ISSN 2319 - 5975

Volume 8, No.3, April - May 2019
International Journal of Networks and Systems
Available Online at http://www.warse.org/IJNS/static/pdf/file/ijns20832019.pdf
https://doi.org/10.30534/ijns/2019/20832019

A SURVEY ON MIDDLEWARES USED IN ADVANCED IoT TECHNOLOGIES

Megha S [1], Devipriya S Kumar [2], Sheba Jiju George[3]
[1]KTU University Mangalam College of Engineering, Kerala India, smegha4321@gmail.com
[2] KTU University Mangalam College of Engineering, Kerala India, devipriyasreeragam@gmail.com
[3] KTU University Mangalam College of Engineering, Kerala India, sheba.jiju@mangalam.in

ABSTRACT

This paper proffers a survey on various middleware accessible for the Internet of Things (IoT). IoT is an advanced technology considered as a part of future internet and pervasive computing and it makes a true ubiquitous environment. The middleware for IoT act as a bond joining the amalgamate domains of applications divulging over amalgamate interfaces. Encyclopedic review of the existing middleware systems for IoT is provided to accomplish the better empathetic of the current gaps in this field. Apportionment of various middleware and protocols used are analyzed and our perception in this area is also presented.

Key words: Internet of Things, Middleware, Protocols

1. INTRODUCTION

As we are eloquent towards the Internet of Things (IoT), a jillion of sensors are getting connected to the internet, for making a ubiquitous environment. A challenge arising while attaching these sensors is to determine which one is eclectic to get correct data. The accuracy and precious of sensors will depend on the user needs. The communication between the user and IoT is done by middleware.

IoT is basically expanding interdependence of humans to interact, contribute and collaborate with things around us. It is the inter-networking of devices, vehicles, buildings and other things embedded with electronics, software, sensor and network connectivity which enables the objects to collect and recapture data.

1.1 Benefits of IoT

The first thing that could be as a benefit of having an IoT platform would that we efficiently utilize the resources that are available. If we have a smart system which can interact with everything or has enough computational power or has enough understanding of how things work between each other and quite sure the usage of the resource available will be more efficient as well. This resource could be in terms of monitory, natural resource, could also be input taken up by the things as input and so on. So all this can be more efficient if we have a platform which is smarter and interconnected as well. Apart from this, it minimizes the human effort involved. If our system is smarter enough to do anything we don't need to involve with or minimum.
It saves time. Time is a major factor that can be saved on an IoT platform. If we have a system where all these components and things are interconnected then intern all the security presented each of these things going to get multiplied and it's going to build much more security. Apart from that, the level of security that we would be integrating into the platform itself is going to be quite huge. So the overall security built with respect to everything is going to increase multiple times as well.

2. MIDDLEWARE LAYER

Middleware layer issues censorious functionalities such as filtering and aggregating the data acquired from the hardware devices, accomplishing information discovery and bestowing access control to the devices for applications. This layer provides services that allow sharing between other layers. There are multiple ways to process and integrate information but due to the lack of standards, an important role is played by the middleware layer. This hides the complexities of the lower layers, such as network and operating systems

3. MIDDLEWARE

Middleware is also known as 'software glue' which serves as a link between components of IoT, making elements capable to communicate that are not adept. This is a software which acts as a bridge between the operating system and application on the internet. A piece of middleware software also might manage a connection to cloud-based resources. It has a capability of implementing logic based upon the request made from the user. It plays a role in load balancing, concurrent processing, and transaction management. It has an ability to challenge the users and it requires a secure connection. Its products are also available as a specific cloud service tool, as well as multi-tool suites. They have typically had the capacity to scale vertically and horizontally to help virtual machines over multiple servers.

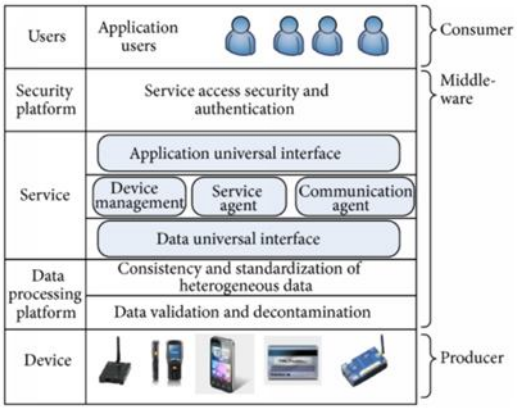

Fig 1 Architecture of Middleware [4]

There are so many acumens why middleware is necessary to connect sovereign devices:

3.1 Device Management
Devices must report their companionship and the services that they implement before connecting. In this case, middleware takes the form of APIs, listing the devices, services, and capabilities. Implementation of IoT in business supervisors. It is a process of configuring, provisioning maintaining and monitoring the firm work of device and software that provide the equipping capabilities.

