Practical Implementation of Industry 4.0 Based on Open Access Tools and Technologies

Ahmed Awouda[✉], Khurshid Aliev, Paolo Chiabert, and Dario Antonelli

Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Turin, Italy
{ahmed.awouda,khurshid.aliev,paolo.chiabert,
dario.antonelli}@polito.it

Abstract. Industry 4.0 provides a major breakthrough for innovating processes in contemporary manufacturing companies, it arises from the simultaneous presence of an opportunity and a necessity. The opportunity is the availability of powerful ICT technologies, that everyone uses every day with smart phones thanks to the well-established and expensive infrastructure provided by tech giants. The necessity is the achievement of Lean transformation in western enterprises through a deep knowledge on manufacturing processes and an incessant training of workforce.

Industry 4.0 will provide us with an integrated holistic view of the manufacturing processes and the possibility to control them according to appropriate forecasts and evaluation of performances and results. Industry 4.0 was born in 2011 and it is in its early stage, there are many research gaps which must be studied and large applications that must be fully integrated in the Industry 4.0 framework.

In this paper a simple application of Industry 4.0 is presented with the aim of demonstrating that, using commercially available open source components, it is possible to integrate different technologies belonging to industry (robotics) and commercial components into one eco-system. Moreover, it integrates different I4.0 enabling technologies namely: Robotics, IoT and fog/edge computing. Finally, the proposed framework can serve as an educational and practical tool for demonstration of Industry 4.0.

Keywords: Industry 4.0 · Internet of things · Collaborative robots · Open source · Data acquisition systems

1 Introduction

The Fourth industrial revolution promotes the "smart factory" idea, this idea is not only limited to the factory but is extended to the whole supply chain. Industry 4.0 integrates up-to-date technologies, which are equipped with sensors and intelligent control units able to simultaneously communicate and to autonomously exchange information about machines, plants or products. The underlying concepts of Industry 4.0 is the pervasive connection of devices, applications, operators belonging to the different levels of the automation stack illustrated by ISA 95 standards.

The Cyber-Physical Systems (CPS) promoted by Industry 4.0, can be easily assimilated to the human nervous system: a set of sensors (nerve endings) monitor physical processes and transfer signals to fog computing devices (peripheral nervous system) that control actuators (automated reaction to the stimulus) or transfer the information to the higher decision levels (the human brain) for a more elaborated and proactive answer.

In such Industry 4.0 scenario there are many research gaps which must be studied and there is no standard road map for developing solutions both from academic and industrial point of view.

The paper deals with a small part of the Industry 4.0 scenario: It's concerned with the usage of IoT solutions to monitor different systems belonging to: The Industrial realm and the commercial realm. Section 2 introduces the Industry 4.0 enabling technologies and highlight their use in the proposed solutions. It explores the connectivity of devices and illustrates the specific protocols chosen in this solution. Section 3 describes the implementation methodology: hardware components and supported functionalities are illustrated in more details. Section 4 summarizes the results of the study and highlights the possibilities for future developments.

The adoption of the open-source paradigm promotes reduced implementation costs, widespread portability of solutions and complete owning of process data.

2   Background

2.1   Enabling Tools and Technologies

To develop the fourth revolution of industry the following enabling technologies or techniques are studied [1]. These Enabling technologies (Both hardware and software based) represent the building blocks of Industry 4.0 as shown in Fig. 1, this section starts by introducing the technologies used and implemented in the proposed application and gives a brief description of the other possible technologies that can be added to the system in the future.

Cobots (collaborative robots) are the main hardware tool that has a big demand for implementation in Industry 4.0 applications. Cobots are designed to actively collaborate with human co-worker in the same workplace to complete the assigned tasks but they can work also as autonomous robots. Another advantage of cobots is their ease of programing, deployment and re-deployment to different applications. Therefore, as an example of cobots the UR3 (Universal Robot) has been utilized in the proposed system. The acquired data from UR3 allows to monitor the performance of the cobot and can serve the increase of production productivity.

Internet of things (IoT) means to utilize intelligently connected devices and systems to acquire data from embedded sensors and actuators in machines and other physical objects. Industrial Internet of Things (IIoT) is the subcategory of IoT and IIoT will serve to enhance manufacturing and industrial processes. IIoT allows field devices to communicate and interact both with one another and with more centralized controllers [2]. In the proposed system, IoT is deployed to connect different Industrial devices and sensors. Furthermore, the collected data can be used to perform decentralized analytics and to enable real-time responses.

