ISSN 1870-4069

Architecture of Mobile Services Applied to the Internet of Things

Chadwick Carreto[1], Francisco Cerda[1], Felipe Menchaca[2]

[1] National Polytechnic Institute, School of Computer Science, Mexico
[2] National Polytechnic Institute, School of Mechanical and Electrical Engineering, Mexico
ccarretoa@ipn.mx, fco.cerda.mtz@gmail.com, fmenchac@gmail.com

Abstract. In this paper, we propose a service architecture that integrates an information exchange architecture with the objective of standardizing the communication between different networks of nodes that implement different protocols such as IEEE 802.15.4, Zigbee, DigiMesh, Thread, This is achieved by means of a correct management of the proposed frame for its interconnection, as well as attending different aspects of the communication and establishment of links, this for networks of nodes that allow the handling of information for applications and services applied to the Internet of Things (IoT). This architecture proposes to implement within a system of unification of various medical and academic services.

Keywords: internet of things (IoT), interconnection, exchange, standardization, architecture.

1 Introduction

With the arrival of IoT (Internet of Things) comes the technology of sensor networks which brings us a new problem, one where information and communication systems surround our personal and professional areas which brings with it the generation of enormous amounts of Information and the necessity to have a place to store the information, as well as one to present it, to exploit it and to treat it to an easily interpretable form.

In order for the Internet of things vision to work correctly, the paradigm of computing must go beyond the traditional approaches of mobile computing, in this paradigm we recognize that user is not the only user of a systems, and user generates more information that he believe (pervasive systems).

The IoT requires three primary factors which are desirable in most cases. A shared understanding of the situation of users and their devices. Perceptive software architectures and communication networks, allowing not only to transmit the usual information but also to include contextual information where necessary in order to provide better services.

The existence of analytical tools in order to achieve an autonomous and intelligent behavior of the network, as well as the services provided [1]. In addition to this, there is a need to employ some type of architecture and define a set of protocols and formats of exchange in order to achieve or address the points mentioned above. At present, there

is a set of architectures that allow the development of state-of-the-art technologies, some architectures are used in a particular way for specific systems [2].

There is also the possibility of using hybrid architecture to enhance the best aspects of each one. But this brings with it the problem of standardizing these structures to guarantee intercommunication and interconnection between them.

In the last decade, advances in Information and Communication Technologies (ICTs) have allowed the optimization of services that are used in fields such as industry, education, medicine, environmental protection, etc. Such is the case that in recent years it has been decided to make use of ICTs to automate the use of clinical records and thereby reduce the waiting time for patient care and improve the control of information This is why it is necessary to use an architecture that allows the integration and optimization of various medical services so that they can be used by the right-holders, patients, doctors, researchers, administrators, managers, etc. which are part of the health sector [3].

The following sections describe the existing Services architectures and the proposal to be implemented to test their characteristics.

2 Communication Protocols

The architecture is defined as the structure that defines which components must possess a system as well as the relationships between each one of them. The design of a software architecture represents one of the most critical decisions for the correct functioning of a system, since this architecture will define many of the characteristics that a system can achieve [2].

IEEE 802.4.15 Protocol. IEEE Std 802.15.4-2003 defined the protocol and the compatible interconnection for data communication devices that use low-power, low-power, low-complexity short-range radio-frequency (RF) transmissions in a wireless personal area network (WPAN). This revision expands the applicability of the IEEE Std 802.15.4 market, eliminates ambiguities in the standard, and improves IEEE Std implementations 802.15.4-2003

IEEE 802.15.4 was designed to operate in unlicensed radio frequency bands. (Normally, RF output envelope rules and possibly the duty cycle of a device operating in these bands are still applied). The unlicensed RF, bands are not the same in all territories of the world, but IEEE 802.15.4 employs three possible bands, at least one of which should be available in a given territory. The three bands focus on the following frequencies: 868, 915 and 2400 MHz. [4]

ZigBee. It is a wireless communications standard designed by the ZigBee Alliance. It is a standard set of solutions that can be implemented by any manufacturer. ZigBee is based on the IEEE 802.15.4 standard for wireless personal area networks (wireless personal area Newark, WPAN) and targets applications that require secure communications with low data rate and maximize the life of your batteries [5].

