Research on the Message-Oriented Middleware for Wireless Sensor Networks

Lifang Zhai[1], Chunyuan Li[2], Liping Sun[1]
[1]Dalian Jiaotong University, Dalian 032302, China
[2]University of Science and Technology Beijing, Beijing 100876, China
zlf_dl@yahoo.cn

***Abstract*—The Propagation of information in wireless sensor networks (WSN) is critical to the communication. It is necessary and important to design a suitable communication primitive. Among these communication primitives, one distinguished primitive is Publish/Subscribe Communication Generics. This paper introduces current middleware of wireless sensor networks; analyze layer structure and problems of middleware. Then we design a message-oriented middleware system named WMOS (Wireless Message-Oriented System) which is developed on TinyOS. WMOS has three features mainly: first, WMOS is a distributed middleware system which is well-adapted to scalable system. WMOS utilizes conception of fully distributed and modular to make it appropriate for practical implementations. Second, we supply self-adapt multilevel QoS service to save power effectively. Third, we introduce topic/content double mode based on XML matched in WSN application. It guarantees effective query information than other middleware.**

***Index Terms*—wireless sensor network, publish/subscribe, middleware, TinyOS**

1. INTRODUCTION

With the rapid development and increasing maturation of low-power consumption Mobile communications technology, Embedded Computing technology and Micro-Sensor technology, it becomes reality that large amount, low-cost micro-sensors comprise sensor networks via mobile links. Compared to traditional wired sensor networks, wireless sensor networks have characters such as placed flexible, extended and simplicity. Its potential applications prospects make it has been repid developed in health care, environmental monitoring and military fields [1], [2], [3].

While wireless sensor networks have the advantage as above, considering WSN is a network which has limited energy, the nodes are usually static and large, and it is also uneasy for nodes to recharge or change batteries. So in the wireless sensor network applications, the most important problem is to deal with energy consumption of sensor nodes. Only we reduce the energy consumption of sensor nodes effectivly, wireless sensor networks can ensure connectivity and extend the network lifetime. From this, the sensor nodes should detect data, process and communication under the minimized energy.

In solving the resource constraints, many studies are focused on how to improve the MAC layer protocol effectively, increse utilization of resourse and then solve the energy problem effectively [4], [5], [6]. Although these protocols can peolong the lifetime of wireless sensor networks, but it is still an unsolved problem how to communicate between the protocol layer and application layer to enable developers use well. On the other hand, design and implementation of appropriate middleware layer, communication between application layer and protocol layer is still belonging to the front stage. The traditional middleware system such as CORBA [11], Java RMI [12], EJB [13], because the required computer memory and computing is large, and they are not consider the issue of energy consumption, therefore these design model are not suitable for WSN.

The main application of WSN is event-based, so traditional request / response mechanism is not suitable for WSN middleware. In most applications of WSN especially the environmental monitoring, only when response event occurs or user query via sink node, can data transmission be triggered. The Publish/Subscribe Communication Generics is asynchronous, loosely-coupled and many-to-many communication, enable the participants decoupled completely in the communication space, time, and control flow, which meet the demond of application environment that wireless sensor networks is based o data query.

For these reasons, we constructed event-based WSN middleware named WMOS (Wireless-Middleware Publish Subscriber System) using the Publish/Subscribe Communication Generics. It is developed on the TinyOS and realizes the content/topic self-adaptive double-mode based on the XML matching. It has grade QoS and provides data gathering services such as Pub / Sub middleware, and it also provides a set of API interfaces for application layer. WMOS can not only provide developers with a convenient interface, but also reduce the energy consumption of each sensor node effectively.

Currently, there are a lot of work been done related to the middleware of handheld devices the most used operating system of handheld devices are Symbian OS[14], Windows CE[15], Tiny Linux[16], Palm OS[17 and so on. However, the middleware on sensor node has not been developed as handheld devices, which need our study and practice. We have a lot of research about WSN already, which is focused on the characteristics and problems WSN, its network protocols and potential applications, including [7], [8], [9], and [10]; but there is no

doi:10.4304/jcp.6.5.1040-1046

description about WSN middleware. Recently, researchers have designed a variety of middleware. We will analyze TinyDB[18,19],Mate[20,21], Mires[22], SINA[23], Cougar[24]and MiLAN[25] on the second part. There are many other middleware such as EnviroTrack[26], DSWare[27], Hood[28], Impala[29], TinyCubus[30], Agillap[31] , TinyLime[32]and Smart Messages[33].