Cloud computing models will serve to store and analyze increased datasets of Industry 4.0 applications and enable data sharing across sites and company boundaries by

dividing the physical resources with virtualization tools. Cloud computing performances will improve respond times (milliseconds) between machines by deploying data to the cloud and enabling data driven services for production systems [3]. Cloud services have been partially implemented into the proposed system by enabling storage data of machines, sharing data across sites and integration visualization tools.

The enabling tools of industry 4.0 are not limited to the abovementioned technologies but also include: *Big data, Augmented reality, Additive manufacturing, Simulation, Horizontal integration, Vertical integration* and *Cyber security* [4–7]. These technologies and tools are not addressed in this work but can be added to this work as a future improvement to the proposed system.
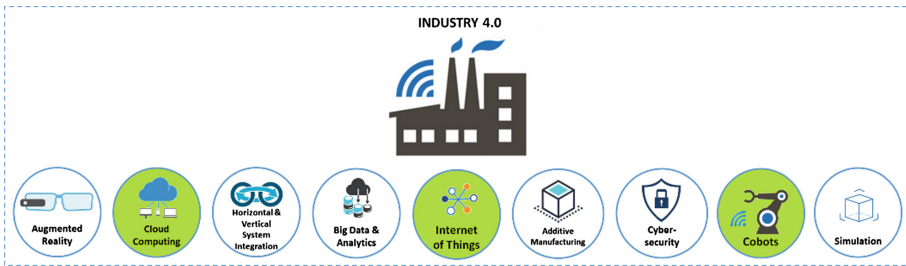
Fig. 1. Implementation of Industry 4.0 technologies are colored with green and future tools are indicated with a white color. (Color figure online)

2.2 Connectivity

Connectivity is the backbone of IoT [14], in the vision of I4.0 advanced connectivity of anything such as devices, systems and the granted access to the users at any time in any place to these systems is a must [15]. Communication protocols provide the set of rules for such devices to communicate and coordinate decisions.

There is a big variety of protocols used in today's industrial scene and the choice among these different protocols is totally application dependent and device specific. These industrial communication protocols fall into two main categories; Fieldbus protocols (such as Modbus RTU, Profibus DP and many other non-ethernet based protocols), and Industrial ethernet protocols (such as Modbus TCP and Profinet).

Industrial ethernet is practically the use of standard ethernet in industrial environments with protocols that satisfy the deterministic and real-time control requirements of factory floor and other industrial environments [8]. The growing popularity of Internet technology and IT usage stimulated a new wave of ethernet based industrial networks that borrowed basic technology from the IT world and tailored it towards usage in an industrial setting. According to a study conducted by [9] in 2018, industrial ethernet networks now hold a greater market share (52%) than traditional fieldbus technologies (48%) in factory automation with an annual growth rate of 22%. This transition to industrial Ethernet is driven by the need for high performance integration between factory installations and IT, interoperability requirements of industrial automation and control devices, as well as the growing adoption of Industrial IoT-systems.

Standardization of IoT connectivity is far from fruition and there is no clear de facto standard for IoT communication systems, however discussions within industry alliance

are active such as the IIC (Industrial Internet Consortium) [17] and IEC (International Electrotechnical Commission) [18]. Nevertheless, the MQTT protocol is becoming a de facto standard in IoT applications especially for applications of remote device connectivity. Another protocol that stands out in the realm of industrial communications and is widely adopted in legacy factory floor systems is the Modbus protocol which has been the de facto standard in factory floor settings for many years. For this reason, MQTT and Modbus have been chosen as suitable Protocols for our study since this will allow our framework to combine legacy systems of the factory floor with the emerging technology paradigm of IoT. This allows the creation of a true cyber-physical systems which are the primary enabler of Industry 4.0 [19].

Modbus TCP (Also Modbus TCP/IP) is a variant of the modbus protocol where the modbus message frame is incapsulated in a TCP/IP wrapper [10]. TCP/IP refers to the Transmission Control Protocol and Internet Protocol, which provides the transmission medium for Modbus TCP messaging. The primary function of TCP is to ensure that all packets of data are received correctly, while IP makes sure that messages are correctly addressed and routed.

