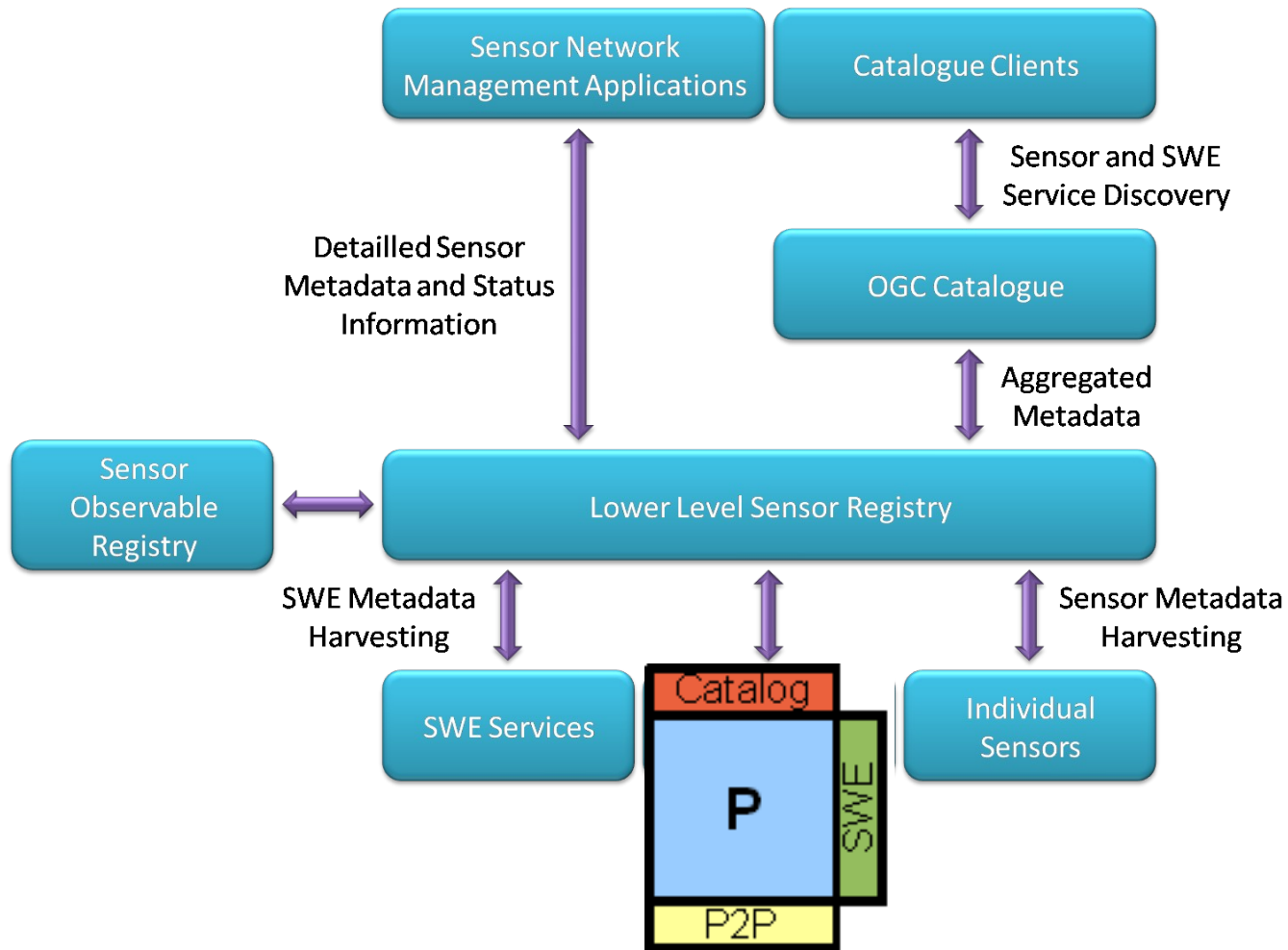
# Challenges of Sensor Discovery

- **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

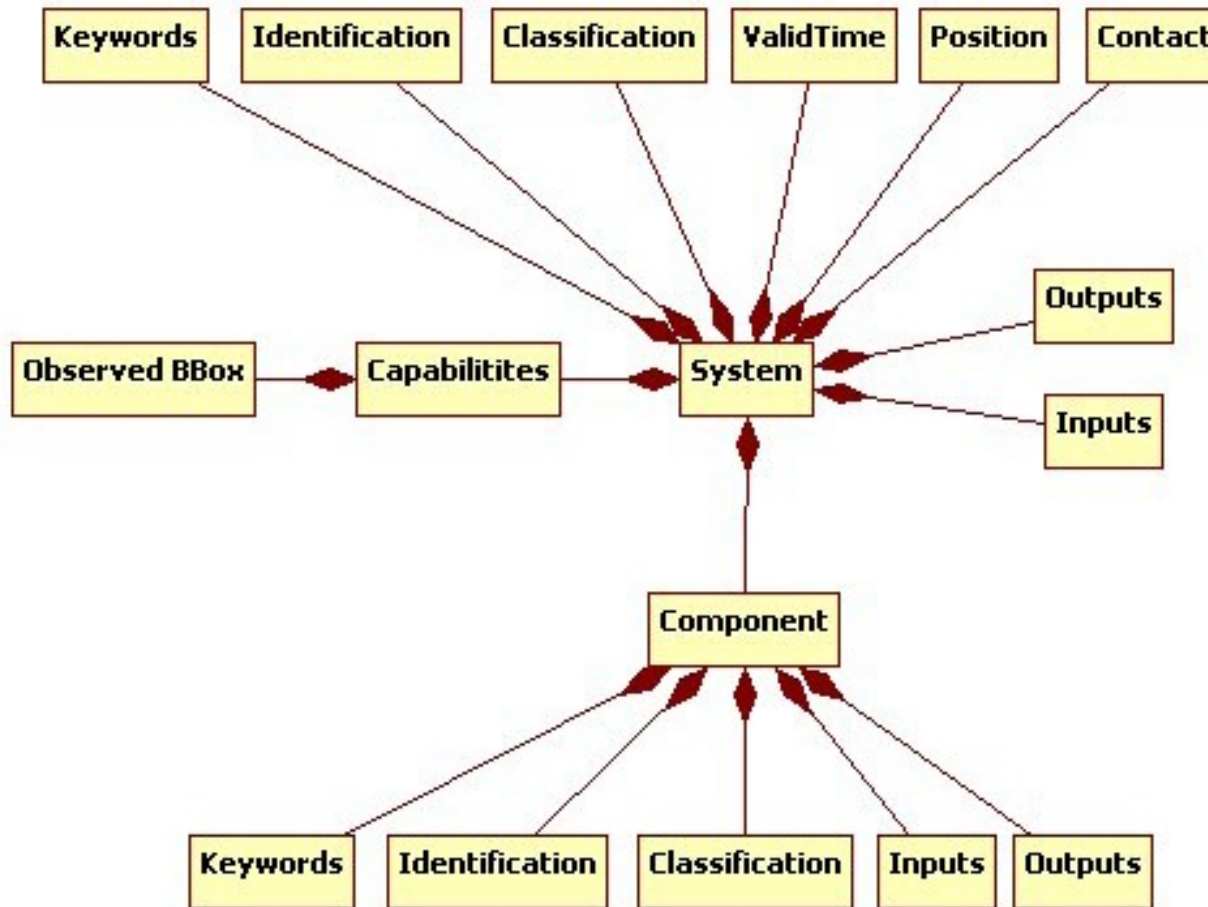




# SensorML Discovery Profile

- 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



Harvest Service Request - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

http://giv-genesis.uni-muenster.de:8080/SIR2/pages/harvestService.jsp

Suchen Freigeben Sidewiki Lesezeichen Rechtschreibprüfung Übersetzen AutoFill Anmelden

