Research and Implementation of Typhoon Forecast Negotiation System on Mobile Platform

Chaozhen Guo, Shaolin Liu, Binyang Zhang
Department of Computer Science and Technology, Fuzhou University, Fuzhou, Fujian, 350002, P.R.China
guocz@fzu.edu.cn , guocz@263.net

Abstract

Communication technology has been developing at a rapid speed, and mobile phone usage is now popular. At the same time, mobile phones are considered not only as a simple communication device, but also as a device with extra functionality. In fact, mobile phones are now developing into a digital information terminal. A Typhoon forecast negotiation system is a special kind of group decision support system, which enables meteorological experts to take part in the typhoon forecast process on a uniform platform. This article analysed the workflow of a typhoon forecast negotiation system on a mobile platform, and put forward a general design and implementation based on the J2ME platform. The platform included three basic elements: UI, key input and mobile network communication. This article researched the communication interface protocol between server and client, and put forward a mobile service framework. Moreover, this article introduces the design and implementation of the client component, server component, cooperative work model, image transfer process.

Keywords: Typhoon Forecast Mobile Platform; Negotiation System; GDSS; J2ME

1. Introduction

On one hand, the field wireless mobile communication has achieved success in recent year and has gradually changed people's lives. Due to faster processing speeds, greater network bandwidth and more powerful intelligent operating systems, mobile terminals are used for more than communication alone. The Java language is of importance due to its platform-independent feature. Through using Java as the core of the wireless application layer all kinds of wireless data business applications that focus on Java emerge, which enrich the functionality of the mobile phone and provide a personalized electronic device.

Fujian province is an area in which typhoons typically experiences every year. Typhoon forecast has become a responsibility of the provincial Meteorological Observatory. Typhoon forecast is a negotiation process where many meteorological experts work together to forecast a typhoon's future path and its area of influence. However, typhoon forecast has many problems including the uncertainty of information sources, mutability of information and conclusion inconsistency. Therefore, experts should co-operate and utilize computers within the negotiation process. Computers direct the process of negotiation and adopt a preference algorithm to correct the decision to arrive at a reasonable conclusion. This method avoids the situation in which an expert dominates the decision process guaranteeing a reasonable and objective decision. Pervasive computing focuses on people [6]. Experts are likely to access the system with differing kinds of devices at anytime and anywhere. Unlike experts sitting in front of a computer, these experts need real-time system services using handheld devices connected via a wireless network. Therefore, mobile phones are a good choice. However, we need to

The project was supported by the Natural Science Foundation of Fujian Province of China (No.A071007)

978-1-4244-2020-9/08/$25.00 ©2008 IEEE

integrate the mobile phone platform into the IGDSS (Intelligent Group Decision Support System) to build a complete system. As a result, we can overcome the drawbacks of a fixed geographical location and device type restrictions.

2. Requirement analysis of typhoon forecast negotiation system on a mobile platform

In a real-world natural disaster forecast situation, experts need to use system services to take part in the decision process at anytime and anywhere with different kinds of computing devices such as computers, PDAs and intelligent mobile phones. With mobile phones supporting Java, experts are able to enter the system, edit the typhoon path on a map, submit their schemes, question other experts' schemes and score schemes accordingly.

The typhoon forecast negotiation system on a mobile platform is supported by wireless mobile communication technology, which has advantages over traditional networks with regard to constraints on both time and space. Traditional IGDSS ignore the diversity of client devices in user-machine interaction. In a real-world natural disaster forecast situation, experts need to take part in the decision process to make real-time decisions at anytime and anywhere with multiple computing devices. The system is a cooperative application platform based on mobile networks and utilises modern information techniques. Its final goal is to communicate with servers and fulfill as many requirements of the typhoon forecast as is possible. Three main functions were implemented:

(1) Check login: The system is mainly used by meteorological experts, so users need to be checked before entering the system.

(2) Browse meeting information: Provides a list of meetings which the current online experts have the right to participate in.

(3) Meeting process: Includes editing the typhoon path on a map, editing typhoon attribute parameters, evaluating schemes, viewing scheme scores and review the meeting conclusion.

In conclusion, the purpose of building a typhoon forecast negotiation system is to provide a platform for experts without a PC to hand, for example, at an outdoor location, to take part in the decision process. Thus, it is possible to overcome the restrictions on the objective condition.

The workflow of the negotiation system is shown as follows in Figure 1 according to a one-round meeting process.

