The Development of a Mobile Terminal Middleware Platform Based on HTML5

Xiaolin Yi

Beijing University of Technology/ College of Computer Science, Beijing, China
Email: yixiaolin@bjut.edu.cn

Yishuang Ning

Beijing University of Technology/ College of Computer Science, Beijing, China
Email: ningyishuang@emails.bjut.edu.cn

Abstract—**With the rapid development of wireless communication technology and the proliferation of mobile devices, mobile applications are more and more widely used. Depending on the characteristics that the existing mobile applications need long development cycles depending on a variety of mobile devices, this paper puts forward an HTML5-based mobile middleware on the basis of the traditional middleware and mobile middleware. The mobile middleware uses the B/C/S hybrid architecture and shields the differences between the hardware and software of mobile devices, which enables developers to develop applications that are suitable for mobile terminal platforms according to MVC pattern. Compared with other middlewares, HTML5-based mobile middleware has the advantages of short develop cycle, low cost and low difficulty as well as good expansibility.**

Index Terms—**mobile middleware, HTML5, B/C/S, cross-platform**

I. INTRODUCTION

With the acceleration of the informatization process and the rapid development of various wireless access technologies, accessing the Internet and exchanging data through mobile terminals such as iPad, PDA and other mobile devices has become more and more popular among users. Traditional mode of office working can no longer meet the business or governmental office requirements. The research and design of the mobile middleware that are suitable for various terminal platforms has become an important topic in the area of mobile Internet. As one of the most cutting-edge Web development technologies in the next decade, HTML5 [1,2] will turn over the existing Web application development model[3]. Along with the development of the WebKit-based mobile browsers and the launch of a large number of mobile Web development platforms, the mobilization of the Web applications, the delivery form of the new applications and the foundation of a large number of developers have become the focus of the markets, which allows the smooth migration from the native system to mobile development [4, 5].

The middleware [6,7,8] means the general services between the platform (hardware and operating system) and the applications. These services have standard programming interfaces and protocols [9,10]. For different operating systems and hardware platforms, they can have different implementations conforming to the interface and protocol specifications. The programming interfaces provided by the middleware define a high-level application environment which is relatively stable. However the underlying computer hardware and system software have changed, we need only to update the middleware and keep the external interface definition of the middleware unchanged, the application software need not to be modified in most cases, thus protects the significant investment in application software development and maintenance [11,12].

The mobile middleware platform proposed in this paper, adopting a B/C/S mixed structure, supports developers to download all the pages to mobile devices through a B/S guidance page. During the running process, they can communicate with the server through AJAX technology, thus create a C/S structure or localized and even network-independent applications. Furthermore, according to the underlying characteristics of the implementation for Android, it can encapsulate the system functions into expansive JavaScript interfaces. Developers need only to call these interfaces in the Web pages to achieve the corresponding functions. The research results show that this mobile middleware can enable developers to use HTML5+CSS3+JavaScript as well as a wealth of third-party components to develop mobile applications, and in all aspects of the software lifecycle such as development, deployment, management, and maintenance, this can reduce the threshold, increase efficiency, help to extend the IT applications to the mobile terminals quickly with low-cost and low-risk, and at the same time take advantages of the new features of the mobile terminals to conduct business innovations.

II. RESEARCH OF MIDDLEWARE TECHNOLOGY

A. Current Situation of Middleware Technology

Middleware technology started to develop and grow up

after overcoming the common problems of the complex network applications. These problems can be summarized in four aspects:

- **From the point of view of computing environments:** what the middleware confronted is a complex and constantly changing computing environment, which requests that middleware technology should have sufficient flexibility and expansibility;

- **From the perspective of resource management:** the resources with finite types and quantities that operating system and database management system administrated are very limited. However, the resources, such as statistics, services and applications, which middleware need to manage are abundant and the boundaries of resources to be extended are divergent;

- **From the point of view of application support:** middleware needs to provide the overall operation model of the entire life cycle, which is distributed application development, integration, deployment and operation management;

- **From the point of view of applications:** the applications that use middleware to be accomplished are always complex enterprise ones, which have a complicated relationship with each other. For instance, the customer relationship management system needs to integrate multiple internal applications, while supplying chain management involves the integration of different enterprises.

