The Design and Implementation of Combined System Based on Internet Resource Model

Shuzheng Luo[1, a], Chuanchang Liu[2, b]

[1]The State Key Lab of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing 100876, China;

[2] The State Key Lab of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing 100876, China.

[a]shuzhengLUO2013@126.com, [b]lcc3265@bupt.edu.cn

Keywords: Internet of Things; Combined System; Implements and Services.

Abstract. For the past few years, more and more applications and services need to interact with information from real world. Therefore, many researchers are focus on theories and techniques about the Internet of Things and more enterprises have already adopted this new-generation information technique marked with the Internet of Things. In this paper, we design a combined system to enhance the supervision and configuration in the Internet of Things. In order to realize the smart and efficient interoperation among devices, we encapsulate services into widgets and process their interactions using resource modelling in semantic technology.

1. Introduction

With the increase of users of Internet of Things (LoT), existing system could not totally satisfy various requirement of users. For example, common combined system just provide 2D view to users, but they would not like to get 3D view which is more vivid. Then, users always need to get a client of this software and set corresponding configurations before acquiring the view from common the system. However, some users just need to supervise the view provided by combined system instead of all the functions of this software. So it is a waste of hardware costs and deployment costs to install the whole system.

Therefore, it is quite meaningful to design one combined system which could provide 3D real-time supervision and control and convenient network access. To meet various requirement of different industries and different levels of users, the system provide the design and operation features of both 2D and 3D visualization and convenient network supervision and control features and diverse service for different requirement of display. It shows the essential of LoT which is that LoT provides free interaction between users and software in any situation at any time.

The rest of this paper is organized as follows. Section 2 describes some work related by other researchers. Section 3 introduces the whole architecture of the combined system. Section 4 will be filled by the detailed implementation of the system. The last section will make a conclusion to the whole paper.

2. Related work

So far, a lot of work has been done on combined system about service composition. Service composition in graphical user interface has been fully developed in some domain-specific graph configuration software [1]. They provide flexible composition between UI components and services by using SCA and DTS models, which decouples business from user interface and models user interface in different aspects. However, this kind of UI composition has been built on C/S architecture, in most cases, users need to install a full set of the software although they only want to use a small fraction of functionality in the whole software.

Another research on widget orchestration has provided a UI composition in B/S architecture [2]. They encapsulate service into widgets so that they can work everywhere as long as the users have a web browser. This approach, nevertheless, suffers from the problem that their sematic reasoning is built upon common ontology, which is unable to clearly describe the complex relations in some domain-specific environment in the Internet of things.

3. Architecture of Combined System

The combined system mainly divides into two independent modules according to different user groups and functions. An overall illustration of the whole system is shown in Fig 1.

