IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Overview Of Iot-Enabling Technologies And Protocols

[1]P.M.G.Jegathambal, [2]B.Yamini, [3]P.Sheela Gowr

[1]Assistant Professor, [2]Assistant Professor, [3]Assistant Professor

[1]Computer Science and Engineering,

Vels Institute of Science, Technology and Advance Studies, Chennai, India

Abstract: This chapter presents an overview of the Internet of Things (IoT), focusing on enabling technology, protocols, and application concerns. The Internet of Things is enabled by the latest breakthroughs in RFID, smart sensors, communication technologies, Internet protocols. The core assumption is that smart sensors interact directly without human intervention to deliver a new class of applications. It describes about the different IoT enabling technologies like wireless sensor networks, cloud computing, big data analytics, communication protocols, embedded system. These technologies together enable development of IoT applications. It describes a Wireless Sensor Network (WSN) is a collection of devices which communicate through wireless channels and a WSN consists of distributed devices with sensors which are used to monitor the environmental and physical conditions with some examples of WSNs used in IoT. It also describes Cloud computing is a computing model in which applications and services are delivered over Internet. The resources provisioned by cloud can be compute, networking or storage. Cloud allows the users to access resources based on utility model. Along with this, the characteristics of the cloud computing, three basic service models (Infrastructure-As-A-Service, Platform-As-A-Service, Software-As-A-Service) and four cloud deployment models like public cloud, private cloud, community cloud, hybrid cloud are also explained. Additionally, it covers big data analytics which involves collecting, processing, and analyzing large, diverse datasets. In big data analytics the six step data analytics frameworks are summarized. In addition to these, it mentions communication protocols that allow devices to exchange data over networks and explains about the embedded systems which are specialized computer systems designed to perform specific tasks. Also explains about the characteristics and three components of the embedded system. It also discusses IoT protocols for data communication and connection models. It describes the key pillars of IoT protocols as being device, connectivity, data, and analytics. It also outlines various types of IoT data protocols like AMQP, DDS, XMPP, and WebSocket that establish end-to-end communication. Additionally, it covers IoT network protocols like Bluetooth, LPWANs, ZigBee, Z-Wave and others that facilitate secured communication between IoT devices over the internet. It also features the list of communication protocols that are suitable for Internet of things. IoT Layered architecture protocols help to establish Communication between IoT Device (Node Device) and Cloud based Server over the Internet. It helps to send commands to IoT Device and received data from an IoT device over the Internet. It features from http to CoAP in four layer categorization i.e., link layer protocol uses 802.3, 802.11, 802.16, etc, Network layer discuses about IPV4, IPV6, 6LoWPAN, Transport layer protocol explains about TCP, UDP and finally application layer protocol describes the HTTP, CoAP, MQTT. Finally, it elucidates the logical design of IoT i.e., IoT functional blocks that provide the system capabilities for identifying, sensing, actuation, communication and management. The IoT communication models discusses about request response model, publish subscribe model, push pull model and exclusive pair. In addition to these it discusses about two different types of IoT communication API's i.e., rest-based communication API's and web socket-based API's.

Index Terms: AMQP, DDS, XMPP, 802.3, 802.11, 802.16, HTTP, CoAP, MQTT, TCP, UDP, IOT, WSN

I. Introduction

The Internet of Things (IoT) is becoming a reality as more physical devices connect to the Internet at unprecedented rates. A simple illustration of such Objects include thermostats and HVAC (heating, ventilation, Monitoring and control systems for air conditioning Enable smart homes. There are alternative domains. situations in which the IoT can play a spectacular role. Enhance the quality of our life. These applications include transportation, healthcare, industrial automation, and emergency response to natural and man-made disasters that need complex human decision-making. The Internet of Things enables physical items to see, hear, and think. Collaborate on tasks by facilitating communication and sharing information and coordinate decisions. The IoT transforms these products are transformed from ordinary to smart by utilizing their underlying technologies, like ubiquitous and pervasive. Computing, embedded devices, and communication technologies

Sensor networks, Internet protocols, and applications. Smart The domain is made up of items and their supposed functions. Specific applications (vertical markets), while widespread Computing and analytical services form the application domain. Independent services (horizontal market). Figure 1 demonstrates the overall notion of the IoT in which every domain-specific application is interacting with domain independent services, whereas in each domain sensors and actuators communicate directly with each other.

