Volume 2, Issue 10, October-2015, pp. 902-908 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

DEVELOPING AN ENTERPRISE ENVIRONMENT BY USING WIRELESS SENSOR NETWORK SYSTEM ARCHITECTURE

P. G. V. SURESH KUMAR[1], SEELAM SOWJANYA[2]

[1]Centre for ITSC, School of Electrical & Computer Engineering

Addis Ababa Institute of Technology, Addis Ababa University, Ethiopia

[2]Priyadarshini Institute of Technology and Science, Chinthalapudi, India

Email: pendemsuresh@gmail.com

Abstract: - The greatest advance in building monitoring is through the use of wireless sensor and a most important aspect of efficient facilities management is that of monitoring the surrounding environment and collecting accurate and up-to-date information regarding the status of the facilities. In this paper we propose to exploit a Service Bus for developing an enterprise networking environment that is used for integrating services with other networking and SOA for the purpose of information sharing and monitoring, controlling, and managing the enterprise environment. In this proposed WSN System architecture improves scalability, reliability and continuous data availability.

Keywords – Building Monitoring System (BMS), Facilities Management, Service Oriented Architecture (SOA), Wireless Sensor Network (WSN), Service Bus

———————————— ◆ ————————————

1. INTRODUCTION

The Building Management Systems (BMS) are used to both improve the indoor climate in buildings and to reduce the operational costs. Originally, BMS mostly consisted of heating, ventilation and air-conditioning (HVAC) systems. To further increase management and reduce costs, lighting, safety, security, and transportation supervision have been integrated into BMS [1]. Wireless sensor nodes are useful for supervision of building management system typically; sensor nodes transport the measured data to a base station using multi hop communication. Sensor nodes are typically powered by batteries. Wireless sensor networks (WSN) consist of small sensor nodes that sense the building environment, perform computations, and communicate with Control unit using building networking system.

Integrating WSN and BMS has a number of advantages. Wiring is avoided. New buildings can therefore be equipped with a BMS based on wireless sensor network technology [2]. It is also possible to extend an existing BMS in order to increase the sensor coverage. Furthermore, wireless sensors can also be installed more easily in unapproachable places such as at high heights. This enables the building manager to have a single point of interaction to control all these facilities in an integrated manner. In this paper propose a Service oriented architecture (SOA) for integrating the Building Monitoring System with Service Bus. The Service bus is integrating the more than one distinct network with corresponding enterprise network.

2. RELATED WORK

Existing building systems are tightly coupled with the sensors they utilize, restricting the generic extensibility of the overall architecture. The emergence of Wireless Sensor Networks (WSNs) has brought significant benefits as far as monitoring is concerned, since they are more cost-efficient (because of the lack of wired installations) compared to their wired counterparts, while additionally they allow for flexible positioning of the sensor devices. In line with the established move towards integrated enterprise architectures, it is beneficial to consider the WSNs within that scope. The WSN architecture should subsequently be designed in such a manner, so as to allow its straightforward integration to the building enterprise infrastructure.

The existing system provides a detailed architectural framework for WSNs to be integrated seamlessly with enterprise applications over an SOA infrastructure, within the scope of facilities management. An overall WSN architecture serves a multitude of facets such as follows:

• optimizes the information flow and accessibility, allowing all authorized systems and applications, which require a given source of information to have shared access to that information stream;

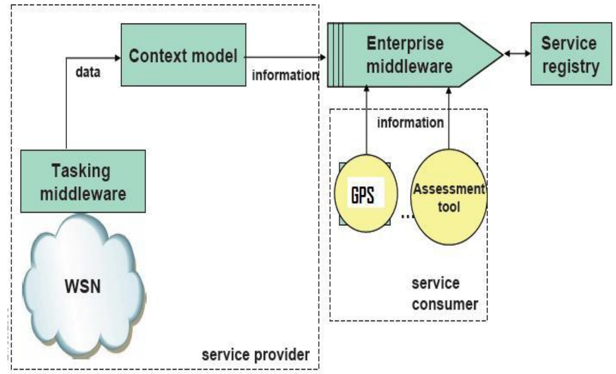
• increases the coverage, resolution, and accuracy of the information awareness made available to human and automated decision makers, by developing an information web, which all systems and applications can access and exploit through standard interfaces;

• develops a new design approach to high-level facilities management applications, by allowing composition of data, information and operational services into added value monitoring and decision making support tools.