3.2 Interoperation
Devices that have independently developed components can interact and cooperate with each other. They can exchange data. In this case, there is an increasing importance of network latency, autonomy, and mobility. Interoperability models are reusable, visual software artifacts that model the behavior in a lightweight and technology independent manner. These are used to help developers create and run a system that exactly interoperate.

3.3 Platform Portability
Platform portability in terms of middleware is a measure of how easily a platform can be transferred from one environment to another. A platform is considered portable to a new environment if the effort required to accept it to the new environment is within reasonable limits.

3.4 Context Awareness
Context awareness is the ability of a system or a system component to congregate information about its environment at any given time and acclimate behavior accordingly. In this case, contextual information falls into a vast range of divisions including time, location, device, user, role, activity, task, process and nearby devices or users. Changing requirements dynamic environments are drivers for a context-aware application.

3.5 Security and privacy
IoT devices have a robust amalgamation into our personal lives. As the sale of smart watches, smart phones, smart TVs, smart pillow, security and privacy issues should be also addressed. The middleware aiding IoT should have convinced security controls, including access control management and user authentication. IoT middleware has security one of their main challenges as these systems will be the target of new security threats.

4. INTERFACE PROTOCOLS IN IoT

Interface protocols are commonly used for software that facilitates communication and management of data in supported applications. Some of the interface protocols used in middleware are:

4.1 Zigbee
Zigbee is a standards-based wireless technology advanced to implement low cost, low power wireless machine-2-machine, and the internet of things networks. Zigbee is for low power applications, low data rate and is an open standard. This hypothetically, facilitate the mixing of implementations from different manufacturers, but in pragmatic, Zigbee products have been protracted and customized by hawkers and beleaguer by interoperability controversy. In disparity to Wi-Fi network used to connect endpoints to a high-speed network. It uses a mesh networking protocol to prevent hub devices and develop a self-healing architecture. It has a standard IEEE 802.15.4 and its WPANs operate on 24GHz, 900MHz, and 868 MHz frequencies.

4.2 RFID
Radio frequency identification (RFID) is a wireless use of the electromagnetic field to identify things. One of the main purposes of RFID is to hide and manage a broad range of device readers. Two basis solution of override is to define the specification that must be accomplished for every RFID reader inside the network and to define an abstract layer to translate propriety protocols and specific characteristics from a vendor. If we attach RFID to the terminal of the internet then, the user can track, identify and monitor the object to which the tag is connected. We practice this technology in an IoT platform since it is a security protocol. Bands RFID runs on: 120-150 KHz (10cm), 3.56 MHz (10cm-1m), and 433 MHz (1-100m).

4.3 Wi-Fi
Wi-Fi is a wireless local area network (WLAN) that exploit the IEEE 802.11 standard through 2.4 GHz to 5 GHz frequencies. It is the standard way computers connect to wireless networks. Since Wi-Fi is a wireless networking standard, any device with a “Wi-Fi Certified” wireless card should be recognized by any “Wi-Fi Certified” access point and vice verse. It is an interface protocol which we utilize for fast data transfer. High-quality data transfer can be done from anywhere rather than to a particular location.

4.4 Bluetooth
Bluetooth is a PAN protocol. Bluetooth is a short-range communication technology which is very imperative for computing. The distance of data transmission is small in comparison to other modes of wireless communication. It is

used in small gadget communication like a wireless keyboard, wireless headphone. It is a momentous protocol to IoT applications for data transfer. It has been developed to offer momentously reduced power consumption. The maximum coverage of it is 10m. Bluetooth sensors access the internet using 6Lo WPAN connectivity. Its frequency range is 2.4 GHz.