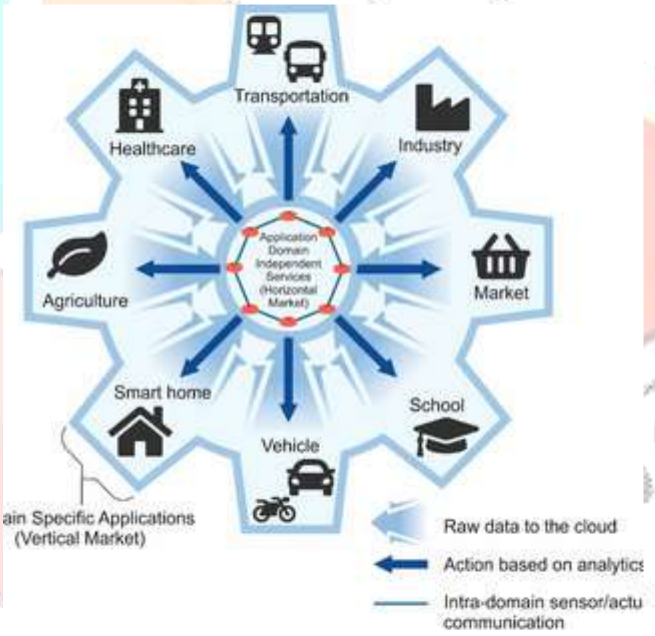
Transportation
Healthcare
Industry
Agriculture
Application Domain Independent Services (Horizontal Market)
Market
Smart home
School
Vehicle
ain Specific Applications (Vertical Market)
Raw data to the cloud
Action based on analytics
Intra-domain sensor/actu communication

Fig. 1. The overall picture of IoT emphasizing the vertical markets and the horizontal integration between them

II. IOT ARCHITECTURE:

The IoT should be capable of interconnecting billions or trillions of heterogeneous objects through the Internet, so there is a critical need for a flexible layered architecture. The ever increasing number of proposed architectures has not yet converged to a reference model [15]. Meanwhile, there are some projects like IoT-A [16] which try to design a common architecture based on the analysis of the needs of researchers and the industry. From the pool of proposed models, the basic model is a 3- layer architecture [3] consisting of the Application, Network, and Perception Layers. In the recent literature, however, some other models have been proposed that add more abstraction to the IoT architecture [2, 3]. Fig. 2 illustrates some common architectures among them is the 5- layer model (not to be confused with the TCP/IP layers) which has been used in [3]. Next, we provide a brief discussion on these five layers:

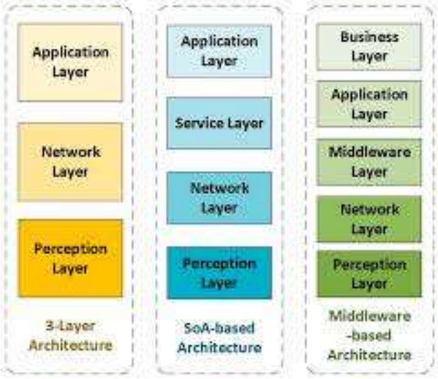

Fig. 2. The IoT architecture.

A. Objects Layer

The first layer, the Objects (devices) or perception layer, represents the physical sensors of the IoT that aim to collect and process information. This layer includes sensors and actuators to perform different functionalities such as querying location, temperature, weight, motion, vibration, acceleration, humidity, etc. Standardized plug-and-play mechanisms need to be used by the perception layer to configure heterogeneous objects. The perception layer digitizes and transfers data to the Object Abstraction layer through secure channels. The big data created by the IoT are initiated at this layer.

B. Object Abstraction layer

Object Abstraction transfers data produced by the Objects layer to the Service Management layer through secure channels. Data can be transferred through various technologies such as RFID, 3G, GSM, UMTS, WiFi, Bluetooth Low Energy, infrared, ZigBee, etc. Furthermore, other functions like cloud computing and data management processes are handled at this layer .

