A Semantically Enabled Service Architecture for Mashups over Streaming and Stored Data

Alasdair J G Gray[1], Raúl García-Castro[2], Kostis Kyzirakos[3], Manos Karpathiotakis[3], Jean-Paul Calbimonte[2], Kevin Page[4], Jason Sadler[4], Alex Frazer[4], Ixent Galpin[1], Alvaro A A Fernandes[1], Norman W Paton[1], Oscar Corcho[2], Manolis Koubarakis[3], David De Roure[4], Kirk Martinez[4], and Asunción Gómez-Pérez[2]

[1] University of Manchester, United Kingdom.
[2] Universidad Politécnica de Madrid, Spain.
[3] National and Kapodistrian University of Athens, Greece.
[4] University of Southampton, United Kingdom.
http://www.semsorgrid4env.eu

Abstract. Sensing devices are increasingly being deployed to monitor the physical world around us. One class of application for which sensor data is pertinent is environmental decision support systems, e.g. flood emergency response. However, in order to interpret the readings from the sensors, the data needs to be put in context through correlation with other sensor readings, sensor data histories, and stored data, as well as juxtaposing with maps and forecast models. In this paper we use a flood emergency response planning application to identify requirements for a semantic sensor web. We propose a generic service architecture to satisfy the requirements that uses semantic annotations to support well-informed interactions between the services. We present the SemSor-Grid4Env realisation of the architecture and illustrate its capabilities in the context of the example application.

Keywords: Sensor web architecture, semantic sensor networks, sensor network ontology, mashups, use case

1 Introduction

Sensor networks promise to bridge the gap that, for too long, has separated computing applications from the physical world that they model and in which they are ultimately embedded. Many scientific and technological challenges need to be tackled before sensor networks can be exploited to their full capacity for aiding decision support applications. Additionally, as more and more sensor networks are independently developed and deployed, it becomes increasingly important to support their reuse in applications that were not foreseen or that transcend their original purpose. This will facilitate the use of sensor network technology to support decision-making that requires on-the-fly integration of data of differing modalities, e.g. sensed data with data stored in databases, as

well as the *ad hoc* generation of mashups over data stemming from computations that combine real-time and legacy historical data. This, in turn, will enable the enacting of decisions based on such real-time sensed data.

One area that has seen a massive increase in the deployment of sensing devices to continuously gather data is environmental monitoring [9]. For example, metocean data, i.e. wave, tide, and meteorology data, for the south coast of England is measured by two independently deployed sensor networks—the Channel Coastal Observatory (CCO)[5] and WaveNet[6]—as well as meteorological data from the MetOffice[7]. More broadly, data from sensors is being used worldwide to improve predictions of, and plan responses to, environmental disasters, e.g. coastal and estuarine flooding, forest fires, and tsunamis. The models that make these predictions can be improved by combining data from a wide variety of sources. For example, in a coastal flooding scenario, details of sea defences, combined with current wave information, can be used to identify potential overtopping events—when waves go over the top of a sea defence. When responding to a flooding event additional data sources such as live traffic feeds[8], and details of public transport infrastructure can help inform decisions.

Enabling the rapid development of flexible and user-centric decision support systems that use data from multiple autonomous independently deployed sensor networks and other applications raises several technical challenges, including: (i) Discovering relevant sources of data based on their content, e.g. features of interest and the region covered by the dataset; (ii) Reconciling heterogeneity in the data sources, e.g. the modality, data model, or interface of the data source, and enabling users to retrieve data using domain concepts; and (iii) Integrating and/or mashing up data from multiple sources to enable more knowledge about a situation to become available. OGC-SWE [3] and GSN [1] are previous proposals that share some of our aims in this paper. However, both require data sources to expose their respective data model, thus limiting the reuse of existing sources from a multitude of domains. Additionally, neither supports integrating data from heterogeneous data sources. Our proposed approach makes extensive use of semantic technologies to reconcile the heterogeneity of data sources whilst offering services for correlating data from independent sources. This enables user-level applications to generate queries over ontologies which are then translated into queries to be executed over the data sources. Data is returned expressed in terms of the user-level ontology and can be correlated and juxtaposed with other data in a meaningful and controlled manner.

The rest of the paper is structured as follows. In Section 2 we provide a detailed description of the flood emergency planning scenario for the south coast of England and identify the set of requirements. Section 3 provides an overview of the ontology network that we have developed to represent the information

---

[5] `http://www.channelcoast.org/` (21 October 2010).

[6] `http://www.cefas.co.uk/our-science/observing-and-modelling/` `monitoring-programmes/wavenet.aspx` (21 October 2010).

[7] `http://www.metoffice.gov.uk/` (4 November 2010).

[8] `http://www.highways.gov.uk/rssfeed/rss.xml` (4 November 2010).

needed in this scenario. We then present our architecture for a semantic sensor web in Section 4, and describe the role of ontologies and semantic annotations in the architecture. Section 5 describes a prototype deployment of our architecture for the flood emergency response use case. We discuss related work in Section 6 and present our conclusions in Section 7.