The ZigBee protocol defines three types of nodes: Coordinators, Routers and Final Device, with a requirement of a Network Coordinator. Although all nodes can send and receive data, there are differences in the specific functions performed by each of the different types.

Coordinating nodes are the most capable of the three types of nodes, there is a coordinator per network and it establishes the origin of the network is able to store and/or manage network information, including security information.

Nodes Coordinators.- act as intermediaries between nodes, and transfer data from other devices.

Final Device.- Those devices that have sufficient capacity to speak with higher hierarchical nodes (roots) and cannot communicate with other devices, so as to reduce cost are seen with diminished functionality [6].different rates of data. [4]

Digi Mesh. Digi Mesh is a protocol that only uses a node type, which generates a homogeneous network, this means that all nodes can route information and are interchangeable since they do not have a relation to their root nodes. DigiMesh is a proprietary networking topology for use in wireless endpoint connectivity solutions. It supports advanced networking features, including sleeping routers and dense mesh networks. DigiMesh supports multiple network topologies such as point-to-point, point-to-multipoint and mesh networks. With support for sleeping routers, DigiMesh is ideal for power-sensitive applications that rely on batteries or energy-harvesting technology [6].

Thread. It is an open standard for reliable, cost-effective and low power D2D (device to device) communication. It is specifically designed for connected home applications where IP-based networking is desired and a variety of application layers can be used in the stack. Here are the general characteristics of the stack of wires and the network:

- Simple network setup, commissioning, and operation: Simple protocols for forming, joining, and maintaining wire networks allow systems to autoconfigure and correct routing problems as they occur.
- Secure: Devices do not join the subnetting network unless they are authorized and all communications are encrypted and secure.
- Small and large networks: Home networks vary from several devices to hundreds of devices that communicate transparently. The network layer is designed to optimize network operation based on expected usage.
- Range: Typical devices along with mesh networks provide enough range to cover a normal home. Extended spectrum technology is used in the physical layer to provide good immunity to interference.
- There is no single point of failure: the battery is designed to provide safe and reliable operations even with the failure or loss of individual devices.
- Low power consumption: Host devices can normally operate for several years on AA-size batteries using suitable duty cycles [7].

3 Middleware

Middleware is software that sits between an operating system and the applications running on it. Basically, it works as a hidden translation layer to allow communication and data management in distributed applications can be referred to as “plumbing”, as it connects two applications so that data and databases can be easily passed through one “canalization”. The use of middleware allows users to make requests such as submitting

forms in a web browser or allowing a web server to return dynamic web pages based on a user's profile [8].

A Middleware can be seen as a set of reusable, expandable services and functions that are commonly used by many applications to function well within an interconnected environment [9].

Some common examples of middleware are database middleware, application server middleware, message oriented middleware, web middleware, and transaction processing monitors. Each program typically provides messaging services so that different applications can communicate using messaging frameworks such as Simple Object Access Protocol (SOAP), Web Services, Representational State Transfer (REST), and JavaScript Object Notation (JSON). While all middleware performs communication functions, the type a company chooses depends on the service being used and the type of information to be communicated. This can be security authentication, transaction management, message queues, application servers, web servers, and directories. The middleware can also be used for distributed processing with actions that occur in real time [8].

4 Services Architectures

The Architecture is defined as the structure that defines which components must possess a system as well as the relationships between each one of them. The design of a software architecture represents one of the most critical decisions for the correct functioning of a system, since this architecture will define many of the characteristics that a system can achieve [2].

Service Oriented Architecture. Mental elements for the development of solutions, a service will be described as a computational element agnostic to the platform that supports a fast and low cost, that make up distributed applications. Services perform functions, ranging from simple requests to complex business processes. Services allow organizations to expose their core competencies programmatically over the Internet (or intranet) using standard languages and protocols (based on XML, JSON or some other exchange format), and implemented through a self-describing interface based on open standards [10].

Architecture PipeLine. A pipeline connects components (filters) through connectors (pipes), so that the data is processed and executed as a flow. The data are transported through the pipes between the filters, gradually transforming the inputs into outputs [11].

Architecture Peer to Peer. We can consider a P2P system as a multilayered architecture, namely: The network level represents the lowest level of the architecture, offering the basic capabilities of communication between computers, either through IP networks or through Ad-Hoc networks. The level of administration of the P2P layer, responsible for routing messages and performing overlay maintenance tasks. The level of features, which supports network functionalities such as security, fault tolerance or resource management. The level of services, which is in charge of providing functionalities at the application level [1].