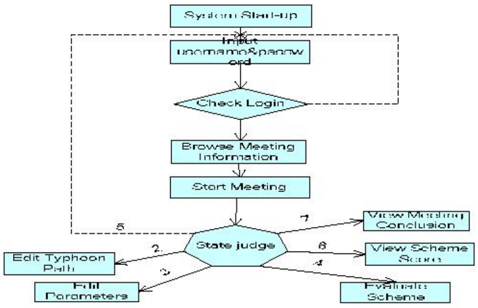


Figure.1 The workflow graph of the negotiation system

3. The general design of a typhoon forecast negotiation system based on J2ME

3.1 Introduction to J2ME

In order to adapt to the development of mobile application data and promote the development of wireless e-commerce business, J2ME (Java 2 Micro Edition), that is, Java for the embedded systems has been introduced into the wireless field. J2ME provides a new model to mobile networks which allows a mobile phone to download applications from the internet and run these applications offline in the executable environment provided by J2ME.

J2ME is one of the products sponsored by Sun Microsystems. It is mainly used in embedded business systems, such as the mobile phone, PDA, Set-Top Box [1]. J2ME provides a series of solutions for efficient network programming in micro systems. It guarantees that the device manufacturers, network transporters, and program developers provide the latest applications and services to their customers. In order to fulfill consumer needs and deal with the diversity of embedded market, J2ME adopts modular and extensible design architecture. The architecture is implemented by a three-tier model built on the local operating system.

According to the resource features of various devices, the J2ME architecture is divided into three layers: profile, configuration and Java Virtual Machine.

(1) Profile Layer: Defines a minimum set of APIs for specific devices.

(2) Configuration Layer: Descriptor of specific hardware.

(3) JVM Layer: Core of the J2ME and the implementation of Java Virtual Machine focusing on the operating system of the local device. Provides support for the specific J2ME configuration.

3.2 The architecture of tbe typhoon forecast negotiation system based on J2ME

The Typhoon forecast negotiation system is divided into 9 modules: Identity Validation, Meeting Process, Browsing Meeting Information, Editing typhoon Path, Editing path parameters, Viewing meeting conclusion, Evaluating Scheme, Viewing Scheme Score, and Monitoring Meeting State. It is shown in Figure 2.

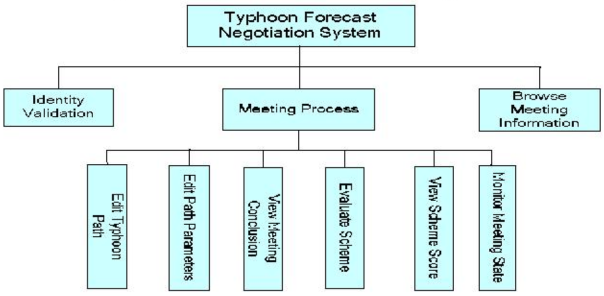


Figure.2 Modules of Mobile Phone Client

The typhoon forecast negotiation system is implemented by the Tomcat + Servlet + Jsp design pattern. Java server pages (Jsp) constitute the presentation layer of the system, and the mobile phone is used for sending wireless requests and displaying responses. The administrator can therefore maintain the system through Java server pages. Meanwhile, we use SQL Server 2005 as database server and JDBC for database access.

The system is based on the Client/Server model, with J2EE web layer at the server side and mobile phone supporting Java at the client side. The Clients access servlets on the server side through the HTTP protocols. Servlets handle requests from the clients by triggering database operations and send wrapped responses back to the clients. This process is shown in Figure 4 as follows.

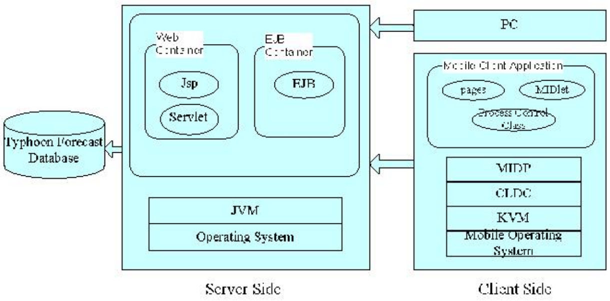


Figure.3 The architecture of the typhoon forecast negotiation system

In Figure 3

(1) Mobile Client Application: Responsible for three aspects: UI, data communication with server and cooperative work with other client applications.