3. PROPOSED WORK

In this respect, the notion of facilities management can be defined as an integrated process to support and improve the effectiveness of the primary activities of an organization by the management and delivery of agreed distributed support services for the appropriate environment that is needed to achieve its set objectives

The most prominent approach towards an enterprise-wide, open framework, which satisfies the aforementioned requirements, is that of service oriented architectures (SOAs).The proposed WSN system Architecture Shows Fig.1, it describes about the service provider, service consumer and enterprise middleware with service registry. The main duty of service provider is to collect the sensor data from Wireless Sensor Network (WSN) through the tasking middleware and stored in to web server database (DB).



GPS-Gas Pipeline System

Fig. 1 Proposed WSN System Architecture

3.1. Wireless Senor Networks (WSN)

Wireless Sensor Networks which is a type of wireless network consist of small nodes with capabilities of sensing physical or environmental conditions, processing related data and send information wirelessly. The ideal wireless sensor is networked and scalable, consumes very little power, is smart and software programmable, capable of fast data acquisition, reliable and accurate over the long term, costs little to purchase and install, and requires no real maintenance. A key feature of any wireless sensing node is to minimize the power consumed by the system. Generally, the radio subsystem requires the largest amount of power. Therefore, it is advantageous to send data over the radio network only when required. This sensor event-driven data collection model requires an algorithm to be loaded into the node to determine when to send data based on the sensed event. Additionally, it is important to minimize the power consumed by the sensor itself. Therefore, the hardware should be designed to allow the microprocessor to judiciously control power to the radio, sensor, and sensor signal conditioner.

3.2 Tasking Middleware

The tasking middleware is the architectural entity that implements the functionality exposed by the WSN service interface. It is actually client-server architecture with the client side residing on sensor nodes and the server side existing on the Virtual Gateway. The tasking middleware receives as input high-level service requests (known as WSN queries) from enterprise entities via the enterprise networking architecture, i.e., WSNWS interface, determines at the Virtual Gateway what data and processing are required to provide the service, tasks the relevant sensor nodes to perform sensing and processing (known as sensor tasks), collects the resulting data at the sensor node, sends the data back to the Virtual Gateway where it is contextually processed and stored it in a database (DB) for record keeping reasons and finally responds to the original service request.

3.3. Context Model

To ensure scalability and reliability of the WSN we consider that there might be the need to have more than one gateway for a single WSN (as shown in Fig. 2 for example), since for example the number of nodes could rise significantly or the main gateway could potentially fail. Another reason to allow for multiple gateways is the survivability of the WSN, since there would be a more power efficient distribution of resources consumption. The reason for this is the fact that not all nodes should use the same path to route their data towards the gateway, which would drain the resources of particular nodes rapidly and hence lead to node failures. In short, gateway is the core of the proposed WSN architecture as it allows for sensor data collection and reporting between Wireless sensor Networks (WSN) domains to web server.

3.4. Enterprise Tasking Middleware

In the light of aforementioned development process, an essential construct in SOA named enterprise service bus (ESB) [1] is required to facilitate the integration and choreography of diverse services and support event-based messaging among services as well. ESB is a reference architecture which can be implemented by middleware technologies to facilitate communication among different services and applications. ESB not only includes a collection of key services to assist the SOA developers in building and managing the services but it also encloses a set of services to support better decision-making with real-time information. Since the main objective of ESB is to support the development of SOA applications and to provide a flexible connectivity for services and applications, the components or services in the ESB can be varied according to the system or customer requirements. The following illustrates some services that the ESB could or should have. An ESB should contain a set of infrastructural services to support the other services in the architecture by optimising system throughput, availability and performance. In the ESB, it is essential to have an appropriate building management that can manage and secure services, applications, and resources.

3.5. Service Registry

An optional service registry can be used within the architecture to help the client automatically configure certain aspects of its service client. The service provider pushes changes regarding the service's details to the registry to which the consumer has subscribed. When the changes are made, the service consumer is notified of these changes and can configure its service client to talk to the service. It provides a standards-based mechanism for dynamic discovery of services and their associated policies during runtime and supports heterogeneous services from any vendor.

