

# DartDataFlow: Semantic-Based Sensor Grid

Huajun Chen, Zhiyong Ye, and Zhaohui Wu

College of Computer Science, Zhejiang University,  
Hangzhou, China  
{huajunsir, yezhy, wzh}@zju.edu.cn

**Abstract.** We propose DartDataFlow, a semantic-base Sensor Grid to manage sense data. In our system, the data and subscriptions are all represents as RDF graph, so we provide a RDF Graph Filter to filter RDF graph to meet subscriptions. And also, we design an intelligent data dissemination framework to support different cases of data dissemination. This system integrates Semantic Web technology into Sensor Grid and can be used in intelligent transportation systems, military, weather forecast, environment monitor, etc.

## 1 Introduction

Today, plenty of sensor nodes in sensor grid generate rapid, continuous and large volumes of stream data .It is one of the most important issues how users communicate and understand data when sharing and receiving data from sensor grid [1]. A solution for this issue is for each sensor to publish data schema based upon a shared ontology using Semantic Web technology [2]. Therefore, when a sensor node publishes data, users that receive data will be able to interpret the contents without ambiguity. A solution for this issue is for each sensor to publish data schema based upon a shared ontology using Semantic Web technology. Therefore, when a sensor node publishes data, users that receive data will be able to interpret the contents without ambiguity.

On the other hand, publish/subscribe (pub/sub) [1, 3] is a loosely coupled communication paradigm for distributed computing environments. In pub/sub systems, publishers publish data to brokers in the form of event, subscribers subscribe to a particular category of data within the system, and brokers ensure the timely and reliable delivery of published data to all interested subscribers. The advantage of pub/sub paradigm is that information producers and consumers are full decoupled in time, space and flow, so it is well suitable for large-scale and highly dynamic distributed systems.

In this paper, we introduce a semantic-base Sensor Grid named DartDataflow (DDF) to fuse and disseminate sensor data. In DDF system, publishers and subscribers share a RDF-based ontology. This releases the constraints on value-based publish/subscribe systems that publishers and subscribers must share the same data schemas. Different from relational pub/sub systems, the data correlation in our system is transparent to subscribers.

The remainder of the paper is organized as follows. In Section 2, we discuss related work. In Section 3, we introduce Resource Description Framework which is the preliminary of this paper. In Section 4, we introduce the system architecture of the DDF system. In Section 5, 6, we describe Fusion Layer and Dissemination Layer detailedly in the DDF system. Finally, in Section 7, we conclude the paper with a summary.

## 2 Related Work

In the past several years, many data dissemination schemes have been proposed for sensor networks. Based on where data generated are stored, these schemas [4, 5] are divided into three categories: local-storage, data-centric storage-based and external-storage.

- 1) External storage-based data dissemination relies on a centralized base station, which is external to the sensor network, for collecting and storing sensing data. In this schema, data must be sent back and forth between the sensors and the base station.
- 2) Data-centric storage-based data dissemination stored the sensing data at certain nodes within the network. In this scheme, data are still pushed in a predefined manner regardless of queries.
- 3) Local storage-based data dissemination is the dissemination schemes which a source sends data to a sink only when the sink has sent a query for the data. These schemes need a sink-source matching mechanism to facilitate a sink to find the source holding the data of interest.

These schemes include directed diffusion [4], two-tier data dissemination [5], etc.

Pub/sub [3] systems are generally divided into two categories: subject-based and content-based.

- 1) The earliest pub/sub systems are subject-based. In those systems, each data belongs to one of a fixed set of subjects (also called topics, channels, or groups). Publishers are required to label each data with a subject name; consumers subscribe to all data under a particular subject. The techniques for subject-based pub/sub systems have already matured and there are many successful products and specifications such as TIB/Rendezvous [6] from TIBCO [6] and MQSeries from IBM [7].
- 2) The topic-based publish/subscribe variant represents a static scheme which offers only restricted expressiveness. In consequence, as improvements to topic-based solution, content-based publish/subscribe systems are proposed. In these systems, data are no longer divided into different subjects. The subscriber defines a subscription condition according to the internal structure of data; all data that meet the condition will be sent to the subscriber. The internal structures of data are defined as data schemas. Compared with the subject-based pub/sub systems, the content-based systems are more expressive and flexible; it can enable subscribers to express their interests in a finer granularity. Known prototype systems include Gryphon [8], Siena [9], etc.