2   Motivating Scenario and Requirements

This section presents a flood emergency planning scenario for the south coast of England, drawn from the SemSorGrid4Env project[9], that illustrates the use of data coming from sensor networks together with traditional stored data sources, e.g. relational databases, and map images. Requirements for a software infrastructure aimed to support the use case are identified. Although the requirements have been identified for a specific use case, they are representative of general requirements for a software infrastructure to enable web applications to generate data mashups over heterogeneous data sources, including sensor data.

2.1   Flood Emergency Planning Scenario

A flood emergency response planner would like to receive an alert, by email or SMS text message, when a flooding event is predicted to occur, or has been detected to have occurred, for their area of responsibility. Such a prediction may stem from a model that forecasts the future tidal patterns based on the current sea state, as measured by sensor networks such as the CCO[5] and WaveNet[6], and predicted weather from a web feed such as that provided by the UK Met Office[7]. Alternatively, it could arise from a sea defence, the details of which are stored in a database such as the UK Environment Agency's National Flood and Coastal Defence Database (NFCDD)[10], being overtopped, as measured by some sensor. This requires characterising the overtopping event that, in turn, requires stored and sensed data to be correlated.

When the emergency response planner receives an alert, they need to plan a suitable response based on the likely severity of the flood, its location, and the likely impact on the public, the environment, and industry. To enable them to respond appropriately, they need to dynamically find relevant data sources based on the thematic and spatiotemporal coverage of the data. Once relevant sources have been identified, suitable mechanisms for retrieving, correlating, and displaying the data in terms of the flooding domain are required that hide the complexities of the heterogeneity of the sources, data modalities, and terminology used. For example, the manager responsible for the Portsmouth region would need details of the shipping due to arrive (available from web feeds), as well as details of the ships and their cargo (available from databases). They would also need the weather forecast (available from web feeds), details of the current sea

---

[9] http://www.semsorgrid4env.eu/ (24 November 2010).
[10] http://www.scisys.co.uk/casestudies/environment/nfcdd.aspx (21 October 2010).

state (available from sensors), and a forecast of how the latter are likely to evolve in the region (available from predictive models). This would enable them to make more accurate decisions about the likely effects of the sea-state on shipping. Similarly, they must assess the risk to the public. To aid this, they need details of the transportation infrastructure (available from stored data), populated areas (available as maps), and the predicted effects of the flood (available from models based on the sensor data).

2.2   Requirements

The following general requirements can be drawn from the scenario.

R1 *Accurate characterisation of conditions that define an event.* It should be possible to describe the data of interest and allow the system to discover how to retrieve the data. This is shown in the example scenario by the need to automatically identify when the conditions characterising an overtopping event are met so that an alert can be generated.

R2 *Correlation of data of differing modalities.* It should be possible to correlate streaming data from sensors with stored data such as stream archives and databases, and visual data such as maps. This is shown in the scenario by the need to detect overtopping events that combines sensor data with data stored in a database and display it on a map.

R3 *Integrating data coming from heterogeneous data models.* It should be possible to mediate data coming from autonomous sources regardless of the source data model, i.e. the conceptualisation and terminology used in the data, and representations (e.g. relational, RDF, and XML). This is shown in the scenario by the need to use a wide variety of independent sources.

R4 *Discovery of relevant data sources.* It should be possible to identify potentially useful sources of data based on the spatiotemporal and thematic coverage of the data provided. This is shown in the scenario by the need to find sources of data when planning the response to a flooding event.

R5 *Presentation and control of information.* It should be possible for users to discover, relate, juxtapose, and display information from a variety of sources without needing to understand the sources, and in response to an evolving situation. This is shown throughout the scenario.

3   Modelling Semantic Sensor Web Information

The requirements presented in the previous section indicate that there is a clear need to support the *ad hoc* responsive evolving use of an information space. The data resources of the information space will contain various forms of heterogeneity, including data modality (i.e. sensed, stored, and graphical), data model (i.e. terminology), and data representation (e.g. relational, RDF, and XML), which need to be reconciled into a single coherent conceptualisation. We use ontologies to represent the common data model for the information space since they
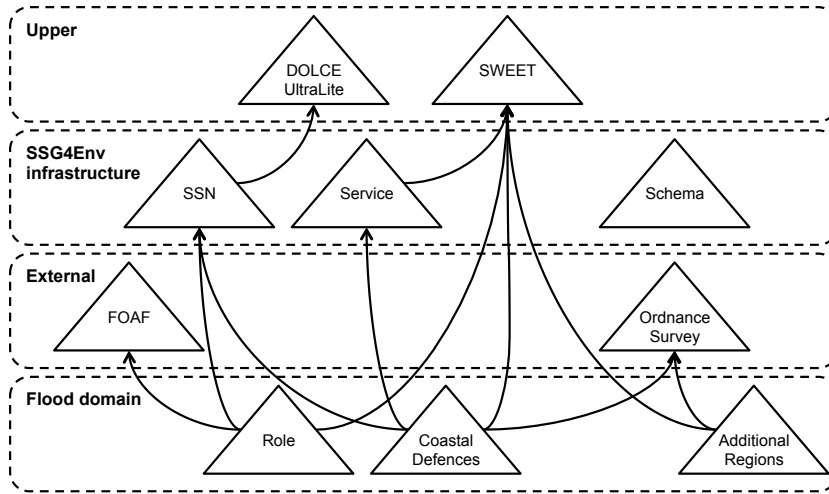
Fig. 1. The SemSorGrid4Env ontology network in the flood emergency planning scenario. The arrows indicate ontology reuse.

