International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 8 Issue 1 –APRIL 2014.

Smart computing using context aware middleware

J.Madhusudanan[1] V.Pavithra[2] R.Preethi[2] P.Tharika[2]

[1] Research Scholar, Department of Computer Science & Engineering, Pondicherry University, puducherry.

[2] UG Students, Department of Computer Science Engineering, Sri Manakula Vinayagar Engineering College, Puducherry.

Abstract: **Pervasive computing is an emerging trend that makes computers physically available but effectively invisible to use. It is mostly used in banks, logistics, hospitals, researches and also in various other fields.**
Most of the research work of pervasive computing deals only for specific application such as smart banks, smart homes etc., the middleware will handle the context and device information together which will lead into inconsistencies and there is a complication for the middleware to handle device and context information.
The inference from the survey reveals that applications are more specific and from this we came to know that there is a need for generic middleware. Thus the proposed system is to develop a generic middleware with intelligent function that could be implemented for various environments. The main objective of the generic context middleware is to provide the best service to the user based on their varying context.

Keywords**:** generic middleware, context middleware, intelligent function, context information.

1. INTRODUCTION

Pervasive computing is a post-desktop model of human-computer interaction in which information processing has been thoroughly integrated into everyday objects and activities. Other names given to this trend are Ubiquitous computing and Ambient Intelligence. The concept was introduced by Weiser in 1991. When primarily concerning the objects involved, it is also called physical computing or the Internet Of Things. A smart environment consists of a set of networked devices that have some connection to the physical world. Unlike smart devices, the devices that comprise a smart environment usually execute a single predefined task, e.g., motion or body heat sensors coupled to a door release and lock control. Embedded environment components can be designed to automatically respond to or to anticipate users' interaction using iHCI (implicit human computer interaction), e.g., a person walks towards a closed door, so the door automatically opens. Hence, smart environments support a bounded, local context of user interaction. Devices are indispensable in building any pervasive application. Pervasive computing environment consists of many independent devices capable of sensing, actuating computing and communicate.

Pervasive computing goes beyond the realm of personal computers: it is the idea that almost any device, from clothing to tools to appliances to cars to homes to the human body to your coffee mug, can be embedded with chips to connect the device to an infinite network of other devices.

The goal of pervasive computing, which combines current network technologies with wireless computing, voice recognition and artificial intelligence, is to create an environment where the connectivity of devices is embedded in such a way that the connectivity is unobtrusive and always available.

The structure of the paper is organized in the following manner. The chapter 2 contains the detailed description about Smart Environment. Chapter 3 deals with pervasive Computing and its detailed description. The chapter 4 contains about context awareness in smart computing. Chapter 5 contains existing system and its issues. Finally, chapter 6 contains our proposed system named as Smart Computing using context aware middleware.

2. SMART ENVIRONMENT

In a smart environment, computation is seamlessly used to enhance ordinary activities. Cook and Das [1] refer to a smart environment as 'one that is able to acquire and apply knowledge about the environment and its inhabitants in order to improve their experience in that environment'. A smart environment consists of a set of networked devices that have some connection to the physical world. Unlike smart devices, the devices that comprise a smart environment usually execute a single predefined task, e.g., motion or body heat sensors coupled to a door release and lock control. Embedded environment components can be designed to automatically respond to or to anticipate users' interaction using iHCI (implicit human computer interaction), e.g., a person walks towards a closed door, so the door automatically opens. Hence, smart environments support a bounded, local context of user interaction.

Smart environment devices may also be fixed in the physical world at a location or mobile, e.g., air born. Smart environments could necessitate novel and revolutionary upgrades to be incorporated into the environment in order to support less obtrusive interaction, e.g., pressure sensors can be incorporated into surfaces to detect when people sit down or

walk. A more evolutionary approach could impart minimal modifications to the environment through embedding devices such as surface mounted wireless sensor devices, cameras and microphones.

Smart environment is a term that is not very commonly used in research community. However, some definitions do exist. In most of them the key issue is that smart environment is a physical space full of devices and services that function autonomously on behalf of an individual user by providing an ambient and continuous context and reacting and responding to ongoing activities. Ubiquitous and pervasive computing, ambient intelligence or Internet of Things are basically used to describe the same idea that is also behind the Smart Environments.