As seen in Fig. 2, Modbus TCP uses a Client server model where the Client sends a request to the server to initiate a transaction, this transaction can be a data read or write. The server receives the request as an indication, performs the necessary action required from the transaction and then replies with a response. Finally, the client receives the response in the form of a confirmation.
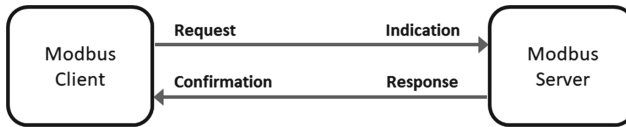
Fig. 2. Use case of Modbus TCP: Client-Server model where the Client sending a request to the server to initiate a transaction

MQTT provides connectivity between applications and users at one end, network and communications at the other end. MQTT is designed to minimize network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery, also MQTT protocol is a good choice for wireless networks that experience varying levels of latency due to occasional bandwidth constraints or unreliable connections [11].

As shown in Fig. 3, MQTT uses the Publish/Subscribe Model which consists of three main components: publishers, subscribers, and a broker. Publishers are the lightweight sensors and devices that connect to the broker to send their data and go back to sleep whenever possible. Subscribers are applications or devices that are interested in a certain topic, or sensory data, so they connect to brokers to be informed whenever new data is received. The brokers classify sensory data in topics and send them to subscribers interested in those topics only. A device can behave as a publisher and a subscriber at the same time by publishing to specific topics and subscribing to others, the term MQTT client is used to distinguish publishers/subscribers from brokers [12].
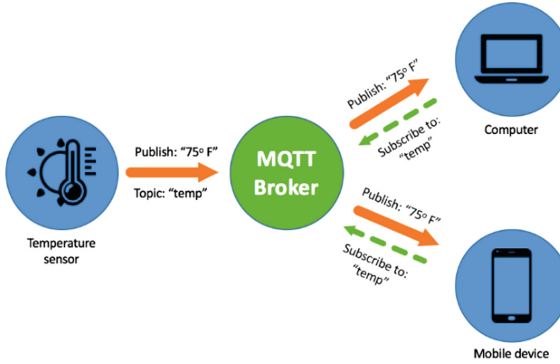
Fig. 3. MQTT Publish/Subscribe model between sensor & devices [13]

3 Implementation Methodology of Proposed Architecture

3.1 Hardware and Software Components

The internet of things allows devices to communicate and coordinate with each other in order to be able to make consensus decisions that benefit many applications. The proposed framework is a system able to collect data from different devices and allow this data to be accessible from anywhere in the world (via the internet) in the form of a dashboard. Figure 4 shows the proposed architecture and highlights its different components; the description of the components is as follows:
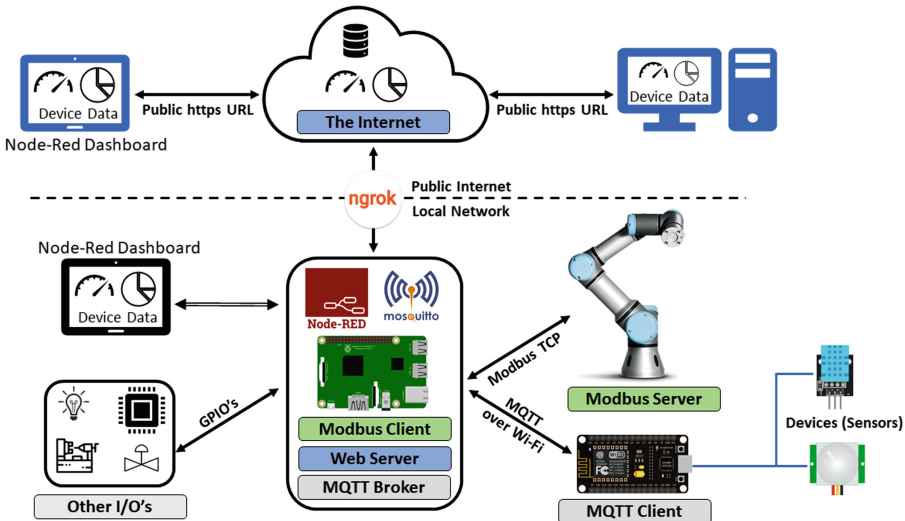

Fig. 4. Proposed architecture

Raspberry Pi is a single board computer that runs on open source software, it contains a Broadcom system on chip (SoC) with an ARM compatible CPU. In the proposed system the raspberry pi plays many roles: Modbus Client for the UR3 (Modbus Server);

Hosts the MQTT Broker (Mosquitto); Web Server for the Dashboard UI; Communication Gateway for GPIO connected Devices.

