Sensor Web Enablement

Implementation of SOS 1.0 service
Traffic noise data

Overview
1. What is
   - Sensor network
   - Sensor web
   - Differences between conventional and sensor data sources
2. OGC SWE
   - Concept
   - Functionality
   - Framework
3. OGC SOS
   - Concept
   - Observation model
   - Functionality
   - Encoding
   - Example application (traffic noise)

Sensor network
- computer accessible network
- many, spatially distributed devices using sensors
- monitor conditions at different locations,
  - Temperature, pressure, humidity,
  - Sound (noise), vibration,
  - Motion
  - Pictures
  - ...

Sensor Web
- Web accessible
  - sensor networks
  - archived sensor data
- Can be
  - discovered and
  - accessed
- using standard protocols and application program interfaces (APIs).

Differences between conventional and sensor network data sources (1/2)
- highly dynamic sensor metadata
  - Position, state of sensor quickly change
- time dependent data availability
  - Temporarily deployed at certain location
  - Mobile sensors
- sensor management
  - More detailed amount of metainformation, functionality necessary
  - Knowledge about individual sensors necessary

Differences between conventional and sensor network data sources (2/2)
- handling sensor status information
  - relevant to sensor management
  - can be used for improving the quality of sensor discovery
- specific metadata formats
  - SensorML as metadata input
- Semantics
  - taking into account semantic relationships the quality of a discovery solution can be improved
  - allows for example to discover sensors that measure similar or equivalent phenomena to a given one
2.8.2011

Sensor web concept - SWE

- All sensors reporting position
- All connected to the Web
- All with metadata registered
- All readable remotely
- Some controllable remotely

OGC SWE FW

Services
- Observation
- Tasking
- Alerting
- Analysis

Registries & Dictionaries
- Sensor Description Language
- Phenomenon Description Language

OGC SWE 2.0

Sensor Web Layer

Application Layer
- Sensor Web Portals

Sensor Web Layer
- Sensor Web Infrastructures

Sensor Layer
- Sensor Network Management Systems

internet of Things

SWE 2.0 interface model

SWE Standards framework
Standard Models and XML encoding schemas

- 1. Observations & Measurements Schema (O&M) – Encodes observations and measurements from a sensor, both archived and real-time.

- 2. SWE Common - defines low level data models for exchanging sensor related data between nodes of the OGC® Sensor Web Enablement (SWE) framework.

- 3. Sensor Model Language (SensorML) – Describes sensors systems and processes; provides information needed for discovery of sensors, location of sensor observations, processing of low-level sensor observations, and listing of taskable properties.
**SWE Standards framework**

**Standard web service interfaces**

- 4. Sensor Observations Service (SOS) - for requesting, filtering, and retrieving observations and sensor system information. This is the intermediary between a client and an observation repository or near real-time sensor channel.
- 5. Sensor Planning Service (SPS) – for requesting user-driven acquisitions and observations. This is the intermediary between a client and a sensor collection management environment.
- 6. Sensor Alert Service (SAS) – for publishing and subscribing to alerts from sensors.
- 7. Web Notification Services (WNS) – for asynchronous delivery of messages or alerts from SAS and SPS web services and other elements of service workflows.

**SWE 2.0**

- 8. SOR - WSI for accessing phenomenon definitions and for exploring semantic relationships between different phenomena.
- 9. SIR - WSI for managing sensor metadata; this includes the collection of sensor metadata, management of sensor status information as well as functionality for pushing sensor metadata into OGC Catalogues.
- 10. SES – Sensor Event Service – (Instead of SAS) Integration and leveraging of existing standards for realizing publish/subscribe interface and encoding event data, filtering and processing functionality.
- 11. EML – Enables Event Processing functionality for detecting patterns in (sensor) data streams and deriving new, higher-level information.

**Deployment scenario of SWE**

**SOS Concept**

**Observation**

- act associated with
  - discrete time instant or
  - period through which
    - a number,
    - term or
    - other symbol
  is assigned to a phenomenon (any observable occurrence)

**Observation Offerings**

- Collections of related sensor system observations
- Analogous to a “layer” in Web Map Service
- Each OO is constrained by (i.e.)
  - Specific sensor systems
  - Time period(s)
  - Phenomena
  - Geographical region (containing sensors or FOI)
SOS workflow

• Consumer’s point of view

Information for discovery SOS from OGC Catalogue

• Time period of observations
• Phenomena captured by observations
• Spatial extent of observations
• GML names used in observation offerings
• GML descriptions used in observation offerings
• Additionally: feature of interest, procedure/sensor, sampling rate...

GetCapabilities() Response

• Service Identification
• Service Provider
• Operations Metadata
• sos:Filter_capabilities
  – Spatial, temporal, scalar, ID filter capabilities
• sos:Contents
  – Observation offering list
DescribeSensor() Response: **SensorML**

- Identification (Unique ID)
- Capabilities (status, mobile)
- Sensor position (reference frame, X, Y, Z)
- Inputs (definitions of observable properties)
- Outputs
  - Name and ID of phenomena
  - Name and ID of offerings
- ...

GetObservation() request

- Mandatory
  - Service
  - Version
  - Offering
  - Observed property
  - Response Format
- Optional
  - srsName
  - Event time
  - Procedure
  - FOI
  - Result
  - Result model
  - Response mode

SOS workflow

- Producer’s point of view
RegisterSensor() request
- Service
- Version
- Sensor Description (SensorML)
- Observation Template (O&M)

InsertObservation()
- AssignedSensorID
- Observation
- Service
- Version

Other requests
- GetObservationByld()
- GetResult()
- GetFeatureOfInterest()
- GetFeatureOfInterestTime()
- DescribeFeatureType()
- DescribeObservationType()
- DescribeResultModel()

SOS concept example: Noise
- Procedure observes noise, which is emitted from airport.
- Procedure also produces observation...
Sensor Observation Service

- **52N-SOS-3.1.1 (SOS schema 1.0.0)**
  - Java 1.6, Apache Tomcat 7, PostgreSQL 9 + PostGIS 1.5, Apache Maven/Ant, XmlBeans, JTS Topology Suite
  - Basic + enhanced functionality of OGC SWE
    - GetCapabilities, GetObservation, DescribeSensor, RegisterSensor, InsertObservation, GetFeaturesOfInterest, GetResult
  - Filtering (spatial, temporal, scalar)
  - O&M types: Observation, Measurement, Category Observation

Testing

- Stable, semimobile and mobile sensor data simulation
- Registration of sensors and data input via
  - Transaction model
  - Direct record into DB
  - Checking of characteristics and constraints
- Searching data according to spatial, temporal and thematic attributes

Problems

- Missing definition of measurement interval
- Problems with communication between Postgre and SOS
- Unavailability of compatible clients and FW
  - Should be done until end of year
- Service downgrade
  - Small adaptation of DM (tab. Offering...)
  - Limited functionality of some parts (TestClient)