

Design of Smart Home System Based on Service Encapsulation and Device Abstraction

Quan-Feng YAN^{1,a}, Yue-Bin WANG^{1,b,*}, Yan XU^{1,c}, Li-Qiang LIU^{1,d},
Bo FANG^{1,e}

¹College of Computer Science, Hunan Institute of Science&Technology, Yueyan, 414006, China

^aacmfcc@163.com, ^byuebingw@163.com, ^cxuyan@163.com, ^dliuliqiang2@163.com,
^ebandfan@163.com

*Corresponding author

Keywords: Smart home, Device abstract, Service encapsulation, UPnP.

Abstract. Through research and analysis about smart home, a smart home system based on service encapsulation and equipment abstract solutions have been proposed. It's combining pervasive computing and mobile computing technique centered on home intelligent gateway, through the device abstract and service encapsulation technology, changes home applications into protocol-independent devices and provides a platform where mobile terminal equipment can conveniently monitor home environment and control home devices intelligently ultimately. Experiments demonstrate that this smart family can largely facilitate the people to manage family devices.

Introduction

As the digitization process continuously advances, the embedded technology has gradually penetrated into people's daily lives, and thus all kinds of information appliances emerge. Although these information appliances have facilitated people's lives to some extent, they are often independent from each other and cannot be connected with the Internet for information transmission and resource sharing, and therefore they are not smart in the true sense. In order to overcome the barrier of the "information silo" and achieve the mutual connection and cooperation among smart home appliances, the Internet of Household Things [1] comes into being.

In view of the current status of domestic researches on home network, this paper conducts researches of the basic principle, operation mechanism and implementation method of the typical UPnP technology [2]. On the one hand, the technology of device abstraction and service encapsulation is applied to build the Internet of Household Things; on the other hand, because UPnP is restricted within the environment of local area network, the XMPP [3] protocol is used under the technological framework of M2M, which serves applications on the Internet of Things, to realize information interaction between inner and outer networks, so as to realize remote monitoring and enable users to gain timely information of household environment from the outer network. Targeting ordinary families, this scheme involves the integration of local control and remote control, and has a great value for popularization and application.

The organizational struct.of this paper is as follows: the first section mainly introduces relevant protocols; the second section analyzes service encapsulation and device abstraction; the third section presents the diagram of the frame of the whole household system and expounds the design and implementation of related modules; the fourth section states situations regarding the implementation effect; and the final part is a summarization of this paper.

Analysis of Related Protocols

In this system, the UPnP protocol is applied for building a family inner network, and the XMPP protocol is used for building the information channel between the family and the device server. Mainly used for realizing the interconnections among smart devices, UPnP involves such

mechanisms as device discovery, device inquiry, device control, device subscription, etc., demands no driver, and can run on almost all the existing operating system platforms. UPnP describes devices and services by using XML, and the communication mode of the XMPP protocol is also based on XML[4], thus the decoding of data between the client and the smart gateway can become very easy when remotely accessing home appliances.

UPnP

An UPnP network usually consists of three basic parts: device, service and control. The UPnP protocol is an application layer protocol running on the IP layer, and its specific implementation process does not pertain to the underlying network media. UPnP achieves such functions as device discovery, device inquiry, device control, event notification, etc. by defining protocols among devices, between devices and the control, and among controls.

XMPP

The XMPP protocol is an XML-schema-based and highly extensible open protocol, which has inherited the flexibility in the XML environment and deals with users’ requests by sending structured information based on XML.

The framework struct.of XMPP service defines three roles: XMPP client, XMPP server and gateway, among which the server assumes multiple tasks such as recording client information and connecting the routes of management and information while the gateway is responsible for connecting to heterogeneous instant communication systems.

Service Encapsulation and Device Abstraction

Home network contains a variety of digital home appliances, which are based on different communication protocols and thus cannot interconnect with each other or receive unified smart control and management. To solve this problem, this paper introduces the technology of service encapsulation [5] and device abstraction.

Service Encapsulation

The process of service encapsulation is conducting uniform descriptions according to the properties and functions of physical home appliances for the purpose of indiscriminately treating access of applications in the upper layers. The UPnP protocol uses XML to describe devices, thus device information can be easily decoded by merely invoking the API of UPnP.