This paper is organized as follows: The 2nd part describes the existing WSN middleware. The 3rd part describes in detail the structure and complex problem of WSN and the architecture of WMOS middleware. The 4th part proposes the application scene of the coal mine monitoring to demonstrate WMOS middleware. Finally, the 5th part summarized and some future work.

II. RELATED WORK

In this section, we introduce current WSN middleware. Our emphasis is on the detail of different middleware design method and the evaluation of these middleware.

A. TinyDB [18, 19]

TinyDB system was a sensor network data management middleware system developed by the University of California, Berkeley. TinyDB provides users with a simple, easy to use, SQL like application programming interface. Users can query the TinyDB system for sensor network data using the traditional relational database systems, without knowing the details of the sensor network, which makes the architecture transparent to the user. When TinyDB receives query submited form the user, it collects data from each node of sensor network, scheduling the all sensor nodes deal with the query using distributed processing, and then pass the result via base station node. TinyDB's main features are: 1. Provide metadata management; 2. Support the described query language; 3. Provide effective management of the network topology.

Evaluation: TinyDB can support multiple queries on the same node simultaneously; In addition, when expand the sensor network, we can simply install standard TinyDB code into a new join node, as a result, the node automatically joined to the TinyDB system. However, TinyDB does not provide many functions for the middleware, most of the services need to be developed upon it.

B. Mate [20, 21]

Mate is realized by using virtual machine technology. It implement manipulation and solve the problem occur in WSN by the term of abstraction layer method. This project is developed by the University of California, Berkeley, which is mainly focus on creating a new communication generics to overcome the constraints, such as bandwidth limitation and energy consumption caused by network activity. Mate puts forward a method to process program in order to achieve updating via VM method instead of adjusting simple parameters.

Evaluation:Mate is aimed at providing better interaction and adapting to the changes of sensor networks, which certificates that the use of information activities can update the network protocols and inserted parameters. Mate program is small and has friendly interface, which makes the network dynamic, flexible and easy configurated. Mate provides effective access to networks and sensors via VM. However, because of energy limitation, Mate application is only suitable for sleeping application. For complex applications, it consumes much energy because of the interpretation of complexity. Mate is only a prototype research; it needs more advanced language and generics to develop detail application of sensor.

C. Mires [22]

Mires raises an information-oriented middleware which fits for traditional distributed system. Mires provides an asynchronous communication mode, which is suitable for WSN applications. In most cases, WSN applications are event-driven, and have more advantages on the traditional corresponding models. Mires is written by using NesC and runs on TinyOS. It uses a component-based programming model, and employs dynamic information to achieve Pub / Sub -based communication architecture.The core components of Mires are (pub / sub service), a routing component, and some additional services, such as data aggregation.

Evaluation: Mires demonstrates successfully that it is applicable to use the conventional, information-oriented middleware in WSN middleware. It uses pub / sub system as asynchronous communication model which considers energy limitation. Mires uses multi-hop protocol which doses not include resource discovery mechanism. In addition, we must estimate the performance of network affected by Mires.

D. SINA [23]

SINA usea Data-centric approach mostly. Delaware University developed the system information network architecture in order to realize the self-adaptive, self-organizing sensor networks. SINA design this network as a large number of distributed objects. Its core provides an effective mechanism to make the network scalable, decrease energy consumption of network nodes. Its system querys and monitors using the database based on a spreadsheet. Logical data table is composed of tuples, which represents attributes of sensor nodes. The name of each tuple is unified. The entire data of the table is composed of the imformation from each sensor node. This system provides a request form similar to SQL. There are four ways to access and update information of tuples, they are: requesting content retrieval, content learning, periodic updates and triggered updates.

Evaluation: Compared with Cougar, SINA has advantage that sensor nodes are hierarchical clustering to make data aggregated effectively. However, compared with the Cougar, it can not run on a variety of distributed systems.

E. Cougar [24]

Cougar is a sensor database system developed by the

University of Cornell. The basic idea of Cougar is to query and process data within the sensor network as much as possible in order to reduce communication overhead. In the precedure of querying and processing, the data only relevant to a query can be extracted from the sensor network. Cougar system provides a query language similar to SQL. Cougar system consists of three parts, the first part is the user graphical interface GUI, the second part is the customer front-end system, and the third part is the query agent.