5. HARDWARE v/s SOFTWARE IN IoT

Managing hardware products requires very different skill than managing software. The more we understand more about how hardware works, its nuances and its nomenclature the more empowering will be to have a smart conversation with devices. Many IoT applications along with IoT entrepreneurs, it would be difficult to discern a hardware architecture disregarding of the application all IoT devices contribute some commonalities as shown below:

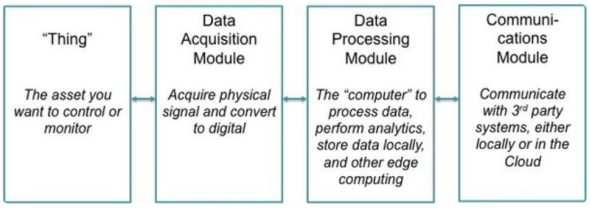

Fig 2 Building blocks of IoT device hardware [5]

5.1 Building Block 1: Thing

It is the asset that we want to control of control or monitor. In IoT products, 'things' is integrated fully into smart devices. For example, products like a smart pump and autonomous vehicles control and monitor themselves.

5.2 Building Block 2: Data Acquisition Module

This module focuses on amassing physical signals from 'thing' and convert it into digital signals which can be employed by a computer. A module contains more sensors though. It includes the necessary hardware to convert sensor signal to digital by signal conditioning, analog to digital conversion, interpretation, and scaling

5.3 Building Block 3: Data Processing Module

This is a data processing module which performs local analytics, store data locally and processes the data. The most two considerations to focus on are is one amount of local area storage and processing power.

5.4 Building Block 4: Communication Module

The last fundamental block of hardware is a communication module. It enables communication with a cloud platform and with the third party systems. This module may include communication port such as USB, CAN or Modbus. It also may contain radio frequency foe wireless communications such as Wi-Fi, Zigbee etc.

6. TYPES OF MIDDLEWARE

6.1 HYDRA

Hydra [1] is an IoT middleware that intent to consolidate wireless devices and sensors into ambient systems. HYDRA encompass a Context-Aware Framework (CAF) which equips the capabilities of both high level and powerful reasoning. CAF consist of two preeminent components. First is the Context Manager (CM) which is culpable for context management, interpretation and awareness and the second is Data Acquisition Components (DAqC) which is culpable for attaching and recapturing data from sensors. HYDRA detects context reasoning rule engine, context querying, context storage, and event management as the key integral of a context-aware framework [2].

6.2 ISMB

Middleware is adopted to clarify the problems in a specific application which means if any security related problem arises we use middleware. Since we are approaching a different technology to use there may be chances of high risk while interconnectivity. In IoT most probably we use this for real-time messaging and advanced message patterns. First, four levels of ISO/OSI stack is for static internet.

6.3 UBIWARE

In UBIWARE, smart semantic middleware is used for ubiquitous computing. Here RFID and sensor technologies are adapted to attach with the physical world and IT infrastructure. RFID and sensors are interface protocols. Sensors and RFID are used to realize about IoT. Some additional middleware is used for amalgamate components.

6.4 SOCRADES

On the basis of web services, the internal transmission and its communication with external entities is an event established approach. Hypothetically, each integral could be introduced to different locations. As a precedent, parts of architecture could be functioning on cloud contributing high conducting services whereas parts collaborating locally with devices could function even on an embedded system located at the device layer. With an ascendable in mind, the design and implementation were done. However, in the future, we intend to conduct performance tests and extended ascendable. Validation trials with real-world devices are planned in the automation and energy domain [3].

7. CONCLUSION

All the above-recorded middleware abutment devices exploration and management. Based on platform portability, HYDRA uses Java and XML, ISMB uses any Java docile platform, UBIWARE uses J2SE and J2ME and SOCRADES uses SAP Net Weaver Platform as well as DPWS. Context-aware functionality is backed by HYDRA and UBIWARE. Ubiquitous computing is backed by HYDRA and ISMB. On the other hand, HYDRA and SOCRADES are some examples of middleware enforcing security and user privacy in their architecture. On the substratum of our survey,

HYDRA is the most popular and well-documented middleware in comparison with the above-mentioned middleware.

REFERENCES

[1] W3.org, "Semantic sensor network xg final report: W3c incubator group report," June 2011, http://www.w3.org/2005/Incubator/ssn/XGR- ssn-20110628/ [Accessed on: 2012-09-25].
[2] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Ca4iot: Context awareness for internet of things," in IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, and Conference on Cyber, Physical and Social Computing (iThing), Besanon, France, November 2012.
[3] P. Norvig, "Teach yourself programming in ten years," 2001, http: //norvig.com/21-days.html [Accessed on: 2012-04-16].
[4] http:// www.researchgate.net/figure/The-IoT-middleware-architecture-based-on-SOA_fig1_281165297.
[5]Jose Antonio Cerrada https://www. researchgate.net/publication/260318390_Data_EPC_Acquisition_System_Middleware_Device_Manager_DEPCAS_MDM