facilitate: (i) Describing the different infrastructure services and data sources as well as any domain-dependent information; (ii) Having a shared vocabulary to interoperate both across the internal infrastructure services, and between that infrastructure and external sources that adopt alternative approaches, e.g. OGC-SWE based ones [3]; and (iii) Discovering, accessing, and integrating information that is shared within the infrastructure.

Fig. 1 illustrates how the ontology network used in the flood emergency planning scenario is composed of different ontologies that can be classified in different layers according to whether the ontology represents: domain-specific information required for the scenario, information required for the infrastructure, or upper-level information used to facilitate interoperability among the other ontologies. These ontologies satisfy different knowledge representation requirements extracted during the development of the architecture and of the scenario prototype. (i) To represent sensor networks and their observed information about properties of certain features of interest. This is covered by the SSN ontology, developed by the W3C Semantic Sensor Network Incubator Group[11]. The SSN reuses the DOLCE+DnS UltraLite upper ontology[12]. (ii) To represent the services provided by the infrastructure and the datasets they provide access to. This is covered by the Service module that reuses the SWEET upper ontologies [15] and includes concepts from the ISO19119 standard on geographic information services [11]. (iii) To represent schema metadata about relations and relational streams. This is covered by the Schema module that extends, and

---

[11] `http://www.w3.org/2005/Incubator/ssn/` (11 November 2010).
[12] `http://www.loa-cnr.it/ontologies/DUL.owl` accessed 11 November 2010.

corrects, an ontology for relational data and schema components [14]. (iv) To represent the geographic and administrative regions of the south coast of England. This is covered by the Ordnance Survey ontologies[13], which include the regions from Great Britain, and by the Additional Regions ontology, which includes other regions needed in our scenario. (v) To represent those features of interest and their properties that are specific to the flood emergency planning scenario. This is covered by the Coastal Defences ontology. (vi) To represent the different roles involved in a flood emergency planning scenario. This is covered by the Roles ontology.

All the ontologies[14] have been implemented using OWL. While some of the ontologies presented here are specific to the flood warning scenario, e.g. Role, the architecture proposed in the next section is generic. Thus, it can be adapted to other situations by replacing the flood domain ontologies.

4   Semantic Sensor Web Architecture

The proposed service architecture gives rise to a semantic sensor web for environmental management that aims to meet the requirements identified in Section 2. The architecture (Fig. 2) comprises a set of services that can be composed into orchestrations to deliver the desired functionality. The architecture can interact with existing frameworks such as OGC-SWE [3], either as applications that exploit our services or as concrete resources that are wrapped to provide additional functionality, e.g. query-based access or semantic integration. A sample orchestration from the flood scenario is given in Fig. 5 and described in Section 5. The architecture is structured into three tiers, although a service may call any other service regardless of which tier they appear in. The *data tier* enables the publication and querying of data in its native format, i.e. a relational database can be queried using SQL and sensor data through a continuous query language such as SNEEql [7]. The *middleware tier* supports the discovery of relevant data as well as the reconciliation of data models and querying over these reconciled models. The *application tier* provides domain specific services and supports the transition from service-oriented web services to REST services [6]. The interfaces[15] offered by the web services in our architecture are given in Table 1. A full specification of the services, and the operations they support, can be found in [8]. The service architecture uses, where possible, existing web service standards.

The specification of the service interfaces, i.e. the WSDL definition of the service-oriented services, supports *well-formed* interactions with the services. However, it is anticipated that there will be multiple services which, at a functional level, satisfy a user's needs. To enable the user to make a *well-informed* choice between these services, the architecture uses *semantically annotated property documents* which describe the non-functional properties of a service, e.g. the

---

[13] `http://www.ordnancesurvey.co.uk/oswebsite/ontology/` (11 November 2010).