Evaluation: Cougar database method suit well for large sensor data collection, and provide implement method for all the different network operators. However, it uses a lot of resources to transmit large amounts of raw data from device end to database server. For large sensor network, it suffers from potential risk of the communication link failure. Moreover the dynamics of large-scale sensor networks will cause the problem that Cougar access information center using global network.

F. MiLAN [33]

MILAN system emphasis much on that middleware can connect application layer and network layer, the main feature is support interface based on application activities on the effect of real network. It is developed by the University of Rochester. MILAN has the following characteristics: 1. Specificate sensor network applications according to QoS Requirements; 2. when QoS demand is continuing, it can adjust the network characteristics to prolong application life cycle. To achieve these functions, it needs the following information: 1. Individual applications: using association method of different sensor nodes meets QoS requirements; 2. Systems and users: the relative importance between different applications; 3. Network: the used sensors and resources such as energy and bandwidth.

Evaluation: Milan depends on the application-driven network management which is suitable for network applications, and it also handles QoS requirements well. However, Milan lacks innovation on the architecture to suit processing model of WSN, providing the diversity of supporting applications and hardware.

G. Other middleware

Then, we also introduced some other research related to sensor network middleware. However, the method most of them used is similar to the middleware as described above. Agillap [31] is based upon Mate and made extantion. Agillap has additional functionality of introduction of mobile code on the sensor network and smart move of mobile agents. Hood [28] is also the sensor network middleware, which is mainly focused on data stream management.

III. OVERVIEW OF MIDDLEWARE AND WMOS

A. Architecture of middleware

WSN middleware architecture is shown in Figure 1, it is divided into four levels: network adaptation layer, Foundation software layer, application development layer, and application services adaptation layer. Here WSN nodes embedded software (deployed in the WSN nodes) is composed of network adaptation layer and Foundation software layer. WSN application supporting structure (to support the development and implementation of business applications) is composed of application development layer and application services adaptation layer.

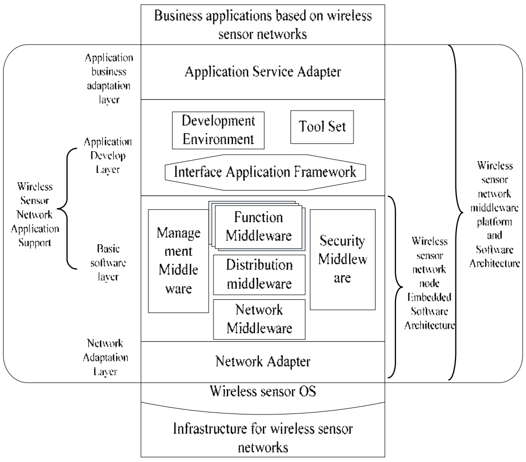
Business applications based on wireless sensor networks
Application business adaptation layer
Application Service Adapter
Application Develop Layer
Development Environment
Tool Set
Interface Application Framework
Wireless Sensor Network Application Support
Basic software layer
Manage ment Middle ware
Function Middleware
Distribution middleware
Network Middleware
Security Middlew are
Wireless sensor network node Embedded Software Architecture
Wireless sensor network middleware platform and Software Architecture
Network Adaptation Layer
Network Adapter
Wireless sensor OS
Infrastructure for wireless sensor networks

Figure1 structure of WSN middleware

Figure 1 shows that, WSN middleware and platform software have hierarchical, modular architecture, making it more adapted to requirements of WSN application. Middleware makes it possible to achieve applicating and developing easily at the cost of its self-complex. And middleware technology can meet application needs clearly. Therefore, flexibility and scalability of WSN middleware and platform software guarantees the security of WSN, improving data management capabilities and energy efficiency, reducing the complexity of application development [11].

B. Complex Issues of WSN Middleware

WSN is booming in recent years [10]. It has many advantages compared to traditional networks, and can be applied to many fields [9]. For example, environmental monitoring and forecasting, health care, complex machine control and space exploration and so on.