Node-Red is a flow based open source programming tool built upon Node.js that is used to connect hardware devices, API's and other online services belonging to the realm of IOT. Node-Red provides a browser-based flow editor which can be used to create JavaScript functions in the form of interconnected blocks that together construct a flow. One of the biggest advantages of Node-Red is its ability to run at the edge of the network on low-cost hardware such as the Raspberry Pi, in the cloud and locally on a standard PC. In the proposed framework Node-Red is ran on the Raspberry Pi and the editor is accessible via any web browser on the local network.

As seen in Fig. 5 which shows the developed node-red flow, Node-Red has been used to create the interactions between devices, sensors, robot ("things") and then to present the live data acquired from the devices on an interactive dashboard.
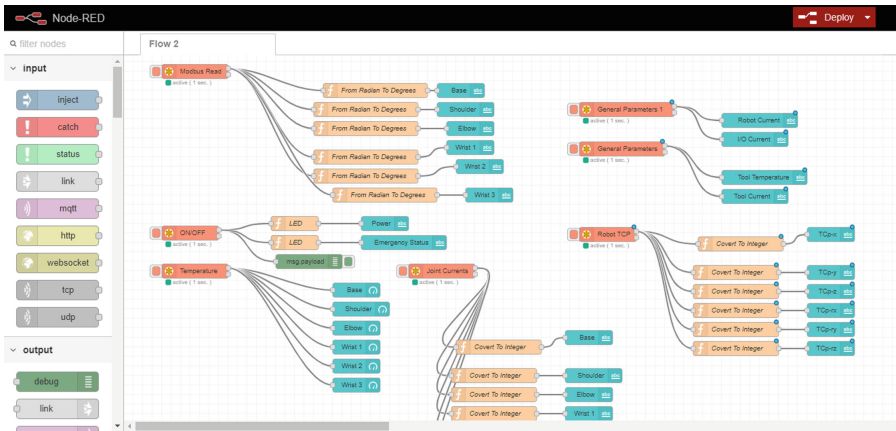

Fig. 5. Node-red flow image

The Node-Red dashboard is an add on module of Node-Red that is used to create and present a live data graphical user interface on a web browser. Figure 6 shows the developed dashboard, it can be seen from the figure that the dashboard package allows the addition of many UI components such as buttons, sliders, leds and gauges.

Mosquitto is an open Source message broker that implements the MQTT protocol, it can run on many devices from single board computers to powerful servers. It has been used to control the MQTT message flow between the ESP8266 and the Raspberry Pi but its limitations are much further as it can also be used manage MQTT communication between a magnitude of devices.

The ESP8266 (also known Node_MCU) is a low power Wi-Fi module with a full TCP/IP Stack and microcontroller capability. As seen in Fig. 4, it has been operated
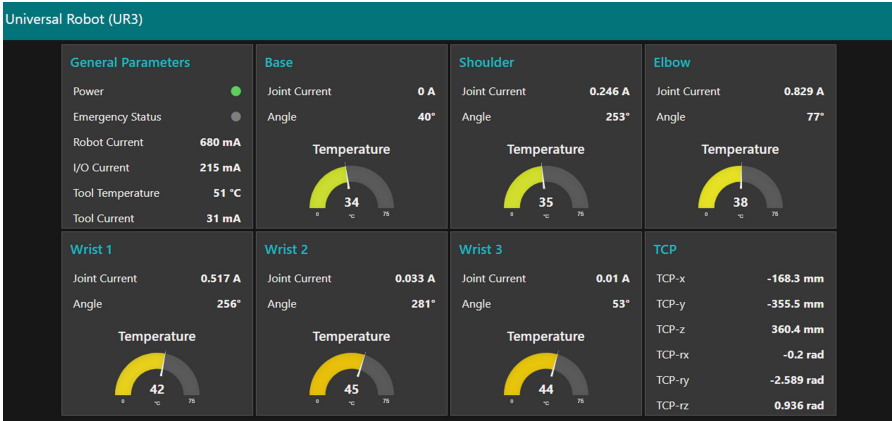
Fig. 6. Node-red dashboard

as an MQTT client that acquires sensor data from the sensors connected to it and then transmits the data to the Raspberry Pi via MQTT protocol.

NGROK is a tunneling, reverse proxy software that creates a secure tunnel on a local machine along with a public URL that can be used for browsing the local site hosted on the local machine (Raspberry Pi), it exposes local servers behind NATs and firewalls to the public internet over secure tunnels. It is used to allow the Node-Red Dashboard to be accessible via a public URL on the internet.

