SCIREA Journal of Computer Science and Technology

http://www.scirea.org/journal/Computer

December 21, 2016

Volume 1, Issue 2, December 2016

Basic Design of Smart Object Communication in Agent based Approach

V SAI MANOJ[1], M BALAJEE[2]

[1]B.Tech, 3rd year Student, GMR Institute of Technology, Rajam

manojvissavajhula@gmail.com

[2]Sr. Asst. Prof., Dept. of CSE, GMRIT, Rajam, INDIA.

balajee.journal@outlook.com

ABSTRACT

In this paper I discuss about a basic design of communication of smart objects. There are many designs for providing medium for communication of smart objects. It includes several components on background. But for both reliability and security we have to choose the best architecture for the design of communication of smart objects. Mostly Agent based architecture provides all the requirements and has most advantages. The basic implementation and design of smart objects through agent based architecture is the integration of three components: Agent, Cloud and Middleware. Agent-based computing can support development of decentralized, dynamic, cooperating and open IoT systems in terms of multi-agent systems. Cloud computing can enhance the IoT objects with high performance computing capabilities and huge storage resources. In general, middleware can ease a development process by integrating heterogeneous computing and communications devices, and supporting interoperability within the diverse applications and services.

Keywords: Smart, Agent, Framework, Communication, IoT.

INTRODUCTION

Smart objects are nothing but the devices which can communicate with each other or with the cloud service which forms the internet of things. The Internet of Things (IoT) represents a world-wide network of heterogeneous cyber-physical objects such as sensors, actuators, smart devices, smart objects, RFID, embedded computers. These objects, which have identities, physical attributes, and communication interface for service provision, are uniquely addressable and based on standard communication protocols. By enabling easy access of, and interaction with, a wide variety of physical devices or things such as, home appliances, surveillance cameras, monitoring sensors, actuators, displays, vehicles, machines and so on, the IoT will foster the development of applications in many different domains, such as home automation, industrial automation, medical aids, mobile healthcare, elderly assistance, intelligent energy management and smart grids, automotive, traffic management, and many others.

Middleware provides a set of fundamental mechanisms for SO naming, discovery, high-level interaction, and management. The existing middleware solutions are : Event-based, Service-oriented, VM-based, Agent-based, Tuple-spaces, Database-oriented, Application-specific .In general, service-oriented, agent-based, and VM-based design approaches address more IoT requirements than others. The agent-based design approach is good at resource and code management because of its mobile and distributed nature.

The development of IoT smart objects require suitable models, techniques and technology. For this purpose, there are two complimentary mainstream paradigms for large-scale distributed computing : (i)Agent-Based Computing and (ii)Cloud Computing Paradigms.

Literature Survey

In different types of communication like device to device, device to cloud or other, the framework used and middleware plays an important role. There are many middleware solutions available. But out of all those, it has to be chosen cleverly depending upon the requirements. The main middleware solutions are Event-based, Service-oriented, VM-based, Agent-based, Tuple-spaces, Database-oriented, Application-specific middleware.

There are many factors to consider while judging which middleware solution is best. All those can be framed into two types: 1) Service requirements and 2) Architectural requirements.

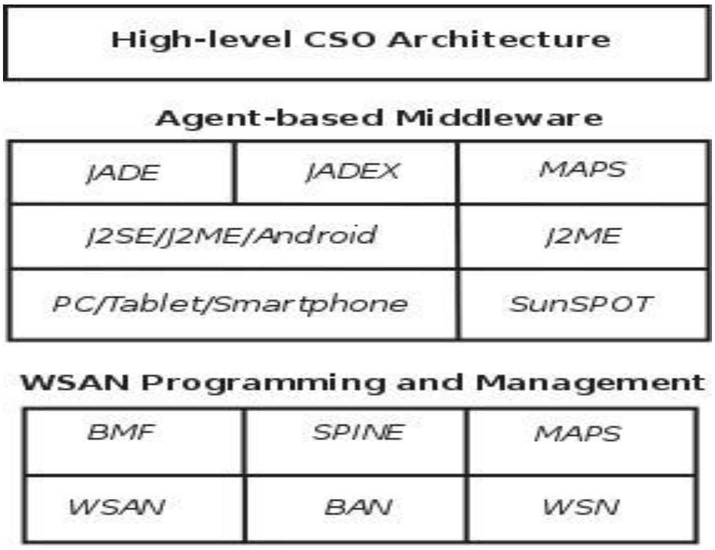
Middleware service requirements for the IoT can be categorized as both functional and nonfunctional. Functional requirements capture the services or functions (e.g., abstractions, resource management) a middleware provides and nonfunctional requirements (e.g., reliability, security, and availability) capture QoS support or performance issues. These include Resource discovery, Resource management, Data management, Event management, Code management, Scalability, Real time or Timelessness, Reliability, Availability, Security and Privacy, Popularity.

The architectural requirements included in this section are designed to support application developers. They include requirements for programming abstractions, and other implementation-level concerns. That include Program abstraction, Interoperable, Service based, Adaptive, Context-aware, Distributed.