(2) MIDlet: The interface between the application declaration and the running environment, which is controlled by the application manager. It implements the three abstract methods needed by the application manager to manage life cycle of MIDlet: startApp (), pauseApp () and destroyApp (). As soon as MIDlet starts, the startApp () method is called. The pauseApp () method is called before MIDlet stops, and the destroyApp () method is called before MIDlet is destroyed.

(3) J2ME architecture in mobile phone client:

① MIDP(Mobile Information Device Profile): Profile layer of J2ME architecture, composed of a series of API which provide a complete running environment for mobile devices together with CLDC (Connected Limited Device Configuration).

② Configuration: Configuration layer of J2ME architecture, which describes the specific hardware configuration.

③KVM: Java Virtual Machine which is designed specially for micro and resource-limited devices. It is also acts as the minimum JVM for the J2ME platform and CLDC.

4. The implementation of typhoon forecast negotiation system on mobile platform based on J2ME

According to our system design, we choose JBuilder 9.0 as the IDE. It is a good choice for developing J2ME applications as it contains the mobile development packages (Mobile Set) from the 9th to 2005 edition. ,and the J2ME Wireless Toolkit (abbreviated WTK). WTK2.2 is widely used in our system, so we install it first, and then integrated it into JBuilder. For the purpose of testing, WTK2.2 supports MIDP2.0 as well as a mobile phone runtime simulation environment. Moreover, we choose JDK 1.5, Tomcat 5.0 and SQL Server 2005 as our server side components.

4.1 The communication mode of the typhoon negotiation system on a mobile platform based on J2ME

(1) Network communication protocol.

In order to leverage the mobile equipment to full potential, we need to write programs for a network environment. Theoretically, we can choose between Socket and Datagrams protocols. However, problems may occur when programs are ported as some manufacturers' MIDP devices may not support these protocols. Therefore, we choose the HTTP protocol which is agile, easy to implement, and is widely used on Internet. The HTTP protocol makes full use of many of the basic existing facilities on the server side. In addition, HTTP transmission is synchronous, that is, a package that is not requested will not arrive at MIDP device, so it is quite easy to use the MIDP device to implement the network interface. HTTP is chosen as the transmission protocol on the MIDP device, and is implemented as a connection type in the MIDP standard. At present, all mobile information devices that support Java support the HTTP protocol.

After we use the HTTP protocol to obtain a connection object, we can use the Connector.open () method to get a network connection. Then some relevant methods are called to acquire essential information. Usually, we can use the StreamConnection and HttpConnection interfaces. The latter is inherited from the former and provides extended functionality. Take for example the StreamConnection interface; the development process includes 4 steps: firstly, create a network connection of the class StreamConnection c= (StreamConnection) Connector.open (url). Secondly, call the method openInputStream () to gain the data stream s= c.openInputStream (). Thirdly, use some method (read () for example) to read data obtained from the data object processing to a StringBuffer variable, then output it through the method println(). Finally, close the connection object upon finishing data process.

(2) The way to implement the network connection

The network connection is established when needed. The data is transmitted once the connection is set up. After finishing the data transmission, the connection should be closed immediately. When an error occurs during connection or transmission, the user should be made aware of the error cause, and the possibility of either trying again or exiting. The only disadvantage of this approach lies with a possible delay when establishing a connection each time, On the other hand it guarantees the quality of the connection and improves server performance.

Some problems may occur when deploying the l program on a mobile phone as a single-threaded program may pause while waiting for the connection. The duration of the delay depends on the quality of the network. This, for example, may result in a white screen without any display or response. In order to avoid this situation, we must introduce a technique utilising multi-threads in J2ME to establish the connection. A thread class is represented by java.lang.Thread class which defines methods to set and get the thread attributes and the run () method to start the thread.

4.2 Principle of the system's cooperative work

As the system is a type of negotiation system, the entire meeting process should be controlled by a meeting holder. In the meeting process, experts need to be kept abreast of developments therefore the client mobile phone should obtain the meeting status dynamically, and act accordingly. In this way, the mobile phone platform can be kept consistent with the browser. The class Listener implements a monitor, the underlying principle is as follows: The thread refreshes periodically, in each cycle, the mobile phone reads the status from the server. If the status read from the server is inconsistent with the current status, then the mobile phone is redirected to a new page representing the updated status. This solves the problem of the synchronization as the new page reflects the new status of the meeting. That is the meeting holder can “write” the status and the experts can “read” it. The workflow graph of this process is shown in Figure 4 below.