WSN middleware design and development is very difficult. We need consider both the characteristics of WSN and the applications features. The main issues are as follows:

a. hardware resources

Considering the small size of sensor nodes, its battery carrys limited power. It is not practical to change battery because of the large number of sensor nodes, the low cost, the wide distribution, and the complex of deployment regional environment. How to use limited energy efficiently to maximize the network life cycle is the primary challenge faced by the WSN middleware. Sensor nodes generally have three basic parts: sensors, processors and wireless communication modules. It cost more power when sensor nodes transmit information than calculation. Therefore, we should pay more attention of the efficency of communications of sensor middleware, reducing unnecessary forwarding and receiving, and change to sleeping status quickly when there is no communication.

b. dynamic network

Sensor network topology may changed because of the following factors: 1) environmental factors or the sensor node failure caused by energy depletion. 2) Bandwidth changes of wireless communication link rose by environmental conditions, 3) mobility of sensor, perceived object and the observer, 4) adding of new nodes. These require middleware must meet changes of this dynamic fault-tolerant, self-organizing and self-adaptive.

c. diversity of Application network

Sensor network is aimed to sense the objective physical world, obtain the amount of information of physical world. There are variety kinds of physical quantities. Different sensor network applications concern a corresponding physical quantity. Therefore we have a wide range of requirements to the application of sensor. Different application background need sensor networks.

d. Real-time

Most sensor network applications require real-time feature, and it is critical take into account of change of time and space. Therefore, the middleware must provide real-time service to adapt the update of data.

e. data-centric network

Sensor networks are task-based networks. When user queries events via sensor network, he can notice the network about the events he interest in directly, rather than a notice to a specific node. This is the reason that sensor network is a data-centric network. As a result, the middleware must adapt to this network.

C. Overview of WMOS

According above, WSN applications collect and integrate data from a lot of logic and emanative sensor nodes. In wireless sensor networks, a large number of sensor nodes exchange data and stop only until query information shows that the source node and sink node disappear simultaneously. Therefore, the traditional request / response communication mechanism does not meet this demand. For example, while only customer requesting and responsing information synchronously, can they establish a connection. This method will lead to a lot of the energy consumption of sensor nodes, which does not comform to the key design of wireless sensor network middleware.

The pub/sub communication generics satisfy the application model of wireless sensor network. In this communication mechanism, information publishers can publish messages for many subscribers. In this basic model, information is transmitted through the contents matchingand the connection. In addition, rhw subscribers and publishers are fully decoupled in the space, time and control flow, which would reduce the energy consumption effectively. This loosely-coupled is the most main superiority in the ad-hoc and pervasive environment (such as WSN).

WMOS is a WSN middleware realizing publish/ subscribe communication. It has three stages to achieve communication. First, advertises information into network when the WSN nodes access useful content (such as temperature, etc.); then, advertising information using multi-hop routing algorithm to route to sink nodes, user connect sink nodes to choose and subscribe the wanted topic. Finally, the subscribed information will be broadcast to the network nodes. After receiving the subscribed topics, the node will publish the data they collect to the network. More details will be analyzed in the following paragraphs.

C.1. WMOS architecture

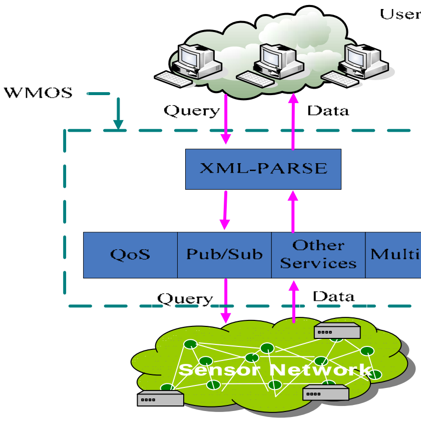

Figure2. WMOS architecture

Figure 2 shows the WMOS' architecture. Nodes of WSN establish the connection through middleware WMOS and upper user. WMOS is developed and controled based on TinyOS. The services provided by TinyOS can be accessd via standard interfaces. WMOS at the top of TinyOS, which provides the user interface and application services to user applications. WMOS' hardware is structured by pub / sub equipment and routing equipment. The main component of WMOS is pub / sub equipment, user send queries content as subscriptor. Sensor nodes of WSN publish information collected to the sink nodes as publisher. WMOS is responsible for the content matching between them. The match information will be successfully transmitted to the subscriber. Only information relevant to the topic can be transmitted, which reduce the energy consumption of sensor nodes.