[14] `http://www.semsorgrid4env.eu/ontologies/` (7 December 2010)

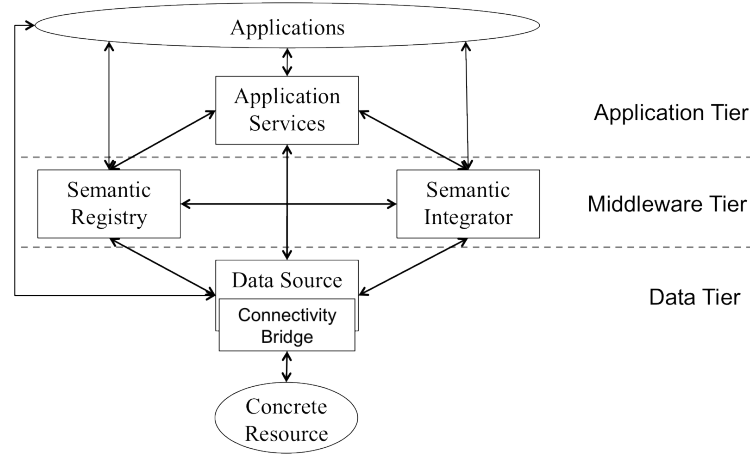[15] We use the term *interface* to mean a logical group of operations.

Fig. 2. Conceptual view of the service architecture. Boxes denote **SemSorGrid4Env** services, ovals denote external services/entities, arrows denote caller-callee relationships.

Service Name	Interfaces Offered	Notes
Data Source Service	Service, *Integration*, *Query*, *Data Access*, *Subscription*	Must provide at least one of query, data access, or subscription interfaces.
Semantic Registry Service	Service, Registration, *Discovery*, *Query*, *Data Access*, *Subscription*	Either the discovery or the query interface must be provided.
Semantic Integrator Service	Service, Integration, Query, *Data Access*, *Subscription*	One of the data access or subscription interface must be offered to support queries over streams.
Application Services	REST	Interface design focuses on identifying and structuring web resources.

Table 1. Services and their interfaces. Interfaces shown in italics are optional.

datasets and their features, that are available from the service. The property document can be retrieved through the service interface operations, provided by all services, and their content is specified using the ontology network (Fig. 1).

In the following sections, we describe the services of the architecture and the role of the property document in supporting their activities. We illustrate these interactions with the property document for the CCO sensor data service, excerpts of which are shown in Fig. 3. Note, these declarations make use of stRDF [12], a spatiotemporal extensions for RDF that defines URIs of the form &term.

4.1 Data Source Services

Data source services provide the mechanism to publish data: either coming from a sensor network or some other data source, e.g. a database or another data service. Depending on the interfaces supported by the data service, operations are provided for querying, retrieving, and subscribing to data. A distributed query processing service can be offered, using the integration interface, which

```
1   <service:WebService rdf:about="#cco-ws">
2       <rdfs:label>Channel coastal observatory streaming data service</rdfs:label>
3       <service:hasInterface rdf:resource="service:ssg4ePullStream"/>
4       <service:hasDataset rdf:resource="#envdata_SandownPier_Tide"/>
5       <service:hasDataset rdf:resource="#envdata_SandownPier_Met"/>
6       ...
7   </service:WebService>
8   <sweet:Dataset rdf:about="#envdata_SandownPier_Tide">
9       <rdfs:label>envdata_SandownPier_Tide</rdfs:label>
10      <service:coversRegion rdf:resource="&AdditionalRegions;SandownPierLocation"/>
11      <time:hasTemporalExtent rdf:datatype="&registry;TemporalInterval">
12          [2005, NOW]</time:hasTemporalExtent>;
13      <service:includesFeatureType rdf:resource="&CoastalDefences;Sea"/>
14      <service:includesPropertyType rdf:resource="&CoastalDefences;TideHeight"/>
15      <service:includesPropertyType rdf:resource="&CoastalDefences;WaveHeight"/>
16      <service:hasSchema rdf:resource="#envdata_SandownPier_Tide_Schema"/>
17      ...
18  </sweet:Dataset>
19  <schema:Stream rdf:about="#envdata_SandownPier_Tide_Schema">
20      <schema:extent-name>envdata_SandownPier_Tide</schema:extent-name>
21      <schema:hasAttribute rdf:resource="#HMax"/>
22      <schema:hasAttribute rdf:resource="#Tp"/>
23      ...
24  </schema:Stream>
25  <schema:Attribute rdf:about="#HMax">
26      <schema:attribute-name>HMax</schema:attribute-name>
27      ...
28  </schema:Attribute>
29  ...
```

Fig. 3. Snippets from the CCO sensor data web service semantic property document expressed in stRDF [12] using XML notation. We assume appropriate namespace declarations and represent omitted parts of the document with '...'.

consumes data from other services that may only support the data access or subscription interfaces.

Data source services publish a property document about the data that they provide, and the mechanisms by which it may be accessed. The first part of the property document in Fig. 3 describes the interaction mechanisms provided, i.e. the interfaces (line 3) and operations supported by the data service. The rest of the property document describes the data that is available through the service, which is not covered in the WSDL definition of the service.

