Interoperability for Smart Home Environment Using Web Services

[1]Thinagaran Perumal, [1]Abdul Rahman Ramli, [1]Chui Yew Leong, [2]Shattri Mansor, [2]Khairulmizam Samsudin
[1]Intelligent Systems and Robotics Laboratory,Institute of Advanced Technology,Universiti Putra Malaysia, Serdang,Malaysia
[2] Faculty of Engineering,Universiti Putra Malaysia, Serdang, Malaysia
e-mail: thinagaran@hotmail.com, arr@eng.upm.edu.my, chuiyewleong@hotmail.com, shattri@eng.upm.edu.my, kmbs@eng.upm.edu.my

Abstract

Recent advances in computing and communication technologies paved the growth for applications and devices in smart home environment. A typical smart home is highly characterized by heterogeneity elements that need to perform joint execution of tasks in an efficient manner. Although there are huge growth of services, applications and devices in smart home environment, the interoperability elements still seems ambiguous. Being a distributed architecture, smart home environment needs certain degree of interoperability to manage sub-systems comprising of different platforms. Generally, these sub-systems are developed in isolation and consist of different operating system and tier of services. There is need for a cross-platform interoperability that could make the sub-systems 'talk' each other and operate in an interoperable fashion within smart home environment. Web Services seems to be the emerging technology that could lead the way in providing greater interoperability. In this paper we describe the potential of Web Services technology using Simple Object Access Protocol (SOAP) in addressing the interoperability requirements for smart home environment. The SOAP protocol provides data exchange mechanism as well as optimized performance for interoperation among sub-systems residing in smart home environment. The proposed system performance is evaluated to demonstrate a complete, bi-directional real-time management of sub-systems in smart home environment.

1. Introduction

The evolution of ubiquitous computing, Internet and consumer electronics have stimulated the rapid growth of services and applications in smart home environment. As Mark Weiser [1] argued that "the most profound technologies are those that disappear" seems suits well the smart home environment characteristics as we could feel the blend of intelligence and technology hiding in the backend, providing comfort to home dwellers with many services and applications. The homes are not any more a place where a number of appliances carry on simple executions and tasks but a distributed system with many entities working together. Smart home environment consists of sub-systems which acquire and apply the knowledge about the home dwellers to meet the goal of achieving comfort and efficiency [2]. The sub-systems defined in smart home environment are often heterogeneous in nature and developed in isolation. These heterogeneous sub-systems consist of seven main application domains in smart home environment. Figure 1 below shows the sub-systems defined in smart home environment.

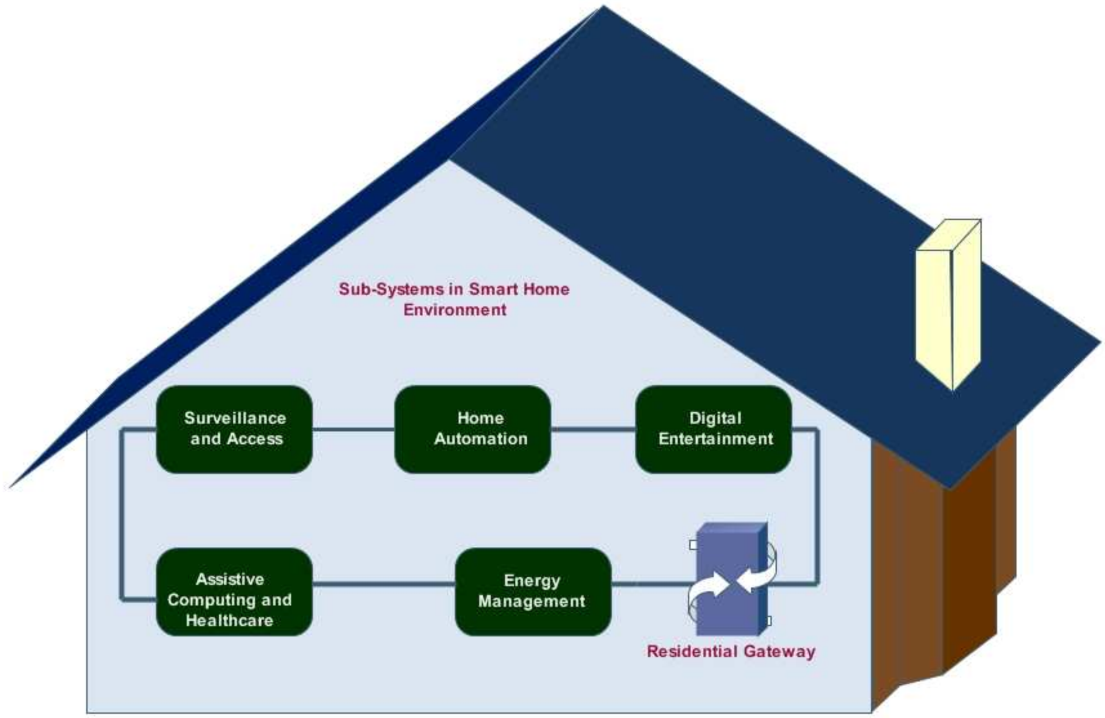

Figure 1: Sub-systems in smart home environment

There are five main sub-systems defined in smart home environment as Figure 1 shows. Those sub-systems are:

a) Surveillance and Access Control

b) Home Automation Systems

c) Digital Entertainment Systems

d) Assistive Computing and Healthcare

e) Energy Management Systems

In smart home, different layers of operation governed by standards, physical media, home networks and middleware equipped with different communication protocols enables automation and management of these sub-systems. Power line technology like X10[3] and Lon Works [4], wireless technologies, fiber optics cable and CAT5 wires are commonly used for audio, video and data communication in smart home environment. On the other hand, middleware such as Jini [5], HAVi [6], and UPnP [7] plays their role in connecting home entertainment sub-systems and in some scenario for appliances connectivity. These physical media, devices and services configured in smart homes are revolutionizing smart homes towards data-intensive environment, resulting in few operational problems. The first problem is that the functionalities of smart homes depend on higher heterogeneity of sub-systems built with different specification and protocols. Managing these different sub-systems has been difficult and residential gateways are needed with respect to the number of systems to be connected. The second complexity is the interoperability factor, due to differences in operating systems, programming language and hardware for those sub-systems. Devices and systems come from different vendors with different network interfaces but still need to interoperate. Interoperability for those systems involves not only by providing system interconnectivity with multiple entities but also in achieving join execution of tasks. The core difficulty in achieving interoperability in smart home environment is due to the isolation of sub-systems and the tendency of system developers deploying those sub-systems without leveraging the requirement for joint execution of tasks. This factor leads to systems developed operating with different architectures and operating systems. These problems are generating great demand for interoperability solution and it is clear that today's challenge for smart home environment is interoperability. On the push to support interoperability, a number of research works are underway with the goal to implement total interoperability. TAHI or know as The Application Home Initiative is working specifically on interoperability issues for the home [8]. A mechanism for supporting joint execution of tasks is needed among sub-systems should be developed for home dwellers to manage those systems efficiently. Interoperability could solve many application and device level integration among sub-systems in smart home scenario. This paper is associated with the design and implementation of interoperability for sub-systems in smart home environment regardless of their level. We also make reference in this paper to the technologies associated with the proposed architecture and show the results of deployment in addressing the interoperability for smart home entities. The rest of this paper is organized as follows. Section 2 discusses the related work and technologies. Section 3 describes the middleware technologies evaluation whereas Section 4 and 5 will address the system and performance evaluation. Finally, conclusion and recommendation for future works are included in Section 6.

2. Related Works

In the context of smart home environment, interoperability means the ability of systems, applications and services to work together reliably in predictable fashion [9]. Interoperability is defined as the ability of two or more systems exchange information and use the information that has been exchanged [10] [11]. Interoperability allows information and resource exchange between defined sub-systems. The most important thing is to ensure these sub-systems interwork seamlessly and provide home dwellers integrated applications without modifying underlying systems platform or protocols. Some generic models of interoperability have been proposed by researchers [12], [13], which focused on syntactic, network and basic connectivity interoperability. These sub-divided tiers are derived from the seven layer OSI model are seems to be the foundation in achieving higher degree of interoperability in smart home environment. Those interoperability tiers for smart home environment are shown in the Figure 2 below:

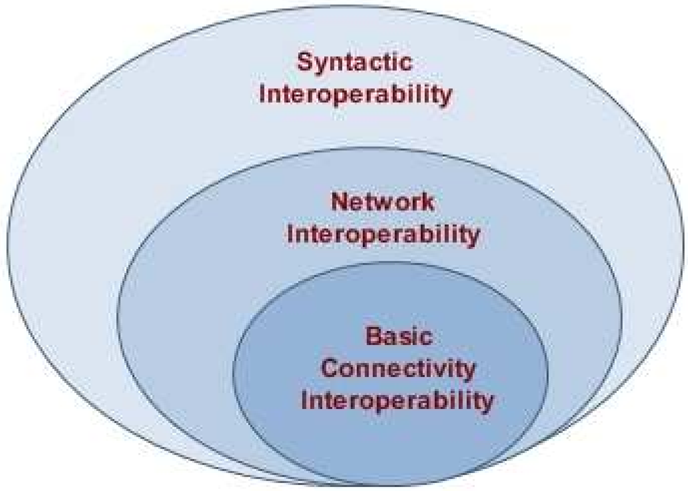

Figure 2: Interoperability Tiers in Smart Home Environment

Each of the tiers describes concerns on interoperability especially during joint tasks execution of two or more sub-systems in smart home environment. The descriptions of these three tiers are presented below:

2.1 Basic Connectivity Interoperability

Basic connectivity interoperability as first tier provides common standard or path for data exchange between two sub-systems and established a communication link. The basic connectivity interoperability can be achieved by common agreement of data transmission medium, low-level data encoding and rules of accessing the medium. Basic connectivity tier is represented by the physical and data link layers of the seven layer OSI model. Ethernet, WI-Fi, and PPP, are examples of common standards for basic connectivity. In smart home environment, sub-systems like home entertainment are interfaced using basic connection to complement with structured residential cabling like CAT5 wire for audio, video and data communications.