The Assessment tools are devices or activities such as charts, reports, or brainstorming sessions, that help in monitoring the effectiveness of the system. Since the field of environmental assessment tools for buildings is vast, the aim of this study is to clarify that field by analyzing and categorizing existing tools. For example, the tools are designed for assessing different types of buildings, and they emphasize different phases of the life cycle.

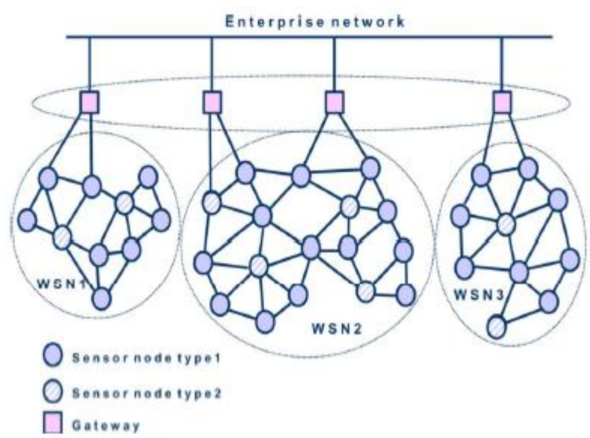


Fig. 2. WSN exemplary deployment architecture.

4. ANALYSIS

In this proposal Gas Pipeline System utilizes wireless sensor networks. However, especially in distributed Gas Pipeline System processes we expect a growing need for information from multiple sources of specialized sensor networks. In the following we explore how sensor networks can contribute to a Gas Pipeline System and what technical problems still have to be solved to enable an effective use of WSN technology in this application area.

4.1 APPLICATION ANALYSIS

Transmission pipelines for gas and oil are an important part of national energy-transportation infrastructure vital to the national economy. Because these Pipeline Systems are operated at high pressure, pipeline failure can cause severe damage to human health and property and interruption of gas or oil supplies.

Pipeline inspection technologies using sensor networks have drawn significant attention, for example, in the applications of natural gas pipeline inspection and monitoring by acoustic sensors [3], [4]. In this paper, we discuss how sensor networks can detect, localize, and quantify bursts, leaks and other anomalies in general pipeline systems. In general, pipeline defects can occur in the manufacture, construction, and operation processes. In this paper, we focus on the operational defects that encompass internal corrosion, external corrosion, erosion, fatigue, third party damage, denting and buckling. The leading cause of pipeline incidents is damage by digging near existing pipelines. Corrosion sometimes results from excavation damage, which, while not severe enough to trigger a puncture or failure of the pipeline, could create weaknesses in the pipeline. Such a weakness later renders the pipeline more susceptible to corrosion. To ensure the continued safe operation of the transmission pipelines, continuous monitoring or periodic assessment of the integrity of the pipelines is necessary. In pipeline monitoring and inspection, the ultimate objective is to identify the locations that have defects, and obtain an accurate measurement and assessment of the defects so that human operators can take appropriate actions to prevent further damage. The key objective is to develop a SOA to investigate the feasibility of developing a continuous, remote, and real-time monitoring and inspection system using sensors on web that can provide early detection and early warning of defects, such as corrosion and leaks, for pipeline systems.

A pipeline monitoring and inspection system has a long list of tasks to accomplish. For example, [3], for natural gas pipelines, these tasks include:

1. Measuring wall thickness;

2. Detecting gas contamination in pipeline;

3. Measuring velocity and flow of gas;

4. Detecting presence of gas leaks;

5. Determining the variation in pipe cross-section;

6. Determining structural defects in pipes, etc.

To achieve these goals, we rely on various sensors and each type of sensors has its unique feature and operational condition.

Sharing sensor networks yields several advantages for Gas Pipeline System. Sensor networks allow a continuous and direct monitoring of enterprise network process. Sensor networks may automate processes by seamlessly share detailed process information at the point where it is really needed. Next, monitoring and automatic processing of current data from sensor data across large sites allows manual inspections to be reduced to a minimum in Gas Pipeline System. Manual action is then only required in cases of unrecoverable exceptions. Each part of the process resides in a different location and utilizes different types of sensor nodes. A deployed sensor network of sensors around the items monitors, that certain ranges are met.