C. Service Management Layer

Service Management or Middleware (pairing) layer pairs a service with its requester based on addresses and names. This layer enables the IoT application programmers to work with heterogeneous objects without consideration to a specific hardware platform. Also, this layer processes received data, makes decisions, and delivers the required services over the network wire protocols [3].

D. Application Layer

The application layer provides the services requested by customers. For instance, the application layer can provide temperature and air humidity measurements to the customer who asks for that data. The importance of this layer for the IoT is that it has the ability to provide high-quality smart services to meet customers‘ needs. The application layer covers numerous vertical markets such as smart home, smart building, transportation, industrial automation and smart healthcare [3]

E. Business Layer

The business (management) layer manages the overall IoT system activities and services. The responsibilities of this layer are to build a business model, graphs, flowcharts, etc. based on the received data from the Application layer. It is also supposed to design, analyze, implement, evaluate, monitor, and develop IoT system related elements. The Business Layer makes it possible to support decision-making processes based on Big Data analysis. In addition, monitoring and management of the underlying four layers is achieved at this layer. Moreover, this layer compares the output of each layer with the expected output to enhance services and maintain users‘ privacy [3].

III. IOT COMMON STANDARDS

Many IoT standards are proposed to facilitate and simplify application programmers‘ and service providers‘ jobs. Different groups have been created to provide protocols in support of the IoT including efforts led by the World Wide Web Consortium (W3C), Internet Engineering Task Force (IETF), EPCglobal, Institute of Electrical and Electronics Engineers (IEEE) and the European Telecommunications Standards Institute (ETSI). Table III, provides a summary of the most prominent protocols defined by these groups. In this paper, we classify the IoT protocols into four broad categories, namely: application protocols, service discovery protocols, infrastructure protocols and other influential protocols. However, not all of these protocols have to be bundled together to deliver a given IoT application. Moreover, based on the nature of the IoT application, some standards may not be required to be supported in an application. In the following subsections, we provide an overview of some of the common protocols in these categories and their core functionality

A. Application Protocols

1) Constrained Application Protocol (CoAP)

The IETF Constrained RESTful Environments (CoRE) working group created CoAP, which is an application layer protocol for IoT applications. The CoAP defines a web transfer protocol based on REpresentational State Transfer (REST) on top of HTTP functionalities. REST represents a simpler way to exchange data between clients and servers over HTTP . REST can be seen as a cacheable connection protocol that relies on stateless client-server architecture. It is used within mobile and social network applications and it eliminates ambiguity by using HTTP get, post, put, and delete methods. REST enables clients and servers to expose and consume web services like the Simple Object Access Protocol (SOAP) but in an easier way using

Uniform Resource Identifiers (URIs) as nouns and HTTP get, post, put, and delete methods as verbs. REST does not require XML for message exchanges. Unlike REST, CoAP is bound to UDP (not TCP) by default which makes it more suitable for the IoT applications. Furthermore, CoAP modifies some HTTP functionalities to meet the IoT requirements such as low power consumption and operation in the presence of lossy and noisy links. However, since CoAP has been designed based on REST, conversion between these two protocols in REST-CoAP proxies is straightforward

CoAP aims to enable tiny devices with low power, computation and communication capabilities to utilize RESTful interactions. CoAP can be divided into two sublayers, namely: the messaging sub-layer and the request/response sub-layer. The messaging sub-layer detects duplications and provides reliable communication over the UDP transport layer using exponential backoff since UDP does not have a built-in error recovery mechanism. The request/response sub-layer on the other hand handles REST communications. CoAP utilizes four types of messages: confirmable, non-confirmable, reset and acknowledgement. Reliability of CoAP is accomplished by a mix of confirmable and non-confirmable messages. It also employs four modes of responses as illustrated in. In CoAP's non-confirmable response mode, the client sends data without waiting for an ACK message, while message IDs are used to detect duplicates. The server side responds with a RST message when messages are missed or communication issues occur. CoAP, as in HTTP, utilizes methods such as GET, PUT, POST and DELETE to achieve Create, Retrieve, Update and Delete (CRUD) operations. For example, the GET method can be used by a server to inquire the client's temperature using the piggybacked response mode. The client sends back the temperature if it exists; otherwise, it replies with a status code to indicate that the requested data is not found. CoAP uses a simple and small format to encode messages. The first and fixed part of each message is four bytes of header. Then a token value may appear whose length ranges from zero to eight bytes. The token value is used for correlating requests and responses