Therefore, due to the complexity of network applications, in particular their distributed, heterogeneous and autonomous characteristics, thus determines the morphological diversity of middleware technologies and products. The academic field has proposed corresponding mobile middleware technology in order to meet the requirements of mobile computing. These middleware technologies adopt different strategies to meet the demands of mobile middleware as much as possible. Combining their architectures, characteristics and limitations of these mobile middleware, mobile middleware can be divided into the following four categories: the reflection-based middleware, the context-aware middleware, the event-based middleware and the message-based middleware.

B. Research of Existing Problems

There is a great distance between the principles that the middleware system must follow and its current situation. Most popular middleware services use proprietary API and protocols, which makes it difficult to manipulate interactively among different manufacturers. And middleware services only apply implementations for some platforms, thus limit the transplantation between heterogeneous systems. Besides, Application developers also need to take risks to build their own applications over these services, because they often need to rewrite their systems as technology develops. Middleware services to improve the degree of abstraction of

distributed computing, application developers need to face many difficult design choices.

Because of the limitation of the processing power of mobile terminals, the middleware that supports mobile applications must be lightweight and must not take up too many system resources. However, at the moment, traditional middlewares are heavyweight and lack of flexibility, which make them fail to meet the requirements of mobile applications. Therefore, it brings challenges to the smooth migration of mobile applications from IT systems.

Practice shows that it is feasible to transplant the existing middleware platforms to the mobile network on a technical level. However, the platforms having poor performance can't support the adjustment for the adaption and the asynchronous interactions of mobile applications, and need to spend large costs for the development and maintenance. Therefore, it is particularly significant to investigate the mobile terminal middleware platform based on HTML5. The new solution will be discussed in the following section.

III.   MOBILE MIDDLEWARE BASE ON HTML5

A.   Design Principle

The Android implementation mechanism in the HTML5-based mobile middleware adopts the development mode of Android that developers can use HTML pages as software interface. The WebView control realizes the communication between JavaScript and Java code, so that the Android software interface can be developed by using the HTML web technology. Mobile JavaScript, Asynchronous JavaScript and XML (AJAX) [14] can enhance the interactivity of the dynamic mobile Web sites, reduce its impact on the mobile network and improve its availability in smart phones and other device equipped with advanced mobile browser [15]. At the same time, the platform supports developers to download all the pages to the mobile devices through a B/S guidance page. During the running process, they can communicate with the server through AJAX technology to create a C/S structure or localized and even network-independent applications. This method maintains not only all the benefits of the B/S structure, but also that of the C/S structure.

B.   Design Scheme

In this paper, the solution of the mobile terminal application middleware platform is shown in Figure 1.
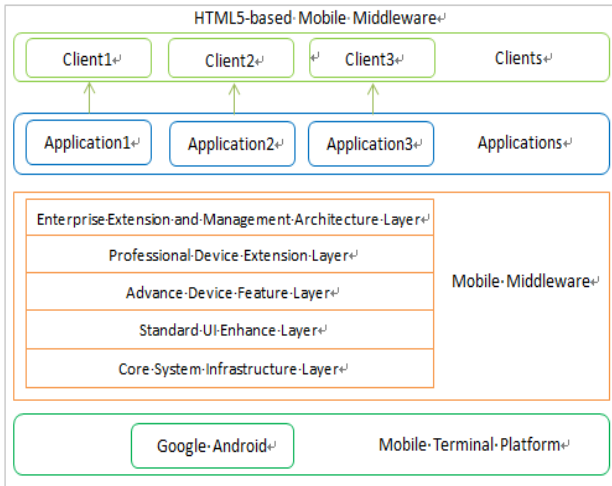
Figure 1   Platform framework of the application middleware of the

mobile terminal

The middleware, located upon the operating system of client/server, manages the computer resources and network communications. It is used to connect two independent applications or systems. Two connected systems, even though they have different interfaces, are able to exchange information through middleware. A critical method to execute middleware is information transmission. With the middleware, applications can work in multi-platforms or OS environments.

The mobile middleware is divided into five layers. Besides the Core layer, the remaining four layers focus on different functional directions and areas:

The CSI (Core System Infrastructure) layer: this layer concentrates on establishing an extensible, standard and efficient system infrastructure, and the integration of the WebKit kernel and the third-party frameworks or components.

The SUE (Standard UI of Enhance) layer: this layer concentrates on the extension of the expression layer. It enables the developers to establish user interfaces that are more suitable for the terminal characteristics.

ADF (Advance the Device the Feature) layer: this layer concerns the use of the functions related to the mobile terminal equipments, and supports the business innovations that take advantages of the new functions of the terminals.