2.2 Network Interoperability

The second important tier is the network interoperability. Network interoperability tier enables message exchange between systems across a variety of networks in smart home environment. It defines on agreement of addressing the issues rising from information transfer between heterogeneous systems across multiple communication links. Network interoperability is represented by the network, transport, session and application layers of the OSI model. Examples of common network interoperability standards are Transport Control Protocol (TCP), User Datagram Protocol (UDP), File Transfer Protocol (FTP), Address Resolution Protocol (ARP) and Internet Protocol (IP/IPv6). Many consumer devices for smart home environment are usually IP enabled as they need to connect to various home networks. However, there will be variation of connectivity due to different networks deployed for each sub-system.

2.3 Syntactic interoperability

The final tier, syntactic interoperability refers to agreement of rules that manage the format as well as the structure on encoding information exchanges between sub-systems. Syntactic interoperability refers to ability of two or more components to work together with data and messages passed between them are understood by each component. This final tier provides a mechanism to understand the data structure in messages exchanged between two entities in smart home environment. Syntactic interoperability is represented by the application and presentation layers of the OSI model. Some of the functions provided by this tier are message content structure, such as Simple Object Access Protocol (SOAP) encoding [14], Representational State Transfer (REST)[15] encoding, message exchange patterns such as Asynchronous Publish/Subscribe and translation of one character data from one format to another. Syntactic interoperability is crucial to ensure smooth message transition between heterogeneous sub-systems in smart home environment.

3. Middleware Technologies for Interoperability

There are several middleware technologies and languages that address interoperability issues in smart home environment. These middleware technologies are built to cater the interoperability tiers suiting the smart home requirements. The common selected approaches that drive interoperability in smart home environment are Common Object Request Broker Architecture (CORBA), Microsoft Component Object Model (COM), .NET Framework, Sun's Java 2 Enterprise Edition (J2EE) and World Wide Web Consortium's (W3C) extensible Markup Language (XML) based Web Services. In the following section, we evaluate the interoperability approach using mentioned technologies and selected SOAP and .NET Web Services as our interoperability solution for smart home environment.

3.1 Common Object Request Broker (CORBA)

Common Object Request Broker Architecture (CORBA) is an architectural framework established by The Object Management Group (OMG) [16] as part of standard in Object Management Architecture (OMA). The OMA set of standards consists of Object Services, Object Request Broker (ORB) function, common facilities, application objects and domain interfaces. CORBA is structured to allow integration of wide range of object systems and provides mechanisms to find object implementation for a request, to prepare implementation to receive request and finally communicate with the data that making up the request. CORBA provides a mechanism to define interfaces between components, and specifies standard services such as persistent object services, directory and naming services and transaction services which describes the interoperability feature for CORBA compliant applications. Researchers from University of Texas at Arlington developed a smart home architecture called MavHome [17]. MavHome architecture was built using CORBA interface catering software components and power line control for managing systems in smart home environment. Although CORBA interface could resolve interoperability by providing interoperation feature for managing disparate systems, it also has some drawbacks which may not be ideal for implementation in smart home environment. Modification is needed to enable joint execution of tasks among heterogeneous sub-systems, especially if the systems are not in compliant with CORBA specifications. Any modification of legacy systems in smart home environment could be costly and time consuming. Therefore a framework that will enable interoperability in managing heterogeneous systems without requiring modification to the existing systems is highly needed.

3.2 Component Object Model (COM)

Component Object Model or widely known as COM was introduced by Microsoft which enables applications built from binary components defined by software vendors. COM's successors are Distributed COM (DCOM) and COM+. These technologies are aimed to provide generic mechanism in integrating the components on Windows based platforms. In terms of interoperability, Component Object Model (COM) technologies provide similar features as CORBA. The difference is that COM address interoperability among binary software components while CORBA tackles at the source code level. However, the drawback of COM in providing interoperability is that it requires information of the remote systems before functioning and eventually leads into modifying the legacy systems that are not complied with COM standards. Similar to the CORBA's outcome, modification of legacy systems in smart home is not desirable for developers and home dwellers.

3.3 .NET Framework

.NET Framework was developed by Microsoft to provide a set of standard components and languages which is compliant to ECMA-335 Common Language Infrastructure (CLI) standard. Common Language Infrastructure (CLI) enables code reusability in single or multiple operating system platforms. The main advantage of the .NET Framework is the Common Language Runtime (CLR) mechanism that allows objects used in one language can be used in another language. CLR depends on Microsoft Intermediate Language (MSIL) in producing managed code. All language development tools could produce the same MSIL regardless of the language used to write particular codes. The MSIL code produced by language development tools are then compiled by a Just-in-time (JIT) compiler to produce the actual machine code that executes for specific applications. .NET Framework supports interoperability by providing cross-language platform where classes and objects are interchangeable and reusable without using a specialized Interface Definition Language (IDL). .NET also enables integration of .NET programs and legacy codes. Interfacing legacy codes is supported by enabling applications that are part of managed code environment generated by CLR to access unmanaged Dynamic Link Library (DLL) functions. Legacy codes support are one of the contributing factor in promoting interoperability for smart home environment. .NET Framework could become an ideal solution for smart home technologies especially with its language and platform independence features for application development. Figure 3 shows the potential integration of sub-systems using .NET.

