

Design of Application Platform for Wireless Sensor Networks

Shuqian Wang^{1, a}, Wei Zhang^{1, a}, Zhangwei Zhu^{1, a}, Chengquan Hu^{1, b}, Lili He^{1, a}

¹College of Computer Science and Technology, Jilin University, Changchun 130012, China

^ahelili@jlu.edu.cn, ^bhucq@jlu.edu.cn

Keywords: Application Platform, Wireless Sensor Networks, Socket, Browser/Server Structure

Abstract. One of the bottlenecks that slow down the development the wireless sensor networks application is that people have to design different programs according to the different applications. A wireless sensor network application platform is presented which can be used to rapidly deploy an application by setting simple configuration parameters. The platform consists of two modules: Base station Module and Server Module. Base Station Module receives data from the coordinator by RS232 port, and sends them to the server through socket protocol. It also receives commands and parameters from server and delivers them to the coordinator. The platform also provides human-computer interactive function for users in the Browser/Server structure.

Introduction

As an emerging technology in the 21st, wireless sensor network is a novel technology about information acquiring and information processing. It is the combination of micro-intelligent sensor technology, embedded computing technology, distributed low-power consumption information processing technology, less-energy depend technology, wireless communication technology, etc. The WSN can real-time monitoring, perceiving, collect various information of different test objects in the network distributed area in team. It has been widely used in many fields for its low-power, low-cost, ad hoc network, and distribution^[1]. But now the most research for WSN is focused on the network establishment, communication protocol, information fusion, and node power^[2, 3]. So it is lack of necessary tools for the upper application^[4, 5, 6]. In this paper information processing application platform is designed for WSN. And after some simple configuration is changed on it, a new platform according to specific application requirement of different sensing network can be designed rapidly. This platform can provide a real-time monitoring and real-time control for WSN and be help to analyze and manage the sensing data.

Application platform design

Overall design. This platform is designed for sensor data in WSN storage management and analysis. The overall design chart is shown in Figure 1. It includes sensor network, base station, the server, and the client. Between the coordinator in WSN and base station the data communicate through the RS-232 serial communication interface. And between base station and application server, between the client and application server the connection is through the Internet. In this platform, the client is designed to adopt B/S structure, and the user interaction is mainly designed in the server.

3) Design for data processing: This part is responsible for analyzing data that in the buffer pool. The program deletes the error data according to the protocol format. To some message, for example time synchronization request, base station responds through the serial port, and sending the sensor data and network topology information to the server through the established socket connection. This part also includes the program to download command, the specific process is monitoring the socket connection until receives command message from the server and then sending them to the coordinator in WSN (shown in Figure 4).

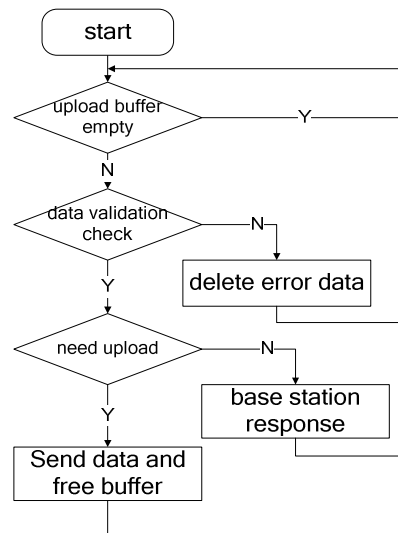


Fig. 4 Data processing flow

The server design. The server design is the core part of the data storage and processing. The design adopts B/S structure as the network structure model. The server uses Apache Tomcat 7.0 version as the network server, uses Java Servlet technology to support application service, uses Scalable Vector Graphics combine with JavaScript to design browser page.

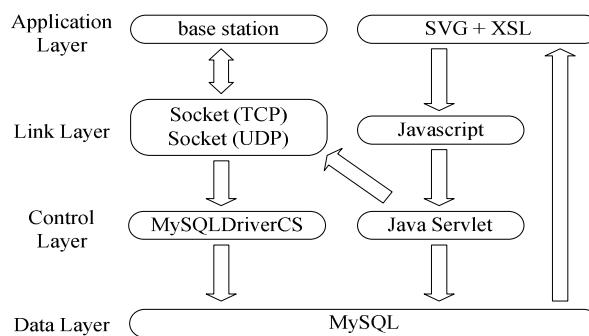


Fig. 5 Server software design chart

1) Design for Database: The design uses MySQL relational database management system to storage data. Because of the need to real-time keeping the latest status information of the sensor node, when inserting the sensor data into the database, in the meantime the server maintains a database table to store the latest information through a database trigger. In B/S structure, the data that returned to the client is got from this table. On the side, some new tables can be added to the database according to the concrete application. For example, in the later experiment, it is required to coordinate localization the mobile node base on RSSI value. So a table named 'locnode_inf' is built to store node's coordinate.