Harvest Service Request

**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 ebRIM Transformation](#) [Catalog Push Information](#)

**Harvest Service Request**

Service URL

Service Type:

Build request

```

1 <sir:HarvestServiceRequest xmlns:sir="http://sws1.uni-muenster.de/sir"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" service="SIR" version="0.3.0"
  xsi:schemaLocation="http://sws1.uni-muenster.de/sir http://giv-genesis/schemas
/sir/sirAll.xsd http://www.opengis.net/sensorML/1.0.1 http://schemas.opengis.net/sensorML
/1.0.1/sensorML.xsd">
2   <sir:ServiceURL>http://www.uni-muenster.de/mySOS</sir:ServiceURL>

```

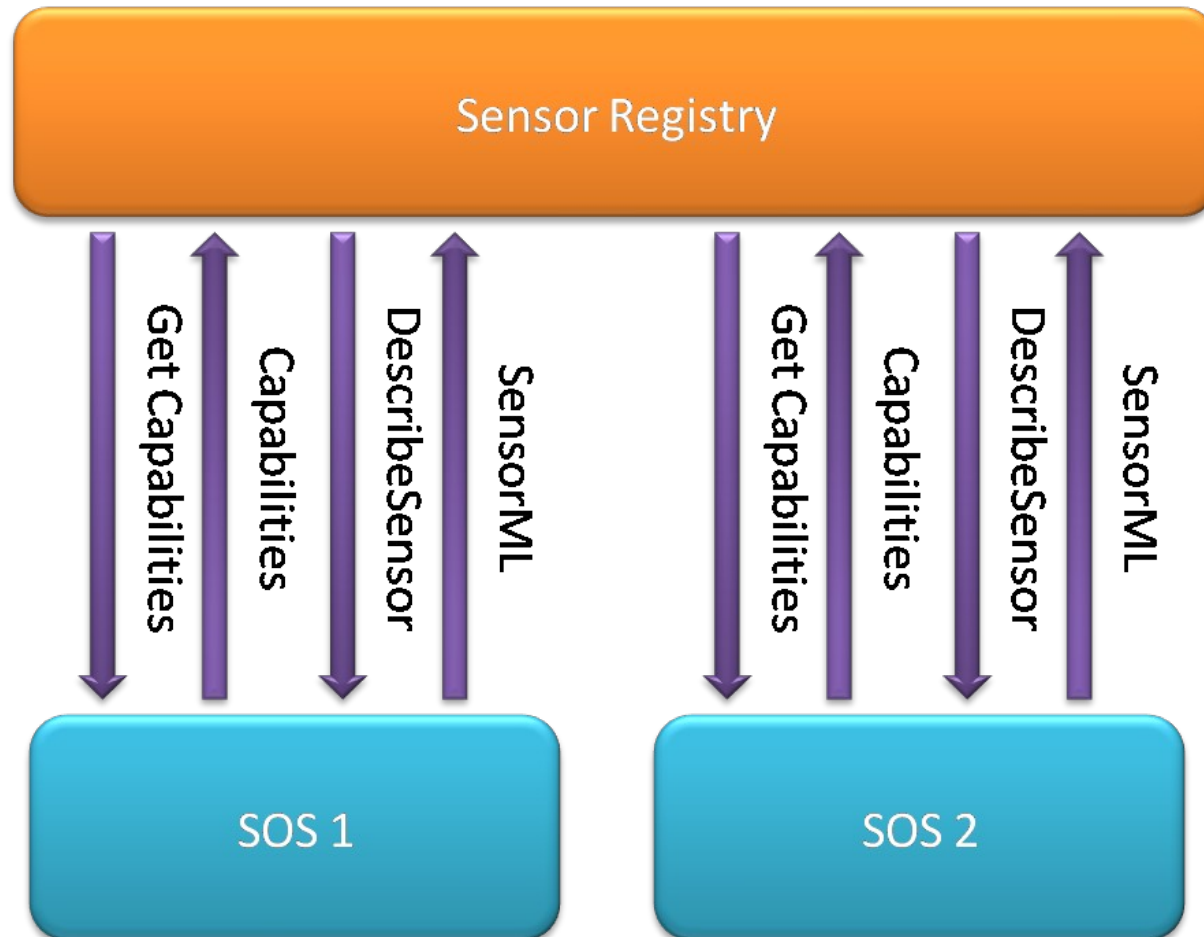
Send request

```

1
2
3
4
5
6
7

```

# Metadata Harvesting: SOS



# 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





Connect To Catalog Request - Mozilla Firefox

http://giv-genesis.uni-muenster.de:8080/SIR2/pages/connectToCatalog.jsp

Suchen Freigeben Sidewiki Lesezeichen Rechtschreibprüfung Übersetzen AutoFill Anmelden

**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 ebRIM Transformation, Catalog Push Information

### ConnectToCatalogRequest

Catalog URL:   
 Push Interval:  (seconds, '0' for single catalog connection)

Build request

```

1 <sir:ConnectToCatalogRequest xmlns:sir="http://sws1.uni-muenster.de/sir"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" service="SIR" version="0.3.0"
  xsi:schemaLocation="http://sws1.uni-muenster.de/sir http://giv-genesis/schemas