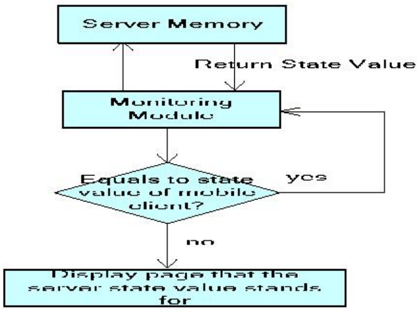

Server Memory
Return State Value
Monitoring Module
Equals to state value of mobile client?
yes
no
Display page that the server state value stands for

Figure.4 The Workflow Graph of the Monitoring Module

4.3 The interaction between server and mobile phone client

The components on the server side are responsible for data communication with the mobile phone; a Servlet is suitable for this task. Through HttpServletRequest and HttpServletResponse, we can receive and send data of data types int or string. The methods for sending and receiving data are divided into two groups: read(), write() for int type communication, and readUTF(), writeUTF() for String type communication. We implement network communication with the doPost() method. Different actions are taken according to operation information received from the mobile phone client and the response is sent back to the client.

The process of the Servlet handling the request is shown in Figure 5 below:

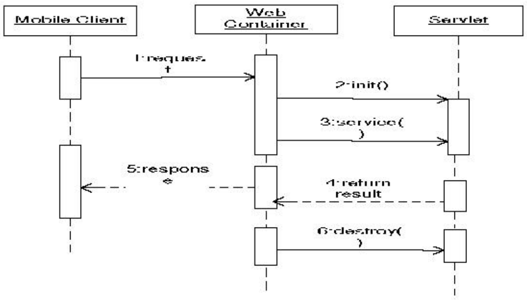


Figure.5 The Process of Servlet Handle Mobile Phone Request

4.4 Image Transmission and Processing

The image downloaded by the mobile phone client is transmitted in the form of binary data. The binary data is stored in a byte array (byte []) or InputStream object. After downloading the image, we can use the CreateImage () method to reconstruuct the image object, and display it in the image widget.

The flow chart of image transmission and processing is shown in Figure 6 below:

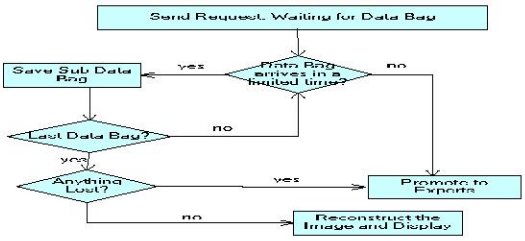

Send Request, Waiting for Data Bag
Data Bag arrives in a limited time?
yes
no
Save Sub Data Bag
Last Data Bag?
no
yes
Anything Lost?
yes
Promote to Experts
no
Reconstruct the Image and Display

Figure.6 The flow chart of image transmission and processing in mobile phone client

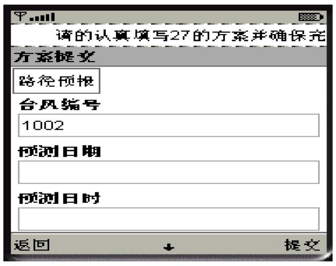
4.5 The Database Connection

The mobile phone client, based on J2ME platform, is allowed to open network connections and access the remote server. However it does not have the ability to directly access the database. Therefore, we need to write a set of programs on the server side which accepts instructions from the mobile phone client. This is achieved by deploying servlets that support the HTTP protocol on the server side to execute database access operation. To connect to the database the system used JDBC. JDBC is a technique for connecting databases on the J2EE platform.. The JDBC API supplies a unified programming interface for Java developers. SQL statements can easily be sent to nearly any kind of database for processing by JDBC. However, the database connection must be established before using JDBC.

5. System running and testing

The goal of the system is to allow mobile phone users to participate in a typhoon forecast negotiation meeting. Therefore, a key point is to implement basic network connectivity, including text transmission, image transmission and the cooperative work between mobile phone clients and PC clients. The meeting holder uses the IE browser to enter the system, while experts use the mobile phone simulator to enter the system. The meeting holder controls the meeting and the experts are directed by the holder to do tasks such as editing the typhoon path, editing typhoon parameters, evaluating schemes, viewing scheme scores, viewing meeting conclusions and so on.

During the editing of the typhoon path experts can query the current typhoon attributes through reading the latest typhoon map. At the stage of submitting the scheme, the experts can edit their schemes according to the scheme panel. The scheme submission panel is shown in Figure 7.