4.2 Technical Analysis

4.2.1 'SOA' and the Enterprise Service Bus

The Service Oriented Architecture (SOA) provide s an approach to describe, discover, and invoke services from heterogeneous platforms using XML and SOAP standards. The term 'service' not only represents a software system but also refers to hardware and any devices that can be used by human beings.

The SOA will communicate WSN for reading sensed data and communicate to ESB for transferring sensor responses. The SOA will handle the request as well as response in the form of XML.The ESB handle the request and response in the form of XML.

4.2.2 Manageability

The likely distributed nature of the ESB has obvious implications when it comes to management of deployed systems. Whilst the product is distributed, support is best provided from a single point of control, from which system definitions can be maintained, adapted and then distributed across the full ESB deployment. When the system is running, it is obviously important to monitor the state and behaviour of the system, and to this end some form of ESB management must be provided that is able to monitor and identify potential problems before they become critical issues. Crucially, in B2B or interdepartmental deployments this information must of necessity be available across these boundaries. In a Gas pipe line system we have no of sensors and no of different types of users are available for sharing the sensed Information. Those system is managed by separate module to monitor all activities in the system

4.2.3 Robustness, Fault Avoidance and Tolerance

Give the pervasive nature of the ESB, and its position at the heart of so many operations and business processes, the need for a truly robust solution is a given. This 'robustness' manifests itself in two ways:

- Fault avoidance by which the ESB aims to prevent problems occurring.
- Fault tolerance by which problems, when they do occur, have little impact on levels of service.

4.2.4 Scalability and Performance

As a key element of the IT infrastructure and a platform upon which multiple solutions will be based, the scalability and performance of an ESB are of critical importance. After the initial adoption of an ESB it is common to 'discover' new requirements and candidates for integration solutions, usually due to the initial success of the first integration projects. Thus adoption can accelerate rapidly across the organization. The ESB that may be initially chosen for a small project must be equally capable of supporting this increased workload. A number of features and technologies can assist in this area and should be considered when selecting an ESB: Support for asynchronous messaging and multi-threading, thus enabling business processes to occur in parallel and consequently with greater efficiency and intelligent load-balancing capabilities to deal with spikes in demand and unusually large document or data volumes within the system.

In future the sensor will add on our system and also new user can get those details. Loose coupling model will provide this provision. The new sensor data will easily capture by the application server. The application server is responsible for giving those data to user end. In our system we are maintaining the proxy system for critical situation to handle server failure. So we can improve data always available

4.2.5 Security

Clearly any solution that spans multiple departments and organizations must implement or support security measures that protect the integrity of the information and business processes within the system. The issue of security is a central concern in the IT industry. There are a number of solutions in the marketplace which any ESB should be able to use for the purposes of, for example, user authentication – ensuring that only those with relevant privileges can access the information within the solution, and indeed control the deployment of the ESB itself through system management tools. The Request and response are handled in form of XML with encrypted format so it provides more security.

4.2.6. Authentication

When designing systems that communicate data across multiple users, and thus security domains, controlling the information flow is an obvious problem. Therefore, we finally see the need for data centric access control, that ensures that only data critical to the distributed process is forwarded. The registered user can give the request and can receive the response.

4.2.7. Context Aware

The end user can get the details of sensed data anywhere and anytime. The user may not present at the sample place in the plant. The users may roam on inside the plant. In inside the plant the user need to get the sensed data through Mobile, PDA, laptops and other internet enabled devices. This can be achieved by the system is deployed on an external web server.

5. IMPLEMENTATION

5.1. Service Layer and SOA

The different view of SOA [9] is the essential infrastructure that supports the Service Layer and plays a very important role in presenting the core middleware components as services for client access. The main reason for Sensor Web relying heavily on SOA is because it simplifies integration of distributed heterogeneous systems which are loosely coupled. Moreover, SOA allows the services to be published, discovered and invoked by each other on the network dynamically. All the services communicate with each other through predefined protocols via a messaging mechanism which supports both synchronous and asynchronous communication models. Since each sensor network differs from each other, trying to put different sensors on the web, and providing discovery and accessibility requires the adoption of SOA.