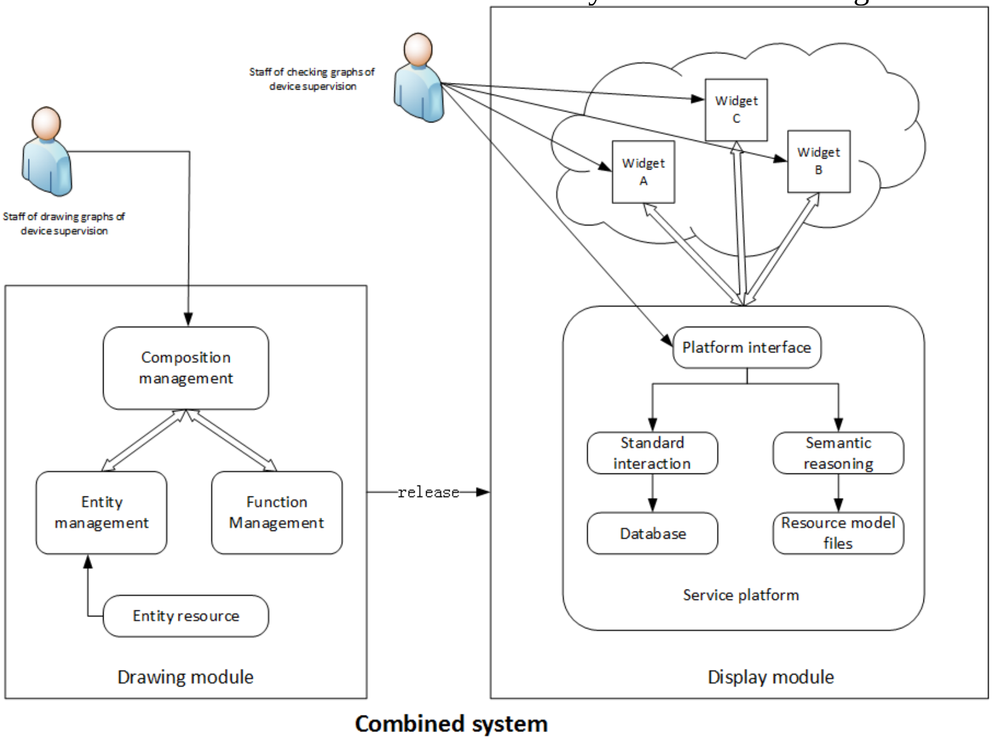

Fig. 1 Architecture of Combined System

The drawing module is based on C/S architecture, providing equipment monitoring diagram editor drawing function. The drawing module consist of two parts, one is 2D graphics (Jgraphx) and 3D graphics (SweetHome 3D). Editors monitoring could release the diagrams to the display module after completion of equipment monitoring on the way of publishing to the web page.

The display module is based on B/S architecture, user groups are equipment monitoring diagram viewers. They can watch the states of the diagram and even control it in the display module only if they have the privileges. Convenience is, they could do it just in the web browser.

The combined system, drawing and display module, separates to allow the users in accordance with the requirements for installation. The equipment monitoring diagram editors is fewer usually, only they need to install the drawing module. On the other hand, equipment monitoring diagram viewers do not need to install any software, just checking the system by a normal browser. Thus, not only can we reduce the hardware investment, but also realize more convenient equipment view modes.

4. Implementation of Combined System

4.1 Drawing Module

The module needs to have the function of making 2D and 3D equipment monitoring diagram. However, it is hard to achieve for such large software cost from scratch, so we are based on the two open source software – Jgraphx and SweetHome 3D. The original software just offer some simple operates for SVG or OBJ files, thus we improve them to achieve more functions such as connect with the database and optimize some operates to make diagrams more conveniently and so on.

We can watch the Fig 2 to realize the whole architecture of drawing module.

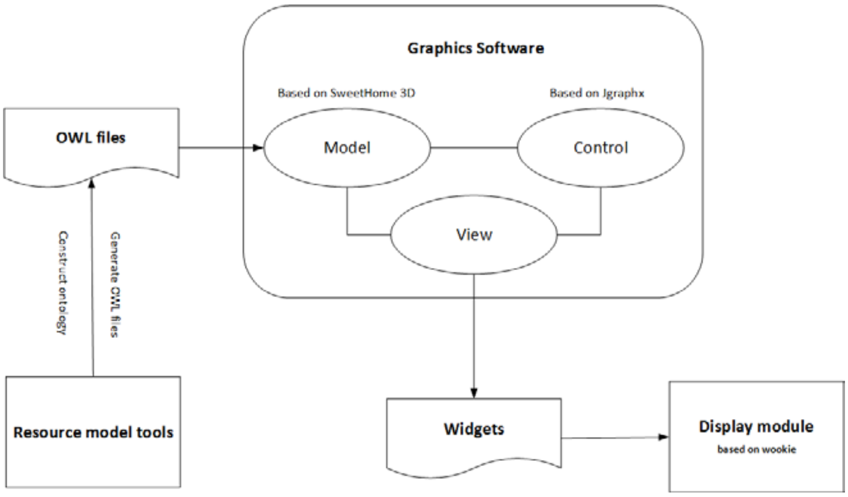

Fig. 2 Architecture of Drawing Module

OWL API is an open source Java library that implements the W3C Web Ontology Language standard. It provides the ontology files import, modify, save and other basic functions [3]. The OWL files that created by related resources model is the overall description combined system using specific information in the field of domain ontology. At the same time, it will be effectively reused and shared for organization in the field of knowledge.

The SVG files is made by Jgraphx while the OBJ files by SweetHome 3D. In this step, we can define our animation, data content or other content in our diagrams. The users could release them to the display module.