Data in the UR3 has been accessed via the modbus TCP protocol where the cobot is a Modbus Server and the RPi is a Modbus Client, the client sends requests to read specific registers available on the cobots internal memory, the cobot responds by providing the value of the requested register. Registers can hold discrete variables such as On/Off status and also analog values such as joint velocities, angles and also robot temperature and input current. The values of these registers have been aggregated in the raspberry pi and presented on the Node-Red Dashboard.

3.2 Functions Supported by the Demonstrator

The proposed framework is a generic, open source data acquisition system able to collect data from multiple devices and provide easy accessibility for this data via the internet. As seen previously in Fig. 4, this system is based on a Raspberry Pi that acts as a data collection buffer by holding the device data temporarily and as a gateway to the internet by presenting all the data on a web page GUI. By exploiting the multiple interfaces available on the raspberry pi one is able to have higher flexibility when it comes to data sources.

The presented system will allow (i) the collection of real-time data from various devices using different technologies and protocols of data transmission which in turn grants higher flexibility and lowers the limitations faced when acquiring device data (ii) remote monitoring of processes and sensor data from anywhere in the world via the internet (iii) usage of low cost devices and sensors to present a practical implementation

of IoT based on open access tools that can be used for educational purposes and other applications.

By exploiting the flexibility of the open source Node-Red packages, several nodes can be used to serve different applications, In the presented work the system acquires data via Modbus TCP, MQTT and hardwired GPIO's.

As seen in Fig. 7; a different set of nodes is used for each method of data acquisition. For the Universal robot a modbus tcp node is used, by defining the IP address of the Modbus TCP server (the robot), the function code and the data length, node-red can access the internal registers of the universal robot. From the robot point of view, a set of registers containing various parameters are present, by knowing the parameters held in each register (also known as the modbus map) one can read or write to those registers. An example of these parameters are joint angles, velocities, joint motor currents, temperatures, robot state and many other parameters measured by the robots internal sensory. In addition to the existing registers, the UR3 contains a set of general-purpose registers that can be customized to store other parameters or variables.
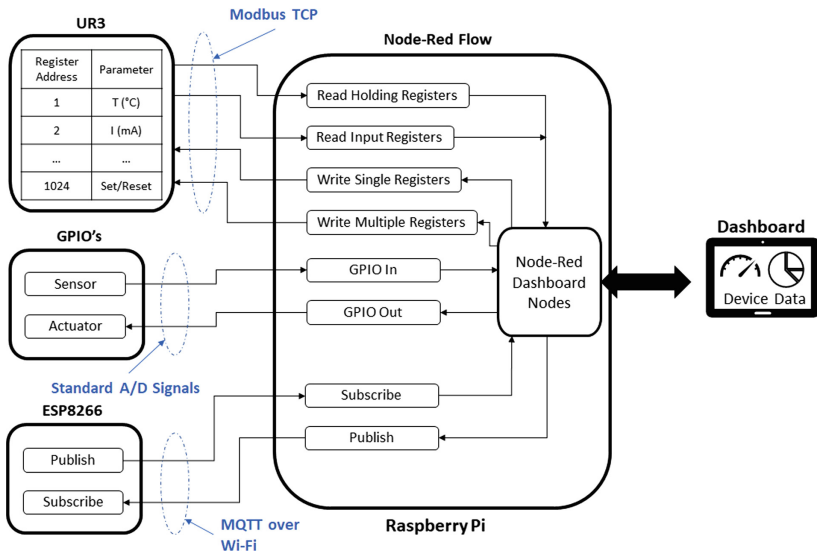

Fig. 7. Functional diagram of the proposed system.

For the ESP8266, the MQTT protocol over Wi-Fi is employed, the ESP8266 publishes messages to specific topics that the raspberry Pi is subscribed to, the Pi can also publish messages to topics that the ESP is subscribed to, messages can contain the sensor readings or commands to the ESP. By using this "Publish/Subscribe" model of MQTT a sort of 2-way communication can exist between the Raspberry Pi and the ESP8266.

The GPIO's onboard the Raspberry Pi are also used to acquire data. By connecting sensors and actuators used in standard small-scale electronics directly to the Raspberry Pi, Node-Red can read/write from/to these devices using an existing RPi library that

provides access to all the hardware interfaces on the Raspberry Pi such as USB ports, sense HAT modules and GPIO's.