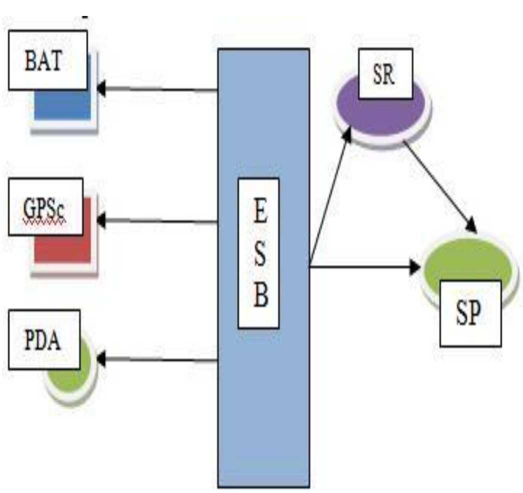
5.2. Sensor Design using TtinyOS

The prosed WSN System architecture has the ability of dealing with heterogeneous sensor networks that may adopt quite different communication protocols including radio, blue tooth [5], [6] and ZigBee/IEEE 802.11.4 [7], [8] protocols. As a result, it is quite desirable that the operating system level support for sensor networks can largely eliminate the work of developing device drivers and analysing various protocol stacks directly in order to concentrate on higher-level issues related to the middleware development.

TinyOS is an operating system specifically designed for sensor networks. It has a component-based programming model, provided by the nesC language, a dialect of C. TinyOS application [10] is not an OS in the traditional sense. It is a programming framework for embedded systems and set of components that enable building an application specific OS into each application.

5.3. Enterprise Service Bus(ESB) using Mule and Service Registry

Mule is a light-weight Enterprise Service Bus (ESB) and integration framework [11]. It can handle services and applications using disparate transport and messaging technologies. The platform itself is Java-based but can broker interactions between other platforms such as .Net using Web Services or Sockets. The Fig.3 shows dataflow between client and SOA via ESB.



SR-Service Registry SP-service Provider

BAT-Building Assessment Tool

GPSc-Gas Pipeline System clients

Fig. 3. Dataflow between clients and SOA via Enterprise Service Bus (ESB) with Service registry

The architecture is a scalable, highly distributable object broker that can seamlessly handle interactions across legacy systems, in-house applications and almost all modern transports and protocols. Enterprise service buses are the preferred tools for integrating systems with heterogeneous data interchange interfaces and based on a wide array of technologies, from COBOL to CORBA to JEE.

Oracle Service Registry provides a 'DNS'-like reference for SOA. A fully compliant UDDI v3 registry, Oracle Service Registry provides a standards-based interface for SOA runtime infrastructure to dynamically discover and bind to deployed service end points. As part of the Oracle SOA Governance solution, Oracle Service Registry bridges the gap between the design time and runtime environments through automated synchronization with Oracle Enterprise Repository and Oracle SOA Suite.

CONCLUSION AND FUTURE WORK

We have discussed in this paper an architectural framework that enables the integration of wireless sensor networks in overall facilities management enterprise architecture. Scalability, extensibility, and reliability that are extremely important in the wireless domain have been taken into account; while in parallel security should also be supported.

Our future work focuses on system level evaluation of the proposed architecture in our experimental WSN test bed in order to validate the development efforts and furthermore to examine areas of potential deployment of such an approach. The overall aim for the SOA-based architecture has been to create an appropriate building services environment that can maximize benefits, reduce the costs, and be reliable, stable and usable.

REFERENCES

[1] F. Österlind, E. Pramsten, D. Roberthson, J. Eriksson, N. Finne, and T.Voigt, “Integrating building automation systems and wireless sensor networks,” in *Proc. 12th IEEE Conf. Emerging*

Technol. Factory Autom. (ETFA), Sep. 2007, pp. 1376–1379.

[2] S. Hadim and N. Mohamed, "Middleware: Middleware Challenges and Approaches for Wireless Sensor Networks," in IEEE Distributed Systems Online, vol. 7, no. 3, 2006.

[3] H. W. Park, H. Sohn, K. H. Law and C. R. Farrar, "Time reversal active sensing for health monitoring of a composite plate," Journal of Sound and Vibration, vol. 302, no. 1-2, 17 April 2007, pp. 50-66.

[4] Agrawal, "A theoretical, numerical and experimental investigation of guided wave propagation in hollow cylinders." Master thesis, Carnegie Mellon University, September 2008.

[5] Bisdikian, "An overview of the Bluetooth wireless technology," IEEE Commun. Mag., vol. 39, no. 12, pp. 86–94, De