4.2 Display Module

The module display the graphics from the files made by drawing module in the form of a widget. Users only need to know the corresponding url, then they can check the 2D or 3D graphics published in the system. The module provides a convenient access method only through the browser doing the job. We can watch the Fig 3

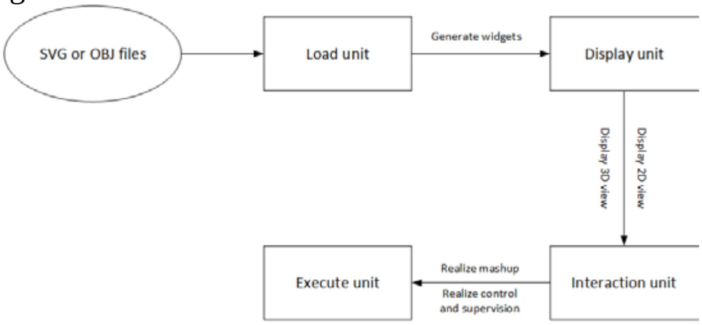

Fig. 3 Architecture of Display Module

A. Widgets

Services are encapsulated into widgets following the W3C Widgets specification. This specification makes basic definitions of widget in terms of packaging format, the configuration document, reserved files and the processing of a widget package [4].

According their different authentications, users may access and control widget, widget instances and participants through the REST API supported by this system [5].

Once a widget is published, it can be presented in different environment. It can be embedded into other web applications, offering the basic functionality built inside, like, displaying shared information and communicate with other widgets. Moreover, it can interact with other widgets on our widget platform using its frontend features. We can watch the Fig 4.

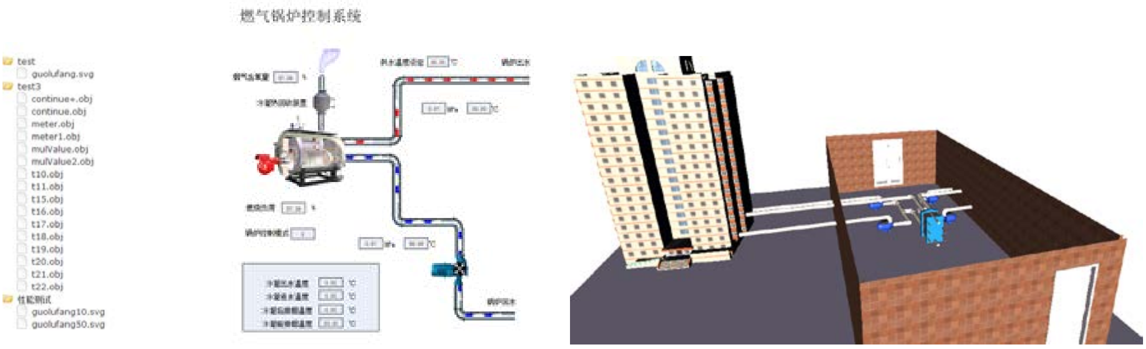

Fig. 4 Tree Widget, 2D Graph, 3D Graph

B. Frontend Features

Defined as JavaScript functions, frontend features reside in our widget platform, aiming to enhance graphical widget interactions. And there are two kinds of frontend features: the shared features and widget-specific features.

Shared features provide common functions that every widget can use in order to interact with others, such as snapping when two widget with certain relations are dragged close to each other.

Widget-specific features are functions to boost certain types of widgets. In terms of interactions, the widgets described in this system falls into two categories: graph widgets and control widgets.

5. Summary

This paper introduces the combined system based on Internet Resource model. It can be used in the supervisory and management of devices in the Internet of things. Drawing module provides a way to produce 2D or 3D graphics while the display module provides a convenient way to show the graphics. Currently, the system do not offer a flexible way to interoperate widgets that encapsulate different services, and other hand, future work may be achieved to reduce the granularity, resulting in the interactions among certain values in widgets.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (Grant No.U1536112).

References

[1]. Youyu Chen. A Graph Configuration Method Base on UI Service Composition. 2011 Seventh International Conference on Mobile Ad-hoc and Sensor Networks. 2011. p. 339 – 340.

[2]. Ahmet Soylu, Fridolin Wild. Mashups and Widget Orchestration Proceedings of the International Conference on Management of Emergent Digital EcoSystems. 2011.11.

[3]. G. Meditskos, N. Bassiliades. A combinatory framework of Web 2.0 mashup tools. OWL-S and UDDI Expert Systems with Application 2011, 38(6).

[4]. Yong Liu, Xiuquan Qiao, Xiaofeng Li. Telecommunications Network Capability Services Model Based on Mashup Technology in Cloud Computing Environment Computer Science 2012, 39(1).

[5]. Buqing Cao, Jianxun Liu, GuangRong Wang. Mashup services network and its topology analysis. Journal of Hunan University of Science & Technology (Natural Science Edition). 2012, 27(2).