A data source may publish one or more datasets, as per the WS-DAI standard [2]. Lines 4 and 5 show that the CCO sensor data service publishes multiple datasets including the two identified as #envdata_SandownPier_Tide and #envdata_SandownPier_Met. Each dataset is described in terms of its spatiotemporal and thematic coverage, and (where appropriate) its schema. Lines 8 to 18 provide details of the #envdata_SandownPier_Tide dataset. Specifically, lines 10 to 12 describe the spatiotemporal range of the dataset as providing data for the 'Sandown Pier' location and that the time range is from 2005 until the current time, represented with the distinguished literal 'NOW'. The types of features covered by the dataset are declared by the statements in line 13, which state that the #envdata_SandownPier_Tide dataset contains information about the CoastalDefences concept Sea. Lines 14 and 15 give the property types covered

as the CoastalDefences concepts of TideHeight and WaveHeight. Where appropriate, e.g. for relational data sources, the property document also includes an ontological description of the schema of the dataset using the Schema ontology. Line 16 declares that the `#envdata_SandownPier_Tide` dataset has a schema described by the resource `#envdata_SandownPier_Tide_Schema`. Lines 19 to 28 describe the relational schema of the `#envdata_SandownPier_Tide` data stream: its name, attributes, types of the attributes, primary key, and timestamp attribute. It is this information that enables a distributed query service, which itself can be seen as a data service, to support queries over external data sources.

4.2   Semantic Registry Service

The registry service supports the discovery of relevant services. The registration interface enables the registration of service descriptions, viz. the information contained in the service property document. Since a registry service can be seen as a data service, it provides the same interfaces, with the data access and subscription interfaces providing support for long-lived queries. In the SemSorGrid4Env implementation, the registry service data store is populated with the content of the service property documents. It supports query-based access through the spatiotemporal SPARQL extension stSPARQL [12]. External services, such as those defined by the OGC [3, 5], can be manually entered.

The information contained in the property document enables applications to discover relevant data sources using application domain specific terms. For example, a flood response manager responsible for the Portsmouth region would like to discover sensor data sources for that region `&AdditionalRegions;Solent`. From Fig. 3, we see that the CCO sensor data web service `#CCO-WS` exposes a stream interface service:ssg4ePullStream (line 3) taken from the Service ontology, and that the `#envdata_SandownPier_Tide` dataset has the spatial coverage `&AdditionalRegions;SandownPierLocation` (line 10) using the Additional Regions ontology. Due to the use of spatial constraints in the definitions of the regions in the ontology, the registry is able to deduce that the `#envdata_SandownPier_Tide` dataset is relevant for the application. This is because the location `&AdditionalRegions;SandownPierLocation` is contained in the region defined for the Solent `&AdditionalRegions;Solent`. Thus, `#CCO-WS` is a relevant source for the flood response manager.

4.3   Semantic Integration Service

The integrator supports the creation and querying of an information space over independent heterogeneous data sources. The integration interface enables the creation of an integrated data model by supplying a mapping document relating the data sources to the global model. The query interface enables ontology-based access to the data sources [4]. That is, a user or application can express a query in terms of ontological concepts and the integration service translates it into a set of queries over the relevant sources, retrieves the answers, and translates them into ontological instances. Note, the integration service is a data source,

thus it provides the same interfaces, with either the data access or subscription interface being offered to support long-lived continuous queries over sensor data.

The integrator uses the property documents of the data sources involved in an integration to select the appropriate interaction mechanism. That is, for each data source, the property document informs the integrator of the interfaces and query language supported. From the example document given in Fig. 3, the integrator can infer that the `#CCO-WS` only supports data retrieval through the pull-stream interface (line 3), i.e. it does not support queries, and that the schema of the `#envdata_SandownPier_Tide` is as described (lines 19-28). The integrator can also query the registry to discover suitable distributed query processing services to invoke in answering queries over the integrated data resource. The property documents of the data sources also aid the integrator in the creation of the property document that describes the integrated data model. In particular, its spatiotemporal and thematic content. Note that the semantic representation of a source schema, as provided in lines 19-28 of Fig. 3, can help mapping tools in understanding the schema of a data source and, therefore, in the creation of mappings between source schemas and an ontology for the domain of interest.

4.4   Application Services

The services in the application tier of the architecture provide support for web-based applications and mashups to interact with the services and data sources in the architecture. On the whole, web-based applications and mashups interact through a resource-oriented approach [6], i.e. they interact through HTTP calls, viz. GET, POST, PUT, and DELETE. As such, the application services bridge the gap between a resource-oriented viewpoint that prevails for user fronting applications and the service-oriented viewpoint that is preferred in middleware and back-end system components. By exposing a REST interface, the application services enable web-based applications to request content based on formats that they can process and display to the user, e.g. GML, GeoJSON, and HTML [13].