Since wireless sensor networks have been gradually applied in various fields, the sort of node acquisition information and the sort of upper-user collection information can not solve the actual query situation well only through defining the topic matching. The Content-based Pub/Sub generics can gather wanted information more effectively. In content-based model, the event is no longer confined to specific groups, it is the query and predication defined by subscriber that decide where an event to. Given that Content-based Pub/Sub system has the advantage of flexibility of subscribers. Subscribers do not need to know the defination of subject topic name before subscription. But the disadvantage is that there will be a great burden because of the large number of matched events based on the subscribed condition. The number of the subscription must be more than the number of topics of the system based on topic, leading to the high match efficiency. Meanwhile, the content-based Pub/Sub system will increase the energy consumption of sensor nodes. In solving this core issue, this paper uses double-model pub/sub mechanism based on content/topic. On the sensor node end, middleware use topic-based modelwhile, on the user end it applies content-based model. The match between them is completed by the WMOS using XML.

In addition, the key principle of designing wireless sensor networks is reducing energy consumption, so WMOS use graded QoS to consummate the Subscribe/Publish communication system. On the user subscriber end, WMOS will convert user subscriber into topic-based model automatically when the QoS level is low; when QoS level is high WMOS adopt content-based model. Similarly, on the sensor node end, WMOS custom QoS rank according to the energy of sensor nodes, self-organize and connect ton the most suitable subscription / publishing links.

In order to send information of the sensor node matched to the sink node more effectively, sub/pub services use multi-hop routing algorithm. Routing algorithm acts as a component implementation in WMOS. Developers can adjust corresponding routing algorithm according to different application environments, which improves WMOS scalability.

C.2. Pub/Sub services

Before analyzing WSN middleware WMOS in detail, it need to emphasis that WMOS is implemented based on TinyOS. This environment is a component-based programming model, using nesC language which is a high-level language to be applied in constructing structure module. TinyOS can be considered as component-set, each component availabe support and utilize external interfaces, which is constituted by command lineand events. Command line is implemented of the interface provider. When the interface providers send out an event, this procedure is achieved on the interface user. Communication of network nodes is based on the dynamic message rules. Each message has the ID to call the target node and parameters of data load. Event-based and message-oriented communications principle determines that TinyOS is good platforms to build publish/subscrib communication mechanism.

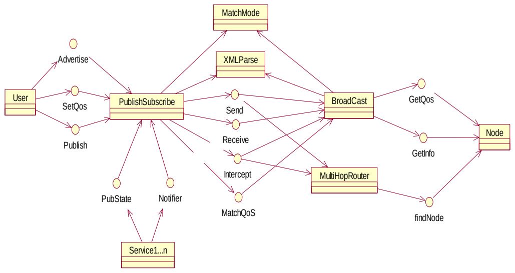

Figure3 Pub/Sub class structure

Figure 3 shows the component structure of WMOS middleware. Among them, the Pub/Sub component provides broadcasting and publishing interfaces. Pub/Sub component connect the sink nodes through BroadCast link, provideing send, receive, trigger, match the QoS and other interfaces. It also provides extending component ServiceX to access Pub/Sub status and response interfaces.

The following sequence diagram shows interaction between Pub/Sub component and other components. In Figure 4, the user broadcast information to Pub/Sub components, after Pub/ Sub components has matched with conponent and XML has processed information, Pub/Sub conponent send information to the multi-hop router. Figure 5 describes the interaction procedure of information subscription. Sensor node transmits collected information to the sink nodes, after XML handle the message, and the message will be sent to the Pub/Sub component, and calls the corresponding data aggregation services.

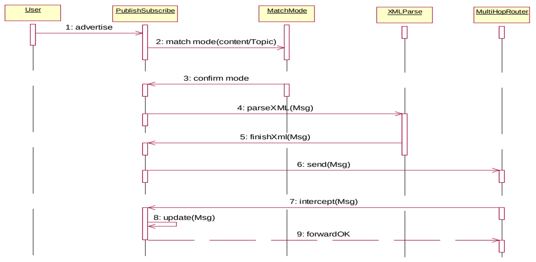

Figure4 content broadcasting sequence diagram

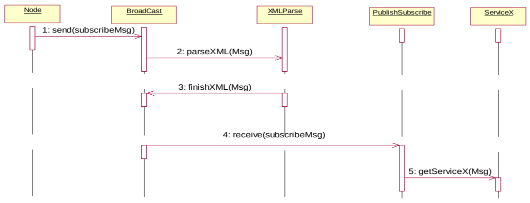

Figure5content subscribing sequence diagram

In the next section, we will discuss the utilization of WMOS environmental monitoring application, and analyze the concrete use of the grades QoS services on the user end and on the nodes end.