The fields in the header are as follows: Ver is the version of CoAP, T is the type of Transaction, OC is Option count, and Code represents the request method (1-10) or response code (40-255). For example the code for GET, POST, PUT, and DELETE is 1, 2, 3, and 4, respectively. the Transaction ID in the header is a unique identifier for matching the response. Some of the important features provided by CoAP

- Resource observation: On-demand subscriptions to monitor resources of interest using publish/subscribe mechanism.

- Block-wise resource transport: Ability to exchange transceiver data between the client and the server without the need to update the whole data to reduce the communication overhead.

- Resource discovery: Server utilizes well-known URI paths based on the web link fields in CoRE link format to provide resource discovery for the client.

- Interacting with HTTP: Flexibility of communicating with several devices because the common REST architecture enables CoAP to interact easily with HTTP through a proxy.

- Security: CoAP is a secure protocol since it is built on top of datagram transport layer security (DTLS) to guarantee integrity and confidentiality of exchanged messages.

As an example of how an application protocol works in an IoT environment. Since the cloud service for this project, Nimbits, does not support CoAP currently, we used HTTP REST to integrate with Nimbits.

2) Message Queue Telemetry Transport (MQTT) MQTT is a messaging protocol that was introduced by Andy Stanford-Clark of IBM and Arlen Nipper of Arcom (now Eurotech) in 1999 and was standardized in 2013 at OASIS. MQTT aims at connecting embedded devices and networks with applications and middleware. The connection operation uses a routing mechanism (one-to-one, one-to-many, many-to-many) and enables MQTT as an optimal connection protocol for the IoT and M2M. MQTT utilizes the publish/subscribe pattern to provide transition flexibility and simplicity of implementation as depicted in Fig. 3. Also, MQTT is suitable for resource constrained devices that use unreliable or low bandwidth links. MQTT is built on top of the TCP protocol. It delivers messages through three levels of QoS. Two major specifications exist for MQTT: MQTT v3.1 and MQTT-SN (formerly known as MQTT-S) V1.2. The latter was defined specifically for sensor networks and defines a UDP mapping of MQTT and adds broker support for indexing topic names. The specifications provide three elements: connection semantics, routing, and endpoint

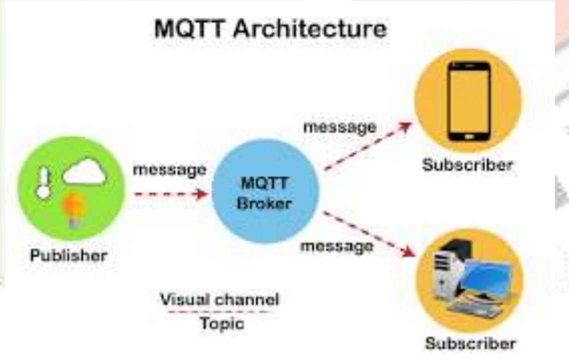

Fig 3. The architecture of MQTT.

3) Extensible Messaging and Presence Protocol (XMPP)

XMPP is an IETF instant messaging (IM) standard that is used for multi-party chatting, voice and video calling and telepresence.. XMPP was developed by the Jabber open source community to support an open, secure, spam free and decentralized messaging protocol. XMPP allows users to communicate with each other by sending instant messages on the Internet no matter which operating system they are using. XMPP allows IM applications to achieve authentication, access control, privacy measurement, hop-by-hop and end-toend encryption, and compatibility with other protocols.