The PDE (Professional, Device Extension) layer: this layer concerns the further exploration, expansion and extension of the relative functions of mobile devices.

The EMA (Enterprise the Extension and Management Architecture) layer: the layer concerns the enterprise expansion of the objects and is used to establish the framework among systems management, equipment management, remote control and the integrations or collaborations of the server sides.

C.   Technical Architecture

- **System architecture diagram of the mobile middleware**

The bottom of the HTML5-based mobile middleware is the system core layer, which is used for the integration of the WebKit kernel and the third-party components; the next layer is user interface layer that can be used to establish expressive user interfaces according to the terminal characteristics; the third layer is the device layer, which supports developers to conduct business innovations according to the terminal characteristics; the top two layers are the JavaScript interface layer and the enterprise management layer, which are used for the integrations of the server sides and self-defined JavaScript engine as well as the third-party frameworks. Developers can reasonably use them to call the JavaScript interfaces provided by the system to implement business functions. The system architecture diagram of the middleware is shown in Figure 2.
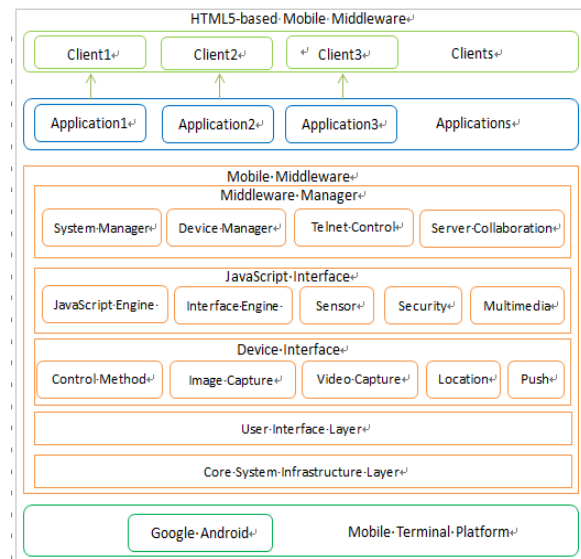

Figure 2   System architecture diagram of the mobile middleware

- **Startup process of the mobile middleware**

The startup process of the mobile middleware mainly involves the following four classes: DratekMapAbcActivity, DratekLayout, DratekBrowser and DratekUrl. When the middleware is started, it will read the main activity (DratekMapAbcActivity) defined in the AndroidManifest.xml file. It will call the onCreate() method extended from its super class to complete the initialization operation when the Activity is created. Then the system will create a layout object from the context, which will load the index address of the application defined in the DratekUrl class and then load this address into the browsers of the mobile devices. The system will call the init () initialization method first to register the JavaScript objects defined by the middleware as the plugins of the interface lists of the browser. Finally, it will call the initUserAgent() method to initialize the User-Agent headers in the device and the main Activity will call the load() method to load the Intent obtained from the context. The startup process of the mobile middleware is shown in Figure 3.
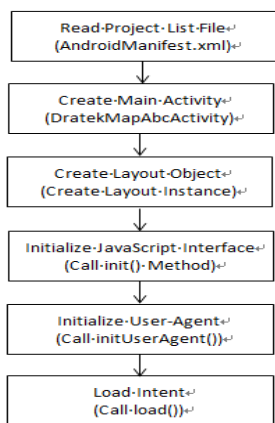
Figure 3 Startup process of the mobile middleware

IV. RESULTS AND DISCUSSIONS

A. Function Discription

The HTML5-based mobile middleware supports engineers to develop mobile applications through standard HTML5+CSS3+JavaScript and abundant third-party components. At the same time, the mobile middleware extends the JavaScript scripting language greatly. On the one hand, it provides friendly supports to the third-party frameworks such as AJAX, Dojo and JQuery, and on the other hand, it supports developers to implement the interaction between the browsers and servers with the self-defined JavaScript engine, laying a solid foundation for the business expansion and innovation based on the terminal capacity. Table I summarizes the functions of the mobile middleware.

