Internet of things: a survey of challenges and issues

Qusay Idrees Sarhan

Department of Computer Science,
College of Science,
University of Duhok,
Duhok, Kurdistan Region, Iraq
Email: qusay.sarhan@uod.ac

Abstract: Internet of things (IoT) is the promising and future internet. The IoT is a network of connected sensors, actuators, and everyday objects that are used in various domains, such as healthcare, airports, and military. As it connects everything around us to the internet, the IoT poses a number of severe challenges and issues as compared to the conventional internet. Currently, there are massive studies on the IoT, these studies mostly cover IoT vision, enabling technologies, applications, or services. So far, a limited number of surveys point out comprehensively the challenges and issues of the IoT which considered unique to this future internet and which must be faced and tackled by different research communities. In this paper, well-known IoT challenges and issues (e.g., reliable cooperation, standards, protocols, operational, data, and software) have been surveyed alongside many directions. Furthermore, the paper also raises awareness of work being achieved across a number of research communities to help whoever decided to approach this hot discipline in order to contribute to its development.

Keywords: internet of things; IoT; wireless sensor; actuator networks; smart objects; smart environments; IoT applications; IoT services; research challenges; survey.

Reference to this paper should be made as follows: Sarhan, Q.I. (2018) 'Internet of things: a survey of challenges and issues', *Int. J. Internet of Things and Cyber-Assurance*, Vol. 1, No. 1, pp.40–75.

Biographical notes: Qusay Idrees Sarhan received his BSc degree in Software Engineering from the University of Mosul, Nineveh, Iraq in 2007 and the MTech degree in Software Engineering from the Jawaharlal Nehru Technological University (JNTU), Hyderabad, India, in 2011. Currently, he is a faculty member at the University of Duhok, Duhok, Kurdistan Region of Iraq. He has a couple of national and international publications and his research interests include software engineering, web services and technologies, wireless sensor networks, internet of things, distributed computing and systems.

1 Introduction

Technologies shape our modern life in one way or another. Out of many promising technologies is the internet of things (IoT). The term 'IoT' formed and appeared in 2002 from the title of a Forbes article by Kevin Ashton, when he said: "We need an 'internet-for-things', a standardised way for computers to understand the real world" (Köhler et al., 2014). The IoT extends intelligence computing and communication capabilities to everyday things or objects, such as traditional tools, sensors, cameras, cars, and appliances which are normally not considered as computing devices. Allowing these things to communicate and work with each other collaboratively to achieve common goals with minimal human intervention (Zanella et al., 2014). These interconnected devices link our physical world to the digital world to make the life easier and smarter. In 2011, the number of interconnected things or devices exceeded the number of people on the planet (Gubbi et al., 2013). The IoT is growing rapidly, it is expected that by 2025, the IoT will connect every object of our daily life to the digital world as stated by the US National Intelligence Council (NIC) (2008). As IoT consists of a huge number of heterogeneous and interconnected devices, this changes the way we live in many aspects. These devices are able to generate and consume different types of data. As a result, this provides significant development of a various number of applications never found before. The massive scale and different types of data produced by IoT heterogeneous devices will be used by these applications to provide new generation of services. IoT applications and services will cover many aspects of our practical life, such as energy management, inventory management, traffic management, home control and automation, industrial automation, healthcare, battlefield, and many others (Bellavista et al., 2013). However, IoT heterogeneous devices, applications, and services pose several challenges and issues which can be considered as a major barrier between the conceptual IoT and its full implementation, deployment, and adoption into our daily life. So, to fully implement, deploy, and utilise the IoT concepts, applications, and services in the practical life, a lot of research efforts and contributions are still required along many directions (e.g. technological, economical, legislation/regulation, and social) (Tan and Wang, 2010).

This survey paper contributes to the following:

- Provide in-depth study towards the state-of-the-art of major IoT challenges and issues that need to be tackled in order to fulfil the requirements of full, functional, and safe deployment of IoT scenarios in our daily activities.
- Provide readers of what have been done or proposed to address those IoT challenges and issues and what still remains to be addressed.

The rest of this paper is organised in many sections as follows. Section 2 presents the main challenges and issues that limit the reliable global use of IoT. Section 3 presents a number of IoT standards and protocols along with their challenges and issues. Section 4 presents the technical and operational IoT challenges and issues. Section 5 presents data and software challenges in IoT. It is worth mentioning that many of these different types of challenges and issues overlap with each other in some points. Finally, overall conclusions of this survey work and some future work are provided in Section 6.

2 Global reliable cooperation

2.1 Privacy

The main aim of privacy in the IoT is to prevent abuse or disclosure of data. Various efforts have been made to identify the issues of IoT privacy and provide possible solutions, either for user-centric privacy or device-centric privacy. For example, some of them are presented in Bandyopadhyay and Sen (2011):

a People often do not like their data to be accessed by public; so to ensure data privacy, there is a need for controlling over user's personal information.

b There is a desire that people should not be tracked without their permissions; so to ensure location privacy, there is a need for controlling over user's physical location and movements.

c To ensure the right of privacy, there is a need for privacy protection laws as clear legislative frameworks.

d To ensure privacy management, there is a need for standards, methodologies, and tools to help in this context.

Moreover, privacy issues have been expanded to user's devices as presented in Said and Masud (2013):

a Who collects devices data?

b How devices data are collected?

c What is the right time to collect devices data?

d Why devices data are being collected?

However, both user-centric data and device-centric data that are collected should be stored in authorised servers and accessed only by authorised individuals or parties (Chan and Perrig, 2003). In the IoT world, different systems communicate and interact with each other, each having a set of different privacy policies. As a result, conflict of policies and inconsistencies may arise as a big issue across these systems. Thus, new solutions are required for policies consistency checking, notifying, and resolving (Stankovic, 2014). Hence privacy polices considered a real issue that may limit the interaction between different IoT systems, there is a desire also from researchers to create a new and unified language to describe privacy policies in each of such systems. For example, in the traditional internet, platform for privacy preferences (P3P) works well as a language to express privacy policies. Unfortunately, traditional internet privacy languages have many drawbacks to be adopted in the IoT as they do not support real-time dynamic changes in the policies and do not support expressing different types of dynamic data and contexts (Olurin et al., 2012). However, another possible solution is the delegation mechanism as a privacy preservation approach. A simple scenario example of this mechanism can be given as a collaboration process of a smart fridge belongs to a private user network domain and an intelligent shopping service (belongs to another different network domain) provided by a retail shop. The intelligent shopping service can suggest a list of food items based on their expiration dates or availability in the smart fridge. To do so, it needs a privilege to access the smart fridge. To overcome this issue, a valid privilege will be delegated from the user network domain to the service network domain in order to allow the service to read the information required by it to make its list of suggestions (Roman et al., 2011). When a user in the IoT is able to use a radio-frequency identification (RFID) reader in mobile phones to scan the RFID tags embedded in objects (e.g. a visa card) and downloads their privacy policy in order to use and interact with them; that means an interesting technique called the privacy coach is used (Broenink et al., 2010). In this solution, if the downloaded privacy policy of an object does not match the user's preferences, the user can decide not to use that object. On the other hand, whenever there is an attempt by an RFID reader to read data from the user's mobile phone, the phone can check the privacy policy of the reader and ask for user permission. Another usage of the privacy coach is to protect the private physical space of the user (e.g. house or office). This type of protection is achieved by performing a scanning process for unwanted or malicious objects, such as sensors left at a house to do monitoring without the permission of its owner (Radomirovic, 2010). Sometimes, users want to provide information about them but without providing too much. For example, the user can locate someone near to his/her location who likes the same type of songs that he/she likes without providing his/her own location and songs preferences to that near person (Oleshchuk, 2009). Besides the previous example, a huge number of occasions will be there for collecting different types of user data. On top of that, the low cost of data storage which nowadays is about $0.03/GB or even cheaper, makes these data to be stored and memorised indefinitely. Therefore, it will be impossible for users to control their data personally (Atzori et al., 2010). However, remembering user data raises many other privacy issues as they may be used in many negative ways (e.g. defamation and disclosure). Therefore, there is a need for digital forgetting mechanisms to address the concerns in this respect by periodically delete user data that are no longer used for the purpose they were produced for. Digital forgetting ensures that data are memorised only when they are strictly needed (Singh et al., 2015). Recently, a number of software tools that support digital forgetting have been developed and released for the public use. For example, both *drop.io* and the guest pass feature of *Flickr* website allow their users to upload and share different types of data files (e.g. pictures) over the internet with the guarantee that their files will expire on a specific date and then be deleted (Thompson, 2009). It is clear that digital forgetting has been considered as a critical and important privacy protection technique. Unfortunately, studies regarding this technique and practical contributions to its development are still at the beginning (Mayer-Schönberger, 2009). In many other situations, users cannot limit or control what data are being gathered about them. For instance, users getting in a building equipped with a sensor network (e.g. composed of IoT cameras). This situation can be avoided by not entering the building so no image can be taken for users, but in most cases nowadays users have to enter buildings equipped with sensors. Restricting the ability of the IoT network deployed in such buildings is one possible solution in this respect. Restrictions can be applied on a detail level of gathering data that could not affect privacy in any way. For example, blurring can be applied on images of individuals in order to hide some important information in those images, thus protecting individual's privacy. In some critical situations, the image of a relevant individual can be reconstructed again to get some important information or details for further procedures but this can be performed only by law enforcement personnel (Wickramasuriya et al., 2004).

It is worth mentioning that in the traditional internet, privacy problems mostly appear only for users who are playing active roles. Whilst in the IoT, individuals face privacy problems even if they are not using any application or service. Privacy preserving and its acceptance by users considered one of the key requirements for the wide adoption of the IoT in our life. Therefore, concerns of privacy in the IoT must be taken seriously into account, given more attention by researchers, and should be considered and supported in the design of any IoT based solution.