The core features and attributes that are present in most Smart Environment definitions are following:

1. Smart Environment is a physical space consisting of various embedded systems and electronic devices that are interconnected by wired or wireless techniques. Together these devices integrate the information of the physical world and form a digital virtual space. This physical space orientation separates Smart Environments from pure web-based environments where the actual location of the service is usually irrelevant.

2. Smart Environment should have means for perceiving, measuring and storing the context information of the entities in the environment. Context is defined as any information that can be used to characterize the state and situation of an entity. The context information gathered by an individual device should be made available everywhere in the Smart Environment without human assistance.

3. The applications and devices in the environment should have means to make smart actions based on the information available in the space. The actions to be performed are only limited by one's imagination.

3. PERVASIVE COMPUTING

Pervasive computing is a rapidly developing area of Information and Communications Technology (ICT), including distributed computing, mobile computing, sensor networks, human –computer interactions and artificial intelligence. Pervasive computing has many potential applications, from health and home care to environmental monitoring and intelligent transport systems. Increases integration of ICT into people's lives and environments, made possible by the growing availability of microprocessors with inbuilt communications facilities. It provides services accessible via natural interaction. There are "Context-aware" smart appliances and smart spaces. Pervasive computing builds a "virtual world" out of the "physical world". It senses their physical environment, and adapts their behavior accordingly.

4. CONTEXT AWARENESS IN SMART COMPUTING

In order to make smart environment to react autonomously to the context, context awareness plays an important role. By sensing the situation of the users and environments, the contexts are generated. Without an explicit user intervention the informations are gathered from the environment based on the context understood by the applications and this idea is referred as context awareness.

An entity is a place, person or object and any information that can be used to characterize the situation of an entity is known as context .The aspects of the context are (1) where you are;(2) who are you with;(3) what resources are nearby configuring sensors, devices ,user and associating set of rules among them.

5. EXISTING SYSTEM AND ITS ISSUES

The Context middleware allows agents to acquire contextual information easily, reason about it using different logics and then adapt themselves to changing contexts. Most of the research work of context middleware handles only a specific application. The Middleware designed is to handle context and devices in the same layer this leads to complications like inconsistency. There is no generic context middleware that suits all types of application. In existing system using ontology based model for representing context there exist inconsistency of data.

The specific applications that exist so far works with the device and context layer put together. So there occurs some inconsistency in the context data that has to be passed on correctly to the respective device. Some of the issues are as follows:

Adaptability: Ability of a software entity to adapt to the changing environment, changes in applications and user's request may require the presence of adaptation mechanism within the middleware

Heterogeneity: Able to operate across different homogeneous environment, seamless integration of devices and environment, taking on new contexts when a new resource becomes available.

Inconsistent data: missing of available data that is needed to perform the operation.

Scalability: Ability to cope up and perform under an increased or expanding workload. A system that scales well will be able to maintain or even increase its level of performance or efficiency when tested by larger operational demands.

6. PROPOSED SYSTEM

Our system provides a generic middleware to support various applications. IN which the middleware splits the scenario as context information. The Xml based language support is provided to the middleware. Along with the generic middleware intelligent function is included. If the previously given scenario is been used frequently, then those scenarios can be identified and the already generated Xml file can be reused instead of starting the process from the beginning.

Inorder to include intelligent function we have to maintain a history of scenarios that are being used.

The proposed system also involves another important property, priority. The system has priority for each and every devices present in the environment. So that if any two devices has been given instructions to be performed at the sametime, the system uses priority checker and executes the instructions of the devices as per the priority. And also the system can perform multiple scenarios at the sametime whereas in existing system only one scenario can be performed at a time.

When the scenario is given as input, the context from the scenario is identified and the corresponding Xml file for the scenario is generated.