For a physical device, information received through the gateway sensor is merely a hexadecimal numeric string, containing its address, type, data, calibration data, and so on. The format of the data is shown in Table 1. To achieve undifferentiated disposal of the upper applications requires transforming the properties of the devices and their services into XML description documents. First of all, the address of the device is mapped as UUID, which serves as the unique identifier of this device; then the name of the device type is defined according to its specific type, and used as a node of XML. Before the design of device data, a service node should be first set up to encapsulate information such as service type, service ID and service URL, and also the specific action name, state variables and parameter list for the service should be defined in the extended URL.

Tab. 1 Sensor instruction table

Address	deviceType	deviceData	CRC
---------	------------	------------	-----

Device Abstraction

Furthermore, in order to facilitate undifferentiated management over the devices, a device that can be recognized by the application program should be established after the service encapsulation. Physical home appliances are abstracted first to category based on the object-oriented principle, and then to member variable in the light of the functions of physical home appliances. Some methods

are devised to realize the operation towards the member variable, such as member variable settings, updating, inquiring, and so on. Meanwhile, an interface will be set up to interact with the lower protocol. Then, by using such mechanisms provided by the UPnP protocol as device addressing, device discovery, device control and event notification, the device is released on the network. Thus, it seems as if such a device really existed from the perspective of the control, and all the control and inquiry commands from the control to the virtual device will be accurately transmitted to the physical home appliance; updates of the statuses of physical home appliances are sent to the control in the form of UPnP protocol after data decoding and re-encapsulation and clearly displayed at the control, so as to realize smart control of the devices.

Overall Design and Implementation of the Smart Home System

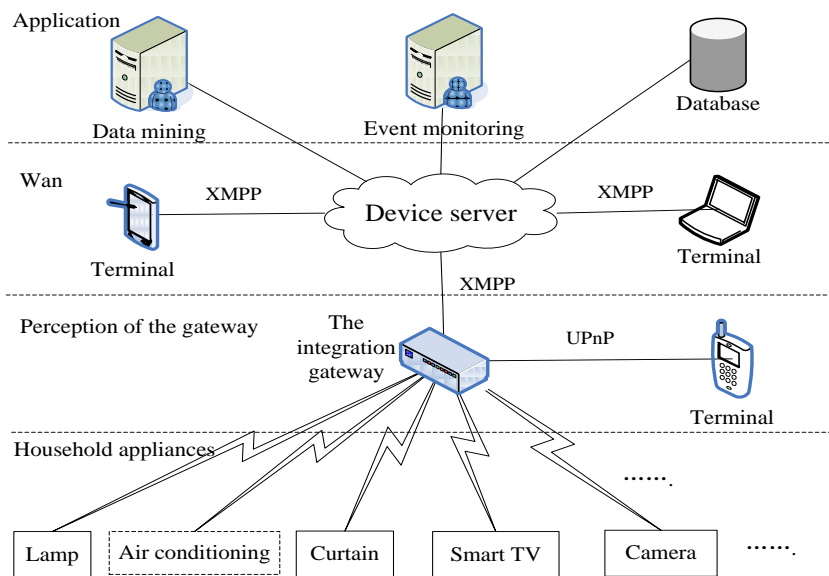


Fig.1 The overall system framework

Based on the above research of UPnP and XMPP, this paper proposes a smart home scheme based on service encapsulation and device abstraction. According to the scheme, this system is mainly composed of the sensing layer, the network layer and the terminal. The architect.of the whole system is shown in Fig.1:

The role and function of each module is as below:

The sensing layer contains the smart home integration gateway [6], which is the core part, and ubiquitous devices [7] (e.g., smart appliances, sensors, etc.). It has the following functions: (1) integrating all the ubiquitous devices, so as to realize the seamless access to the devices; (2) building a device pool[8] for the centralized management of the ubiquitous devices;(3) achieving protocol conversion between the outer network and the inner network, so as to realize the remote control; (4) retransmitting data, including the control and inquiry commands sent by the control as well as uploaded data resulting from device updates.