2.2 Security

Privacy and security in the IoT overlap and reinforce each other in many aspects. However, security is a conclusive challenge for IoT core physical components [e.g. wireless sensor network (WSN) and RFID devices] due to the resource limitations, computing constraints, deployment nature, small storage space, low battery energy, and limited wireless channel bandwidth of these components. As sensor devices and networks in the IoT can be used in many sensitive applications, they can be attacked if they are deployed in unsecured environments (Sarhan, 2013). Any attack on a sensor node in the IoT may lead the entire sensor network to be compromised. However, software and hardware enhancements may address this issue in some cases. But to handle the issue properly and comprehensively, sophisticated countermeasures have to be applied, such as malicious node detection techniques, lightweight encryption algorithms, secure key management mechanisms, and secure routing protocols (Kocher et al., 2013). However, well-known WSN attacks and notable countermeasures that meet the requirements of protecting WSN in the IoT (Zia and Zomaya, 2006; Zhao and Ge, 2013) are summarised in Table 1.

Table 1 WSN attacks and countermeasures

Layers	*Attacks*	*Countermeasures*
Application	Subversion, data access, malicious nodes	Malicious node detection and isolation
Network	Sinkholes, Sybil, routing loop, wormholes	Secure routing key management
Data Link	Jamming	Encryption
Physical	DoS, fake node, node capture	Adaptive antennas, nodes monitoring, spread spectrum

To develop a potential security solution for the IoT, it is very important to grasp the security constrains presented in Table 2 for WSN as they form the main building block for the IoT (Kocher et al., 2013).

Table 2 Security constraints for WSN

Constraints	*Examples*
Limitable resources	Limited code storage space, limited data memory size, limited battery energy
Unreliable wireless channel	Unreliable data transfer, data conflicts, data processing latency
Unattended environment operation	Unattended deployment, natural disasters

Also, new applications will be needed to immediately notify users in case of any object moved or taken from a restricted area without any authorisation. Notifications triggered by such applications can be in many forms, such as sending SMSs, emails, and voice records (Atzori et al., 2010). IoT applications and services should be able to do their tasks continuously in the presence of security attacks and also should be able to do recovering from them in real-time. Recovering from security attacks involves detecting the attacks, analysing them, and then deploying suitable countermeasures against them or providing proper alternative solutions. However, all these processes must be performed in a lightweight manner with considering the security constrains given in Table 2. In many cases, especially when unanticipated attacks occur, recovering from them and healing require re-programming of devices or nodes. To do so, healing instructions need to be securely delivered to the appropriate nodes in order to be executed inside them (Stankovic, 2014). Besides what mentioned earlier, the other major problems related to security concerns in the IoT are data integrity, authentication, and confidentiality (Sarhan, 2013) as presented below:

2.2.1 Data integrity

Integrity of data means the ability to ensure that messages are not modified in any way during their travel in the network. In many cases, injecting a message with additional data can change the whole data stream in the network. Therefore, before making any critical decision on the collected data, it is required to check that the data are originated from the right sources (e.g. sensors and devices) without any modification.

RFID tags used in many IoT scenarios arise new issues in this context as they are unattended most of the time and they cannot be enabled with high level of intelligence (Juels, 2006). From security perspective, the data stored in RFID tags memory and RFID tags generated-data that travel in the network can be modified by attackers (Atzori et al., 2010). To protect data stored in tags or sensor nodes from attacks, programming errors, and memory corruption; many memory protection techniques have been proposed to help in this respect (Kumar et al., 2007). For example, EPCglobal class-1 generation-2 tags protect their memory from reading and writing operations by using passwords. On top of that, memory in this type of tags is divided into five areas, where each area can be protected independently from reading and writing operations by using a password. In many cases, using passwords is not feasible as most tags only support short password length which does not provide a strong level of protection. In case of tags supporting longer passwords, the process of their management considered as a major challenge. Of course, the issue of managing passwords is more obvious when different IoT entities belong to different organisations try to communicate and interact with each other to achieve a common goal.

2.2.2 Data authentication

Authentication of data is important to verify that the IoT data being sent to a receiver is produced from a specific sender. So that, the sender will not be able to claim later that the data sent is not from sender's side. There are many potential techniques have been proposed to overcome this issue. For example, authentication can be achieved using the keyed-hash message authentication code (HMAC) scheme (Krawczyk et al., 1997). The HMAC uses a secret key shared between the parties in form of integrity check value or cryptographic checksum which ensures authentication between communicating parties. It is worth mentioning that the HMAC scheme is often used with a hash function (e.g. MD5 or SHA) to perform the authentication process.

2.2.3 Data confidentiality

Confidentiality of data is to ensure secure messaging in the network and prevent messages from attacks. As the IoT consists of a massive number of sensors, it is important to ensure several sensor-related requirements:

a Data produced by a specific sensor should not be known to its surrounding sensors.

b Providing a secure communication channel, as sensors may exchange sensitive data such as secret keys distribution.

c Encrypting the messages, public keys, and identities of sensors will provide protection against different types of traffic analysis attacks.

To ensure confidentiality, many access control techniques have been studied and proposed to help in this respect. In role-based access control (RBAC) technique (Sandhu et al., 1996) for example, permissions to access certain IoT devices and services are associated with roles. So each IoT user is given a role and then is made a member of an appropriate permission and access control level. By this given role, each user knows exactly what and when to use certain IoT devices and services. In the IoT perspective, the RBAC presents a major advantage as its access permissions and rights can be updated dynamically in real-time whenever there is a change in role assignments. However, integrating access control techniques, real-time data stream management systems, secure data aggregation protocols, encryption techniques, and key distribution schemes can provide an acceptable level of confidentiality in many IoT scenarios (Miorandi et al., 2012).

In IoT based applications, many scenarios require new code to be installed on sensor nodes in order to address specific issues or pre-installed ones require to be periodically updated. Often, this process is achieved by applying the remote wireless re-programming mechanism to all nodes in the network. In traditional networks, data dissemination protocols are used for re-programming. These protocols distribute data to all members of a specific network without performing any authentication process, which considered as a big security issue. Also, cryptographic methods in these networks do not protect members from the internal malicious attacks in most cases. However, to tackle this issue, non-cryptographic methods are required. For example, secure re-programming protocols, such as Deluge protocol can be used to allow the nodes to perform authentication process on every code update and prevent malicious installation (Gubbi et al., 2013). Distributed and unattended nature of sensors deployments in the IoT make them susceptible to different types of physical attacks that could be achieved in many ways (Wang et al., 2004). For example, entering a house or an office where sensors are located and then detect their electronic signals (e.g. magnetic, heat, radio, and visual signals) through the use of signal-detection equipment. Thus, the location of deployed sensors can be determined based on the properties of the received signals, after which they may physically be destroyed (e.g. using physical force, heat, or counterfeit the associated circuitry on them), disabled, or even stolen. The losses due to physical attacks are not able to be undone or altered as the sensors are damaged permanently. In many other cases, physical attacks help attackers to access the cryptographic secrets and spoof or modify software code in the sensor nodes or even replace some nodes with other malicious nodes which would pose a clear challenge to the security of IoT applications (Alsaadi and Tubaishat, 2015).

2.3 Trust and governance

There is no single agreed definition of the term 'trust' in the IoT (Daubert et al., 2015). However, it can be defined as security policies and credentials that manage access operations to resources (Blaze et al., 1996). In the IoT world, trust mechanisms must be able to meet the following requirements (Roman et al., 2011):

- Decreasing the level of uncertainty of objects while interacting with each other.
- Helping objects choose a trusted partner for achieving their goals.
- Providing objects with dynamic and collaborative trust environments.
- Understanding the effects of IoT on the feeling of users while interacting with its scenarios. Feeling of being under external unknown control can greatly limit the deployment of IoT applications and services. Therefore, users must have the power to control their own services and must have tools that accurately describe all their interactions in the IoT world (Roman et al., 2011).
- Providing objects with a language for trust negotiation that simplifies credential polices and specifications in an easy and effective way.

In the IoT, trust can be classified into four main categories (Daubert et al., 2015), as follows:

- Device trust: it refers to the need of interacting with trusted devices only. To achieve this goal, trusted software and schemes have to be applied.
- Processing trust: it refers to the need of dealing only with meaningful and correct data. To achieve this goal, accurate data gathering, suitable data analytics, and data fusion have to be utilised.
- Connection trust: it refers to the need of exchanging right data with only right service providers. To achieve this goal, data integrity, authenticity, and confidentiality have to be taken into consideration.
- System trust: it refers to the need of providing a dependable overall system. To achieve this goal, transparency system workflows, processes, and underlying technology have to be provided and described with their contexts.

Besides the previous trust categories, trust negotiation is also required. Trust negotiation is the credential exchanges between devices and users that allow trusted access operations to be performed on resources to prevent misusing (Miorandi et al., 2012). In literature, there are many trust management protocols and systems that have been proposed in this context. In Bao and Chen (2012) for example, a trust management protocol for the IoT has been proposed. This protocol considers many trust attributes, such as honesty, cooperativeness, and others. Using the proposed protocol enables each node to apply trust detection only towards devices of its interest. The protocol also is event-driven, so upon the occurrence of malicious or unwanted interaction events; it immediately updates trust policies between interacting devices. Regarding trust management systems for IoT environments, the research efforts are fewer comparing to the protocols. However, a fuzzy reputation based trust management model in Chen et al. (2011) has been proposed to serve in IoT environments. The proposed model only considers specific wireless sensors with quality of service (QoS) trust metrics such as packet delivery ratio, packet forwarding ratio, and energy consumption.

In the IoT, governance reinforces trust in one way or another. For example, if a malicious action has been performed by a user or agent and has been attributed by someone else, punishing that user or agent should be possible and be quickly. Governance supports decisions, offers stability, and provides enforcement mechanisms to simplify data protection. On contrast, the wrong understanding and management of governance may lead to excessive governance in terms of people are controlled and monitored continuously. However, governance mechanisms in the IoT have to be given more attention from various research communities (Abdmeziem and Tandjaoui, 2014).