However, if users are going to share and receive data from heterogeneous and highly dynamic sensors in the sensor grid, then it must be able to communicate and to understand this data. Obviously, these schemes as introducing above are not sufficient for this application. Our goal is to introduce the Semantic Web technologies [5] into the pub/sub system to support data fusion and data dissemination in sensor grid.

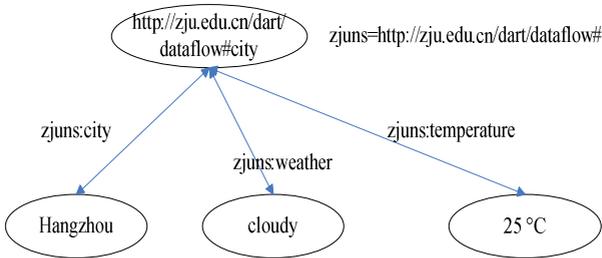
## 3 Resource Description Framework

The Resource Description Framework (RDF) [5] is the basic technology of our system. We first introduce it simply. RDF is a language for representing information

about resources in the World Wide Web .The purpose of RDF is to define the specifications for supporting the use of Metadata about Web resources. This should be accomplished in a manner that will allow a uniform way for the machines to understand and process the information given, together with the flexibility of describing the information for all information publishers. The main concepts of the RDF specifications are the description using Metadata of any kind of resources that can be named via a URI (Uniform Resource Identifier), the interoperability between applications that exchange machine-understandable information on the Web. The specifications also address the ability to enable automated processing of Web resources and finally the capability for different application communities to define their own Metadata Property Set that best serves their needs.

Any expression in RDF is a collection of triples, each consisting of a subject, a property and an object. A set of such triples is called an RDF graph. The nodes of an RDF graph are its subjects and objects. The assertion of an RDF triple says that some relationship, indicated by the property, holds between the things denoted by subject and object of the triple. The assertion of an RDF graph amounts to asserting all the triples in it, so the meaning of an RDF graph is the conjunction of the statements corresponding to all the triples it contains.

Example 1 is a small chunk of RDF in RDF/XML corresponding to the graph in Figure 1:



**Fig. 1.** An RDF graph describing of the weather of Hangzhou

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:zjuns="http://zju.edu.cn/dart/dataflow#">
  <zjuns:City rdf:about="http://zju.edu.cn/dart/dataflow#city">
    <zjuns:name>Hangzhou</zjuns:name>
    <zjuns:weather>Cloudy</zjuns:weather>
    <zjuns:temperature>25 °C</zjuns:temperature>
  </zjuns:City >
</rdf:RDF>
  
```

**Example 1.** RDF/XML describing of the weather of Hangzhou

## 4 Architecture

In sensor grid, data published by heterogeneous sensors differs greatly in formats and semantics. If the pub/sub system is used as a general infrastructure for sensor grid and support different applications, it should have strong expressiveness, i.e.:

- 1) It should support data in different formats and semantics;
- 2) It should provide a powerful subscription language, so that data consumers can easily express their interest in certain data semantics.

Here, we introduce the Semantic Web technologies into the pub/sub paradigm and propose an Semantic-based Publish/Subscribe middleware named Dart-Dataflow (DDF) for manage sensor data .It’s architecture is as Figure 2 shown.

Physical layer is composed of sensor nodes deployed in sensor grid. These sensor nodes collect dynamic and real-time traffic data, weather data, geography data, road surface data and so on. This layer also provides an interface to disseminate these original data into upper layer.

Fusion layer mainly supports to fuse data from heterogeneous sensors and filter data to meet the subscriber. It includes Ontology Service, Semantic Parser Service, Semantic Subscription Service and Semantic Data Filter Service. Services in this layer are provided for sinking data from sensor nodes, querying data schema based on the global ontology and parsing it into RDF-based data and then filtering data to meet the subscribers interesting.

Dissemination layer provides an intelligent and flexible mechanism to disseminating data to the subscriber. It includes Negotiation Service, Performance Monitor Service and Dissemination Service. In this layer, data dissemination is multiple protocols implementation, migration support, easy extensibility.

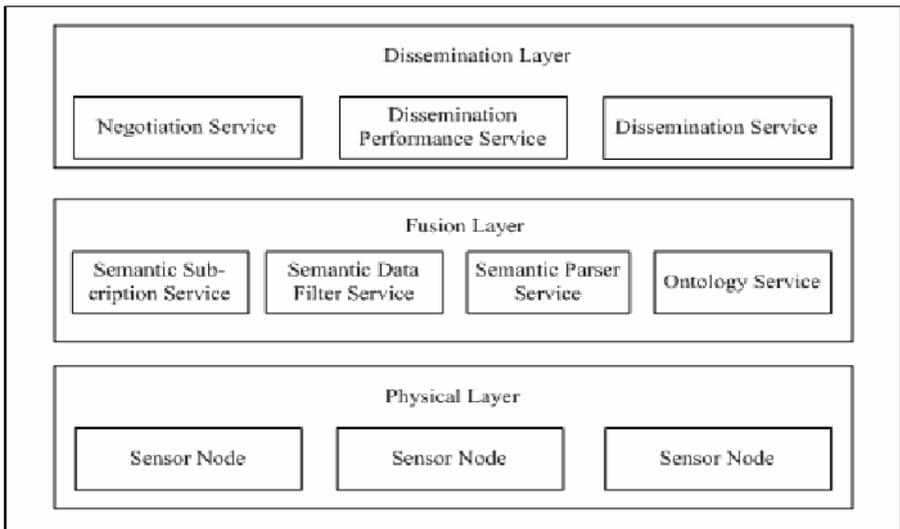


Fig. 2. Architecture of DDF

## 5 Fusion Layer

In DDF, data published by sensors and subscription scripts are both represented as RDF graphs, which is a kind of directed labeled graph and the system filter data with subscriptions both semantically and syntactically. When data is published, it is firstly converted into a RDF graph using Semantic Parser Service based on ontology before further processing. And the same, subscriptions are specified by subscriber and are also represented as RDF graph patterns in Semantic Subscription Service. Then Data Filter Service filters the data to meet the subscriber. As follows we introduce two keys in Fusion Layer.

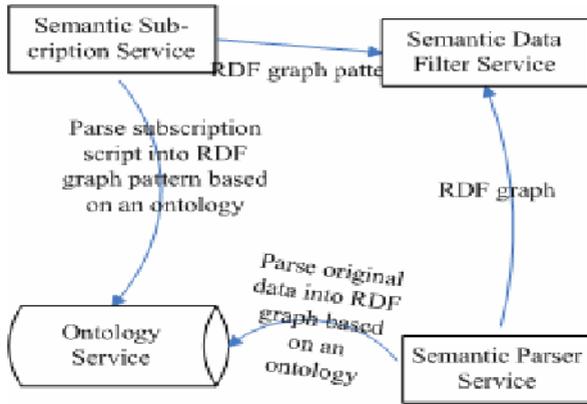


Fig. 3. Mechanism of Fusion Layer

### 5.1 Semantic Subscription Language

In DDF system, data are all represented as RDF graphs; the subscription is in fact a graph pattern which specifies the shape of the graph as well as the constraints on some nodes and arcs in the graph. So we must define a subscription language to support semantic subscription. Here we define a subscription language based on a number of query languages such as SPARQL [10], RDQL [11].

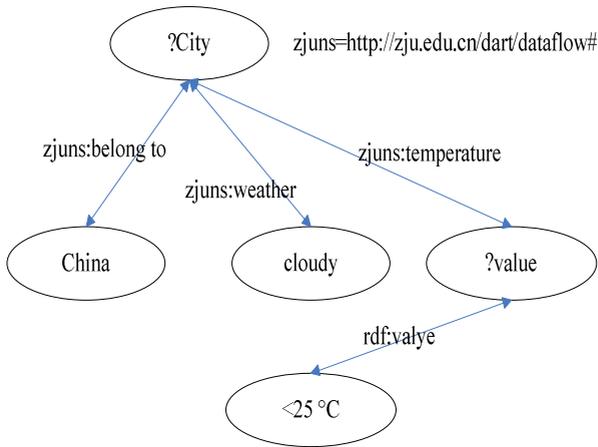
Our subscription language is provided for accessing RDF graphs. A subscription mainly consists of two parts, the Select clause and the Where clause. The Select clause names the variable of interest to the subscriber; and the Where clause has one or more triple patterns which are composed of a subject, predicate and object delimited by parentheses, and for confining the value of object, the Where clause always defines some filter expressions.

For example, in the Weather Forecast system, if someone want to know all cities which are in China with the temperature is below 25 centigrade thermometer and the weather is cloudy, he can express it as:

```

Select ?city
Where (?city zjuns:belongto China)
      (?weather zjuns:weather cloudy)
      (?temperature zjuns:temperature ?value)
And ?value<25
Using zjuns for <http://zju.edu.cn/dart/dataflow>
    
```

**Example 2.** An example of a subscription language



**Fig. 4.** An example of graph for the subscription language

As above mentioned, each subscription can represent as a RDF graph pattern in our system, so preceding subscription can be represent as the corresponding RDF graph pattern in Figure 4.

### 5.2 RDF Graph Filter

In the DDF system, we must filter the incoming data to meet the subscription .We design a RDF Graph Filter to support this task. Subscriptions and data are all represented as graphs in our system, and if every node and arc in a data graph can be mapped to a corresponding node and arc in the subscription graph, the data is said to meet the subscription, otherwise the data can not meet the subscription. Therefore, the filtering problem is a specific kind of graph isomorphism problem.

Following we focus on our graph isomorphism arithmetic. For example, our subscription graph  $G(s)$  is as Figure 5 shown.

We suppose a RDF graph can show as:

$G = (H, V, E)$ , where

$H = \{H\}$ ,  $H$  is the home vertex;

$V$  is the set of the vertexes of the RDF graph and while a vertex has no outgoing edge, we define it is a specific values or an expression;

$E = \{R_i = E(u,v): u,v \text{ belong to } V, \text{ and the direction of the edge is from } u \text{ to } v. \}$ .

Then we can represent  $G(s)$  as

$G(s) = (R_s, V_s, E_s)$ , where

$R_s = \{H\}$ ,

$V_s = \{H, A, B, C, D, E\}$ ,

$E_s = \{R_1 = E(H,A), R_2 = E(H,B), R_3 = E(H,C), R_4 = E(B,D), R_5 = E(B,E), R_6 = E(C,E), R_7 = E(D,E)\}$ .

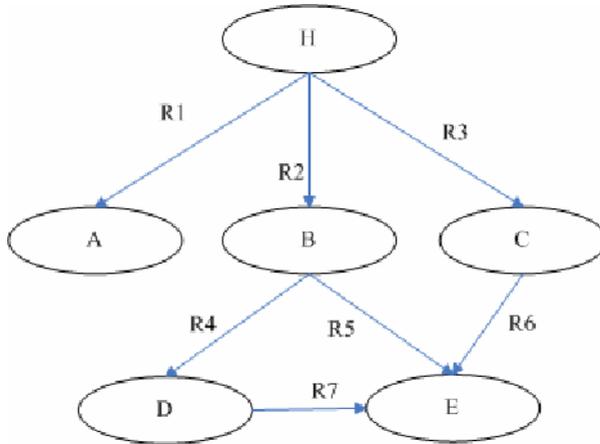


Fig. 5. A subscription graph

**Arithmetic(G):**

- for each vertex  $u \in V_d$
- do flag(u) ← false
- Visit(H)
- for each vertex  $u \in V_d$
- do if flag(u) = false
- return fale
- return true

**Visit(u):**

- do flag(u) ← true
- for each  $v \in \text{adj}(u)$
- do if outdegree(v)=0
- then do if exp(v) = false
- break;
- else do flag(v) ← true
- else Visit(v)
- 

Fig. 6. The filtering arithmetic

When a data which it is represented as RDF graph is coming, we suppose it express as  $G_d = \{H_d, V_d, E_d\}$ . Then we can introduce our graph isomorphism arithmetic as Figure 6 show.

$exp(v)$  is a boolean expression which judge whether the  $v$  in  $G(d)$  meet the constraints of the corresponding vertex in the  $G(s)$ .

## 6 Dissemination Layer

In DDF system, Dissemination Layer is used to disseminate the RDF-base filter data to the corresponding subscriber effectively. Following we introduce the design of the Dissemination Layer detailedly.

As we all know, different transport protocols are suited for different cases. For example, TCP works best where reliable delivery is at a premium but if subscriber can sustain losses in delivery and more concentrate on the latencies, UDP is a better choice. And also, data dissemination in network is influenced by different condition and different network. So in Dissemination Layer, we design more than one Dissemination Services and each service implements a transport protocol.