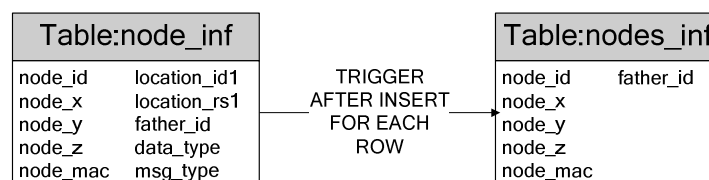


Fig. 6 Database trigger design

2) Design for control layer: The design for control layer consist of facing to base station and facing to the client. The processing procedure for the data from base station includes checking the data in the buffer, whether the data is in accordance with the communication protocol, whether the check digit is correct, and then analyzing data to get reasonable value of the sensor information. The design establishes MySQL database connection through MySQLDriverCS driver, and writes the corresponding SQL statement according to the message type, then executes it and updates the database information. For the client operation, the processing procedure includes verifying client authority, synchronization with the database, responding message to the client, notifying the server to send command to base station, etc.

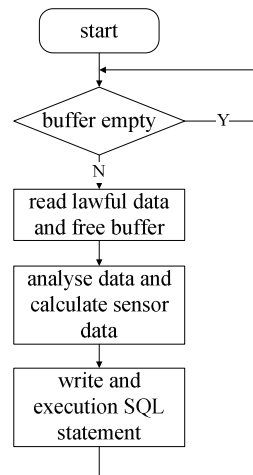


Fig. 7 Processing upload data flow

3) Design for link layer: The design for link layer includes the server data communication design and B/S structure communication design. The first part need to maintain two socket connection: one is based on TCP protocol and uses the server port 9050, it is used to communicate with base station to receive upload data and send download data; the other one is based on UDP protocol and using the server port 9051, it connects with the Java Servlet in the server, used to receive the processing that the client operate to the browser and send it to control layer. The other part is some function files using JavaScript language to write for B/S structure. It is responsible for saving and loading sensor message to the browser and transferring the client operate to control layer.

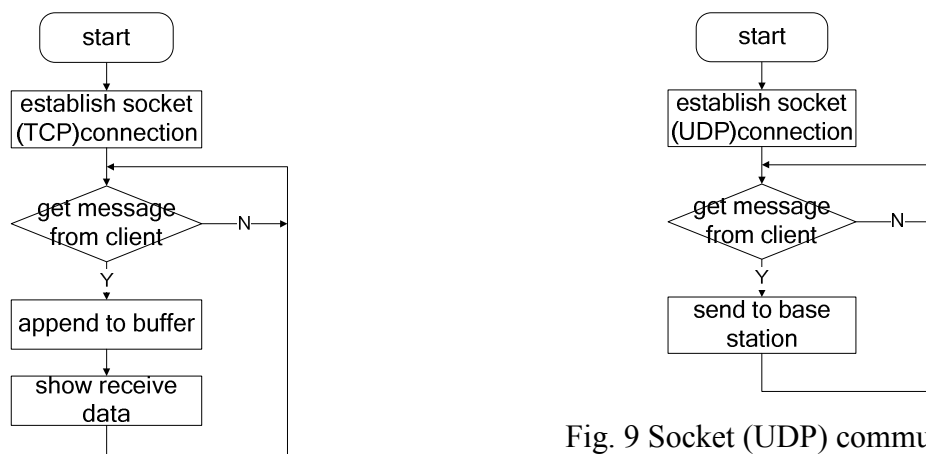


Fig. 9 Socket (UDP) communication flow

Fig. 8 Socket (TCP) communication flow

4) Design for application layer: This part is used to real-time emersion the sensor network information and interacts with the client. The server releases the page through Apache Tomcat 7.0, and uses JavaScript and Java Servlet to realize its function such as modifying the sensor node properties, sending commands, updating the information, etc. The server uses Scalable Vector Graphics (SVG) to design browser page, which is a language for describing two-dimensional graphics

and can use XML to define the graph, describing graph material by descriptive text formatting language. SVG is a W3C Recommendation, so it is good at cross-platform and extensibility like XML, and of course it is easy to read and modify it through embedding XML file. In addition, different from other traditional image format, SVG can operate with JAVA. This design uses SVG to layout browser page that can simulate the WSN environment, dynamic display the location and translocation of the sensor node, use XSL to display the latest information of the sensor node. The operation of the client shown in Figure 10.

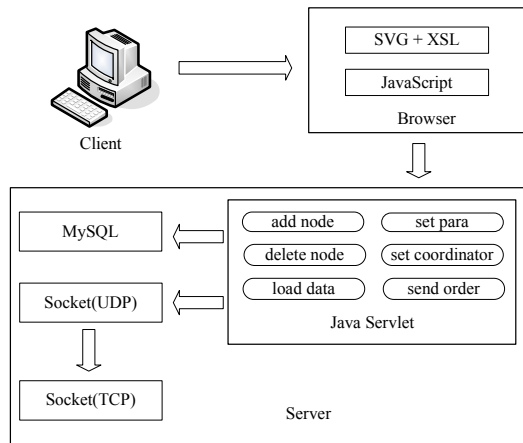


Fig. 10 B/S structure procedure

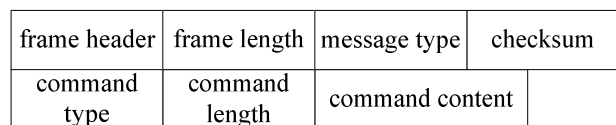


Fig. 11 Command frame format

When the browser page is designed, the event 'onclick' is defined to get operation from the client. So when the event is triggered, the browser executes the matching JavaScript function, if the client wants to modify the database or communicate through internet, the server will call Java Servlet server. For download command operation, the design provides a general format, each part is one byte and the command content is an byte array (shown in Figure 11).