3 Standards and protocols

To fully adopt the IoT in our daily life in order to provide high quality services to users and to accelerate the wide spread of IoT technologies and innovations, a number of standards that address many aspects of the IoT (e.g. interoperability, reliability, and discovery operations) are required. However, many countries, organisations, and research communities around the world are actively contributing to the standardisation of the IoT as in the future it can achieve tremendous economic benefits alongside many other benefits (Da Xu et al., 2014). The most relevant standardisation bodies with their main responsibilities [Roman et al., 2011; The Internet Society (ISOC), 2015] are provided in Table 3. According to (Al-Fuqaha et al., 2015), the main protocols defined by these standard bodies which are being used today to implement the IoT can be classified into categories depending on the aim of each protocol. These categories are: application, discovery, infrastructure, and influential. In the following subsections, a brief yet accurate description of the most dominant protocols in each category is given.

3.1 Application protocols

Application protocols provided in this section are needed to handle the communication among the IoT devices, gateways, internet, and applications. These protocols are used to update online servers with the latest end-device data streams alongside carrying commands from applications to end-device actuators (Karagiannis et al., 2015).

Table 3 IoT standardisation bodies

Standardisation body	*Goal*	*Website*
Alliance for Internet of Things Innovation (AIOTI)	Interaction among IoT players, deployment IoT solutions in Europe, innovation and standardisation policies.	aioti.eu
AllSeen Alliance	Interoperability for IoT.	allseenalliance.org
European Telecommunications Standards Institute (ETSI)	Data processing, data management, data security, data transmission, standards for M2M systems and sensor networks.	etsi.org
International Society of Automation (ISA99)	Developing ISA/IEC 62443 standards to provide security for automation and control systems.	isa99.isa.org
IEEE Standards Association (IEEE-SA)	Standards, projects and events that are related to creating the environment needed for a vibrant IoT.	standards.ieee.org
European Research Cluster on the Internet of Things (IERC)	Coordinating the convergence of on-going activities.	internet-of-things-research.eu
Internet Engineering Task Force (IETF)	Ensuring consistency by reviewing specifications, monitoring the activities of IoT in other standardisation bodies.	ietf.org
Industrial Internet Consortium (IIC)	Industrial internet reference architecture, a language for the elements of Industrial Internet systems and their relationships.	iiconsortium.org
Internet Governance Forum (IGF)	Dynamic coalition on IoT.	iot-dynamic-coalition.org
Internet of Things Consortium (IoTC)	Adoption of IoT products and services through consumer research and market education.	iofthings.org
IP for Smart Objects (IPSO) Alliance	Using the internet protocol for the networking of smart objects.	ipso-alliance.org
International Organization for Standardization (ISO)	Technical standards for IoT, such as anti-collision protocol, frequencies utilised, and modulation schemes.	iso.org
Internet Society Special Interest Group (SIG) on Internet of Food	Technical infrastructural standards that will propel food into the digital era.	internet-of-food.org

Table 3 IoT standardisation bodies (continued)

Standardisation body	*Goal*	*Website*
Internet Society Special Interest Group (SIG) on Internet of Food	Technical infrastructural standards that will propel food into the digital era.	internet-of-food.org
International Telecommunication Union (ITU)	Its ITU-T Study Group 20 addresses the standardisation requirements of IoT technologies, specifically for IoT applications in smart cities and communities.	itu.int
Manufacturers Alliance for Productivity and Innovation (MAPI) Foundation	Industrie 4.0 for industrial applications of IoT.	mapi.net
Organization for the Advancement of Structured Information Standards (OASIS)	Open protocols to ensure interoperability for IoT.	oasis-open.org
One Machine to Machine (oneM2M)	Machine-to-machine communications architecture and standards in the view of the IoT.	onem2m.org
Online Trust Alliance (OTA)	Security, privacy, sustainability of the IoT.	otalliance.org
Open Interconnection Consortium (OIC)	Connectivity requirements, ensuring interoperability.	openinterconnect.org
Open Management Group (OMG)	Data distribution service (DDS), interaction flow modelling language (IFML), dependability frameworks, threat modelling, unified component model for real-time and embedded systems.	omg.org
Open Web Application Security Project (OWASP)	Helps manufacturers, developers, and consumers better understand IoT security issues and make better security decisions.	owasp.org
European Committee for Standardization (CEN)	Focusing on the evolution of RFID towards the IoT.	cen.eu
Global RFID Interoperability Forum for Standards (GRIFS)	Improve collaboration and thereby to maximise the global interoperability of RFID standards.	grifs-project.eu

3.1.1 Constrained application protocol

The constrained application protocol (CoAP) is an application layer protocol for resource-constrained devices that represent the basis for IoT applications and services (Bormann et al., 2012). This protocol has been designed by the IETF based on representational state transfer (REST) which considered as a simple way of exchanging data between clients and servers over hyper text transfer protocol (HTTP) (Lerche et al., 2012). REST allows web services to be exposed and consumed by clients and servers using uniform resource identifiers (URLs) and HTTP commands (e.g. GET, POST, PUT, and DELETE). The CoAP is suitable protocol for IoT and M2M communications due to its important features, such as:

- Some HTTP functionalities have been modified by this protocol to meet the IoT requirements (low power consumption, operation in the existing of lossy and noisy links).
- It provides resources monitoring and discovery for the clients (Al-Fuqaha et al., 2015).
- It is bound to User Datagram Protocol (UDP) not transmission control protocol (TCP) to provide lightweight implementation by reducing bandwidth requirements and removing TCP overhead (Keoh et al., 2014).
- It supports unicast and multicast communications.
- As UDP is unreliable, the CoAP achieves reliability by utilising various types of messages (confirmable, non-confirmable, acknowledgment, and reset) and responses (synchronous and asynchronous) (Karagiannis et al., 2015).
- It is a secure protocol as it utilises the datagram transport layer security (DTLS) that runs on top of the UDP. The DTLS provides many features, such as confidentiality, integrity, authentication, key management, and encryption algorithms (Alghamdi et al., 2013).
- It enables CoAP clients to access resources on HTTP servers through a reverse proxy that translates the HTTP status codes to the CoAP response codes using CoAPCHTTP (Palattella et al., 2013).

3.1.2 Message queue telemetry transport

The message queue telemetry transport (MQTT) is a messaging protocol released by IBM and was standardised in 2013 by OASIS (Locke, 2010). The MQTT protocol is built on top of the TCP protocol and provides a lightweight and optimal connection for the IoT and M2M due to its important features, such as:

- It supports the developing of distributed applications and services. Each client can be a publisher that acts as a generator of interesting data at a specific topic or/and a subscriber in order to be informed every time about any new updates on the topic is subscribed to (Lee et al., 2013).
- It is supported and implemented by well-known open source messaging brokers, such as Apache ActiveMQ and Apache Apollo.

- Brokers implementing this protocol can achieve security by requiring authentication of the publishers and subscribers which is handled by the secure socket layer (TLS/SSL) (Hunkeler et al., 2008).
- It ensures reliability through three levels of QoS (Hunkeler et al., 2008).
- It has low overhead compared to other protocols that are built on top of the TCP (Thangavel et al., 2014).
- It supports different routing approaches (one-to-one, one-to-many, many-to-many) (Al-Fuqaha et al., 2015).
- It is designed to use lower network bandwidth and fewer messages processing which extends the lifetime of devices battery-run. Therefore, it has been utilised by numerous applications, such as Facebook messenger and others (Karagiannis et al., 2015).

3.1.3 Extensible messaging and presence protocol

The extensible messaging and presence protocol (XMPP) was standardised by the IETF for instant chatting, voice/video calling, and telepresence regardless of the used operating system (Saint-Andre, 2011). As the XMPP is exist over a decade ago, it lacks the requirements of providing required chatting services for new applications, has no worldwide support, and it is often used by spammers. Therefore, big companies like Google moved away from supporting this protocol (Steven, 2013). However, the XMPP protocol has re-gained a lot of attention as a suitable communication protocol for instant messaging applications within the IoT scope for the following reasons:

- It is built on top of the TCP.
- It provides different messaging systems: synchronous (request/response) and asynchronous (publish/subscribe).
- It is extensible, allowing new functionality to be added and utilised through XMPP extensible protocols (XEP) (Karagiannis et al., 2015).
- It is a secure protocol as it utilises the TLS/SSL. Moreover, the XMPP provides many security features, such as authentication, privacy measurement, and access control (Al-Fuqaha et al., 2015).

3.1.4 Advanced message queuing protocol

The advanced message queuing protocol (AMQP) was designed to ensure reliability of financial transactions. To exchange messages, the AMQP requires a reliable transport protocol such as TCP (Al-Fuqaha et al., 2015). The AMQP is a good option for message-oriented applications in the IoT for its features as presented below:

- It provides an asynchronous publish/subscribe communication (Karagiannis et al., 2015).
- It offers store-and-forward feature to ensure reliability even after network disruptions (Johnsen et al., 2013).

- It is a secure protocol; it uses TLS/SSL protocols over TCP (Karagiannis et al., 2015).
- It ensures reliable communication by providing three message delivery guarantees (at most once, at least once, and exactly once) (Al-Fuqaha et al., 2015).
- It can send a huge amount of messages per second. It has been found that the AMQP can process 300 million messages per day in a distributed environment of 2,000 users (Fernandes et al., 2013).

3.1.5 Data distribution service

The data distribution service (DDS) was designed by the OMG for real-time IoT and M2M communications. The DDS offers many important features (Al-Fuqaha et al., 2015):

- It is a publish-subscribe protocol.
- It relies on a broker-less architecture and uses multicasting to provide QoS and high reliability to its applications.
- It supports 23 polices of QoS, such as reliability, security, priority, and many others.

3.1.6 Representational state transfer

The REST was first defined by Roy Thomas Fielding in his 2000 PhD dissertation (Fielding, 2000) as a software architectural style not really a protocol. As it depends heavily on HTTP, many researchers count it within the protocols. It uses the synchronous request/response HTTP methods GET, POST, PUT, and DELETE to provide a resource-oriented messaging approach.