DDF system provides a Negotiation Service to negotiate the best transport protocol for data dissemination. If negotiate is successful, the system specify the corresponding Dissemination Service to the subscriber.

Considering that the network capability is unstable. So we design Dissemination Performance Service to monitor the dissemination performance. This service can specify a constraint on the performance factors and specify the migration to anther Dissemination Service which implement a more suited transport protocol when this constraint is satisfied. For example where dissemination using UDP is not feasible due to high loss rates user can switch to TCP and similarly, user may switch to UDP by reason of bandwidth and latency constraints.

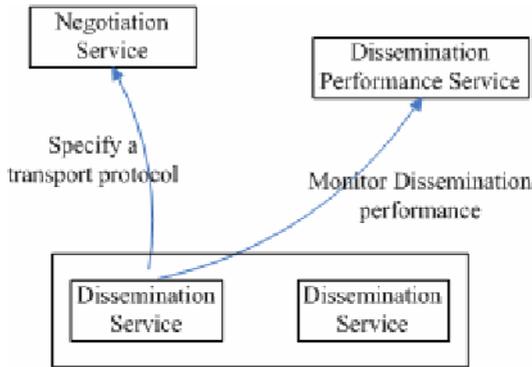


Fig. 7. Mechanism of Dissemination Layer

## 7 Conclusion

DDF system is mainly composed of Physical Layer, Fusion Layer and Dissemination Layer. In Fusion Layer, We define a semantic subscription language to support com-

plex subscriptions. And also, we design a Semantic Data Filter in Fusion Layer to meet the subscriber individual subscription. In Dissemination Layer, we propose a new intelligent data dissemination framework that is appropriate for this semantic-base pub/sub system.

Our future work includes optimization of data filtering arithmetic based on RDF graph to support numerous subscriptions more effectively.

## References

- [1] Bass.T: The federation of critical infrastructure information via publish-subscribe enabled multisensor data fusion. Information Fusion, 2002. Proceedings of the Fifth International Conference on Volume 2, 8-11 July 2002 Page(s):1076 - 1083
- [2] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. Scientific American, 284(5): 34-43. May 2001
- [3] P. Eugster, "Publish/Subscribe," ACM Computing Surveys, vol. 35, no. 2, Jun. 2003, pp. 114-131.2.
- [4] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directeddiffusion: A scalable and robust communication paradigm for sensor networks. In Proc. of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom 2000), Boston, Massachusetts, August 2000.F.
- [5] Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang. A two-tier data dissemination model for large-scale wireless sensor networks. In Proc. of the Eighth ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom2002), Atlanta, Georgia, USA, September 2002.
- [6] TIBCO Corporation, "TIB/Rendezvous", white paper. <http://www.rv.tibco.com/rvwhitepaper.html>.
- [7] IBM. Internet Application Development with MQSeries and Java, February 1997.
- [8] M. K. Aguilera, R. E. Strom, D. C. Sturman, M. Astley, and T. D. Chandra: Matching events in a content-based subscription system. In: Proceedings of the Eighteenth ACM Symposium on Principles of Distributed Computing (1999) 53-61
- [9] Carzaniga, D. S. Rosenblum, and A. L. Wolf: Design and evaluation of a wide-area event notification service. ACM Trans. on Computer Systems 19(3) (2001) 332-383
- [10] SPARQL: Query Language for RDF: <http://www.w3.org/TR/2004/WD-rdf-sparql-query-20041012/>
- [11] RDQL: RDF Data Query Language. <http://www.hpl.hp.com/semweb/rdql.htm>