请的认真填写27的方案并确保充

方案提交

路径预报

台风编号

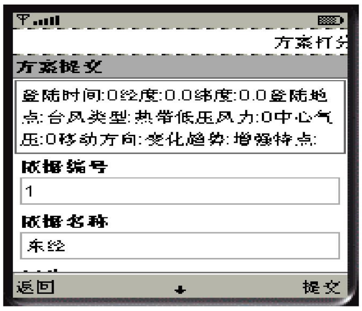
1002

预测日期

预测日时

Figure.7 Scheme Submission Panel

After all the experts have submitted the schemes, the meeting holder directs the meeting onto the editing stage which involves questioning the schemes. Experts can demur at schemes to which they do not agree. Currently, all the experts participate in the discussion in an anonymous way through text and image. Voice and video are possible future extensions. Figure 10 shows the panels of questioning a scheme on the mobile phone client



方案打分

方案提交

登陆时间:0经度:0.0纬度:0.0登陆地点:台风类型:热带低压风力:0中心气压:0移动方向:变化趋势:增强特点:

依据编号

1

依据名称

东经

Figure.8 Panels of questioning the scheme on the mobile phone client

During testing, the mobile phone client connected to the server side very well. Meanwhile, text and image transmission was fluent and the mobile phone client cooperated well with PC client. Figure 9 shows that PC client browser and the mobile phone client arrive at the stage of editing the typhoon path synchronously.

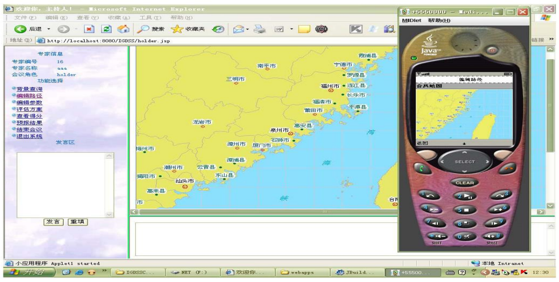


Figure.9 PC client browser and the mobile phone client synchronously editing a typhoon path

6. Summary

This paper proposes a framework for a typhoon forecast negotiation system on a mobile platform. This included the following major work:

(1) Developed a typhoon forecast negotiation system on a mobile phone platform.

(2) Implemented real-time message transmission between the mobile phone client and PC client.

(3) Implemented basic transmission of characters, text and image.

(4) Implemented cooperative work between the mobile phone client and PC client.

The system is based on typhoon forecast at Fujian Observatory. This mobile platform achieves the possibility of having mobile phone users participate in the negotiation of typhoon forecast even when outdoors by means of a wireless network. The system is able to transmit basic path map and attribute data of the current typhoon. Meanwhile, the mobile client can cooperate with the IE browser. However, there are still some drawbacks and aspects worthy of further study:

(1) Due to the size of screen and storage capacity of the mobile phone, it is difficult to zoom and minimize the image and add nodes on the path. Data lose is also possible when transmitting a large sized image.

(2) One possible solution to problem in (1) involves the mobile phone client sending an instruction to the server, the server then sends the image, which modified according to the client instruction, back to the client to display.

When looking into the future mobile phone software based on J2ME is becoming a major trend in developing future mobile applications.

References

[1] Wang Sen. Java mobile phone and PDA programming guide. Beijing: Electronic and Industrial Publishing House, 2004.

[2] James Keogh. Complete J2ME Development. Pan Yin, Wang Lei. Beijing: Tsinghua Publishing House,2004.

[3] Zhuang Dong, Liu Xiao Hua. JBuilderX Wireless Application Development. Beijing: Electronic and Industrial Publishing House, 2004.

[4] Liu Shao Lin. Research and Implementation of Typhoon Forecast GDSS. Fujian:2002.

[5] Lin Zhi Hong. Research and Implementation of Natural Disaster Forecast System based on MAS. Fujian: department of computer Fuzhou university, 2007.

[6] Liu Hong, Wang Yun San, Xiong Wen Long. Implementation of Java mobile phone network communication based on HTTP protocol. Transaction of Wuhan Science University.

[7] Chen Yue. Mobile phone application development based on J2ME platform:[master's paper].Sichuan: Department of Computer, Sichuan University.

[8] Vartan Piroumian, Bill Joy, Mike Clary. Wireless J2ME Platform Programming. Prentice Hall. March 25 2002.

[9] Michael Kroll, Stefan Haustein.java 2 Micro Edition (J2ME) Application Development Same, June 25,2002