The REST plays an important role in the IoT and M2M for its various features (Karagiannis et al., 2015):

- It is supported by all the commercial IoT and M2M cloud platforms.
- It can be implemented easily in smartphone and tablet based applications as it only requires a HTTP library which usually pre-exist in all the distributions of operating systems nowadays.
- HTTP features, such as authentication and cashing can be fully utilised in the REST.
- It is secure as it uses TLS/SSL.
- It can indicate the data format such as XML or JavaScript object notation (JSON) that being received from various clients using the built-in accept header of HTTP.

At the end, utilising all the mentioned protocols in a given IoT application or service is not mandatory as it depends on the application nature and requirements. However, Table 4 provides a brief comparison among the protocols that are presented earlier.

Table 4 Comparison of IoT application protocols

Protocols	*CoAP*	*MQTT*	*XMPP*	*AMQP*	*DDS*	*REST*
Transport	UDP	TCP	TCP	TCP	TCP UDP	TCP
Publisher/subscriber	Yes	Yes	Yes	Yes	Yes	No
Request/response	Yes	No	Yes	No	No	Yes
Security	DTLS	SSL	SSL	SSL	SSL DTLS	SSL
QoS	Yes	Yes	No	Yes	Yes	No
Low power and lossy network	Exc.	Fair	Fair	Fair	Poor	Fair
Dynamic discovery	Yes	No	No	No	Yes	No
Binary encoding	Yes	Yes	Yes	Yes	Yes	No
Real-time	No	No	No	No	Yes	No
Open source	Yes	Yes	Yes	Yes	Yes	No
Architecture style	P2P	Broker	P2P	P2P Broker	Data Space	P2P
Sponsor	IETF	OASIS	IETF	OASIS	OMG	IETF

3.2 Service discovery protocols

To fully utilise IoT devices, there is a need for protocols to discover resources and services offered by these devices in a real-time, efficient, and dynamic way. There are a various number of protocols to help in this respect, but the most prevalent are multicast Domain Name System (mDNS), DNS Service Discovery (DNS-SD), and Resource Directory (RD). Many research efforts have been performed to develop and adopt lightweight versions of these protocols to serve in different IoT environments (Jara et al., 2012).

3.2.1 Multicast domain name system

The mDNS protocol discovers resources based on a request/response mechanism. It sends a multicast message (e.g. inquiry devices that have a given name to replay back) to all surrounding nodes. When a node receives its name in the message, it multicasts a response message including its IP address. Then, all nodes update their nodes information table by the given name and IP address. The mDNS is a good option for IoT devices due to its features (Al-Fuqaha et al., 2015):

- it does not require manual reconfiguration to manage devices
- it is adaptable, as it continues working in case of any failure in the infrastructure.

3.2.2 DNS service discovery

The DNS-SD protocol uses mDNS to multicast standard DNS messages through UDP. In fact, this process involves two major steps: firstly, discovering host names of required services and secondly, pairing IP addresses to their host names using mDNS protocol. The DNS-SD protocol offers many features (Al-Fuqaha et al., 2015):

- It uses host names alongside IP addresses as they may change overtime.
- It keeps host names constant as long as possible to increase trust. For example, if a client knows and use a specific device today, the client will be able to reuse it thereafter without any trust issue.
- It does not require many resources (only the current DNS is needed with servers enabled with IP addressing (Jara et al., 2012).

However, when it comes to resource-constrained devices, the DNS-SD and mDNS suffer from the issue of caching DNS entries which is considered as the main drawback of these two protocols. A possible solution to this issue is timing the information cache or table for a specific interval and emptying it thereafter (Al-Fuqaha et al., 2015).

3.2.3 Resource directory

The RD works as a repository to store web links of resources hosted on smart objects that can be accessed through REST/CoAP interfaces only. Therefore, it does not require any additional protocols to achieve its work. The RD is organised as name domains and sub-domains as in the common DNS but with using CoAP protocol instead of the DNS protocol (Jara et al., 2012). The RD entries are different compared to the traditional DNS, they are stateless. As a result, they continuously require refresh from smart objects. Using the RD provides an approach similar to DNS-SD to perform the query but based on the description of hosted resources, their attributes, relationships, and parameters through the constrained RESTful environments (CoRE) link format (Shelby, 2012). This protocol offers functionalities to maintain the directory entries such as create, delete, and update.

3.3 Infrastructure protocols

In the IoT world, infrastructure protocols are required to establish the communication infrastructure needed to support the work of different IoT applications and services. In this section, the most prominent infrastructure protocols will be reviewed briefly.

3.3.1 Routing over low power and lossy networks

Routing over low power and lossy networks (ROLL) group is interested in the routing issue for IoT scenarios (Weiser, 1999). Recently, they developed the routing protocol for low power and lossy networks (RPL) routing protocol which is expected to be one of the go-to options for routing in the IoT. The main features of RPL are (Al-Fuqaha et al., 2015):

- It supports minimal routing requirements over lossy links.
- It supports different traffic models (point-to-point, point-to-multipoint, and multipoint-to-point).
- It keeps one path to the root at least for each node to perform fast search.
- It supports two operation modes: storing and non-storing modes. In storing mode, downward routing is achieved based on destination IPv6 addresses. Whereas in non-storing mode, the RPL routes messages towards lower levels are based on IP source routing instead of destination.

3.3.2 IPv6 for low power wireless personal area networks

IPv6 for low power wireless personal area networks (6LoWPANs) is developed to make the IPv6 protocol compatible with low capacity devices (Kushalnagar et al., 2007). The 6LoWPAN combines a set of several protocols that facilitate the integration process of sensor nodes into IPv6 networks. The main features of 6LoWPAN are (Al-Fuqaha et al., 2015):

- It provides header compression to reduce the transmission overhead.
- It provides fragmentation technique to meet the requirement of IPv6 maximum transmission unit (MTU).
- It provides forwarding to link layer to support multi-hop delivery.

3.3.3 EPCglobal

It is an initiative from the GS1 standards organisation. The main goals of EPCglobal protocol is to support the wide adoption of a unique identifier called electronic product code (EPC) for each device tag (*The GS1 EPCglobal Architecture Framework, GS1 Version 1.6*, 2014) and to use RFID and wireless technologies to allow everyday objects to be connected to the traditional internet.

3.3.4 Unique/universal/ubiquitous identifier

The unique/universal/ubiquitous identifier (UID) architecture is used to uniquely identify objects and places of the physical world in order to be accessed easily. These unique identifiers help UID based solutions to ensure the global visibility of objects and places (Sakamura, 2006).

3.3.5 Bluetooth low-energy

The Bluetooth low-energy (BLE) has been developed mainly to support novel IoT applications in healthcare, home entertainment, and other domains that do not require high level of data transmission. The BLE has been adopted rapidly by smartphone manufacturers and is now available on most recent versions of them. Currently, many existing mobile operating systems, such as Android, iOS, and Windows Phone support the BLE standard (Frank et al., 2014). The BLE presents many features (Al-Fuqaha et al., 2015):

- It uses short range radio with minimal amount of power compared to the traditional Bluetooth.
- Its coverage range is about 100 metres compared to the traditional Bluetooth which is about 10 metres.
- Its latency is 15 times shorter than the traditional Bluetooth.
- It offers two operation modes for devices, either slaves or masters in a star topology.

3.3.6 Z-Wave

It is a wireless protocol that has been developed by ZenSys (currently Sigma Designs) and improved later by the Z-Wave Alliance for automation in residential and light environments. The aim of this protocol is to allow a control unit to send reliable short messages to one or more nodes in the IoT network to perform automation processes (Gomez and Paradells, 2010). The Z-Wave protocol offers many features (Al-Fuqaha et al., 2015):

- it is a low-power wireless communication protocol
- it covers about 30 metres point-to-point communication
- it supports collision avoidance
- it offers reliable transmission by optional ACK messages.

However, many other infrastructure protocols such as LTE-A, IEEE 802.15.4, ZigBee, INSTEON, and Wavenis have been discussed in Al-Fuqaha et al. (2015) and Gomez and Paradells (2010).

3.4 Influential protocols

Influential protocols are the protocols and mechanisms that influence the acceptability of IoT in terms of privacy, security, interoperability, and many others which all are discussed in this paper in its different sections.

It is clear that standardisations should be considered as an important and critical part of the IoT definition and development process. As noticed earlier, many research efforts and tight collaboration between standardisation bodies, interest groups, and alliances are cooperating on achieving different aspects of the IoT (Santucci, 2009) but are not combined and integrated into a complete and comprehensive global framework (Abdmeziem and Tandjaoui, 2014). However, without a global effort to the standardisation process, the IoT cannot reach a global scale. The technological standardisation of the IoT in most areas is still in its infancy. Therefore, more efforts and collaboration among standardisation bodies, groups, and communities are very urgent and critical to reach the full IoT solution (Tan and Wang, 2010; Yang et al., 2010).