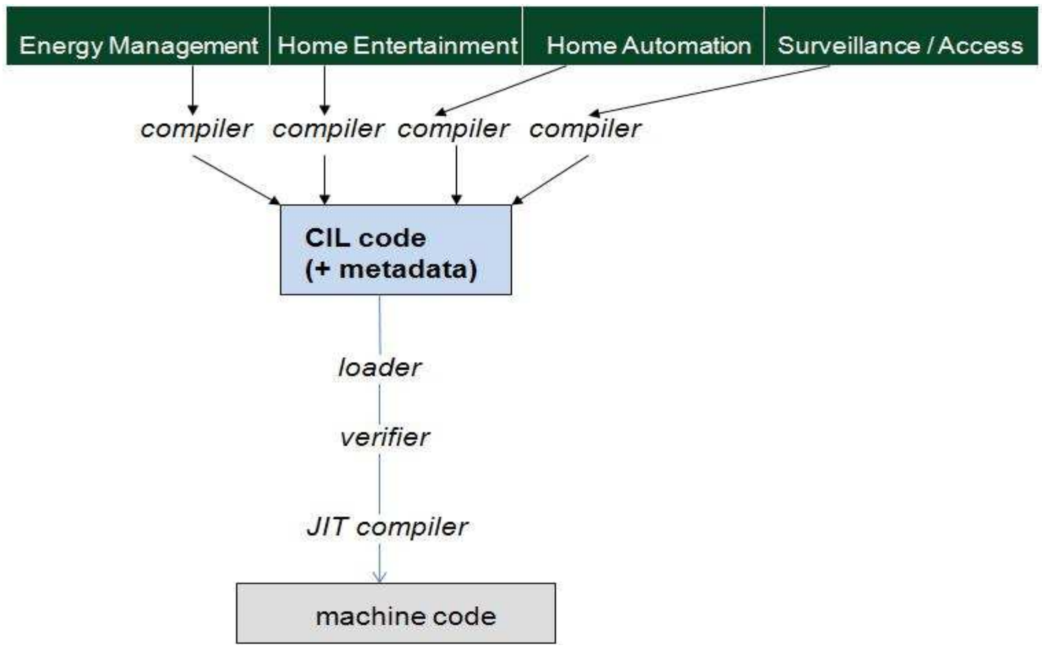

Figure 3: Integration using .NET

Figure 3 shows that several applications of different sub-systems could be developed using different language and compiled into .NET Framework to produce managed code. Having .NET produced code will promote interoperability by eliminating the differences in code implementation. In addition, recent development of Mono Framework, enable porting the .NET features into broad-based interoperability solution by supporting open source based operating systems [18].

3.4 Java Middleware Technologies

Java Middleware technologies support interoperability by providing distributed protocols and APIs that can be used to create an interoperable system. In Java based platform, remote invocation or messaging is the key to achieve interoperability. Java middleware offers Remote Method Invocation (RMI) mechanism that is similar to CORBA-like object oriented middleware layer as distribution protocol. RMI enables objects to be called remotely from other applications in a heterogeneous environment. This feature also extends for interoperation execution between systems and information exchange. One of the implementation of Java Middleware in smart home environment is the OSGi framework [19]. The OSGi Alliance introduced the Open Service Gateway Initiative (OSGI) specification defines a standardized, component oriented, computing environment for networked services.

Work by Rebeca *et.al* [20] focused on service composition using OSGi framework for home environment. Another solution proposed by A.R Al-Ali *et.al* [21] demonstrated the potential of Java Server Pages in managing home appliances over heterogeneous environment. However, all the proposed design requires installation of

Java Virtual Machine (JVM) in the remote systems. Java Middleware presents a competing approach to heterogeneous systems management similar to the one offered by CORBA and COM family. The core advantage of using Java Middleware technologies include its support for interoperability in terms of interoperation execution and information exchange, and full support for modification of existing systems. However, Java Middleware can only be implemented with the presence and requirement of Java Virtual Machine (JVM) in remote and local component of the system involved.

3.5 Web Services

Web Services are collected set of XML based protocols that provides fundamental blocks for creating distributed applications [22]. The functionality of Web Services are based on *publish, discover* and *invoke* that describes standardized concept of function invocation relying on web protocols, independent of any platform (operating system, application server, programming language, database and component model). Web Services consist of three entities [23]:

a) Service Provider – Create Web Services and publish to the external environment by registering through Service Registry
b) Service Registry- Registers and categorizes published services
c) Service Requester - uses Service Registry to find a needed service and bind them accordingly to Service Provider

Figure 4 below shows the three entities of Web Services.