In our design scheme, terminal devices such as mobile phone, tablet and computer act as the control for other devices. The terminal provides users with an operational interface, which details detected devices and their services, facilitating users' operation.

Design and Implementation of the Sensing Layer

Household environment generally contains four types of devices: service devices, control devices, security and protection devices, and daily used devices. The integration gateway is aimed at achieving seamless connection between these devices and the home network. It consists of three sub-modules: engine for protocol conversion, virtual device, and virtual device pool. Fig.2 shows the struct.of the integration gateway:

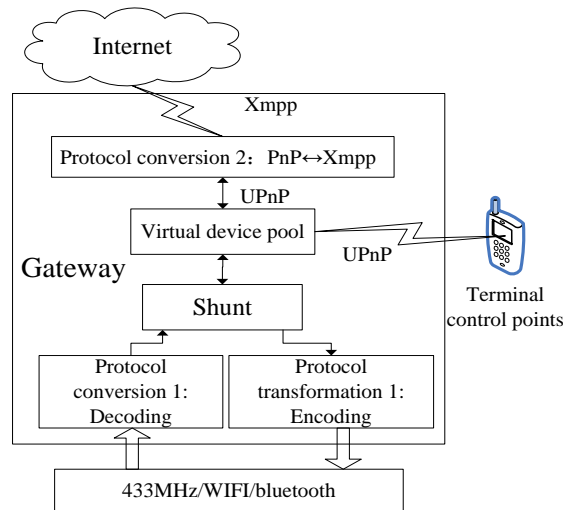


Fig.2 Household integration gateway

Engine for protocol conversion: two kinds of conversion forms are involved here:

(1) 433MHZ/WIFI/Bluetooth ↔ UPnP: because data collected from home appliances cannot directly interact with the upper layer, it must first be decoded for the extraction of device information, and then transmitted to the virtual devices; when a command is forwarded from the control, it must be coded in line with specific devices before being transmitted to physical home appliances through the gateway.

(2) UPnP ↔ XMPP: UPnP messages are encapsulated into XMPP messages and sent to the device server through the HTTP protocol; when the mobile terminal is on the outer network, the forwarded commands are first encapsulated into XMPP messages, retransmitted through the device server to the smart home gateway, where they are converted to UPnP messages, and then sent to corresponding virtual devices. Interfaces for the two kinds of conversions are shown in Table 3:

Tab. 2 Protocol conversion interface

Conversion interface	Describe
UPnPToXmpp	UPnP to XMPP
XmppToUPnP	XMPP to UPnP
UPnPToDev	UPnP to physical data
DevToUPnP	Physical data to UPnP

Virtual devices: service encapsulation must be conducted for functions of home appliances, followed by function extension in the UPnP-defined framework by using UPnP's device virtualization mechanism. The device abstraction function customized in this system first completes the initiation of device service and action, and then sets the service and action monitoring to start monitoring; meanwhile, data from the lower layer is first UPnP encapsulated, and then sent to the control, to update the device status. The codes for the callback function of monitoring the service and action are as below:

```
bool actionControl Received (Action *action) {
// receive action messages, and determine the types of the action, and then decode the data in the
action, and forward it to the protocol conversion engine for encapsulation.}
```

Virtual device pool: It is responsible for device registering, adding, deleting, initiating, etc. When a physical home appliance goes online, whether the device has been registered will be first judged: if it is not registered, the relevant information of this home appliance will be inserted to this table, then the device is added to the virtual device pool, and then this virtual device will be started; if it is found that the device has existed, the only subsequent step is to start the virtual device. Thus the device will exist in the form of physical object in the gateway.

Tab. 3 virtual device pool registry (hex)

Device type ID	Device ID
00(Lamp)	01
00(Lamp)	02
01(Air conditioning)	01
02(Smoke sensor)	01
.....