4 Operational and technical

4.1 Interoperability

The process of managing the work and interaction of heterogeneous data, devices, applications, services, and environments in the IoT to provide interoperability constitutes a major challenge to its popularity. Interoperability can be in two forms: technological and semantic. Technological interoperability should allow heterogeneity at the device level, such as diversity in terms of data communication methods and capabilities (e.g. data-rate, protocol stack, reliability, etc.), storage capability, computational power, energy availability, adaptability, mobility, and many others. And, semantic interoperability should allow heterogeneity at the data consumer level, such as diversity in the way how people ask for information and how they consume it. For example, an individual might ask for archived data while others might ask for real-time data. However, people needs differ in terms of data amount, data type, data location, and data quality (Zeng et al., 2011). For full interoperability, semantic interoperability must be considered even more (Gazis et al., 2015). This consideration is powered by the fact that the IoT consists of a huge number of various sensors and the data they provide is extremely massive. Thus, these data have to be collected, managed, and processed in an understandable manner. Semantic technologies help in this context by making a separation of data and their interpretation to provide better representations and understanding (Kotis and Katasonov, 2012). Semantic interoperability between heterogeneous service providers and service requestors can be achieved in many ways (Bandyopadhyay and Sen, 2011; Toma et al., 2009; Katasonov et al., 2007; Wahlster, 2008; Vazquez, 2009). For example, sharing information models can be used to provide semantic interoperability among different participants. However, the problem of this approach is that it is inflexible specifically when there is a lot of updating on the used information models. Another approach is utilising suitable semantic translators for each participant in an IoT scenario. These semantic translators can provide the required conversion from one information format to another that participants rely on or can understand easily. Besides, there are many attempts from different industry domains to utilise extensible markup language (XML) and develop XML-based standards to provide semantic interoperability at the data level. This leads to provide a standardised way to represent vocabularies of the contract, process, workflow, message, etc. between different IoT domains and scenarios. Publishing these XML based vocabularies as a generalised XML schema or document type definition (DTD) by a specific industry domain to be used, allow all members of other domains to follow the same standardised schema or DTD. These solutions limit the issue of that each domain members use their own set of vocabularies (they only know and understand them) to describe their interactions in the IoT. Another solution is the universal data element framework (UDEF) which aims to provide semantic interoperability between data that are constructed using different vocabularies, different data schemas, and different data dictionaries. This is achieved by using globally unique reference identifiers for data elements that are semantically similar, even if these elements use different XML markup standards for having different names. Web technologies also have been utilised in this respect. For example, semantic web based standards, such as ontology working language (OWL), Darpa agent markup language (DAML) and resource description framework (RDF) are the go-to options in providing semantic foundations in many IoT scenarios. Semantic ontologies should be used to prevent data ambiguities and misinterpretation due to human errors and misusing of different human languages in different regions of the world (Bandyopadhyay and Sen, 2011). There are other major efforts to address the interoperability in the IoT, such as the universal plug and play (UPnP) standard. The UPnP is a collection of networking and web protocols including HTTP, TCP, UDP, simple object access protocol (SOAP), and web services description language (WSDL). Fundamentally, the UPnP is developed for personal network devices to help them discover the existence of each other and then to establish connections among themselves to do required tasks. However, the UPnP has several drawbacks (Duquennoy et al., 2009), such as no authentication is supported which allows any device to configure other devices in the same personal network, it is not strictly standardised, and it is not applicable to some resource-constrained devices as it uses several heavy protocols which usually requires complex processing. Another effort is the IEEE 1905.1 standard (IEEE Standards, 2013), which was developed for providing interoperability between digital home networking devices and heterogeneous technologies supporting both wireline and wireless technologies. This standard provides an abstraction layer or interface that hides the diversity of common home network technologies. So that a combination of data link and physical layer protocols, such as IEEE 1901 over power lines, WiFi/IEEE 802.11 over different radio frequencies bands, Ethernet over fibre cables or twisted pair, and multimedia over coax alliance (MoCA) over coaxial cables can cooperate with each other to achieve common goals. The benefits of IEEE 1905.1 standard include simple setup, and no changes are required in the underlying layers. The interoperability issue between devices and services is still under study. Unfortunately, it is difficult to come up with one unified framework or one-fit-all solution to support the full IoT interoperability as many new various devices will appear in the future.

4.2 Mobility

In the IoT, a huge number of devices are connected to each other via various wireline and wireless technologies. In operation, these devices know their locations and have a set of settings like their constant turn on/off schedules, send data schedules, sleep/wake-up schedules, and many others. As IoT applications rely on devices to offer services to users, the mobility of IoT devices might cause critical problems. For example, in many situations changing a device location means there will be a change in time, date, or even environmental conditions. The data produced from IoT devices are often combined with timestamps (Fujita et al., 2011) and many applications depend on the creation time and date of IoT data to trigger or perform predefined tasks. Moving unexpectedly a device in the same small region may not affect the work of depending applications. But, when a device is moved far away (e.g. from one city to another or even from a country to another) from its original location then many applications which are depending on the data produced by the moved device may fail to deliver their functionalities or services properly. So creating context-aware applications to manage and notify about such changes are critical. Another issue in this respect is the clock synchronisation, which is considered as the most common problem in sensor networks (Lv et al., 2011). Since the appearance of wireless network and distributed systems, clock synchronisation has been studied broadly. Atomic clock, such as the global positioning system (GPS) is considered the classical solution to be utilised in such systems. However, limitations in sensor networks, such as cost and energy make it unfeasible to equip each device or sensor with its own GPS. Another issue that restricts the use of GPS to only outdoor environment applications is the need of sight line to GPS satellites (Lenzen et al., 2009). Moreover, exchanging messages at a very high rate is used by traditional clock synchronisation algorithms, which may be very difficult in WSNs and is not energy-efficient. Algorithms such as the network time protocol (NTP) (Mills, 1991) in are too complex to be used for sensor network applications. As they are designed mainly for the traditional internet and they are not accurate enough. To achieve considerable better results in sensor networks, they require sophisticated algorithms for clock synchronisation. Any solution in this context should consider the limitation of WSN in the IoT, such as the hardware clocks are often simple and may experience significant drift in time synchronisation and the multi-hop character of such networks. However, many other possible solutions are presented in (Lasassmeh and Conrad, 2010).

4.3 Massive scaling

People, animals, and other physical objects are massively getting connected to the internet to form the IoT. The main enabler of this kind of connection is the ongoing advancements in wireless technologies, micro-electromechanical systems (MEMS), and the internet. Eventually, everything around us will be connected to a global network (in many cases without requiring different types of human intervention) to enable collecting and disseminating data for controlling and monitoring purposes. As a result, connected things have a promising future in many domains including smart houses, smart hospitals, smart buildings, industry and manufacturing, distributed robotics, and national security. Moreover, the low cost, small size, and communication capability of sensors have made IoT more viable and have contributed to their increasing popularity as potential solutions to a variety of real life challenges. Unfortunately, how to identify, access, use, maintain, and manage such a massive scale of interconnected devices are some of the major problems in this respect. However, the most dominant problem is how these devices and services can be found in real-time in order to be used to meet daily needs. In order to utilise these services, they must be discovered in an easy and smart way. The inherent limitations of traditional search engines are a major obstacle for discovering IoT devices and services. Thus far, researches have indicated that new IoT devices and services search engines have to be addressed. Therefore, there is a need for better IoT devices and services discovery mechanisms that can combine low operational costs with a high accuracy performance. The research community presented many approaches of designing and implementing search engines for the IoT. Traditional search engines collect metadata from every web document, and store the metadata into an inverted index, this kind of engines mostly support static keyword searches to find such documents (Pokorný, 2004). Information retrieval algorithms (Ahmad and Ansari, 2012) are then used to determine the best answer to user queries. These search engines are not capable of discovering dynamic IoT devices and services as search objects in both types of search engines are very different from each other (Guinard et al., 2010). Search engines and systems providing comprehensive software infrastructures and tools for discovering, acquisition, storage, processing, and visualisation of data produced by IoT devices begun to appear in the last few years. For example, the internet-scale resource-intensive sensor network services (IrisNet) (Gibbons et al., 2003) is probably the first effort that realised the vision ‘world-wide sensor web’, through which multiple of sensors can be made openly accessible to multiple of users. IrisNet enables sharing of sensor data over a local platform to be easy used and configured. Also, the SensorBase (Chang et al., 2006) offers a data storage and management system that provide a consistent and uniform method to log sensor network data into a central repository. Instead of using a central repository, the global sensor networks (GSN) (Aberer et al., 2006) project provides a general-purpose infrastructure to facilitate the development and deployment of sensor networks. The GSN infrastructure is designed to integrate heterogeneous sensor networks and allow these sensors to be accessed by any computer interested in interacting with them. A further effort led to the development of online global sensor directories for providing generic platforms to share, query, and visualise sensor data. These platforms provide various tools for both data owners and users, for owners to publish the data easily and for users to easily make queries over already published and registered data sources, and then visualise data sources on a geo-based web interface. The most well-known sensor directories available today are SenseWeb (Santanche et al., 2006), Shodan (https://www.shodan.io/), and Thingful (https://www.thingful.net/) which are considered advanced engines for searching IoT devices and services. Localisation based searching for sensors and smart devices extended to involve distances between things as proposed by the authors in (Knierim et al., 2012). Their proposed system Find My Stuff (FiMS) is a simple search engine for locating physical objects in indoor environments using relative positioning of objects to find any lost stuff by checking its last position to the positions of all objects in the same place. From the information presented earlier and others in literature, the author found that most of researchers did not focus on the software architectures that have to be used for designing of such search engines and its big impact on the performance of searching and discovering IoT devices and services. Also, some works use very simple contextual based search such as using locations and fixed preselected contextual metadata about IoT devices and services without considering their relationships between each other. However, considering all the aforementioned factors will have a big impact and improvement on developing search engines for IoT devices and services.

4.4 Human interaction

In many IoT services and applications, humans will be closely involved to make them better serve everyday needs. Humans will interoperate with IoT devices to provide new opportunities to a broad range of application domains, such as automobile systems, energy management, and health care (Stankovic, 2014).To achieve this, there will be a necessity to support a tight integration with the human element via human interaction (HI) controls, also referred as the human-in-the-loop (HiTL). The HiTL will consider human intents, emotions, psychological states, and actions deduced from sensory data in the development process of IoT systems. Humans and their behaviours are no longer considered as an external and unpredictable factor but considered as a major element in the IoT (Sousa Nunes et al., 2015). Self-driving cars are a great example of the HiTL. Recently, Tesla Motors launched a car that mostly drives itself on highways with insisting that the human driver keeps hands on the steering system. So when the car senses that it has a doubt about a construction, snow, or something unusual on the road, it returns the control back to the driver (Biewald, 2015). However, utilising the HiTL in the IoT presents its own set of disadvantages. For example, modelling human behaviours is extremely hard due to the complexity of psychological and physiological aspects of them.

