Standards Based Middleware and Tools for the Coastal Sensor Web

Surya Durbha, Roger King, Santhosh Rajender, Shruthi Bheemireddy, and Nicolas Younan,
Geosystems Research Institute (GRI)
Dept. of Electrical and Computer Engineering
Mississippi State University
Acknowledgement:
This work is currently funded by the Northern Gulf Institute (NGI)
Coastal Sensor Networks

- The Integrated Ocean Observing System (IOOS) (The U.S. contribution to the GOOS and GEOSS)
  - Buoys and other ocean platforms
- Presently national networks
  - NDBC, GoMOOS, TAO, etc.
- Data discovery and conversion problems due to
  - Syntactic, structural, and semantic heterogeneity in the datasets.
- Ongoing implementation of a Services-driven Sensor Web Enablement framework for resolving these heterogeneity problems.
Coastal Sensor Web Enablement

Sensor Web Enablement (SWE)

- Discovery
- Access
- Tasking
- Alerts

Web services (Sensor observation Service, O&M, etc.)
Encodings based on open standards

Decision Support Tools (monitoring, control, emergency response)
- Emergencies Support and Management
- Risk & Vulnerability assessments

SWE Clients

Heterogeneous Network Sources
(Various monitoring sensors)

Flood forecasting
Storm Surge Visualization
Sensor Web Architecture

Sensor Web Enablement (CosemWare)

Observables Dictionary

SensorML mapping
- <Sensor Group>
- <TempSensors>
- <Salinity sensors>

Definition of the geometric, dynamic, and observational characteristics of a sensor

Observations and Measurements (O&M)

References

ConstrainedBy

Observation XSD

Sensor Observation Service (SOS)

Access to Collection of Sensors

Buoy Sensors Gateway

GetCapabilities

GetObservation

DescribeSensor

Sensor Planning Service (SPS)

Sensor Alert Service (SAS)

Registry/Catalog (CS-W)

WMS

CosemWare (AJAX Client)

GetObservation

DescribeSensor

Observables Dictionary

SensorML mapping
- <Sensor Group>
- <wind direction sensor>
- <wind speed sensor>

SensorML mapping
- <Sensor Group>
- <salinity sensor>
- <Waves sensor>

SensorML mapping
- <Sensor Group>
- <TempSensors>
- <Salinity sensors>

Metadata

Metadata

Metadata

Observables Dictionary

SensorML mapping
- <Sensor Group>
- <TempSensors>
- <Salinity sensors>

SensorML mapping
- <Sensor Group>
- <wind direction sensor>
- <wind speed sensor>

SensorML mapping
- <Sensor Group>
- <salinity sensor>
- <Waves sensor>

Definition of the geometric, dynamic, and observational characteristics of a sensor

Current System

CosemWare (AJAX Client)
Spatio-temporal data mining from the sensor web

- **Spatial Subsetting**
  - bounding box
  - overlap, containing, intersection

- **Temporal sub-setting**
  - after a time instant, before time instant, during time instant, TEquals, past N sec/min/hrs/days

- **Filtering**
  - comparison operators such as Between, EqualTo, NotEqualTo, LessThan, GreaterThanEqualTo etc.
Semantic heterogeneities

Sea surface Temperature
Ocean Temperature

GCMD
(Global Change Master Directory)

Wind_Speed
Wind_Speed_ve
(Vector averaged wind speed)

DODS

Wind_Speed_sc
(scalar averaged wind speed)

Wind Speed
Water Temperature

NDBC
GoMoos buoy data served through Distributed Oceanographic Data System (DODS)
Semantic Enrichment
SPARQL Querying

SPARQL A-Box Query

- SPARQL is a protocol and query language for semantic web data sources.
- Based on matching graph patterns.
- Graph patterns contain triple patterns.
- Triple patterns are like RDF triples, but with the option of a query variable in place of RDF terms in the subject, predicate or object positions.
- Combining triple patterns gives a basic graph pattern, where an exact match to a graph is needed.