Design and Implementation of the Network Layer

The device server is developed on the basis of the Openfire platform. Based on the XMPP protocol. In the device server, the plug-in is responsible for recording information of the client and the devices, processing data, monitoring events, and providing some business users with service interfaces to implement linkage monitoring. On the one hand, information of family devices can always be sent through the home gateway to the device server for storage, and on the other hand, the remote client can always log in the device server to find his/her own family devices and get aware of the statuses of these devices. At the same time, control commands can be transmitted from the remote terminal, and retransmitted by the server to corresponding devices in users' home, changing the device status. The struct. of the device server is shown in Fig.3.

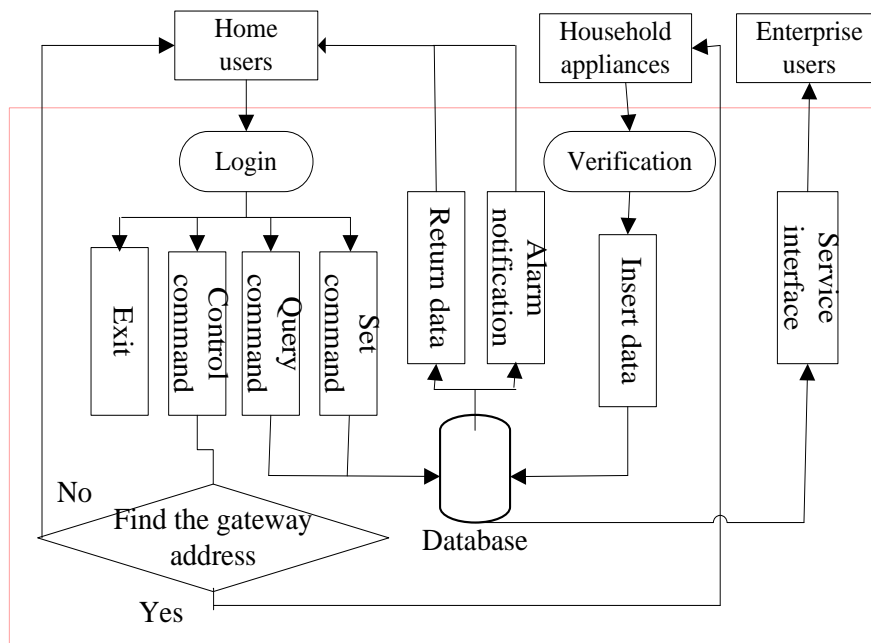


Fig.3 Device server architecture

Design and Implementation of the Terminal

The terminal mainly functions as the control for managing household devices. It is mainly responsible for device information obtaining, device inquiry, event subscription, device control, etc. This terminal in this system is developed mainly based on the Android platform. It has the family mode and the outer-network mode. The family mode: when the users are at home, they can directly log in the home gateway to search for devices, check device status, send commands to devices, etc. The outer-network mode: if the users are on the outside network, they can first log in the device server for validation, and if they get through the verification, the function of the family mode will be available for the terminal, and also messages can be retransmitted through the device server. The proced. of using the terminal is shown in Fig.4:

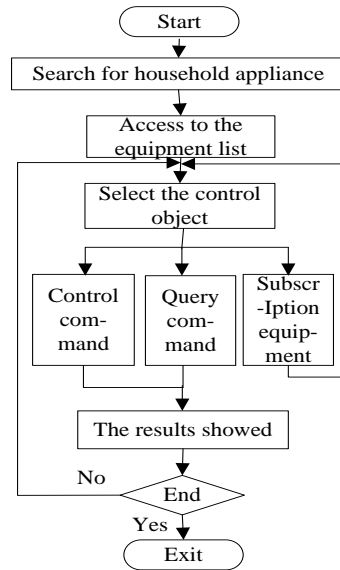


Fig.4 Terminal using the process

Implementation Effect and Analysis

This system uses the Cyberlink [9] open source library to implement the UPnP protocol stack, and uses the Gloop open source library to implement the XMPP protocol. The XMPP server, based on Openfire, is developed by using eclipse on Windows; the integration gateway system is a Linux-based openWRT firmware system, developed on the basis of UPnP virtual devices on Linux; the terminal is set on the Android platform where standard UPnP control is developed. The experimental environment is made up by integration gateway, device server, mobile client and physical home appliances. The system's work proced.in the family mode is as follows:

- (1) First of all, the user logs in the system, the background begins to initiate various components, and the monitoring device goes on-line;
- (2)The home appliances go online, and if a device is new, it will join the virtual device pool, or else the virtual device will be directly started;
- (3)The client receives the information of the device going online, gets the properties of the device, and creates an object depending on the type of the device;
- (4) The client sends commands;
- (5)The virtual devices decode the commands, and send them down to corresponding household appliances;
- (6) Home appliances follow the commands by inversely forwarding their status information to the virtual devices;
- (7) The virtual devices encapsulate the information, and transmit it to the client, causing the updating of device status on the client side.