To overcome such disadvantages, new research efforts are necessary to employ the HiTL controls to system design and to address three key challenges:

a Understanding all types of HiTL controls in order to identify the common underlying factors (principles, requirements and models.

b Creating models of human behaviours (detecting and possibly predicting human nature).

c incorporating the human behaviour models into the IoT system itself (Munir et al., 2013).

However, HiTL applications nowadays cover many aspects of the daily life and each application has its own set of advantages and disadvantages. Generally, these applications are classified into four main categories (Stankovic, 2014), as follows:

- Applications where the functionality of the system is controlled straightforwardly by humans.
- Applications where humans are monitored passively by the system in order to take proper actions in case of need.
- Applications where the physiological states of humans are considered and modelled to perform specific tasks.
- Hybrid applications involving all the types mentioned before.

To enhance the opportuneness, performance, and accuracy of the system that takes into account the human element, accurate modelling techniques have to be utilised that are able to learn, analyse, and predict different types of human behaviours. Bringing the human element in the system is a big challenge as this process requires complex behavioural, psychological, and physiological aspects of human to be considered and modelled. For example, to control and manage tasks at hand, a person state, movements, vital signs, attention level, and many others have to be considered (Sousa Nunes et al., 2015). However, models of human behaviour can be created using many techniques. System identification is one of such techniques that can be used to model the human being using different statistical methods and equations. This technique deals only with static measured input data and it suffers handling dynamic or unknown data. Other modelling techniques, such as data mining, clustering, and inference are considered as the first human physiology and behaviours based models (Huang et al., 2012). To develop smart and sophisticated IoT systems, predictive models will also be required to avoid problems before they occur. In many situations, adaptive models and the HiTL controls will both be required as the system and human behaviours continuously getting evolved (Stankovic, 2014). To achieve the HiTL control, various processes are required (Sousa Nunes et al., 2015), as follows:

- Human existence: initially, human existence must be considered precisely as it is the basis for the subsequent processes. Human presence may be near or far presence.
- Data acquisition: data related to the human individual are collected from various existing sensors, such as wearable sensors, wearable accelerometers, gyroscopes, and many others. Traditional input devices (e.g. mouse and keyboard) have been used for a long time to reflect human needs and desires. These input devices are unpractical

as they involve a series of key presses or mouse clicks which are not intuitive, require a lot of practices to learn and master them, and most important they cannot reflect all types of human needs (various data). However, the IoT data may represent also the physical reality, such as vital signals (heart rate, body temperature, etc.), localisation (GPS positioning, etc.), movement (accelerometers, etc.), sound, and many others. Or, represent non-physical reality that can be obtained for example by analysing communication behaviours (e.g. how phone are used and how SMSs are sent) and social habits of a specific individual. Regarding capturing the physical data of an individual, the research community is extremely active and presented a number of acceptable results; some research efforts have achieved accuracy levels in the range of 9095%. On the other hand, detecting the user psychological states has been massively studies too. For example, smartphones have been utilised in experience sampling method (ESM). The ESM enable participants to respond to short questionnaires in order to give a clear insight into their behaviours and moods as reflects to experiences (Lathia et al., 2013). Then, the obtained results from such methods can be involved in the development of IoT applications and services to provide better functionality to users. Moreover, many other studies in this respect are presented in (Sousa Nunes et al., 2015).

- State inference: the gathered data are then processed to understand different human-related aspects, such as physical/psychological state, intent, emotions, and many others. Some contexts require predicting future states based on the information obtained from the current state and historical data of individuals. The outcome of this process is to help make the right decision in the right time.
- Actuation: eventually, in many scenarios the system performs a certain action based on the current conditions and in other scenarios do not. Moreover, many open-loop systems do not perform direct actuation as their results are simply informative. For example, systems that are passively monitor the human sleep environment in order to provide details about possible reasons of sleep disruption (Kay et al., 2012) have no direct impact on the associated environment. However, these systems are still actuating but by providing information (e.g. information displayed on screens). On the other hand, closed-loop systems actuate directly and have a notable impact on the human or environment to achieve a desired state through specialised devices, such as robots and many others (Schirner et al., 2013).

From what mentioned earlier, it is clear that the HiTL will become much more important in the near future. Even though the HiTL is in its infancy, there are many research efforts regarding data acquisition, state-inference, and actuation which indicate the big technological evolution that may be reached soon (Sousa Nunes et al., 2015).

4.5 Dependability

Dependability can be defined in many ways. For example, it is a system's ability to avoid service failures that are critical and repeated more than an acceptable level (Avižienis et al., 2004). This definition is applicable in the context of IoT applications and services, as failures may result in hazards, such as putting people in danger, financial loss, or environmental damage (Macedo et al., 2014). Dependability encompasses threats (e.g. faults, errors, and failures), attributes (e.g. availability, reliability, safety, confidentiality, integrity, maintainability, scalability, and privacy), and means (e.g. fault prevention, fault tolerance, fault removal, and fault forecasting) that should be taken into account while developing IoT applications and services (Avižienis et al., 2004; Fruhwirth et al., 2015). Dependability approaches and performance-controlled solutions in traditional application domains are X-by-wire systems. For example, in the automotive industry there is steer-by-wire and in the aerospace industry there is fly-by-wire where human life is in a big danger in case of any fail in the control system. Systems in these domains are planned, static, and deployed in well controlled environments where everything is limited in terms of the physical expansion, the number and type of interconnected devices, and the individuals involved (Fruhwirth et al., 2015; Donovan et al., 2010). These limitations and assumptions are obviously not applicable to provide or ensure dependability in the IoT. That is because of IoT environments are different, dynamic, highly mobile, contains different types of faults and failures, and its requirements change at any time. To keep a system performance in such environments under acceptable levels, new adaptable performance-controlled systems have to be developed and both fault tolerant and self-healing mechanisms have to be utilised in the design, development, deployment, execution, and testing processes (Sousa Nunes et al., 2015; Correia et al., 2014). For the IoT to become a reality, wireless sensor and actuator networks which form its backbone must be dependable. However, they do not currently offer an acceptable level of dependable performance as they are often affected by a wide range of environmental conditions. For example, temperature variations can lead to a loss of synchronisation and a degradation of the wireless connection quality. High temperature can drastically change the topology of the network, significantly reduce the performance of data link protocols, and increase the processing delay over the network. Radio interference from electrical appliances and other wireless devices can also impair communications, reduce speed, and lead to high latencies. To handle such issues, there is a need to create new accurate models for capturing the impact of the environment on IoT hardware and protocols. On top of that, developing new protocols that can be configured automatically to meet application-specific dependability requirements have to be considered (European Commission, 2014). Currently, there are many solutions (protocols and technologies) to provide dependable communication in control networks and local area networks (LANs). Also, many communication access methods, such as time division multiple access (TDMA) are used widely in different wired protocols to ensure dependable communication (Fruhwirth et al., 2015). There are notable contributions with the regard to dependability in the IoT also. For example, the authors in Tokuda et al. (2014) proposed a dependability case (D-Case) based tool to monitor, detect, and recover from a number of different IoT faults and failures. The D-Case enables monitoring programs to be executed on a variety of devices using different types of services. Moreover, the tool has many important features, for example, it is capable of monitoring various wireless sensor nodes, local IP networks, and others in real-time. On top of that, it gets evidences from running systems to know whether used networks are running in a healthy state or not. This is very important feature as static evidences like results from classical test cases are not useful always. Another interesting effort has been recently approved, named RELYonIT (2012–2015). It addresses dependability for the IoT by providing a set of testbeds, tools, and realistic environment effects to analyse the impact of surrounding environmental conditions (e.g. temperature, inference, and many others) on the IoT devices. Also, it decreases dependability by designing environment-aware protocols.

The above mentioned efforts made big contributions to facilitate dependability in the IoT, but none of them considered all the attributes of dependability in a complete and comprehensive manner. Moreover, various types of redundancy have not yet been taken comprehensively into account in order to analyse their effects on the application level (Fruhwirth et al., 2015).

5 Data and software

5.1 Openness

The main aim of designing and developing WSNs and embedded systems is to achieve a certain task in a specific industrial, scientific, or engineering domain. In such domain-driven development, the systems work effectively on static targeted scenarios (not unexpected and unplanned scenarios) which results constrained applicability and closed/restricted environments. However, such restriction prevents cost reductions that come with the massive production and development of new technologies nowadays (Rawat et al., 2013). Most of these systems that use sensors are closed systems. For instance, ships, cars, and aircrafts have their own sensor networks that often function internally. On the other hand, these restricted deployments of these systems have their own advantages, for example, they are only used by authorised users who are already familiar with the sensor network capabilities (Wood and Stankovic, 2008). The capabilities of such networks are growing swiftly nowadays. Ships send real-time location information to ships management and control parties. Cars send engine information to manufactures. Therefore, utilising the IoT in this respect will allow two-way communication, control, and exchange information between these entities which provide new service opportunities. To achieve this vision, data openness is required (Alur et al., 2015). The data streams produced from the aforementioned scenarios should be open for the public to be used without centrally controlled and managed. This can be achieved by utilising self-advertising and discovery mechanisms. Thus, gathering, processing, and visualising data should not only target scientists, companies, or administrators; but users mainly (Wood and Stankovic, 2008). The data should be delivered preferentially to users who have a greater need for them. However, It is worth mentioning that when openness is considered, a number of new issues have to be addressed (as the demand for sensor data would be massive and unpredictable), such as modifying the current applications and services composition techniques to account this openness, developing new unified communication interfaces to enable heterogeneous entities to talk to each other smoothly in order to exchange data, and finally accessing data and functionality in face of security and privacy should be considered too (Stankovic, 2014).