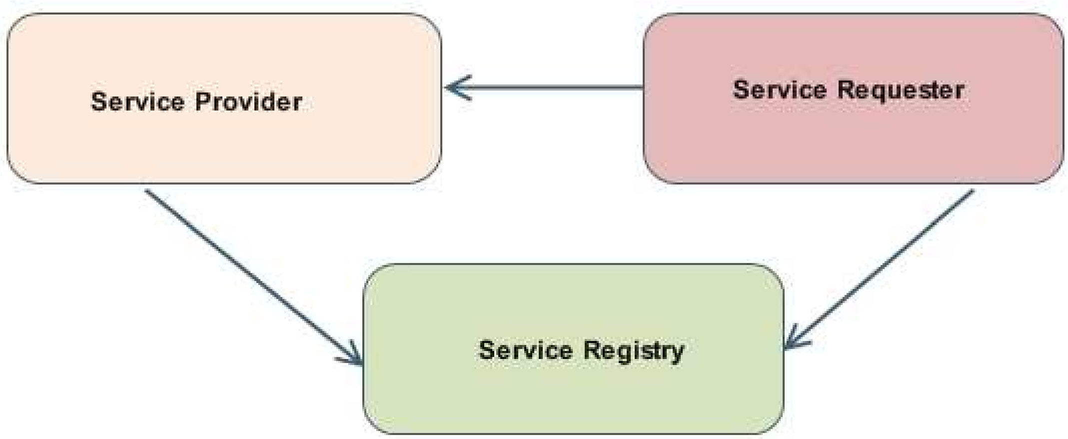

Figure 4: Three entities of Web Services

These entities of Web Services are founded upon three major standard technologies: Simple Object Access Protocol (SOAP), Web Services Description Language (WDSL) and Universal Description Discovery Integration (UDDI). All these standards are based on XML as defined mechanism for data definition, initiated by the World Wide Web Consortium (W3C). Web Services Description Language (WDSL) provides a model along with XML based format in describing the Web services. A WSDL description is done with two levels of stages. One is the abstract stages consist the messages that it sends and receives. On the concrete stage, a binding determines the transport and wire format details for one or more interfaces. Ports or known as *EndPoint* combine the interface bindings information with a network address. Finally a service groups all the Endpoints that implement a common interface. Universal Description Discovery and Integration (UDDI) is the last element needed in providing Web Services implementation. The main goal of UDDI is the specification of a framework for describing and discovering Web Services. UDDI defines data structures and APIs for publishing service descriptions in the registry and querying the registry to look for published descriptions. UDDI is expected to be a service repository of business organization near future towards extending their business information and value added service for smart home environment. In smart home context, Web Services are identified as potential solution for solving interoperability dimension in managing disparate systems. It is also worth highlighting about organization like Open Building Information Exchange Group (OBIX), working towards developing comprehensive standards using XML and Web Services to cater information exchanges between heterogeneous systems in home and buildings [24].

3.6 The Simple Object Access Protocol (SOAP)

Simple Object Access Protocol (SOAP) is an inter-application communication mechanism targeted for exchanging structured information in a distributed environment. SOAP exchanges information using *messages*. In the specification developed by World Wide Web Consortium (W3C), it is also included a method for encapsulating Remote Procedure Calls (RPCs) within SOAP messages.

Ideally, SOAP is created to support loosely-coupled application that could exchange one-way asynchronous messages. SOAP comprises the following elements: an envelope describing the content of the message and the way to process it, a set of encoding rules to express instances, application defined data types and a convention for the representation of remote procedure calls and responses. Figure 5 shows the structure of a SOAP message

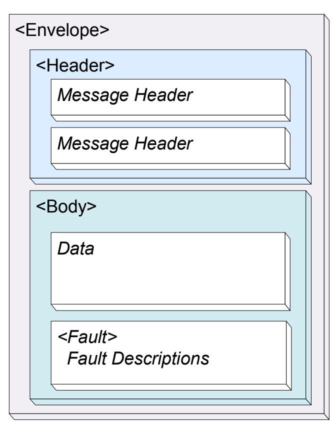

Figure 5: Structure of a SOAP Message

Each envelope consists of a *header* and a *body*. The information intended to be transported will reside in the *body* of the message. Any additional information or value added services will be included in the *header*. SOAP protocol can be used in two different style called document-style and RPC-style. In document style, interaction happens between two applications agreeing upon the structure of documents exchanged among them. While in RPC-style, a SOAP message encapsulate request while another message encapsulate the response. In this paper, we will demonstrate the ability of SOAP in providing generic interoperability mechanism.

4. System Architecture

Sub-systems in smart home environment comprise a number of tasks that are associated with the sequential use of different systems and applications. The need for interoperability in managing these sub-systems has led towards a transition of vendor independence and open systems, taking into account of middleware and Internet technologies. In this paper we propose an architecture that builds upon the general trend towards interoperability for smart home environment. Figure 6 shows the proposed system architecture.