4) Advanced Message Queuing Protocol (AMQP)

AMQP is an open standard application layer protocol for the IoT focusing on message-oriented environments. It supports reliable communication via message delivery guarantee primitives including at-most-once, at-least-once and exactly once delivery. AMQP requires a reliable transport protocol like TCP to exchange messages. Exchanges are used to route the messages to appropriate queues. Routing between exchanges and message queues is based on some pre-defined rules and conditions. Messages can be stored in message queues and then be sent to receivers. Beyond this type of point-to-point communication, AMQP also supports the publish/subscribe communications model.

5) Data Distribution Service (DDS)

Data Distribution Service (DDS) is a publish-subscribe protocol for real-time M2M communications that has been developed by Object Management Group (OMG) . In contrast to other publish-subscribe application protocols like MQTT or AMQP, DDS relies on a broker-less architecture and uses multicasting to bring excellent Quality of Service (QoS) and high reliability to its applications. Its broker-less publish-subscribe architecture suits well to the real-time constraints for IoT and M2M communications. DDS supports 23 QoS policies by which a variety of communication criteria like security, urgency, priority, durability, reliability, etc. can be addressed by the developer

Conclusion

The emerging idea of the Internet of Things (IoT) is rapidly finding its path throughout our modern life, aiming to improve the quality of life by connecting many smart devices, technologies, and applications. Overall, the IoT would allow for the automation of everything around us. This paper presented an overview of the premise of this concept, its enabling technologies, protocols, applications, and the recent research addressing different aspects of the IoT. This, in turn, should provide a good foundation for researchers and practitioners who are interested to gain an insight into the IoT technologies and protocols to understand the overall architecture and role of the different components and protocols that constitute the IoT.

REFERENCES

[1] D. Evans, "The internet of things: How the next evolution of the internet is changing everything," CISCO White Paper, 2011.

[2] L. Atzori, A. Iera and G. Morabito, "The internet of things: A survey," Computer Networks, vol. 54, pp. 2787-2805, 2010.

[3] R. Khan, S. U. Khan, R. Zaheer and S. Khan, "Future internet: The internet of things architecture, possible applications and key challenges," in Frontiers of Information Technology (FIT), 2012 10th International Conference On, 2012, pp. 257-260.

[4] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future Generation Comput. Syst., vol. 29, pp. 1645- 1660, 2013.

[5] P. Lopez, D. Fernandez, A. J. Jara and A. F. Skarmeta, "Survey of internet of things technologies for clinical environments," in Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference On, 2013, pp. 1349-1354.

[6] D. Yang, F. Liu and Y. Liang, "A survey of the internet of things," in Proceedings of the 1st International Conference on EBusiness Intelligence (ICEBI2010), 2010, pp. 358-366.

[7] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton and T. Razafindralambo, "A survey on facilities for experimental internet of things research," Communications Magazine, IEEE, vol. 49, pp. 58-67, 2011.

[8] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. McCann and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," Wireless Communications, IEEE, vol. 20, pp. 91-98, 2013.

[9] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," IDC iView: IDC Analyze the Future, vol. 2007, pp. 1-16, 2012.

[10] S. Taylor, "The Next Generation of the Internet Revolutionizing the Way We Work, Live, Play, and Learn," CISCO Point of View, 2013.

[11] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson and A. Marrs, Disruptive Technologies: Advances that Will Transform Life, Business, and the Global Economy. McKinsey Global Institute San Francisco, CA, 2013. [12] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang and J. Wang, "A first look at cellular machine-to-machine traffic: Large scale measurement and characterization," in ACM SIGMETRICS Performance Evaluation Review, 2012, pp. 65-76.

[13] D. Floyer, "Defining and Sizing the Industrial Internet," Wikibon, 2013.

[14] I. Navigant Consulting, "Commercial Building Automation Systems," Navigant Consulting Research, 2013. [15] S. Krco, B. Pokric and F. Carrez, "Designing IoT architecture(s): A european perspective," in Internet of Things (WF-IoT), 2014 IEEE World Forum On, 2014, pp. 79-84.

[16] (9/18/2014). EU FP7 Internet of Things Architecture project. Available: http://www.iot-a.eu/public