5.2 Big data

Connecting a large number of physical objects (e.g. humans, animals, plants, smart phones, sensors, etc.) equipped with various types of sensors to the internet generates the so-called ‘big data’ (Al-Fuqaha et al., 2015). The analysis of big data gives new chances to extract information and knowledge that will be extremely illustrative and helpful for many domains, such as research and products development [The Internet Society (ISOC), 2015]. The big data in the IoT world are real-time, massive, dynamic, continuous, produced in different formats, located in different places, owned by different people or parties, used for different purposes, and classified into interesting (valuable) and uninteresting (invaluable) data. To address the issue of analysing big data in the IoT, it is required to understand its nature, characteristics, and understanding that the big data exceed the common capabilities of current hardware and software platforms in terms of capturing, analysing, processing, and managing such data within acceptable periods of time (Al-Fuqaha et al., 2015). Big data analytics can be applied on IoT data to achieve data aggregation, correlation, filtering, and analysis. However, aggregating data presents a risk of privacy invasion and potential discrimination for many reasons (e.g. IoT devices can gather information about individuals without considering privacy policies). Both data aggregation and correlation processes can make detailed profiles of individuals which in many scenarios make the individual in face of physical, monetary, or reputational harm. On top of that, big data algorithms and tools are unfairly categorising users and misusing their information and characteristics [The Internet Society (ISOC), 2015]. Since most of the algorithms dealing with data analytics are owned by commercial companies and are not open source to be provided in the public domain, this raises many issues to be addressed [The Internet Society (ISOC), 2015], as follows:

- Detecting unfair practices against users and provide tools to help in this respect.
- Providing sufficient laws regarding discrimination decision, if it is made by a person or by a machine with providing remedies.
- Categorising IoT devices based on the data nature they produce, particularly when they are prone to misuse.

Currently, algorithms that are used for extracting meaningful information from complex sensing environments use shallow learning methods. These methods consider static data and pre-defined events to extract useful information (Kulkarni et al., 2011). The inferring process of activities by using events information that are obtained from shallow learning is considered as the next level of learning. Learning, representing, and modelling of different events and activities simultaneously at multiple complexity levels considered as one of the significant issues in this respect. However, the deep learning algorithms can be used to learn multiple complex layers of data simultaneously and model high level of abstractions in these data (Ravi et al., 2017). Unfortunately, deep learning faces many challenges due to the resource-constrained devices used in the IoT which limit its implementation in such devices. So, there is a need for considering new distributed and adaptive learning techniques (Gubbi et al., 2013). In fact, there is no one-fit-all solution for the issue of analysing and interpreting IoT data. Thus, users might not trust IoT systems if there is uncertainty or ambiguous in the interpreted data. In big data, trust is a key aspect of its usefulness. Both privacy and security are the fundamental elements of trust and they are discussed in detail in other sections of this paper. Wrong interpretation of data or missing data due to using unreliable transport protocols and inaccurate in-field sensor calibration techniques might cause wrong conclusions with bad impacts. Serious safety problems can occur if wrong conclusions drive actuators in sensitive IoT environments, such as laboratories, hospitals, and battlefields. To avoid such issues, users must be informed about how information is derived in such environments. Many IoT applications and services are developed to work for a particular individual, group, or community. In such applications, data association should be achieved correctly to ensure that data collected and subsequent interpretations are related to the right corresponding person, group, or community. In many other situations, real-time, recent, past sensor readings, and the history of a given individual's personal information/activities can be combined all to gain an accurate data association (Stankovic, 2014). These operations on data should be performed using software tools and frameworks. Currently, there are many open-source software frameworks for big data analytics, such as Apache Hadoop which is developed by Apache Software Foundation in 2011 and SciDB which is developed by Paradigm4 in 2008. However, these frameworks do not meet the needs of full IoT big data analysis (Tsai et al., 2014). Many big companies developed their own version of big data frameworks using existing open-source ones. For example, Facebook improved its own version of the Apache Hadoop to handle and analyse billions of messages per day and to offer real-time statistics of user actions (Borthakur et al., 2011). In terms of hardware, a lot of recent smart devices have a high level of computing capabilities that can be used to perform parallel IoT data analytic tasks besides the powerful servers in data centres (Mukherjee et al., 2014). However, standalone analytics are not preferred for IoT big data; it is preferred to have big data analytic which can be delivered as a common service to IoT applications (Al-Fuqaha et al., 2015). For example, recently a good work called TSAaaS has been proposed as an IoT big data analytics service by the authors in (Xu et al., 2014). The TSAaas performs pattern mining on a massive amount of data collected by sensors using time series data analytics. Also, it relies on time series database service and it is accessible by a set of RESTful interfaces to help developers build their own applications based on it. Moreover, it can perform faster data searches than the existing systems as shown by the authors in their evaluations. However, as the IoT big data consist of interesting and uninteresting data, one valuable solution is to keep track of the interesting data only. Many existing approaches can help in this context, such as pattern reduction, dimensionality reduction, feature selection, and distributed filtering methods (Tsai et al., 2014).

5.3 Software development

Software developers consider the 'thing' in the IoT as a type of networked device which is controlled and delivered by embedded software. Thus, functionality of a device is provided as a service. As in the IoT there will be billions of connected devices, there will be billions of services in the internet to form the so-called internet of services (IoS) (Sulistyo, 2013). A service can be defined in different ways. In Koskela et al. (2007) for example, a service is a software functionality that encapsulates a high-level business concept including data, logic, implementation, interfaces, and contract. A smart building can be considered as a typical example of an environment composed of various services. These services control and mange doors, windows, valves, security/surveillance systems, and many others. Combining these independent and different services to build a new application to serve users is a big challenge (Van den Heuvel et al., 2009). This challenge involves coping with a huge number of heterogeneous devices and overcoming different complexities (e.g. the complexity of distributed systems, domain-specific architectural knowledge, designing architecture for the application, applying architectures in application development process, writing code to test the functionality of application, deploying it, and considering proper maintenance in case of need) (Cassou et al., 2012). Therefore, there is an urgent need for developing comprehensive software engineering principles, architectures, frameworks, and tools to support the entire software development life cycle (SDLC) of pervasive computing and service-based applications in the IoT (Van den Heuvel et al., 2009). To do so, there are some key requirements that have to be understood and taken into account (Cassou et al., 2012), as follows.

5.3.1 Abstracting over the heterogeneity

Applications in the IoT interact with heterogeneous physical objects (e.g. sensors, actuators, and webcams). This heterogeneity has a big impact on the applications codes (e.g. a code will be written with much low-level details rather than a simple and high abstract code). This situation leads to write specific code for each physical object. To overcome this issue, it is required to raise the level of abstraction at which these objects are invoked. This can be achieved by developing software frameworks that reduce the details of objects manufacturing in coding applications.

5.3.2 Architecturing the development

Typically, IoT applications and services collect information from various environments, analyse, and perform actions accordingly. So, these applications have so many functional elements, work assignments, and constraints. Therefore, they have to be developed based on well-known software architectures such as microservices that can be considered as their blueprints that describe their internal elements with a high level of abstraction (Butzin et al., 2016). The next level of abstraction can be achieved by considering domain-specific languages. For example, the PervML (Serral et al., 2010) helps in this context. It is a domain-specific language that provides a set of entities and elements to help developers to precisely describe systems in a technology independent way and with no low-level details.

5.3.3 Leveraging domain-specific knowledge

As the IoT includes a growing number of various domains, every domain-specific knowledge has to be shareable and reusable to facilitate and boost the development of new applications and services. Two levels of reusability are required in this context: firstly at the entity level as applications in a specific domain usually share the same classes of entities and secondly at the application level to help developers meet new requirements using existing functionalities.

5.3.4 Covering the SDLC

Traditional software design techniques are very generic and do not fully support the development life cycle of IoT applications and services. To address this issue, new design techniques for the IoT are required. These new techniques should increase the productivity level of IoT applications and services, improve their quality level, and facilitate their maintenance and evolution over time. After successful implementation of applications, all aspect of their deployment should be facilitated too. However, as maintenance and evolution are important aspects for any software system. In the IoT, they are even more important, as new objects may be deployed or removed periodically in real-time and as users requirements may be changed frequently. Moreover, maintenance and evolution processes should be performed and supported by tools to save time and work effort.

5.3.5 Simulating the environment

The deployment and testing of IoT applications and services require a large number of connected devices and objects to be provided, configured, deployed, and then tested in reality. However, some scenarios are very hard to be provided in reality, such as considering fire in a manufacturing building equipped with different types of sensors and actuators (Reynolds et al., 2006). Therefore, various types of simulation tools should be provided to developers to overcome this deployment barrier. These tools should allow assuming different IoT scenarios and should also allow testing applications on these scenarios in a well simulated environment (Fortino et al., 2016).

6 Conclusions

The IoT opens the doors for everyday objects to be as an important part of the internet by allowing them to communicate with each other and work closely in order to provide easier and smarter life. To achieve this vision in a full-scale, many IoT challenges and issues have to be understood and addressed by different research communities. This paper contributes to the understanding of them along with the major efforts introduced to address them by providing a comprehensive survey upon most important studies. However, it is encouraged to do further research in this respect to expand and deepen the scope of this work and to raise the need of more efforts from both industry and academia to tackle IoT challenges and issues in order to promote the progress of the IoT in our practical life. For example, the challenges and issues of implementing the IoT using modern wearable devices, biomedical devices, vehicles, etc. or identifying the limitations of using not standardised IoT solutions that diminish the IoT reliability could be a good choice for a further study. Finally, the author hopes that this paper will be a very useful resource for researchers and academics to approach this field of study and research by assisting them to understand the IoT from different perspectives and also by providing them the prominent citations that will assist them in their further study of this hot topic.

References

Abdmeziem, R. and Tandjaoui, D. (2014) 'internet of things: concept, building blocks, applications and challenges', *Computers and Society*, arXiv preprint arXiv:1401.6877.

Aberer, K., Hauswirth, M. and Salehi, A. (2006) *Global Sensor Networks*, Tech. Rep. LSIR-REPORT-2006-001, 2006, Ecole Polytechnique Fdrale de Lausanne (EPFL), Lausanne, Switzerland.

Ahmad, M.W. and Ansari, M.A. (2012) 'A survey: soft computing in intelligent information retrieval systems', *The 12th International Conference on Computational Science and Its Applications (ICCSA)*, pp.26–34.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M. (2015) 'internet of things: a survey on enabling technologies, protocols, and applications', *IEEE Communications Surveys and Tutorials*, Vol. 17, No. 4, pp.2347–2376.