Data from all the sources presented previously is then passed onto the Node-Red Dashboard Nodes which presents the data in a graphical user interface accessible through a web browser in a real-time manner. Node-red dashboard allows the data to be presented in various forms such as charts, text fields and gauges, Commands are also triggered from the GUI using sliders, switches, text fields and buttons.

4 Conclusions and Future Works

This paper proposed a framework and a practical implementation of an Industry 4.0 application by integrating recent developed tools and technologies. The Proposed technologies are based on open access systems and protocols which gives more flexibility and allows one to reap the benefits of the open source paradigm.

The proposed framework can be an appropriate tool for academics to explain Industry 4.0 in education systems but integration of the system into Industry requires some improvements by substituting components with more robust and certified industrial like tools. Furthermore, the transition into industry also requires suitable methodologies and standards to manage such tools and technologies and a trained work force that has the knowledge to work in I 4.0 systems.

In the future, more industrial devices will be added to the system in order to acquire more data to perform analytics on the cloud in addition to incorporate the other enabling technologies of Industry 4.0.

References

1. Kambarov, I., D'Antonio, G., Aliev, K., Chiabert, P., Inoyatkhodjaev, J.: Uzbekistan towards industry 4.0. defining the gaps between current manufacturing systems and industry 4.0. In: Chiabert, P., Bouras, A., Noël, F., Ríos, J. (eds.) PLM 2018. IAICT, vol. 540, pp. 250–260. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01614-2_23
2. Wollschlaeger, M., Sauter, T., Jasperneite, J.: The future of industrial communication: automation networks in the era of the internet of things and industry 4.0. IEEE Ind. Electron. Mag. **11**(1), 17–27 (2017)
3. Zhang, Q., Cheng, L., Boutaba, R.: Cloud computing: state-of-the-art and research challenges. J. Internet Serv. Appl. **1**(1), 7–18 (2010)
4. Lee, J., Kao, H.A., Yang, S.: Service innovation and smart analytics for industry 4.0 and big data environment. Proc. Cirp **16**, 3–8 (2014)
5. Gattullo, M., Scurati, G.W., Fiorentino, M., Uva, A.E., Ferrise, F., Bordegoni, M.: Towards augmented reality manuals for industry 4.0: a methodology. Robot. Comput.-Integr. Manuf. **56**, 276–286 (2019)
6. Dilberoglu, U.M., Gharehpapagh, B., Yaman, U., Dolen, M.: The role of additive manufacturing in the era of industry 4.0. Proc. Manuf. **11**, 545–554 (2017)
7. Thames, L., Schaefer, D.: Cybersecurity for Industry 40. Springer, Heidelberg (2017). https://doi.org/10.1007/978-3-319-50660-9
8. Jasperneite, J., Schumacher, M., Weber, K.: Limits of increasing the performance of industrial ethernet protocols. In: IEEE Conference on Emerging Technologies and Factory Automation, ETFA 2007, pp. 17–24. IEEE, September 2007

9. The website of HMS networks. https://www.hms-networks.com/press/2018/02/27/industrial-ethernet-is-now-bigger-than-fieldbuses. Access 20 Feb 2019
10. Modicon, I.: Modicon Modbus Protocol Reference Guide, pp. 28–29. North Andover, Massachusetts (1996)
11. OASIS Standard: MQTT version 3.1. 1 (2014). http://docs.oasis-open.org/mqtt/mqtt/v3, 1
12. Salman, T., Jain, R.: A survey of protocols and standards for internet of things. Adv. Comput. Commun. **1**(1), 1–20 (2017)
13. https://www.appcelerator.com/blog/2018/03/api-builder-and-mqtt-for-iot-part-1/. Accessed 1 Feb 2019
14. Samuel, S.S.I.: A review of connectivity challenges in IoT-smart home. In: 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC), pp. 1–4. IEEE, March 2016
15. Cho, J.Y., Lee, H.S., Lee, B.G.: Connectivity issues on IoT business-the Korean case of smart home network. In: The Second International Conference on Electrical Electronics Computer Engineering and their Applications (EECEA2015) (2015)
16. https://developer.ibm.com/articles/iot-lp101-connectivity-network-protocols/
17. Industrial Internet Consortium website. http://www.iiconsortium.org/
18. International Electrotechnical Commission website. https://www.iec.ch/
19. O'Donovan, P., Gallagher, C., Leahy, K., O'Sullivan, D.T.: A comparison of fog and cloud computing cyber-physical interfaces for Industry 4.0 real-time embedded machine learning engineering applications. Comput. Ind. **110**, 12–35 (2019)