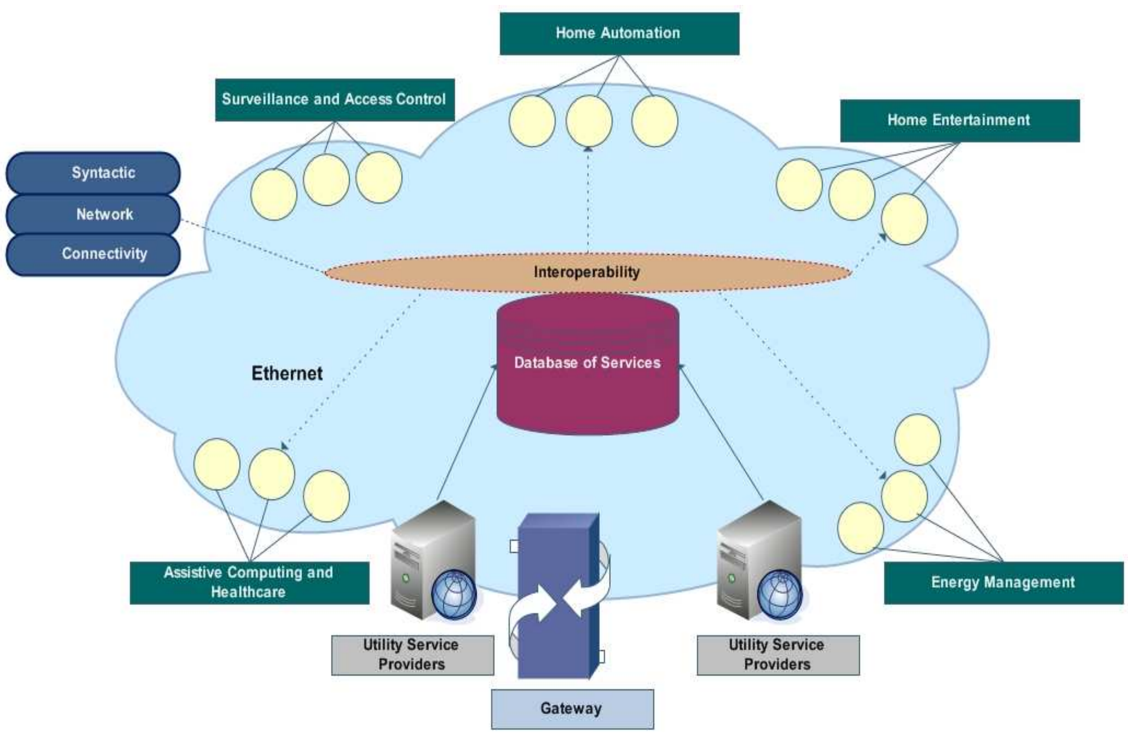

Figure 6: System Architecture

Each tier that defined interoperability has to be fulfilled in order to achieve full interoperability aspect in smart home environment. We propose the utilization of Ethernet cloud as basic connectivity interoperability in smart home environment. Ethernet also seems an ideal solution due to its performance oriented in real-time and also taking into account the existence of Cat 5 cabling based structured wiring in smart home systems. On the aspect of network interoperability, TCP is utilized to perform message exchanges between heterogeneous sub-systems in smart home environment. TCP hides the details of actual interactions between communicating heterogeneous systems from users. The use of TCP here is justified as there is always one distinct approach in the field of smart home environment; to incorporate TCP/IP networking even into simplest consumer devices. Furthermore, the growth of networked based systems is expected to be increased in coming years. On the syntactic interoperability, it is evident that SOAP could be ideal solution as defined structure for message exchanges. We choose XML and SOAP technology as the enabler since both are prime candidates in playing their role as *lingua franca* for interoperability. In our opinion, SOAP is very well suited for providing interoperability among heterogeneous systems due to the standard way of data representation and the format is extensible to deal with changing requirements.

A database of services is configured to handle the queries of SOAP messages of the heterogeneous systems configured in smart home environment. Microsoft SQL Server together with an application gateway is used as an intermediary for the storage and ordering of the messages between systems. An example scenario of interoperation could be explained with operation of two systems. Figure 7 shows the interoperation of heterogeneous systems using SOAP messages

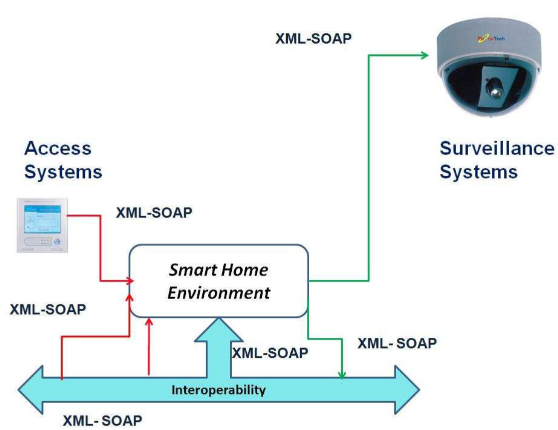

Figure 7: Interoperation of sub-systems

This scenario involves an access control system and surveillance systems that provides security monitoring services for home dwellers. When an intrusion is detected, a triggering signal will be sent to the database of services using SOAP message by the access system. Upon acceptance of the condition, the interoperability tier would need to initialize the associated action by sending commands to other systems. Here, the interoperability tier residing in the core of smart home environment will ensure that message exchange takes place in timely manner. Upon the acceptance of the query, the interoperability tier will send a respond message using SOAP protocol to the surveillance systems or other systems that need to interoperate in order to acknowledge security status to home dwellers. This kind of scenario clearly requires heterogeneous systems to work and integrate together in an interoperable fashion.

5. System Evaluation

In this section, we further elaborate the evaluation of the developed solution and the address significant features of the software engine. Followed by prototype implementation, the performance evaluations are also discussed.

5.1. System Components

Considering the interoperability and scalability features of .NET Framework, C# language together with SOAP service classes are used to operate, manage and control the system operation of sub-systems in smart home environment. The software engine embedded in the application gateway developed using C#. This software engine provides functionality to manage the service requirements using SOAP technology. On the other hand, .NET Framework provides mobility in terms of operation by enabling the use of multiple clients (i.e. mobile phones, desktop PC and personal digital assistants) to manage the systems by taking advantage of one single software engine embedded in application gateway, accessible to home dwellers all the time. The XML-SOAP web services is developed using C# language in managed code using Visual Studio 2005. The implementation of the XML-SOAP Web services resides in a code-behind file and associated with the *.asmx* page using the *Codebehind* attribute, which is part of the .NET Framework. The sub-systems in smart home must be secure in terms of interoperation and reliability while changing states between multiple applications. Realizing the importance of security, the .NET framework itself is defined as a runtime virtual machine where the concept of pointer is not implemented (managed code) on it. Hence, the conventional security attack method such as buffer mal-operation (overflow) would never occur in such case.