Platform application example

The example selects a 70*100m 'E' font structure floor as experiment scene. The scene includes a WSN that consists of some sensor node (more than 20), the route nodes are placed in the corridor ceiling and the mobile nodes are carried by people. Using ZigBee technology as the sensor network protocol, the sensor nodes can construct network themselves. The topology of the network structure is a tree structure and the root node is the coordinator, when the sensor node acquires sensor data, the network can converge them to the coordinator. The type of sensor data includes temperature and humidity sensor information, infrared information, coordinate information of positioning anchor node, the network topology information, and node's time information, etc. The experiment equipment includes a PC as base station, a server with a MySQL database and Tomcat server, and some PC as the client. The browser page can display the latest state information of the WSN, as shown in Figure 12.

In the data acquisition test, after base station and the server open their service program, the administrator can find the sensor data receiving and processing steady and rapid, program running steady, the network communication running normally. In the fault tolerance test, one node is set to upload a misdate data non scheduled and base station can detect and delete it. In addition, the database on the server runs steady and when base station or the server reboot or the network is break for some reason, the program can reconnect and continue to upload the data in the buffer.

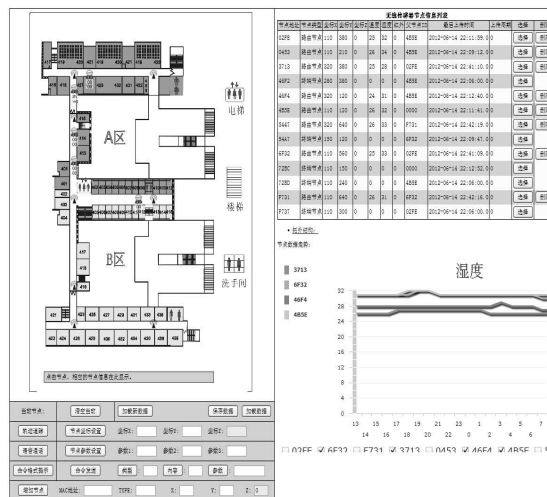


Fig. 12 B/S structure Browser interface

The experiment also includes a test to locate the mobile node using three-point fix based on RSSI value^[7]. According to the floor condition, the location algorithm is modified. When the node needs to be located, it sends request frame on the air, and gets RSSI value from the return message frame that anchor node send. Then the mobile node uploads the RSSI value and the ID of the anchor node. The server completes the location calculate, and the positioning error is about 5m in the test.

In the client operating experiment, the client downloads command or modifies sensor node parameter, and test the synchronism of the database. The result shows that the database running normal and the communication delay less than 1s. All above satisfy the design requirement.

Conclusion

This paper designs an application platform for WSN. It can collect and storage sensor data, set network parameters for a given WSN through some simple configuration. It also provides human-computer interactive function with B/S structure. The application experiment shows that the platform can satisfy application demand of WSN and be deployed quickly and conveniently, operated stably and reliably. The platform can be used as a rapid upper application development tool for WSN.

References

- [1] Limin Sun, Jianzhong Li, Yu Chen, Hongsong Zhu, Wireless Sensor Network [M], Tsinghua University Press, 2005.
- [2] JUN ZHANG, XIAOHUA JIA, Real-Time Data Aggregation in Contention-Based Wireless Sensor Networks [Z], ACM Transactions on Sensor Networks, 2010, 7(1):221-225.
- [3] Zhaochun Wei, Shuigeng Zhou, Jihong Guan, Data Storage and Accedd in Wireless Sensor Network: A Survey [J], Acta Electronica Sinica, 2008,36(10):2001-2010.
- [4] Libo Feng, Yongcheng Li, Yinfeng Wu, Geneeral Monitor Platform for Wireless Sensor Network [J], Instrument Technique and Sensor, 2008(10):55-57.
- [5] Jianzhong Li, Jinbao Li, Shengfei Shi, Concepts, Issues and Advance of Sensor Networks and Data Management of Sensor Networks [J], Journal of Software, 2003,14(10):1717-1727.
- [6] Xin Ke, Jian Shu, Yong Ren, Limin Sun, A Study of Wireless Sensor Network Evaluation Technology and Testbed [J], Computer Science, 2007,34(1):120-122,127.
- [7] Jiangwu Zhang, Lu Zhang, Ying Ying, Feng Gao, Research on Distance Measurement Bssed on RSSI of ZigBee [J], Chinese Journal of Sensors and Actuators, 2009,22(2):285-288.