TABLE I

TABLE OF THE MOBILE MIDDLEWARE FUNCTIONS

Business type	Function module	Detailed description
User Interface	Screen Switching	Set the screen of the applications, including setting the screen directions, get the screen directions, get the current screen directions, get the height and width of the screen and whether to be set the full screen
	Menu Management	Add custom applications
E-Book Special Effect	Flipping	Implement the e-book flip effect, and whether to show or hide the dialog box
Multimedia Support	Animation Play	Control the effects of animation of applications, including playing animation, repeating animation, finishing animation and setting the animation style
	Audio Play	Set audio effect, including playing audio, pausing audio, stopping audio and resuming audio as well as setting the audio volume
	Video Play	Set video effect, including playing video, pausing video, stopping video and resuming video as well as setting the video volume
	Flash Support	Control the Flash file
Barcode Scanning	Barcode Scanning	Provide bar code scanning
Sensor Support	(Brightness, Temperature, Magnetic, Acceleration, Direction and Distance) Sensor	Provide all kinds of sensors function of applications
Security Strategy	Data Encryption or Decryption	Offer a variety of algorithms to encrypt and decrypt the data, including DES, 3 DES, AES, RSA and PBE algorithm
	Digital Abstract	Obtain the summary information for a specific string
	Digital Signature	Conduct RSA signature and verify the signature of the data
PUSH Service	HTTP SMS PUSH	Conduct HTTP PUSH for information
Location Service	SMS PUSH	It can be used to conduct SMS and MMS PUSH for information
	Station Location	Listen to the data of the station and get the station data
	GPS Location	Obtain the location information through GPS technology and calculate the distance of two points
Other Services	AJAX Support	Obtain data from the Internet
	XML Document Parser	Parse the XML file
	Picture and Video Capture	Capture pictures and videos

B. Expected Target

The mobile middleware is the bridge between the software presentation layer and mobile terminal equipment. It should achieve the following objectives:

Investment protection: the mobile middleware can ensure the smooth migration from traditional IT systems to the new mobile IT systems with fewer modifications to the original systems, but more focuses on the adjustment of the presentation layer. During the process of business innovation, this mobile middleware can ensure the smooth embedding of the new business logics into the existing business systems.

Mainstream architecture: the mobile middleware, building mobile application systems based on the multi-layer B/S architecture in this area, can exert the advantages of B/S such as easy to develop, easy to deploy,

easy to manage, easy to integrate and easy to maintain.

Standards to follow: this kind of mobile middleware follows all the standards in this area, avoiding creating self-defined script languages or markup languages.

Maximum Reuse: the mobile middleware provides nice support to the standardized component models and third-party frameworks or components.

Strongest function: the mobile middleware can make full use of the functions and performances of the mobile terminals themselves. No matter smart phones or pads, many mobile terminals provide new functions that personal computers don't possess, such as GPS positioning, bar code scanning and various sensors and so on.

Easy to integrate: mobile application systems should not be isolated ones, but should become an important part of the IT infrastructure of the entire enterprise. This requires that mobile application systems should have the capability to integrate and collaborate with other systems. Moreover, the mobile middleware should ensure that the established mobile application systems can be ready for management and collaboration in the premise of minimum workload.

Minimum cost: the minimum workload, the shortest cycle and the easiest to maintain. Cost is a crucial factor to be considered for the construction of any system. We should not simply consider the construction cost, but should consider the total cost of ownership (TCO) from the point view of the complete software lifecycle, which involves all aspects such as development, deployment, maintenance, management and upgrade and so on. This kind of mobile middleware should guarantee that the created mobile application systems can maintain the lowest cost and the best results in all aspects.

C. Application Design

The technologies to develop HTML5 applications are HTML5, CSS3 and JavaScript. Generally speaking, the HTML5 standard contains these three techniques. The HTML5-based mobile middleware applications are in full compliance with W3C standards, which ensures the fully compatibility with all the third-party frameworks and components. Meanwhile, the platform also supports customized frameworks. Moreover, it can not only improve the loading speed, but also reduce network flow. Developers could design applications with complex functions as long as they are able to use these JavaScript interfaces provided by the platform. This paper will take the E-Book designed by the middleware for example.

- **Function description and operating results of E-Book**

Create a new Android project, put all the resources and pages into the assets directory, and then let the project's main Activity class inherit the DratekMapAbcActivity class of the middleware. In the onCreate() method of the main Activity class, it will call the loadUrl(appUrl) method of the super class and transfer the index link as the parameter. Finally, it will call the onCreate() method to load the link. When the middleware is started, it will

load the index page of the application. In addition, in order to avoid the black screen phenomenon in the booting process, developers must set up the theme for the application. Here it is set to android:theme="@android:style/Theme.