5.2 Prototype Implementation

In order to validate, we designed a prototype designed comprising an embedded CPU and application gateway, providing storage to the entire system. The embedded CPU is configured using Windows Server 2003 and Internet Information Service 6.0 environment. Microsoft SQL Server is configured in embedded CPU for services repository. In addition, .NET Framework 2.0 is installed and configured. The embedded CPU supports inputs and outputs of Ethernet connectivity with 4 ports and 1GB of system memory availability. In smart home environment, managing sub-systems means 24/7 nature and systems are expected to access services all the time. Thus, an embedded CPU configured as application gateway will be an ideal platform for continuous operation, less system downtime, server consolidation and increased application availability. Figure 8 below shows the developed prototype to test the interoperability services in smart home environment.

Figure 8: Application Gateway

5.3 Performance Evaluation

On the basis of the interoperability feature, the software system was developed and integrated with two different sub-systems using the developed prototype. Integration and interoperation of the sub-systems are realized using Ethernet connectivity. Performance test was conducted to verify the response time of the developed systems in order to meet the smart home requirement. Response time is one of the important metrics used to evaluate the QoS of Web Services. Response time indicates the maximum time taken by one XML-SOAP message interoperation between the sub-systems without any interrupts or loads. Response time has considerable impact on the performance of SOAP and verification is necessary to ensure its reliability. A total of 200 samples were measured. Figure 9 below depicts the response time comparison during sub-systems interoperation using XML-SOAP messages.

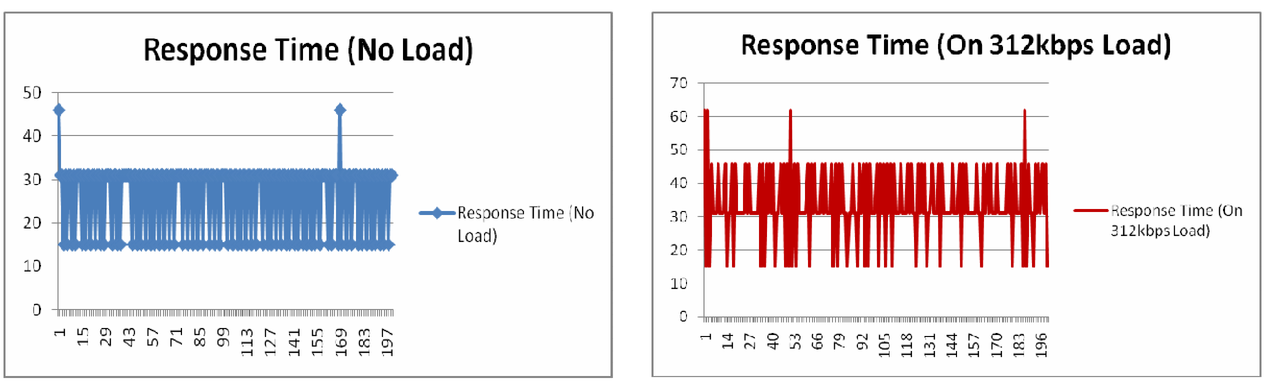

Figure 9: Response time comparison

From the Figure 10 above, the first peak measured at 46ms. The response time of 46ms occurred resulting from compilation time for initial launch of the software engine using Just-In-Time Compiler of .NET Framework. The test result also indicates that an average time of 25.87ms is taken by a single interoperation of the sub-systems. This average time obtained justifies the response time requirement sub-systems interoperation in smart home environment. Standard deviation computation shows consumption time of 7.8ms. For benchmarking purpose, the response time evaluated with load (312kbps) indicates 64ms for first peak. The average response time with load is 33.94ms and with a standard deviation of 11.08ms. The response time calculated above tailored the interoperability requirement of Web Services in smart home and satisfactory for interoperation between sub-systems.

6. Conclusion and Future Works

The work presented in this paper deploys the interoperability requirement for smart home environment namely, the Simple Object Access Protocol (SOAP) with Web Services ability in providing interoperation and scalability for managing sub-systems. In implementing interoperability among sub-systems, data representation must be independent regardless of operating platform. Our work indicates that SOAP protocol maximizes the interoperability and performance of sub-systems. Future research holds a lot of promises especially in extending the interoperability dimension towards semantic and business tiers. A universal schema definition could be defined towards developing a generic abstraction tiers for managing sub-systems in smart home environment.

Acknowledgements

The authors would like to thank all members in the Institute of Advanced Technology for their technical supports. The author also would like to express their highest regards and thanks to Ministry of Science, Technology and Innovation (MOSTI), Malaysia for funding this work under the ScienceFund titled Distributed Embedded System of Multimedia Applications for Ubiquitous Home Entertainment (01-01-04-SF0253).

References

[1] M.Weiser, "The Computer of 21st Century", *Scientific American*, 1991.

[2] K. Tiiu and Kaisa Väänänen-Vainio-Mattila, "Evolution towards smart home environments: empirical evaluation of three user interfaces," *Personal and Ubiquitous Computing.*, vol. 8, no. 3-4, 2004, pp. 234-240.