/sir/sirAll.xsd http://www.opengis.net/sensorML/1.0.1 http://schemas.opengis.net/sensorML
/1.0.1/sensorML.xsd">
2   <sir:CatalogURL>http://www.uni-muenster.de/myCatalogue</sir:CatalogURL>

```

Send request

```

1
2
3
4
5
6

```

# Catalogue Link

**Input**

Service : Sensor Catalogue

Sensor:

Start Date:

End Date:

Keywords:

Service:

Phenomenon:

**Map Viewer**

**Search Results**

Loaded 1 records out of 1

	Name ↕	Date ↕	Abstract	Format	File Name
<input type="checkbox"/>	ID <a href="#">Show Metadata</a> Name Weather station 123 on top of the IfGI building	2010-01-15 2010-01-30	Weather station located on the roof of the Inisitute for Geoinformatics of the University Münster, Germany.		

# 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



SOR Sensor Observable Registry  
Test Client - Version 0.3



sor operations	GetCapabilities	GetDefinitionURIs	GetDefinition	GetMatchingDefinitions
definition handling	InsertDefinition	DeleteDefinition		
other	Textbox-based Test Client	RESTful Web Service		

### Get Matching Definitions Request:

Input URI:

```
urn:ogc:def:phenomenon:OGC::Concentration[CO]
urn:ogc:def:phenomenon:OGC::Concentration[NO]
urn:ogc:def:phenomenon:OGC::Concentration[NO2]
urn:ogc:def:phenomenon:OGC::Concentration[O3]
urn:ogc:def:phenomenon:OGC::Concentration[S02]
urn:ogc:def:phenomenon:OGC::Concentration[Toluene]
urn:ogc:def:phenomenon:OGC::Concentration[Benzene]
urn:ogc:def:phenomenon:OGC::Concentration[Xylene]
urn:ogc:def:phenomenon:OGC::Concentration[PM2_5]
urn:ogc:def:phenomenon:OGC::Concentration[PM10]
```