The application services, and ultimately the applications that rely on them, exploit information contained in the property documents associated with services. Property documents enable application services to locate relevant services through the registry and gain insight into how to interact with them. They also enable applications to integrate data sources through integration services.

4.5   Summary

The property document enables well-informed interactions between the services in the architecture, and is instrumental in all aspects of the functionality offered. It is not a requirement to provide the semantic property document, and no parts of it are mandatory. As such, external services, e.g. those defined by OGC [3, 5], can be incorporated into the architecture. However, by describing the non-functional properties, particularly the spatiotemporal and thematic coverage of its data, in a property document a service can be discovered through the registry, and used by the integrator and application services in a seamless manner.
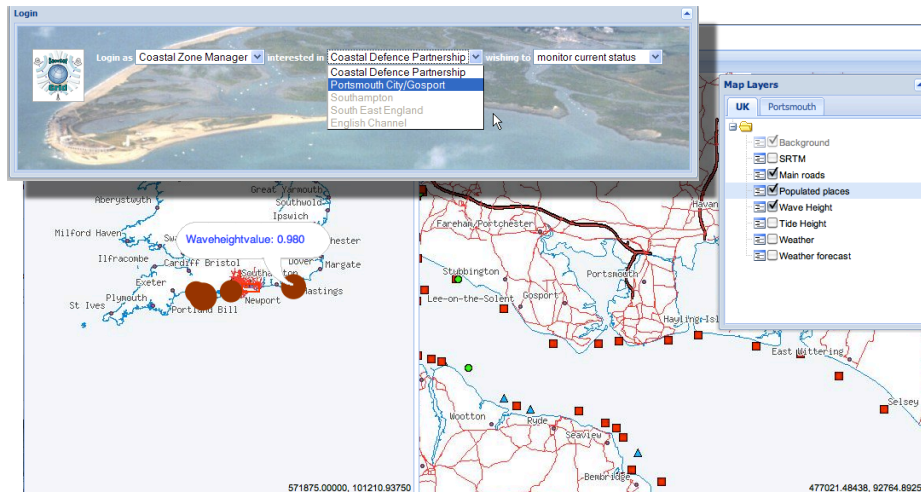
Fig. 4. Screenshots from the flood emergency response Web application available from
`http://webgis1.geodata.soton.ac.uk/flood.html`.

5   The Flood Scenario Deployment

We now show how the architecture enables the flood emergency planning scenario
described in Section 2. We show how a flood emergency planner (the user) can
add a source to identify when an overtopping event is detected. The scenario
assumes the existence of: (i) a semantic registry service at some well known
location; (ii) several data services which have registered their semantic property
documents with the registry service; and (iii) a distributed query service, an
integration service, and application services to support the web application.

The user accesses the web application through the login screen shown in the
top left of Fig. 4. When logging into the application, the user selects their role,
the region they are responsible for, and the task that they wish to conduct.
The options in the selection boxes are populated with terms from the ontology
network (Fig. 1), e.g. the choice of role comes from the concepts in the Role
ontology. This provides an initial characterisation of the data that is relevant
for the interaction, i.e. the values provided parameterise the queries sent to the
registry in order to discover relevant data sources. For the login selection shown,
the registry query is parameterised to discover data sources for the Portsmouth
area for a Coastal Zone Manager who wishes to monitor the current status.

The result of the login process is shown in the main screenshot in Fig. 4. The
user is presented with two map views based on the region selected, viz. Portsmouth.
The left pane shows a zoomed-out map providing context while the right pane
shows a zoomed-in map on the region selected. Both maps have been superim-
posed with layers presenting data from a variety of sources that satisfied the
queries sent to the registry. The available layers are shown in the Map Layers
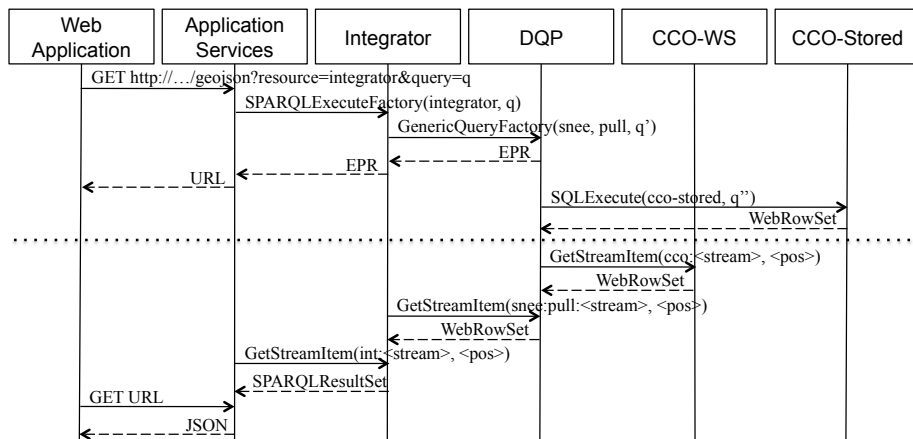pop-up window, from which the user can select the layers they wish to be dis-

Fig. 5. Interaction diagram showing how data is integrated across heterogeneous data sources. Operations below the dotted line are repeated periodically.

played. In the example shown, three layers have been selected for display. Two of these—showing the main roads and the populated areas—have been retrieved from OGC-WMS services [5] that have been manually registered, i.e. details of the service and the dataset have been entered into the registry through a web form. The current wave height values retrieved from the available sensor networks for the region are juxtaposed on top. The values are displayed as red circles: the larger the circle, the higher the wave value measured.