[3] X10 Technology, http://www.x10.com

[4] LonWorks, http://www.echelon.com

[5] R.Gupta, S.Talwar, and D.P.Agarwal, "Jini Home Networking: A Step toward Pervasive Computing," *Computer*, Vol.35, pp34-40, 2002.

[6] B. Yu-Seok, O.Bong-Jin, M.Kyeong-Deok, and K.Sang-Wook, “Architecture of interoperability of services between an ACAP receiver and home networked devices,” *Consumer Electronics, IEEE Transactions on*, vol.52, pp.123-128.

[7] L.Hyun Yong and K. Jong Won,"An Approach for Content Sharing among UPnP Devices in Different Home Networks," *Consumer Electronics, IEEE Transactions on,* vol. 53, pp. 1419-1426, 2007.

[8] TAHI, The Interoperable Services for Smart Home, www.theapplicationhome.com

[9] Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. New York, NY: 1990

[10] Levels of Information Systems Interoperability (LISI), C4ISR Architecture Working Group,1998

[11] P. Young, N. Chaki, V. Berzins, and Luqi, "Evaluation of middleware architectures in achieving system interoperability," in *Rapid Systems Prototyping, 2003. Proceedings. 14th IEEE International Workshop on*, 2003, pp. 108-116

[12] George, T. Aphrodite, and P. Michael, "Interoperability among Heterogeneous Services," in *Services Computing, 2006. SCC '06. IEEE International Conference on*, 2006, pp. 174-181.

[13] GridWise™ Architecture Council, “Interoperability Context-Setting Framework, March 2008

[14] Simple Object Access Protocol (SOAP), http://www.w3.org/TR/soap/

[15] V. Steve, "REST Eye for the SOA Guy," *Internet Computing, IEEE,* vol. 11, pp. 82-84, 2007

[16] Michi Henning and Steve Vinoski, Advanced CORBA Programming with C++, Addison-Wesley, 1999

[17] D. J. Cook, M. Youngblood, E. O. Heierman, III, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, "MavHome: an agent-based smart home," in *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, 2003, pp. 521-524

[18] Mono-Project, http://www.mono-project.com/Main_Page

[19] OSGi Alliance, http://www.osgi.org

[20] R. P. Diaz Redondo, A. F. Vilas, M. R. Cabrer, J. J. Pazos Arias, and L. Marta Rey, "Enhancing Residential Gateways: OSGi Service Composition," *Consumer Electronics, IEEE Transactions on,* vol. 53, pp. 87-95, 2007.

[21] A. R. Al-Ali and M. Al-Rousan, "Java-based home automation system," *Consumer Electronics, IEEE Transactions on,* vol. 50, pp. 498-504, 2004.

[22] T. Perumal, A. R. Ramli, and C. Y. Leong, "Design and implementation of SOAP-based residential management for smart home systems," *Consumer Electronics, IEEE Transactions on,* vol. 54, pp. 453-459, 2008

[23] Gisutavo Alonso, Fabio Casati, Harumi Kuno and Vijay Machiraju, *Web Services: Concepts, Architectures and Applications*, Springer-Verlag Berlin Heidelberg, 2004.

[24] oBIX (Open Building Information Xchange), http://www.obix.org

Authors

Thinagaran Perumal received his B.Eng. in 2003 at Department of Computer and Communication Systems Engineering, Universiti Putra Malaysia. He completed his M.Sc. degree in Intelligent Systems and currently pursuing Ph.D. in Smart Technology and Robotics at Institute of Advanced Technology, Universiti Putra Malaysia. His main interests are smart home systems, middleware technologies and embedded system design. He is a member of IEEE.

Abd Rahman Ramli received M.Sc. degree in Information Technology System from University of Strathclyde, United Kingdom in 1985 and Ph.D. in Image Processing from University of Bradford, United Kingdom in 1995. He is currently an Associate Professor and Head of Intelligent Systems and Robotics Laboratory in Institute of Advanced Technology, Universiti Putra Malaysia. His main interests are imaging, image processing systems and intelligent systems. He is a member of IEEE.

Chui Yew Leong received his B.Eng. Electronics in 2000 from Universiti Malaysia Sabah and M.Sc. specializing in Computer Engineering in 2003 from Universiti Putra Malaysia He obtained his Ph.D. from same university. His main interests are embedded system design, network security and intelligent systems. Currently he is attached as postdoctoral fellow with Institute of Advanced Technology, Universiti Putra Malaysia. He is a member of IEEE.

Prof. Dr. Shattri Mansor, is the Director of Development, Universiti Putra Malaysia. Dr. Shattri maintains a diverse research interest including image processing, remote sensing and GIS. His major research effort includes feature extraction from satellite imagery, spatial decision support system, fish forecasting, oil spill detection and monitoring system, UAV-based remote imaging, disaster management and early warning system. Dr. Shattri has published over 250 articles in journals and conference proceedings.

Khairulmizam Samsudin received the B.E. degree from University Putra Malaysia in 2001. He received the Ph.D. degree in electrical and electronics engineering from University of Glasgow in 2006. He is a faculty member in the Department of Computer and Communication Systems Engineering and leads the Computer Systems Research Group. His research interest includes distributed operating system, high performance computer architecture, biologically inspired computing and mobile-robot agents