IV. CAST STUDY: COAL MINE.

In this section we describe the environmental monitoring application scene of underground mine, demonstratethe realization of the Pub/Sub middleware, and then, we describ the graded OoS services and the connection with WMOS.

A. Scenario

Environmental monitoring scene underground mine should have the following properties: Each environment has a group of sensor nodes, which monitor the underground environment such as temperature, gas concentration, coal particles and so on. The sensor nodes of each region get together. The collected response information must transmit to the sink node of its area. Sink nodes receive queries from end users and broadcast the query informatino to the WSN. After the data of the sensor node matchs the query, the data can be sent to the sink node through the reverse routeing. Hence end user can access the required data on the sensor nodes which monitor the coal mine environment effectively.

B. Graded QoS service

As most fundamental and important design principle of the wireless sensor network is resucing energy consumption of sensor nodes, so WMOS adopt graded QoS service to improve the Pub/Sub communications systems. On the subscribe user end, WMOS will convert end subscriper to topic-based model automatically when the QoS level is low; when QoS level is high content-based model is adopted. On the sensor node, on the sensor node end, WMOS custom QoS rank according to the energy of sensor nodes, self-organize and connect ton the most suitable subscription / publishing links. When subscriber sends query information, there will be many sensor nodes match the query information in a certain region. Then the sink node decides which sensor node provides data according to the QoS level of each sensor nodes, which prolong the lifetime of WSN effectively. Graded QoS services have three basic steps:

a. Collect the QoS level that client set.

b. Sink node collect QoS information of sensor node in the region periodically, and complete the best match of publish and subscrib.

c. Sensor nodes submitted QoS information to the sink node periodically, and self adapt QoS rank according to their energy and other factors.

C. Application example

The WMOS system has four stages: network establishment, Notification, subscription and publication. Sensor nodes send attribute information (such as location, energy, etc.) to the sink nodes through the known routes; routing components of WMOS complete this process. In underground coal mine environment monitoring applications, routing conponents achieve the multi-hop algorithm.

After the establishment of the routing tree, WMOS start notification stage. At this stage, sensor nodes send the collected data (such as gas concentration) through the middleware. WMOS' task is to inform the other sensor nodes of routing tree. Therefore, the upper end application does not need to consider the process of divergence of information in the network. Once the notice information is sent to the sink node through the routes, the information can be passed to the user. In the subscription stage, the user send subscription information to the sink node, WMOS respond to user query and then match message. After subscription stage, the user's subscription information has been arrived the WMOS platform, while the sensor nodes self adapt graded QoS services and connect to WMOS. Finally, under the lowest energy consumption of sensor nodes, WMOS send the collected information to the subscriber.

In a word, users only submit query information. Middleware WMOS receive information and match them with the corresponding sensor node. Subscriber and publisher use asynchronous and loosely-coupled communication in space, time and control the flow, which reduce the gap between application layer and protocol layer.

V. CONCLUSION.

With much widely application of wireless sensor network, it is a great challenge how to design a middleware that reduce sensor node energy consumption and increase network lifetime. This contribution of this paper is not only modify publish/ subscribe more suitable for wireless sensor networks, and also indicate that loosely-coupled asynchronous communication mechanism is more suitable for the application of event-driven WSN than the traditional request / response mechanism. WMOS makes it easier to develop and apply on the wireless sensor network and decrease the energy consumption of sensor nodes. In WMOS, communication links will be established only when collected information of the sensor nodes match user query. WMOS use self-adapt graded QoS services and content /topic model solve the energy consumption of sensor nodes effectively.

The next stage of WMOS is estimating energy consumption, network performance and other parameters, comparing with the major middleware, and improving WMOS architecture design. Also, because WMOS' default routing algorithm is multi-hop protocol which does not include resource discovery mechanism, WMOS will increase the security, resources exploration function gradually.

REFERENCE

[1] I.F.Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “A Survey on Sensor Networks”, IEEE Communications Magazine, pp. 102-114, August 2002.

[2] A. Cerpa et al., “Habitat Monitoring: Application Driver for Wireless Communications Technology”, ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean, Costa Rica, April 2001.

[3] G.J. Pottie, W.J. Kaiser, “Wireless Integrated Network Sensors”, Communications of the ACM, Vol. 43, no. 5, pp. 51-58, May 2000.

[4] W. Ye, J. Heidemann and D. Estrin, An Energy-Efficient MAC Protocol for Wireless Sensor Networks, IEEE INFOCOM, June 2002.