To support identifying an overtopping event requires data from heterogenous sources with different schemas and data modalities, viz. stored and sensed, to be integrated. The required orchestration is depicted in Fig. 5 which shows a web application retrieving data through an integrator that exposes an ontological view of the source data. Note that the orchestration assumes that the integrated resource has already been created, i.e. the mapping document relating the data sources to the ontological view has already been passed to the integrator. The web application supports the user in discovering potential data services for detecting overtopping events based on the contents of the property documents stored by the registry service (not shown in the orchestration in Fig. 5). The web application then supports the user in characterising an overtopping event as a query over the ontological view, hiding all the complexities of the required orchestration. The web application uses a RESTful interface offered by an application service to pass the query as a service call to the integrator. The integrator translates the query over the ontological view into a query expressed in terms of the source schemas. The integrator instantiates a distributed query processing service (DQP) to evaluate the query over the sources. As the query is evaluated, answers are periodically retrieved through the interactions shown below the dotted line in Fig. 5. The rate at which the DQP service polls its sources is controlled by the rate declared in the property document of each source. Simi-

larly, the rates at which the integrator and the application poll their respective source is controlled by the property document declarations.

6   Related Work

We describe related work in its ability to satisfy the requirements identified in Section 2.

The Open Geospatial Consortium Sensor Web Enablement (OGC-SWE) [3] defines a set of XML data models for describing sensors and their data, as well as a set of web services for publishing sensor data, locating data, and receiving alerts about the data. A reference implementation of the framework was developed in the SANY project [16]. The framework can be seen to satisfy R4, and provides support for satisfying R5. However, data access patterns are limited by the service interfaces and there is no support for declarative query languages. As such, it does not fully satisfy R1. Data is published according to their XML data models, which is not always possible with autonomous data sources. Thus, they do not satisfy R3. OGC-SWE does not fully meet R2: there is support for unmediated merging of sensor and stored data but not for correlating it. We note that the GetCapabilities operation provided by the services provide support for the functional properties in our property documents but not the spatiotemporal or thematic properties. Henson *et al.* [10] have extended the sensor observation service by semantically annotating the data. Our proposal goes beyond this by using semantics to support the discovery, integration, and mashup of data stemming from autonomous heterogeneous data sources.

Global Sensor Network (GSN) [1] is a middleware platform for the deployment and programming of sensor networks. It allows for the abstraction of sensor nodes as data sources irrespective of the underlying hardware and provides query processing capabilities within the middleware. It enables a data-oriented view of sensor networks and the processing of queries over that data. GSN satisfies R1 and R2 provided that the data is all published in the same data model. It does not satisfy the other requirements.

Collaborative Oceanography [17] used semantic annotations to support the reuse of oceanographic data. Their approach relied on a centralised triple store containing the annotations and the manual mashup of data based on these annotations. Our approach provides support for semantic integration and mashup of heterogeneous data sources.

7   Conclusions

We have presented a service architecture for providing support to semantic sensor web applications. The architecture provides a semantically integrated information space for sensed and stored data drawn from heterogeneous autonomous data sources. The architecture enables rapid development of thin applications (mashups) over this information space through the use of (i) declarative queries to describe the data need, both for locating data based on its spatiotemporal

and thematic coverage, and for integrating and accessing data, and (ii) semantically annotated property documents which support well-informed interactions between the architecture services.

Five high-level requirements were identified, from the application use case presented, that are considered to be relevant for a broad range of applications.

R1 *Accurate characterisation of conditions that define an event.* This is satisfied by the use of declarative queries to retrieve data. Underlying data sources can be queried through their native query language, e.g. SQL for relational databases and SNEEql for relational streams coming from sensor networks, while the integrator provides for ontology-based queries expressed in a SPARQL extension for streaming data, viz. SPARQL$_{\text{Stream}}$. Data sources that do not support query-based data retrieval can be queried through the use of a distributed query service.

R2 *Correlation of data of differing modalities.* This is satisfied through the use of query languages that provide support for correlating sensed and streaming data, e.g. SNEEql and SPARQL$_{\text{Stream}}$, as well as the use of application services to create layers for juxtaposing data.