The above is the situation when the client takes the initiative to control. In addition, the devices can also take the initiative to upload data: first, data is uploaded to the gateway, decoded, encapsulated as UPnP messages, and sent to the client through the gateway, causing the updating of status on the client side.

In the mode of the outer network, the client first logs in the device server for validation, and perceives the statuses of all family devices from the server. Here the only different part is that: on the one hand, the data uploaded by the devices is transmitted in the form of UPnP messages within the family; on the other hand, the device data, through protocol conversion within the gateway, arrives at the server in the form of XMPP messages, and causes the updating of the statuses of home appliances on the server side. Furthermore, commands can also be forwarded from the client, retransmitted to the home gateway by the server, converted into UPnP messages through protocol

conversion and sent down to the devices, changing the device statuses.

Experiments demonstrate that the virtualization of home appliances by using UPnP's mechanism of service encapsulation and device abstraction can provide users with very direct feelings. Users can quickly inquire and control the statuses of home appliances with a few moves on the client side. At the same time, the mechanism of devices proactively uploading can ensure the constant updating of the statuses of devices on the client side, enabling users to easily and clearly know the household condition. The XMPP-based Openfire server allows users to quickly obtain information of household environment anytime and anywhere, and to make personalized settings as they wish.

Conclusion

This paper proposes a design scheme of smart home system based on service encapsulation and device abstraction, which abstracts devices into platform-independent virtual devices to enhance the integration of the home gateway and heterogeneous devices; meanwhile, the instant messaging protocol XMPP is used for connecting family appliances and the device server so as to monitor household anytime and anywhere, thus effectively solving the problem of space limitation for UPnP. The client visual test of the devices has proved the feasibility of the scheme, which can thus serve as a good reference for the development of the Internet of Household Things.

Acknowledgement

This research was supported by University-enterprise cooperation demonstration base in Hunan province, Scientific research fund of Hunan Provincial Education (14c0512).

References

- [1] Chui M, Löffler M, Roberts R. The internet of things[J]. McKinsey Quarterly, 2010, 2: 1-9.
- [2] UPnP Forum[OL]. Available: <http://www.upnp.org/>, 2013-06-01.
- [3] IETF RFC 3921, Extensible Messaging and Presence Protocol (XMPP): Instant messaging and presence. [S]. Saint-Andre P.2004.
- [4] Hao S, Han X. Heterogeneous relational database integration model based on XML [J]. Computer Engineering and Design, 2010, 31(24): 5285-5288.
- [5] Chen X, Liu X, Zhang X, et al. Service encapsulation for middleware management interfaces[C]//Service Oriented System Engineering (SOSE), 2010 Fifth IEEE International Symposium on. IEEE, 2010: 272-279.
- [6] Valtchev D, Frankov I. Service gateway architecture for a smart home [J]. Communications Magazine, IEEE, 2002, 40(4): 126-132.
- [7] Huang P, Lenders V, Minning P. Jini for ubiquitous devices [M]. Swiss Federal Institute of Technology, Computer Engineering and Networks Laboratory, 200.
- [8] Yin J, Wang Y, Hu M, et al. Application of provenance service in equipment grid, Journal of Tsinghua University (sci&Tech), 2008, Vol, 48, No. 7 (1225-1228).
- [9] Pan F, Wang H, Miao F, et al. The development of instant communication system based on Openfire protocol and XMPP [J]. Computer Age, 2008, 3: 15-19.
- [10] Konno S. Cyberlink development package for UPnP devices for Java [J]. URL: <http://cgupnpjava.sourceforge.net>, 2009.