Translucent.NoTitleBar.Fullscreen" and android: configChanges="orientation|keyboardHidden". For example, the index page of the E-Book application shows a list of poems written by Li Bai. It provides many menu bars to control the applications at the bottom of the screen, which includes the functions of restart, exit and uninstall and so on. All the functions can be implemented just by calling of the JavaScript interfaces provided by the middleware. After clicking bottoms in the list, it will display all the poems in this volume. The effects of the E-Book are shown in Figure 4, Figure 5, Figure 6, and Figure 7.
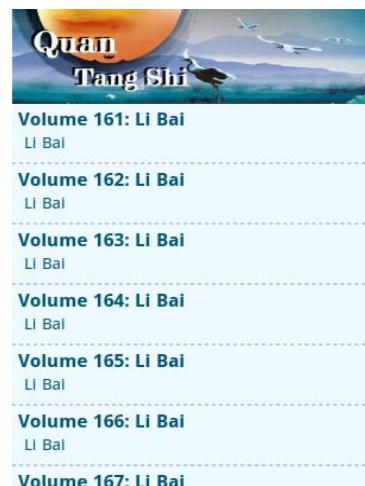

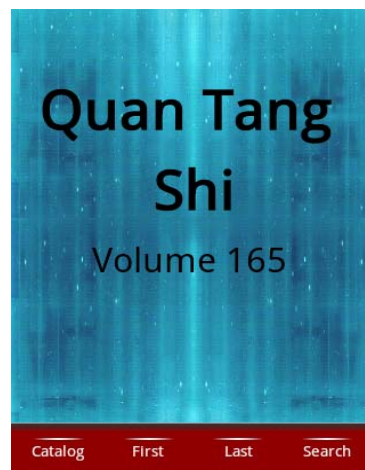Figure 4   E-Book list display

Figure 5   E-Book Details display

Figure 6   E-Book catalog display

Figure 7   E-Book search function display

D.   *Comparison with Other Middlewares*

- **Comparison with Traditional Middleware**

On the one hand, compared to the traditional B/S structure system, the mobile middleware proposed in this paper supports all the local functions of the mobile terminals, dramatically expands the JavaScript language and provides a set of JavaScript interfaces for the applications. The mobile middleware is in full compliance with W3C standard, ensuring the complete compatibility with the third-party frameworks and components. At the same time, the platform also supports self-defined frameworks, which both improves the loading speed and reduces the network traffic.

On the other hand, the traditional mobile middleware divides the system into two levels. The bottom layer is the combination of the basic mobile middleware and the runtime library of the embedded operating system, providing common connection management, data transmission services, content service management, adaptive services and context-aware infrastructure services; the top layer offers a variety of services to achieve the collaboration with stationary applications, fixed services, mobile applications, and universal services. The mobile terminal application middleware platform

proposed in this paper, compared to the traditional mobile middleware, is aiming at designing a universal development pattern for the developers and refining the functions that need to achieve from the system layer to application layer. At the same time, in each of the aspects of the software life circle such as development, deployment, management and maintenance, this scheme not only reduces the technical requirements, improves the development efficiency, but also enables the developers to conduct business innovation by using the new features of the mobile terminal.

- **Comparison with Other Mobile Middleware**

At present, PhoneGap is the most widely studied mobile middleware in this area. PhoneGap is also a rapid development platform that based on HTML5, CSS, and the expansion of JavaScript technology to create cross-platform mobile applications. It abstracts and simplifies the complicated APIs provided by the mobile devices themselves and provides some APIs for developers to call. As long as developers have a good knowledge of HTML5 and JavaScript or Java language, they can call various functions by using the API provided by PhoneGap, thus achieve the application development.

PhoneGap's realization of the callback from Android client to Web server is actually based on the long backpack method. Firstly the Web server launches a cyclical call of AJAX, secondly the Android client starts up a local SocketServer and maintains a JavaScript queue. If the request comes, the Android client will return the JavaScript statement in the queue and then the Web server will execute this JavaScript statement in order to simulate the callback from Android client to the web server. This differentiated from the mobile middleware mentioned in this paper in that PhoneGap needs more system resources as it needs to initiate a cyclical callback of AJAX to the Web server and start a local server from Android client every time. However, the mobile middleware mentioned in this paper can only startup once to complete the application loading, and frequent operations won't lead to slower response speed.

In addition, the table below makes a detailed comparison between this mobile middleware and PhoneGap. Study shows that functions supports by this mobile middleware are more than PhoneGap.