R3 *Integrating data coming from heterogeneous data models.* This is satisfied by the use of ontologies to provide a common vocabulary and data model. Data sources publish semantically annotated property documents that use concepts available in ontologies to describe their datasets and interfaces. The registry, integrator, and application services can automatically exploit the content of the property documents due to the use of ontological terms.

R4 *Discovery of relevant data sources.* This is satisfied by the semantic registry service and the publication of semantically annotated property documents by the data services. The registry service uses the statements contained in the property documents to identify relevant data services by answering declarative stSPARQL queries which characterise the spatiotemporal and thematic data needs of applications and users.

R5 *Presentation and control of information.* This is satisfied by the interaction between web-based applications and mashups with the application services. By providing REST style interfaces, offering data in a variety of formats including mapping layers, and using domain specific concepts drawn from (where possible) standard domain ontologies, the application services enable the rapid development of thin web-based applications without sacrificing the functionality and control offered to the user.

The next steps for the implementation of our architecture are to provide services which can push data from the sources through the architecture, and to provide mechanisms for interacting with existing infrastructures such as OGC-SWE. For future work we intend to investigate offering configurable mechanisms for supporting REST interfaces to integrated information spaces. We will also perform a user evaluation with coastal managers from the Solent region.

References

1. Aberer, K., Hauswirth, M., Salehi, A.: Infrastructure for data processing in large-scale interconnected sensor networks. In: 8th International Conference on Mobile Data Management (MDM 2007). pp. 198–205 (2007)
2. Antonioletti, M., Krause, A., Paton, N.W., Eisenbrg, A., Laws, S., Malaika, S., Melton, J., Pearson, D.: The WS-DAI family of specifications for web service data access and integration. SIGMOD Record 35(1), 48–55 (2006)
3. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC® sensor web enablement: Overview and high level architecture. In: 2nd International Conference on GeoSensor Networks (GSN 2006). pp. 175–190 (2006)
4. Calbimonte, J.P., Corcho, O., Gray, A.J.G.: Enabling ontology-based access to streaming data sources. In: 9th International Semantic Web Conference (1) (ISWC 2010). LNCS, vol. 6496, pp. 96–111. Springer, Shanghai, China (November 2010)
5. de la Beaujardiere (Ed), J.: OpenGIS® web map server implementation specification. Standard Specification 06-042, Open Geospatial Consortium Inc. (2006)
6. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. Ph.D. thesis, Information and Computer Science, University of California, Irvine, California, USA (2000)
7. Galpin, I., Brenninkmeijer, C.Y.A., Gray, A.J.G., Jabeen, F., Fernandes, A.A.A., Paton, N.W.: SNEE: a query processor for wireless sensor networks. Distributed and Parallel Databases 29(1-2), 31–85 (Nov 2010)
8. Gray, A.J.G., Galpin, I., Fernandes, A.A.A., Paton, N.W., Page, K., Sadler, J., Kyzirakos, K., Koubarakis, M., Calbimonte, J.P., Garcia, R., Corcho, O., Gabaldón, J.E., Aparicio, J.J.: SemSorGrid4Env architecture – phase II. Deliverable D1.3v2, SemSorGrid4Env (December 2010), http://www.semsorgrid4env.eu/files/deliverables/wp1/D1.3v2.pdf
9. Hart, J., Martinez, K.: Environmental sensor networks: A revolution in earth system science? Earth Science Reviews 78, 177–191 (2006)
10. Henson, C., Pschorr, J., Sheth, A.P., Thirunarayan, K.: SemSOS: Semantic sensor observation service. In: International Symposium on Collaborative Technologies and Systems (CTS 2009) (2009)
11. Geographic information – services. International Standard ISO19119:2005, ISO (2005)
12. Koubarakis, M., Kyzirakos, K.: Modeling and querying metadata in the semantic sensor web: The model stRDF and the query language stSPARQL. In: 7th Extended Semantic Web Conference (ESWC 2010) Part I. pp. 425–439 (2010)
13. Page, K., De Roure, D.C., Martinez, K., Sadler, J., Kit, O.: Linked sensor data: RESTfully serving RDF and GML. In: International Workshop on Semantic Sensor Networks. pp. 49–63 (2009)
14. Pérez de Laborda, C., Conrad, S.: Relational.OWL: a data and schema representation format based on OWL. In: 2nd Asia-Pacific Conference on Conceptual Modelling (APCCM 2005). pp. 89–96 (2005)
15. Raskin, R.G., Pan, M.J.: Knowledge representation in the semantic web for earth and environmental terminology (SWEET). Computers and Geosciences 31(9), 1119–1125 (November 2005)
16. Schimak, G., Havlik, D.: Sensors anywhere – sensor web enablement in risk management applications. ERCIM News 76, 40–41 (2009)
17. Tao, F.B., Campbell, J., Pagnani, M., Griffiths, G.: Collaborative ocean resource interoperability: Multi-use of ocean data on the semantic web. In: 6th European Semantic Web Conference (ESWC 2009). pp. 753–767 (2009)