PREFIX rdfs:<http://www.w3.org/2000/01/rdf-scheme#>
PREFIX bu:<http://cosem.erc.msstate.edu/ontologies/cosemont.owl#>
SELECT *
FROM <file:/c:/ontologies/cosemont.owl>
WHERE
{
?s bu:hasStationID ?g.
?g bu:latitude ?la.
?g bu:longitude ?lo
FILTER(?WaterTemperature>20)
}
Example SPARQL Query (Scenario: “Find devices that can produce certain output variables”)

```sparql
PREFIX : <http://cosem.erc.msstate.edu/ontologies/cosem.owl#>
SELECT ?hasStationID ?lat ?lon.
FROM atmosphere
WHERE:
  ?x :hasMeasuredBy 'barometer'.
  ?x :hasStationID ?hasStationID.
  ?hasStationID :stationID ?stationID.
  ?hasStationID :latitude ?lat.
  ?hasStationID :longitude ?lon.
```
The hybrid approach, which we proposed is based on the machine learning and name matching techniques.

We utilize the instance data in ontology concepts to enable mapping between concepts.

The architecture of the mapping approach
Support Vector Machines

Linearly separable case; only support vectors (dark circled) are required to define the optimally defined hyperplane.

Linearly nonseparable case; only support vectors (dark circled) are required to define the optimally defined hyperplane. Linear decision hyperplanes in nonlinearly separable data can be handled by including slack variables.

Margin = 2 / ||w||
Semi-automated ontology mapping tool
### Evaluation Metrics

<table>
<thead>
<tr>
<th>Classes</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>windDirection</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Dominant wave period</td>
<td>0.67</td>
<td>0.9</td>
<td>0.77</td>
</tr>
<tr>
<td>wave Height</td>
<td>1</td>
<td>0.93</td>
<td>0.96</td>
</tr>
<tr>
<td>Air temperature</td>
<td>1</td>
<td>0.95</td>
<td>0.97</td>
</tr>
<tr>
<td>Atmospheric pressure</td>
<td>1</td>
<td>0.96</td>
<td>0.98</td>
</tr>
<tr>
<td>windSpeed</td>
<td>0.67</td>
<td>0.9</td>
<td>0.77</td>
</tr>
<tr>
<td>water Temperature</td>
<td>1</td>
<td>0.91</td>
<td>0.95</td>
</tr>
<tr>
<td>wind Gust</td>
<td>1</td>
<td>0.91</td>
<td>0.95</td>
</tr>
</tbody>
</table>
Enable flexible mobile access to distributed web resources for advanced personalization and localization features.

- Automatic discovery and invocation of web services
- Universal Description Discovery and Integration (UDDI) provides a registry of businesses and web services to describe service profiles in human-readable way.

**Android** is a new and promising mobile platform based on the Linux operating system provided by Google.

- Android is not just another Java-based mobile platform but actually the only platform that adopts the results of the mobile middleware research.
Semantic Web Services

- Semantic interoperability is crucial for Web services providing additional features like knowledge-based, location or context-aware information.

- Integration of semantic metadata and the Web services infrastructure results in a service named Semantic Web Services (SWS) that has well-defined semantics.

- OWL-S provides a language to describe actual Web services that can be discovered and then invoked using standards such as WSDL and SOAP in such a way that the descriptions can be interpreted by a computer system in an automated manner.
Ontology for Sensor Concepts and Service Advertisement Propagation.

- **Exact**: If Req\(_{\text{out}}\) and Adv\(_{\text{out}}\) are same. That is, if Req\(_{\text{out}}\) and Adv\(_{\text{out}}\) both point to same concept say WaterTemperature of the ontology.

- **Plug-in**: If Adv\(_{\text{out}}\) subsumes Req\(_{\text{out}}\), then Adv\(_{\text{out}}\) can be plugged instead of Req\(_{\text{out}}\). That is, if Adv\(_{\text{out}}\) points to WaterTemperature and Req\(_{\text{out}}\) points to SeaTemperature of the ontology (Figure 14).

- **Subsume**: If Req\(_{\text{out}}\) subsumes Adv\(_{\text{out}}\), then the provider may or may not completely satisfy the requester.

- **fail**: If there is no subsumption relation between Adv\(_{\text{out}}\) and Req\(_{\text{out}}\).
suryad@gri.msstate.edu
rking@cavs.msstate.edu