A Middleware for Integrating Wireless Sensor Networks with the Sensor Web

Yunfeng Nie, Zhenhua Wu, Pengliu Tan, Zhongmin Li, Sheng Zhang

School of Information Engineering, Nanchang Hangkong University, Nanchang, 330063, P R China
Telephone: (+86 791 8953418)
Fax: (+86 791 8953418)
Email :{ nieyunf@gmail.com, zhuawu@126.com, pltan@mail.hust.edu.cn, zhongmli@nchu.edu.cn, zwxzs168@126.com}

1. Introduction

A wireless sensor network consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. The main goal of a WSN is to cooperatively collect data from the environment and send it to a main location where the data can be observed and analyzed. Until recent years, research in the WSN domain has mainly focused on routing, locating, data aggregation and energy conservation inside a single sensor network while the integration of multiple sensor networks has only been studied to a limited extent. However, as the cost of wireless sensors reduces rapidly we can soon expect that huge numbers of autonomous sensor networks well be deployed and operate by different organizations all over the world. In the real world application scenarios of WSNs, there are many applications need one or more WSN data streams for analyzing and decision makings, so how to integrating heterogeneous sensor networks and its produced huge amount of data streams will soon become an important challenge. In order to address these issues, Sensor Web concept is proposed. A definition of Sensor Web given by NASA is that A Sensor Web is a system of intra-communicating spatially distributed sensor pods that can be deployed to monitor and explore new environments. Sensor Web is a revolutionary concept towards achieving a collaborative, coherent, consistent and consolidated sensor data collection, fusion and distributed systems.

Although the Sensor Web vision has described a high level abstract view that allows users to sharing, accessing and analyzing sensor networks and its produced data streams over the Web, but how to realizing this vision there existed a lot of key technical challenges. Due to the heterogeneity of WSNs, such as various OS, protocol, hardware, and programming language, one of the biggest barriers is how to integrating heterogeneous sensor networks. This requires platforms which enable the dynamic integration and management of sensor networks and the produced data streams.

This work focuses on providing a flexible middleware to resolve the problems mentioned above effectively. Our Wireless Sensor Network over Web (WSNOW) middleware is built upon a uniform set of operations and standard sensor data representations defined by Open Geospatial Consortium (OGC) Sensor Web Enablement (SWE) which includes specifications of interfaces, protocols and encodings that enable discovering, accessing, and obtaining sensor data as well as sensor-processing services. The goal of WSNOW is to allowing a user to access WSNs directly from a web browser based on standard SWE Web Services. At last of this work, the main system's building blocks and functionalities are described.

2. The SWE Framework

As mentioned above, a key challenge in realizing the Sensor Web is how to flexibly integrate different types of data streams that are produced by various sensor networks. SWE initiative initiated by the OGC extends prominent OGC Web services by providing additional services for integrating web accessible sensors and sensor systems. The SWE architecture currently defines four Web service specifications and two models and encodings for observations and sensors respectively. The Web services, known as Sensor Observation service for data access, Sensor Planning Service for sensor tasking, Sensor Alert Service for alerting and Web Notification Service as a data transport protocol transformer describe the operational part of the framework. The data models and encodings Observation & Measurement (O&M) and Sensor Model Language (SensorML) are used as data and metadata exchange protocols. Although the SWE approach provides an open distributed computing infrastructure where sensor resources can be published, automatically discovered and accessed . However there are some major gaps when it comes to dynamic integrating multiple heterogeneous sensor networks and its produced data streams.

3. The WSNOW Architecture

WSNOW adopts a container-based architecture for hosting heterogeneous wireless sensor networks. Like an application servers, WSNOW provides a context in which wireless sensor networks can easily and flexibly be deployed and configured by hiding most of the system complexity in the WSNOW Server. By using the XML deployment descriptor, wireless sensor networks can be deployed and reconfigured in WSNOW instances at runtime. Once deployed, any wireless sensor network can be accessed through a container managed object which is called Virtual Sensor Network (VSN).VSN is the main component in WSNOW, it can execute commands upon a sensor network, then receive and process the data and finally store it. Fig.1 shows the layered architecture of our WSNOW middleware.

Each WSNOW server instance can host a lot of VSN according to its resources capacity and responsibility. The Adaptive Layer provides a series of drivers for communication with the physical sensor networks. A driver is a component that does the data acquisition for a specific type of device via Http, Socket, GPRS or Serial. By providing extensible drivers, WSNOW can receive data streams from various heterogeneous wireless sensor networks, and users can easily develop and deploy their own drivers to the WSNOW server at runtime. The Virtual Sensor Network Layer contains three important components. The Virtual sensor network manager (VSNM) is responsible for providing access to the VSNs, managing the delivery of sensor data steam, and providing the life-cycle management for VSN. The Persistent manager (PM) is responsible for managing the streams, providing and managing persistent storage for data streams. The Event manager is responsible for producing and publishing the data stream related events, such as data missing event, data arriving event and so on.