Matching Type: SUPER\_TYPE

Search Depth: 1

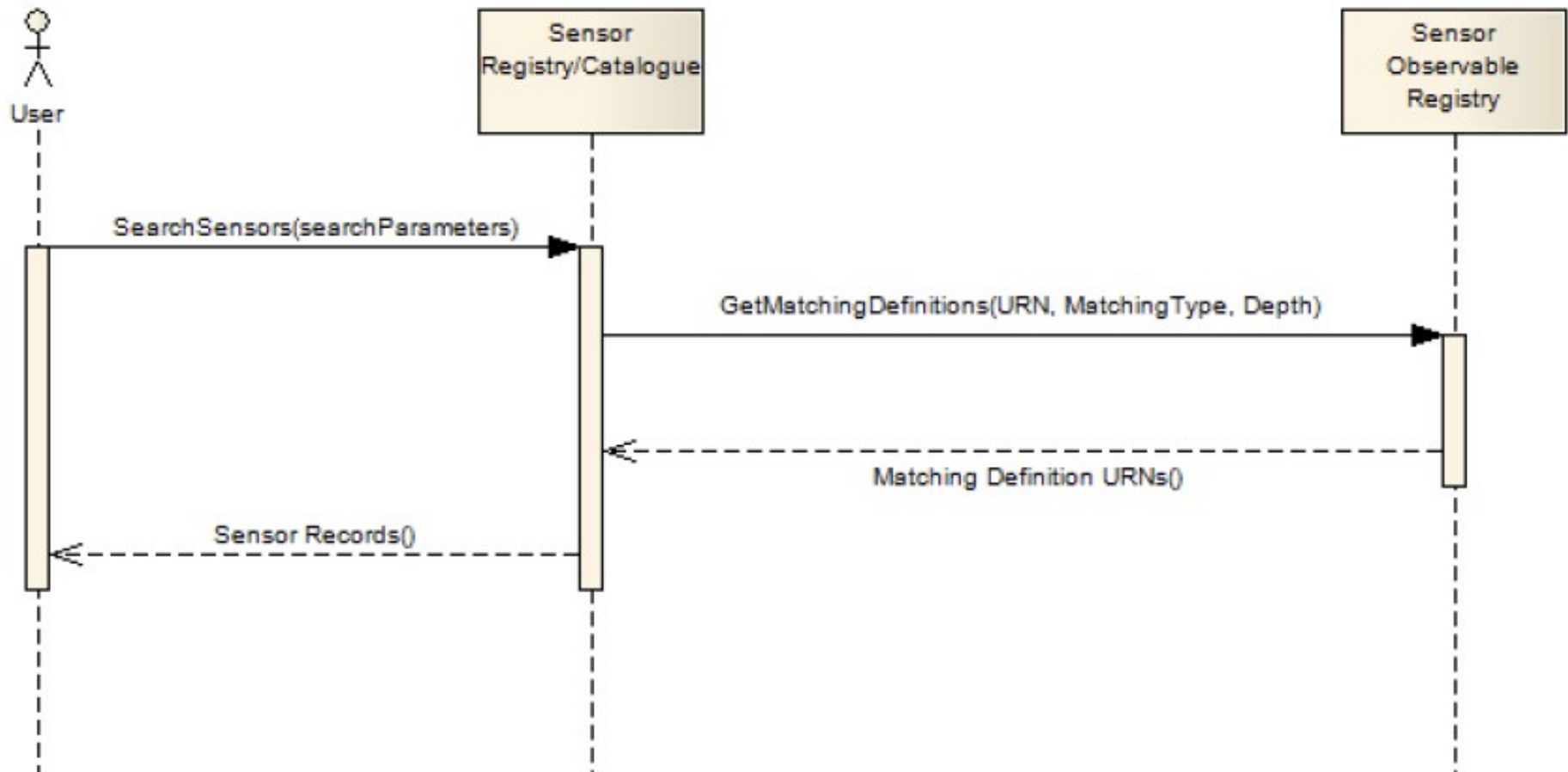
Build request

```
1 <GetMatchingDefinitionsRequest service="SOR" version="0.3.0" xmlns="http://sws1.uni-
muenster.de/sor">
2   <InputURI>urn:ogc:def:phenomenon:OGC::Temperature</InputURI>
3   <MatchingType>SUB_TYPE</MatchingType>
4   <SearchDepth>1</SearchDepth>
5 </GetMatchingDefinitionsRequest>
```

Send request

```
1 <sor:GetMatchingDefinitionsResponse xsi:schemaLocation="http://sws1.uni-muenster.de/sor
```

# Sensor Observable Registry (SOR)



# 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](mailto:jirka@52north.org)