Event Oriented Architecture. Architectures with the following characteristics:

- Broadcast communications. Participating systems broadcast events to any group. More than one party can hear the event and pro-cease it.
- Opportunity. Systems publish events as they occur instead of storing them locally and await the processing cycle, such as a Discontinuous Cycle.
- Fine grain events. Applications tend to post individual events rather than a single aggregate.
- Ontology. The global system defines a nomenclature for classifying events, typically in some form of hierarchy. Receptor systems can often express interest in events or categories of events.
- Complex Event Processing: The system understands and monitors relationships between events, for example aggregation of events (a pattern of events involves a higher level) or causality (one event is caused by another) [12].

Three Level Architecture. Also known as Three Tier, or Three Scheme Approach. The purpose of the architecture of the three schemes is that:

- All users should be able to access the same data
- A user view must be immune to changes made in other views
- Users should not know the physical details of database storage
- DBA must be able to change the storage structures of the database without affecting the view of the users
- The internal structure of the database should not be affected by changes [12].

5 Mobile Services Architecture for IoT

After analyzing the current solutions and after reviewing and analyzing the existing communication protocols of the nodes networks. It seeks to determine in detail the desirable characteristics of the different modules of the proposed architecture, in order to define the necessary elements and the necessary processes within the different modules of the architecture, in order to later implement those modules, once these modules have been developed separately.

Definition of architecture modules. The modules proposed so far will be shown and explained in general below. The proposed architecture will have at least 3 main aspects which are the interconnection module which will be a middleware that will allow the interchange of the communication between the different sensor networks and the service connection architecture.

To achieve this, it is necessary to establish a common frame format which is operable by the hardware and software middleware in order to be able to connect with the service architecture.

Finally, the service connection architecture will be responsible for consuming and exploiting the information generated by the different sensor networks as well as providing services (functionalities) according to the user profile and its requirements. See Fig. 1.

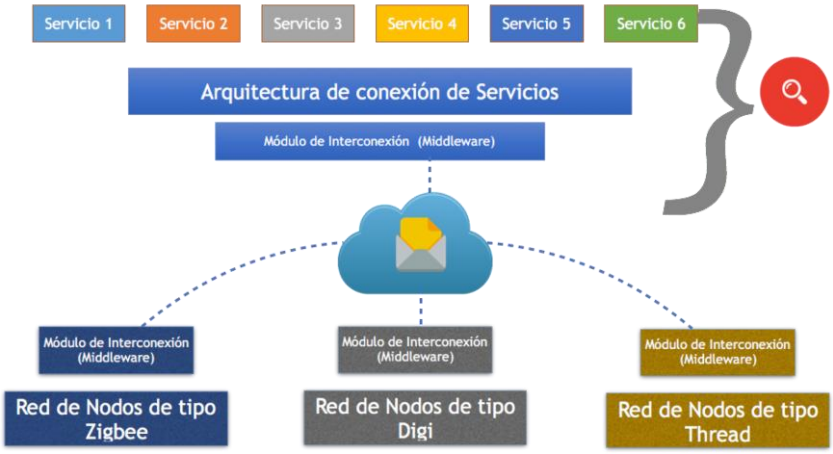

Fig. 1 System architecture.

6 Design and Development of Architecture

The SSA modules are described below:

- **Administration and Configuration Module.** This module is aimed at verifying the work of each of the other modules, as well as updating the databases in terms of configuration of devices and IoT.
- **Services Module.** This module is located both in the client and in the server, it has the functionality of providing and requesting the services and the handling of the information stored in the databases interacting with the user. These services can only be used by authorized users and can view the information they require according to the type of user.
- **Connectivity Module.** This module is one of the most important since it implements what is the connection and disconnection in IoTs between the server and the client, this module presents greater dependence in relation to the authentication module already that in order to establish the connection, it requires authentication beforehand.
- **Authentication Module.** This module is responsible for the validation of users who wish to establish connection, obtains the identifier and the address of the device whether the device assigns one automatically or is provided one in the context of the IoT.
- In this module in the client, the device is responsible for automatically finding a node of the available IoT network or that it is allowed to enter in order to be subsequently provided with the connection, authentication and can be provide the services.
- The authentication module is based on Oauth 2 that implements an access control based on tokens which are generated a refreshed by requests to server.