[5] V. Rajendran, K. Obraczka and J. J. Garcia-Luna-Aceves, Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks, ACM SenSys, Nov 2003.

[6] L. M. Feeney and M. Nilsson, Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment, IEEE INFOCOM, April 2001.

[7] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks”. IEEE Communications Magazine, Aug. 2002，40 (8):102-114.

[8] Q. Jiang and D. Manivannan, “Routing Protocols for

Sensor Networks”. IEEE. 2004

[9] C. Chong and S.P. Kumar, “Sensor Networks: Evolution, Opportunities, and Challenges”. In the proceedings of the IEEE, 91(8), Aug.2003.

[10] D. Culler, D, Estrin and M. Srivastava, “Overview of Sensor Networks”. IEEE Computer Society, Aug.2004.

[11] Object Management Group, “The Common Object Request Broker: Architecture and Specification”, published by the Object Management Group (OMG), Revision 2.3, June 1999.

[12] A. Wollrath, R. Riggs, and J. Waldo, “A Distributed Object Model for the Java System”, Usenix Conference on Object Oriented Technologies and Systems, May 1996.

[13] A. Thomas and P. Seybold, “Enterprise JavaBeans Technology”, available in http://java.sun.com/products/ejb/whitepaper.html, December 1998.

[14] Symbian OS. URL: www.symbian.com

[15] WindowsCE. URL:http://msdn.microsoft.com/embedded/windowsce/default.aspx

[16] Linux PDA. URL: http://tuxmobil.org/pda_linux.html

[17] PALM. URL: www.palm.com.

[18] TinyDB Project. URL: http://telegraph.cs.berkeley.edu/tinydb

[19] R. Muller. “A Query Processing Middleware for Sensor Networks,” Master Thesis, September 30th, 2005.

[20] P. Levis, and D. E. Culler. “Maté: a Tiny Virtual Machine For Sensor Networks,” Architectural Support for Programming Languages and Operating Systems, 2002.

[21] B. Blum. “Mate – VM for Sensor Nets.” URL: http://zoo.cs.yale.edu/classes/cs434/readings/papers/Mate.ppt

[22] E. Souto, G. Guimaraes, G. Vasconcelos, et el. “A message-oriented middleware for sensor networks,” Proceedings of the 2nd Workshop on Middleware for Pervasive and Ad-hoc Computing, Vol. 77, 2004, pp. 127-134.

[23] Srisathapornphat, C.Jaikaeo, and C.C. Shen. “Sensor Information Networking Architecture,” International Workshops on Parallel Processing, 2000, pp. 23-30.

[24] Cougar Project. URL: http://www.cs.cornell.edu/database/cougar

[25] MiLAN Project. URL: http://www.futurehealth.rochester.edu/milan

[26] T. Abdelzaher, B. Blum, D. Evans, et el. “EnviroTrack: Towards an Environmental Computing Paradigm for Distributed Sensor Networks,” IEEE ICDCS , March 2004.

[27] DSWare. URL: http://www.cs.virginia.edu/colloquia/event302.html

[28] K. Whitehouse, C. Sharp, E. Brewer, et el. “Hood: a Neighborhood Abstraction for Sensor Networks,” In Proceedings of ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), Boston, MA, June, 2004.

[29] T. Liu, and M. Martonosi. “Impala: A Middleware System for Managing Autonomic, Parallel Sensor Systems,” Proceedings of the ninth ACM SIGPLAN symposium on Principles and practice of parallel programming, 2003, pp. 107 – 118.

[30] P. J. Marrón, D. Minder, A. Lachenmann, and K. Rothermel. "TinyCubus: An Adaptive Cross-Layer Framework for Sensor Networks,” it – Information Technology, vol. 47(2), 2005, pp. 87-97.

[31] Agilla – A Mobile Agent Middleware for Wireless Sensor Networks. URL: http://www.cs.wustl.edu/mobilab/projects/agilla

[32] TinyLime. URL: http://lime.sourceforge.net/info/tinyLime.html

[33] P. Kang, C. Borcea, Gang Xu, et el. “Smart Messages: A Distributed Computing Platform for Networks of Embedded Systems,” Special Issues on Mobile and Pervasive Computing, The Computer Journal, 2004.

Lifang Zhai, was born at Linshi, Shanxi Province of China in 1992. Now he studies in Dalian Jiaotong University. His research interests include mobile network and WSN.