ARCHITECTURE DIAGRAM

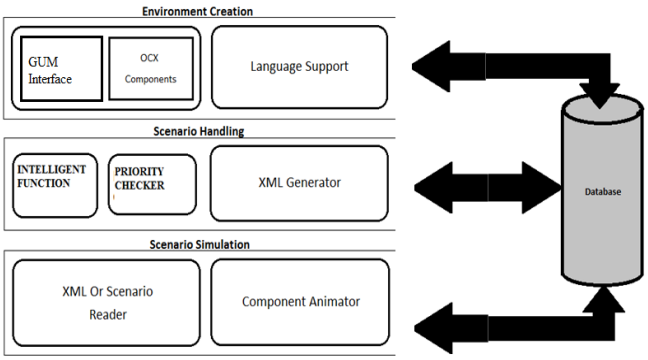

Fig. 1. System architecture

Gum interface: The interface is nothing but a boundary used to communicate and thus gum interface is a boundary for various environments to communicate in a single system.
OCX components: The OCX stands for OLE Control Extensions and OLE refers to Object Linking and Embedding which means that interchange of data or objects between various environments.
Language Support: The system has XML based language support. The language XML is chosen because of its simplicity, extensibility and openness. Thus in the system the contexts are generated as XML file.
Intelligent function: This function identifies the frequently used scenarios and allows to reuse the already generated XML file instead of beginning the process as new.
Priority checker: This is used to check the priority of devices , locations etc., so that instructions to the device are performed based on priority.
Xml generator: The XML generator generates the XML file for the corresponding context information from the scenario.
XML or Scenario reader: The generated XML file is given to the XML or scenario reader which reads the XML file and gives the instructions to the device for which it is meant.
Component animator: This animator is used to animate the components in the environment according to the contexts information from the scenario.

Database: The database stores the scenario and XML file name that are generated. And it also saves the devices and their priority, locations and their priority.

BLOCK DIAGRAM:

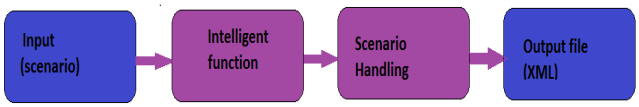

Fig. 2. Block diagram of the system

When the scenario is given to the system, it checks whether the scenario is been already used or not and if it is an existing one then reuses the XML file otherwise, begins the process. The scenario is handled by splitting it into context information and finally XML file for the context information is generated as output.

PSEUDO CODE:

Procedure getContext (Context c_i)

{

For i -> 1 to n

Parse (C_i)

}

Procedure Parse(Context C_i)

{

Device d

Location l

Action a

Time t

Environment E

Status s

Split the context C_i into (C_id, C_il, C_ia,C_it, C_iu,C_is)

Check if C_id € E

then C_id is valid

else

C_id is not valid

Add C_id

Check if C_il € E

then C_il is valid

else

C_il is not valid

Add C_il
Check if C_ia € E
then C_ia is valid
else
C_ia is not valid
Check if C_it = system time
Then
C_it is valid
else
C_it is not valid
Check if C_is = ON or OFF
Then
C_is is valid
else
C_is is not valid
Check if C_iu € E
Then
C_iu is valid
else
C_iu is invalid
Set P(C_id) from the priority list
Set P(C_il) from the priority list
Set P(C_ia) from the priority list
Set P(C_iu) from the priority list
}
Procedure XML(Context C_i)
{
Scenario sc
Environment et
Device dt
Action at
User ut
Time tt
Location lt
Status st
Start sc
Start et
C_iE
End et
For i -> 1 to no of scenario
Start sc i
Start ut
C_iu
End ut
Start at
C_ia
End at
Start lt
C_il
End lt
Start tt
C_it
End tt
Start dt
C_id
End dt
Start st
C_it
End st
End sc i
End sc
}

7. CONCLUSION AND FUTURE ENHANCEMENT

7.1 CONCLUSION

The Generic Context middleware allows agents to acquire contextual information easily, reason about it using different logics and then adapt themselves to changing contexts. Context Middleware handles scenario and context. The contexts and scenarios are modeled using Markup scheme model. Using XML based language the scenarios' and their context are represented. The XML based context representation has several advantages such as it is a universal standard and it is widely used. XML gives the freedom to define your own tags that fit your application needs. XML can also be stored in databases in XML format and human readable format; it is a platform independent language. While data is being exchanged using XML, there will be no loss of data thereby we can achieve consistent data.