– **Device Identification Module.** As the architecture is independent of the devices used it is necessary to verify the capabilities of the device to be accessed and for this the server takes the device identifier, recognizes it and once the type of device was obtained determines how to provide the services in its respective module.

The architecture was tested by setting up a test case at the Higher School of Computer Science of the IPN, divided into a network IoTs a school control, a library domain and an Internet communication service, on different elements and with a central server of backup although it is not necessary for the architecture:

A system was developed that allows the members of the community of the National Polytechnic Institute to have an agile and comfortable access to the medical services offered by the institution, as well as a means of identification that establishes them as members of the community (students, teachers or employees). Likewise, the means in addition to identifying them will contain the vital personal and medical information of the applicant in order that those in charge of providing a service within the institute can offer it in a simple manner and also have a history of the activities of the applicant.

Each of the attention units within the institute where the services for the polytechnic community would be offered will be able to attend to the applicant requesting their card in order to obtain their information and identify him as a member of the IPN, this operation will be carried out through an NFC reader that is you will be connected to a desktop or wireless application where the information necessary to access the service will be displayed.

It will also have a server where the history of all the services that a person has requested and the history necessary for specific cases will be stored, being for example the history of medical consultations for the case of any medical area or the history of loans and debts for the library service.

The system will also have a web portal where users of the community will be able to access with a username and password in which they will have the opportunity to visualize the history of their requests and their general information.

7 Conclusions and Future Work

The main contribution of the proposed architecture is to allow to act with more mobility saving time and effort in the access to the information. The system itself is another way to help people carry out their functions in any area they develop, not only because it facilitates the job but also because it opens a door to innovation and a contribution to society.

The IoT with mobile nodes are becoming more common in both institutions and companies, it has now become a goal to achieve a ubiquitous computation that ensures total interactivity at all times and everywhere, so this project is focused on getting closer to these objectives, aiming to develop an environment capable of providing the services required for specific users in the most transparent way possible.

The system could also be a good basis for developing a ubiquitous and IoT computing environment wherever it is needed, establishing a protocol for the implementation of services for mobile equipment of various types.

The application of this architecture can be in many areas of knowledge, including:

- **Medicine.** In this area, it can be implemented in a hospital where when a patient arrives at reception with his cell phone can make an appointment, or at the same time can obtain information from an inpatient.
- **Education.** One form of application is that they can obtain mini applications on their mobile device such as information services, searches, virtual laboratories, simulation systems, etc.

References

1. Ashton, K.: That 'Internet of Things' thing. RFID Journal (2009)
2. Del Rosso, C.: Continuous evolution through software architecture evaluation: a case study. Journal of Software Maintenance and Evolution: Research and Practice 18(5), 351–383 (2006)
3. Diario Oficial de la Federación: Reglamento Interior de la Secretaria de Salud. pp. 8–40 (2016)
4. IEEE 802.15.4 Stack User Guide (2017)
5. Application Standards: http://www.zigbee.org/zigbee-for-developers/applicationstandards/ (2017)
6. Wireless Mesh Networking ZigBee® vs. DigiMesh. White Paper. Digi International. https://www.digi.com/pdf/wp_zigbeevsdigimesh.pdf (2017)
7. Thread Stack Fundamentals. 1st ed. Thread Group, pp. 3–19 (2015)
8. Microsoft: What is Middleware - Definition and Examples: Microsoft Azure (2017)
9. Aiken, B., Strassner, J., Carpenter, B. *et al.*: rfc2768. Cisco Systems, IBM, Argonne National Laboratory *et al., Ietf.org*. (2017)
10. Sundmaeker, H., Guillemin, P., Friess, P., Woelfflé, S.: Vision and challenges for realizing the Internet of Things. Cluster of European Research Projects on the Internet of Things—CERP IoT (2010)
11. Buckley, J.: The Internet of Things: From RFID to the Next-Generation Pervasive Networked System., Auerbach Publications, New York (2006)
12. G. Hohpe: Programming Without a Call Stack – Event-driven Architectures. 1st ed., pp. 2–3 (2006)