TABLE II

THE COMPARISON OF THE SUPPORT ABILITY OF THE MOBILE MIDDLEWARE WITH PHONEGAP

Type of Middleware / Function	Mobile middleware based on HTML5	PhoneGap
Screen Switching Function	Yes	Yes
Menu Management	Yes	No
Flipping Function	Yes	No
Animation Play Function	Yes	No
Audio Play Function	Yes	Yes
Video Play Function	Yes	No
Flash Support	Yes	No
Barcode Scanning Function	Yes	Yes
Sensor Support	Yes	Yes
Security Strategy	Yes	No
PUSH Service	Yes	Yes
Location Service	Yes	Yes
AJAX Support	Yes	Yes
XML Document Parser	Yes	No
Picture and Video Capture	Yes	Yes
Contacts	No	Yes
Storage	Yes	Yes
Device	Yes	Yes

V. CONCLUSION

This paper puts forward a solution of HTML5-based mobile terminal platform, and gives a detailed description of its design principles as well as the meaning of all the layers and at the same time gives an overview of its supporting capability. In addition, the mobile middleware encapsulates many kinds of services into JavaScript interfaces, which are called reasonably to implement corresponding functions. Therefore, it can shorten the development cycle of the application software, reduce the development costs, improve the efficiency greatly and offer a good scalability.

REFERENCES

[1] Shen Tao. Design and Development of the HTML5 Based Product Demonstration System [D]. Master degree thesis of Beijing Jiaotong University, 2011.
[2] Michael Grady. "Functional programming using JavaScript and the HTML5 canvas element". Journal of Computing Sciences in Colleges. pp. 1-4. December 2010.
[3] Yi Li, Suwei Zhang. Design and Implementation of a Mobile Development Midddleware [J]. Computer Engineering and Design. 2006, 27(23): 4504-4509.
[4] Yuehua Yu & Lei NingWei & zhong Liu. "Combining HTML5 with MVC Framework to Simplify Real-Time Collaboration for Web Development". Mines,vol.2,pp.29-32,2009. Internation Conferance on Multimedia Information Networking and Security, 2009.
[5] Kris Read, Frank Maurer. Developing mobile wireless applications[J]. IEEE Internet Computing, 2003, 7(1):81-86.
[6] Xianghui Liu. Study and Design of the Java Based Mobile Terminal Middleware [D]. Master degree thesis of East China Normal University, 2005.
[7] Yiming Li, Jie Ling. Design of a Safe Message Middleware [J]. Computer Engineering and Design, 2011, 7.
[8] Capra Licia, Emmerich Wolfgang; Mascolo Cecilia Middleware for mobile computing: Awareness vs. Transparency, Proceedings of the Workshop on Hot Topics in Operating Systems-HOTOS, 2001, pp. 164.
[9] T.Renji,C.Shihua,A Finite Automation Public Key Cryptosystems and Diginal Signature[J]. Chinese Journal of Computers,v.8,1985,pp.401-409.
[10] Johannes Behr & Peter Escher & Yvonne Jung & Michael Zoellner. "A scalable architecture for the HTML5/X3D integrations model X3DOM". Web3D'10.1-76.2010.
[11] Amit Parnerkar, Dennis Guster, Jayantha Herath.Secret key distribution protocol using public key cryptography[J]. Journal of Computing Sciences in Colleges,2003,19(1):182-193.
[12] Tran P,Greenfield P.Behavior and Performance of Message Oriented Midleware Systems[C]. Proceedings of the 22th International Conference on Distributed Computing Systems Workshops,Vienna,2002:645-650.

[13] Farhad Arbad, Marcello Bonsangue, Frank de Boer, A Middleware Based on Mobile Channels, Proceedings of the 26th Annual International Computer Software and Applications Conference(COMPSAC'02),0730-3157/02©2002 IEEE,computer socie.

[14] Vichr R, Malhotra V. Middleware smooths the bumpy road to wireless integration[M]. IBM Developer Works, 2001.

[15] D. Auerbach. Controlling Extended Systems of Chaotic Elements. PHYSICAL REVIEW LETTERS. 1994, 72(8): 1184-1187.

Xiaolin Yi was born in July, 1959. She received a B.E. degree from Tsinghua University of China, an M.E. from Beijing University of Technology of China. Her current research is in embedded system, information security, computer architecture and network technology.

She is a senior engineer and speaker professor of the College of Computer Science of Beijing University of Technology. And she is a Senior Membership of CCF and ACM Community and a contributing reviewer of CCCF.

Yishuang Ning was born in September, 1986. He is a graduated student from Beijing University of Technology. His research area is mobile middleware platform based on HTML5.