7.2 FUTURE ENHANCEMENT

The main advantage of this project is that the Generic Context middleware is independent and adaptable to different applications. The Generic Context middleware developed for Context-aware application likes Smart Home, Smart Banking, Smart Hospital, Smart office and so on. The implementation phase of generic context middleware are handled with different applications like smart Home and smart Bank and taken as a sample application to show how our generic context middleware will support for different applications by handling all possible conflicts. Likewise the generic middleware that is created could be deployed in any number of applications without any changes to be done.

8. REFERENCES

[1] Cook and Das, "Smart Environments: Technology, Protocols, and Applications", John Wiley, ISBN 0-471-54448-5, 2005

[2] Poslad, Stefan, "Ubiquitous Computing Smart Devices, Smart Environments and Smart Interaction", John Wiley, ISBN 978-0-470-03560-3, 2009

[3] Dey and Abowd, "Towards a better understanding of context and context awareness," The What, Who, Where, When, Why and How of Context-Awareness Workshop at the Conference on Human Factors in Computing Systems (CHI), 2000

[4] Kaltz, J.W Ziegler, J. Lohmann, "Context-aware Web Engineering: Modeling and Applications", RIA-Revue d'Intellegence Artificielle,Special Issue on Applying Context Management 19(2005) 439-458

[5] Thyagaraju.GS, Umakanth P.Kulkarni, "Design and Implementation of Context Program Compiler for developing Context Aware Applications in Pervasive Computing Environment", International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-1, Issue-4, September 2011

[6] Tam Van Nguyen, Huy Anh Nguyen and Deokjai Choi , "Development of a Context Aware Virtual Smart Home Simulator", International Conference on Ubiquitous Information Technologies & Applications(ICUT'08),2007

[7] Kristian Ellebaek Kjaer, "A Survey of Context-Aware Middleware", 2009

[8] Thomas Strang ,Claudia Linnhoff -Popien, "A Context Modeling Survey", 2007

[9] Mikko Perttunen, Jukka Riekki, Ora Lassila, "Context Representation and Reasoning in Pervasive Computing: a Review, International Journal of Multimedia and Ubiquitous Engineering,Vol. 4, No. 4, October, 2009

[10] Jens H. Jahnke, Yury Bychkov, David Dahlem, Luay Kawasme,"Implicit, Context-Aware Computing for Health Care", 2004

[11] Daniel Minder, Pedro José Marrón, Andreas Lachenmann, Kurt Rothermel, "Experimental construction of a meeting model for smart office environments", Proceedings of the First Workshop on Real-World Wireless Sensor Networks (REALWSN 2005)

[12] Noha Ibrahim, "A Survey on Service Composition Middleware in Pervasive Environments", IJCSI International Journal of Computer Science Issues, Vol. 1, 2009

[13] Zahra Forootan Jahromi, Amir Rajabzadeh, Ali Reza Manashty, "A Multi-Purpose Scenario-based Simulator for Smart House Environments", International Journal of Computer Science and Information Security(IJCSIS),Vol.9,No.1,January 2011

[14] Soma Bandyopadhyay, Munmun Sengupta, Souvik Maiti, and Subhajit Dutta, "A Survey of Middleware for Internet of Things",Recent Trends in Wireless and Mobile Networks,Vol.162(2011),pp.288-296,Springer-Verlag Berlin Heidelberg 2011

[15] Di Zheng, Hang Yan, Jun Wang, "Research of the Middleware based Quality Management for Context-aware Pervasive Applications", International Conference on Computer and Management (CAMAN), page(s):1-4, ISBN: 978-1-4244-9282-4,May 2011

[16] Rosslin John Robles and Tai-hoon Kim, "Review: Context Aware Tools for Smart Home Development",Vol.4, No.1, January 2010

[17] Moeiz Miraoui, Chakib Tadj and Chokri ben Amar,"Context Modeling and Context-Aware Service Adaptation for Pervasive Computing Systems", International Journal of Computer and Information Science and Engineering (IJCISE),Vol.2,Issue:39,Pages:567-576,ISBN:3854032072,2008

[18] Ahmet Soylu, Patrick De Causmaecker, Piet Desmet, "Context and Adaptivity in Pervasive Computing Environments: Links with Software Engineering and Ontological Engineering", Journal of Software, Vol.4, 992-1013, Nov 2009