A few of them are also indirectly linked (black text) to one or more characteristics of the IoT. For instance, the real time behavior requirement is directly related to the application's real-time characteristics and indirectly to the large number of events. Also, a few of the middleware requirements (e.g., resource discovery and resource management) jointly capture the same set of IoT characteristics.

Agent-based Co-Operating Smart Objects (ACOSO) is a middleware providing an agent-oriented programming model for Co-Operating Smart Objects (CSO) and tools for their effective development. The Agent based middleware are modeled as agents that can cooperate with each other and with non-agent cyber-physical entities to fulfill specific goals. Agent based Co-Operating Smart objects currently rely on JADE, which is a software framework fully implemented on JAVA. The main reason to use JADE is it can be distributed across machines (which not even need to share the same OS) and the configuration can be controlled via a remote GUI.

The High-level CSO Architecture layer, which is the abstract layer formalizing CSOs. The Agent-based Middleware layer, which implements the High-level CSO Architecture through a specific agent framework. The Wireless Sensor and Actuator Networks (WSAN) Programming and Management layer programs and manages the network of sensors and actuators embedded in a CSO.



High-level CSO Architecture

Agent-based Middleware

JADE	JADEX	MAPS
J2SE/J2ME/Android		J2ME
PC/Tablet/Smartphone		SunSPOT

WSAN Programming and Management

BMF	SPINE	MAPS
WSAN	BAN	WSN

Main Components of the CSO architecture: -

- The **Task Management Subsystem** manages the reactive and proactive tasks of CSOs.

- Tasks are event-driven state-based software components encapsulating specific objectives like computation , Storage etc.

- In JADE-based CSOs, tasks are defined as JADE behaviours so their execution is based on the mechanisms provided by the basic JADE behavioral execution model.

- The **Communication Management Subsystem** provides a common interface for CSO communications.

- In JADE-based CSOs, the Communication Manager, which is implemented as behaviour in JADE, captures the ACL messages targeting CSOs and translates them into internal events.

- The **Device Management Subsystem** manages the sensing/actuation devices that belong to the CSO. It is organized in a Device Manager and handles several Device Adapters.

- In JADE-based CSOs, two Device Adapters are currently defined: the BMF Adapter, which allows to manage WSANs based on Building Management Framework, and the SPINE Adapter, which allows to manage BANs through SPINE

- The **Knowledge Base Management Subsystem** supports CSOs through a knowledge bas e (KB).

- It consists of a KB Manager, which manages and coordinates different KB Adapters, which manages a Knowledge Base containing the knowledge of the CSO.

- A Knowledge Base can be local or remote information that can be shared among tasks.

LATEST TECHNOLOGY:

Smart agent enhancement :

While the basic smart agent layer is fully supported by ACOSO, the Cloud platform has to provide new functionalities to dynamically create new virtual Cloud-based agents than run on the Cloud-side but are seamlessly linked to the basic smart agents.

Smart object management :

Multitude of smart objects has to be effectively and efficiently managed through mechanisms that scale from localized highly dense to large-scale decentralized CSO systems.

Smart object data stream collection and management :

Data streams coming from highly decentralized smart objects need to be efficiently uploaded onto the Cloud-side and here effectively managed.

The Smart User Agent:

It models human users in the context of smart systems. They therefore provide GUI-based functionalities through which users can formalize and submit service requests.

The Smart Interface Agent:

It defines an interfacing agent such as brokers, mediators, wrappers. Specifically, they are able to coordinate smart object agents and/or wrap components of external IT systems.

The Smart Object Agent:

It formalizes a CSO through a specific agent model embodying hardware and software components.

Cyber-Physical Environment:

It refers to the non-agent-oriented logical and physical context (made up of logical and physical components) in which agents are embedded. It can be modeled in terms of a reactive/proactive environment abstraction that is able to interact with agents according to an event-driven coordination model.

Cloud Computing Platform:

It supports all smart agents, empowering their specific resources. In particular, it allows for the definition of new smart object agents as meta-aggregation of existing smart object agents.

CONCLUSION:

So we have provided the basic design of a IoT architecture of agent based co-operating smart objects. Smart objects could be empowered in terms of processing power and storage resources through the use of cloud computing. Thus, cloud assisted and agent-oriented smart objects could be used as basis for the development of large-scale IoT services and systems. Although the existing middleware solutions address many requirements, some requirements like resource discovery, reliability, security and privacy, interoperability etc. are unexplored. So by comparing all these things, Agent based is effective in all means.

REFERENCES

[1] Giancarlo Fortino, Antonio Guerrieri, Wilma Russo, Claudio Savaglio, "Integration of Agent-based and Cloud Computing for the Smart Objects-oriented IoT ", 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design.

[2] Mohammad Abdur Razzaque, Member, IEEE, Marija Milojevic-Jevric, Andrei Palade, and Siobhán Clarke, "Middleware for Internet of Things: A Survey" , IEEE Internet Of Things Journal, VOL. 3, NO. 1, February 2016.

[3] Chengen Wang, Zhuming Bi, Li Da Xu, "IoT and Cloud Computing in Automation of Assembly Modeling Systems", IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, VOL. 10, NO. 2, MAY 2014

[4] http://arxiv.org/ftp/arxiv/papers/1101/1101.2577.pdf