Command Execution Layer relies on all of the above layers and its core components are Command manager, Query manager and Notification manager. The Command manager is responsible for intercepting client request and forwarding the request to a command object that can process the request and return the finally response to the client. Query manager (QM) is in charge of SQL parsing, query planning, and execution of queries and it also maintains a query repository manages all registered queries (or subscriptions). The Notification manager deals with the delivery of events and query results to registered, local or remote Sensor web applications or users. The Notification manager has an extensible architecture which allows the user to largely customize its functionality, for example, having results mailed or being notified via SMS. The OGC Service Provider layer provides access functions for Sensor Web applications via the OGC SWE web services, such as SOS, SPS, SAS and WMS (Web Map Service).

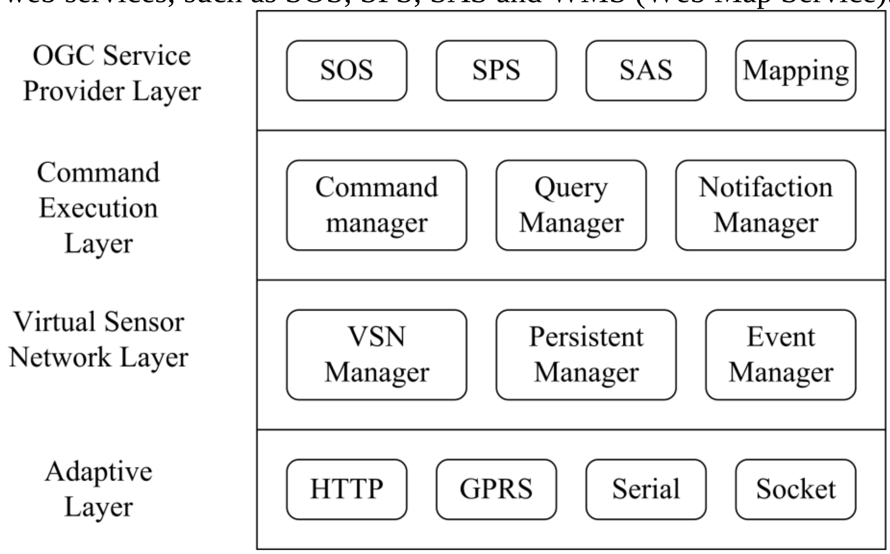

Fig.1 WSNOW middleware architecture

4. Conclusion

In this work, we have presented a Wireless Sensor Network over Web (WSNOW) middleware. we used the OGC Sensor Web Enablement framework as a starting point . The new framework provides far more flexibility than the concurrent system that primarily concentrates on integrating multiple heterogeneous sensor networks and its produced data streams. The Adaptive Layer provides a series of drivers for communication with the physical sensor networks, through these abstract shields the low-level details of WSNs away from upper level applications. The Virtual Sensor Network Layer provides a key abstract called VSN, which is responsible for providing access to the physical WSNs, managing the delivery of sensor data steam, and providing the life-cycle management for WSN instances. Command Execution Layer is responsible for intercepting client request and forwarding the request to a command object that can process the request and return the finally response to the client. The OGC Service Provider layer provides access functions for Sensor Web applications via the OGC SWE web services, such as SOS, SPS, SAS and WMS (Web Map Service).

5. Acknowledgements

The research is supported by the National Natural Science Foundation of China (No. 41101426), the Natural Science Foundation of Jiangxi Province (No. 20122BAB211026) and the S&T plan projects of education department of Jiangxi province (No. GJJ12429).

6. References

Dargie, W. and Poellabauer, C., 2010, Fundamentals of wireless sensor networks: theory and practice, John Wiley and Sons, 168-183.

Sohraby, K., Minoli, D., Znati, T., 2007, Wireless sensor networks: technology, protocols, and applications, John Wiley and Sons, 203-209.

Delin, KA, 2002, The Sensor Web: A macro-instrument for coordinated sensing, Sensors, vol.2, 270-285.

Delin, K.A.,Jackson, S.P,2001, The Sensor Web: A New Instrument Concept, SPIE's Symposium on Integrated Optics, San Jose,CA.

Moodley, D., Simonis, I., 2006, A New Architecture for the Sensor Web: The SWAP Framework, workshop of the 5th international Semantic web conference ISWC.

Botts,M. et al.,2006,OpenGIS Sensor Web Enablement Architecture Document. Open Geospatial Consortium, OGC.

Botts, M.,2007, OGC Implementation Specification: OpenGIS Sensor Model Language (SensorML), Technical Report, Open Geospatial Consortium.

Botts, M., Percivall, G., Reed, C. and Davidson, J. ,2008, OGC Sensor Web Enablement: Overview and High Level Architecture, in S. Nittel, A. Labrinidis and A. Stefanidis (Eds.), GeoSensor Networks, Vol. 4540 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 175–190.

Nath, S., Lie, J., Zhao, F., 2006, Challenges in Building a Portal for Sensors World-Wide. Microsoft Technical Report MSR-TR-2006-133; Microsoft Research: Redmond, WA, USA.

Heinzelman, W.B., Murphy, A.L., Carvalho, H.S., Perillo, M.A., 2004, Middleware to support sensor network applications. IEEE Network 2004, 18, 6-14.