Alghamdi, T.A., Lasebae, A. and Aiash, M. (2013) 'Security analysis of the constrained application protocol in the internet of things', *The 2nd International Conference on Future Generation Communication Technologies (FGCT)*, pp.163–168.

Alsaadi, E. and Tubaishat, A. (2015) 'internet of things: features, challenges, and vulnerabilities', *International Journal of Advanced Computer Science and Information Technology (IJACSIT)*, Vol. 4, No. 1, pp.1–13.

Alur, R., Berger, E., Drobnis, A.W., Fix, L., Fu, K., Hager, G.D., Lopresti, D., Nahrstedt, K., Mynatt, E., Patel, S., Rexford, J., Stankovic, J.A. and Zorn, B. (2015) 'Systems computing challenges in the internet of things', *Computing Community Consortium (CCC)*, pp.1–15.

Atzori, L., Iera, A. and Morabito, G. (2010) 'The internet of things: a survey', *Computer Networks: The International Journal of Computer and Telecommunication Networking*, Vol. 54, No. 15, pp.2787–2805.

Avižienis, A., Laprie, J.C., Randell, B. and Landwehr, C. (2004) 'Basic concepts and taxonomy of dependable and secure computing', *IEEE Transactions on Dependable and Secure Computing (TDSC)*, Vol. 1, No. 1, pp.11–33.

Bandyopadhyay, D. and Sen, J. (2011) 'internet of things: applications and challenges in technology and standardization', *Wireless Personal Communications*, Vol. 58, No. 1, pp.49–69.

Bao, F. and Chen, I.R. (2012) 'Trust management for the internet of things and its application to service composition', *The IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp.1–6.

Bellavista, P., Cardone, G., Corradi, A. and Foschini, L. (2013) 'Convergence of MANET and WSN in IoT urban scenarios', *IEEE Sensors Journal*, Vol. 13, No. 10, pp.3558–3567.

Biewald, L. (2015) *Why Human-in-the-Loop Computing is the Future of Machine Learning* [online] http://www.computerworld.com/article/3004013/robotics/why-human-in-the-loop-computing-is-the-future-of-machine-learning.html.

Blaze, M., Feigenbaum, J. and Lacy, J. (1996) 'Decentralized trust management', *IEEE Symposium on Security and Privacy*, pp.164–173.

Bormann, C., Castellani, A.P. and Shelby, Z. (2012) 'CoAP: an application protocol for billions of tiny internet nodes', *IEEE internet Computing*, Vol. 16, No. 2, pp.62–67.

Borthakur, D., Gray, J., Sarma, J.S., Muthukkaruppan, K., Spiegelberg, N., Kuang, H., Ranganathan, K., Molkov, D., Menon, A. and Rash, S. (2011) 'Apache Hadoop goes realtime at Facebook', *ACM SIGMOD International Conference on Management of Data*, pp.1071–1080.

Broenink, G., Hoepman, J-H., Hof, C.V.T., Van Kranenburg, R., Smits, D. and Wisman, T. (2010) 'The privacy coach: supporting customer privacy in the internet of things', *Proceeding of Workshop on What Can internet Things Do for the Citizen? (CIOT)*, Radboud University, pp.1–10.

Butzin, B., Golatowski, F. and Timmermann, D. (2016) 'Microservices approach for the internet of things', *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*.

Cassou, D., Bruneau, J., Consel, C. and Balland, E. (2012) 'Toward a tool-based development methodology for pervasive computing applications', *IEEE Transactions on Software Engineering (TSE)*, Vol. 38, No. 6, pp.1445–1463.

Chan, H. and Perrig, A. (2003) 'Security and privacy in sensor networks', *IEEE Computer*, Vol. 36, No. 10, pp.103–105.

Chang, K., Yau, N., Hansen, M. and Estrin, D. (2006) 'SensorBase.org: a centralized repository to slog sensor network data', *The Euro-American Workshop on Middleware for Sensor Networks at the 2nd IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS/EAWMS)*.

Chen, D., Chang, G., Sun, D., Li, J., Jia, J. and Wang, X. (2011) 'TRM-IoT: a trust management model based on fuzzy reputation for internet of things', *Computer Science and Information Systems (ComSIS)*, Vol. 8, No. 4, pp.1207–1228.

Correia, L., Tran, T-D., Pereira, V., Silva, J.S. and Giacomin, J.C. (2014) 'Dynmac: a resistant mac protocol to coexistence in wireless sensor networks', *Elsevier Computer Networks*, Vol. 76, pp.1–16.

Da Xu, L., He, W. and Li, S. (2014) 'internet of things in industries: a survey', *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 4, pp.2233–2243.

Daubert, J., Wiesmaier, A. and Kikiras, P. (2015) 'A view on privacy & trust in IoT', *The IEEE International Conference on Communication Workshop (ICCW)*, pp.2665–2670.

Donovan, T.O., Brown, J., Roedig, U., Sreenan, C.J., Doo, J., Dunkels, A., Klein, A., Silva, J.S., Vassiliou, V. and Wolf, L. (2010) 'Ginseng: performance control in wireless sensor networks', *The 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, pp.1–3.

Duquennoy, S., Grimaud, G. and Vandewalle, J.J. (2009) 'Smews: smart and mobile embedded web server', *The International Conference on Complex, Intelligent and Software Intensive Systems (CISIS)*, pp.571–576.

European Commission (2014) *No Dependability, No Internet of Things* [online] https://ec.europa.eu/digital-agenda/en/news/no-dependability-no-internet-things (accessed 5 September 2016).

Fernandes, J.L., Lopes, I.C., Rodrigues, J.J.P.C. and Ullah, S. (2013) 'Performance evaluation of RESTful web services and AMQP protocol', *The International Conference on Ubiquitous and Future Networks (ICUFN)*, pp.810–815.

Fielding, R.T. (2000) *Architectural Styles and the Design of Network-based Software Architectures*, PhD Thesis, University of California, Irvine, USA.

Fortino, G., Russo, W. and Savaglio, C. (2016) 'Agent-oriented modeling and simulation of IoT networks', *Proceedings of the Federated Conference on Computer Science and Information Systems*, Vol. 8, pp.1449–1452.

Frank, R., Bronzi, W., Castignani, G. and Engel, T. (2014) 'Bluetooth low energy: an alternative technology for VANET applications', *The 11th Annual Conference on Wireless On-Demand Network Systems and Services (WONS)*, pp.104–107.

Fruhwirth, T., Krammer, L. and Kastner, W. (2015) 'Dependability demands and state of the art in the internet of things', *The 20th Conference on Emerging Technologies and Factory Automation (ETFA)*, pp.1–4.

Fujita, S., Ishii, Y., Ooishi, K. and Yamaji, M. (2011) 'Identifying and verifying clock synchronization protocol parameters', *Workshop on internet of Things and Service Platforms (IoTSP)*, pp.1–8.

Gazis, V., Goertz, M., Huber, M., Leonardi, A., Mathioudakis, K., Wiesmaier, A. and Zeiger, F. (2015) 'Short paper: IoT: challenges, projects, architectures', *The 18th International Conference on Intelligence in Next Generation Networks (ICIN)*, pp.145–147.

Gibbons, P., Karp, B., Ke, Y., Nath, S. and Seshan, S. (2003) 'Irisnet: an architecture for a worldwide sensor web', *IEEE Pervasive Computing*, Vol. 2, No. 4, pp.22–33.

Gomez, C. and Paradells, J. (2010) 'Wireless home automation networks: a survey of architectures and technologies', *IEEE Communications Magazine*, Vol. 48, No. 6, pp.92–101.

Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013) 'internet of things (IoT): a vision, architectural elements, and future directions', *Future Generation Computer Systems*, Vol. 29, No. 7, pp.1645–1660.

Guinard, D., Trifa, V. and Wilde, E. (2010) 'A resource oriented architecture for the web of things', *The 2nd International Conference on the internet of Things (IoT)*, pp.1–8.

https://www.shodan.io/ (accessed 24 March 2016).

https://www.thingful.net/ (accessed 24 March 2016).

Huang, M., Li, J., Song, X. and Guo, H. (2012) 'Modeling impulsive injections of insulin: towards artificial pancreas', *SIAM Journal of Applied Mathematics*, Vol. 72, No. 5, pp.1524–1548.

Hunkeler, U., Truong, H.L. and Stanford-Clark, A. (2008) 'MQTT-S – a publish/subscribe protocol for wireless sensor networks', *The 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE)*, pp.791–798.

IEEE Standards (2013) *IEEE Standard for a Convergent Digital Home Network for Heterogeneous Technologies*, IEEE Standard 1905.1-2013, pp.1–93.

Jara, A.J., Martinez-Julia, P. and Skarmeta, A. (2012) 'Light-weight multicast DNS and DNS-SD (lmDNS-SD): IPv6-based resource and service discovery for the web of things', *The Sixth International Conference on Innovative Mobile and internet Services in Ubiquitous Computing (IMIS)*, pp.731–738.

Johnsen, F.T., Bloebaum, T.H., Avlesen, M., Spjelkavik, S. and Vik, B. (2013) 'Evaluation of transport protocols for web services', *Military Communications and Information Systems Conference (MCC)*, pp.1–6.

Juels, A. (2006) 'RFID security and privacy: a research survey', *IEEE Journal on Selected Areas in Communications (J-SAC)*, Vol. 24, No. 2, pp.381–394.

Karagiannis, V., Chatzimisios, P., Vazquez-Gallego, F. and Alonso-Zarate, J. (2015) 'A survey on application layer protocols for the internet of things', *Transaction on IoT and Cloud Computing (TICC)*, Vol. 1, No. 1, pp.1–10.

Katasonov, A., Kaykova, O., Khriyenko, O., Nikitin, S. and Terziyan, V. (2007) 'Smart semantic middleware for the internet of things', *The 5th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, pp.169–178.

Kay, M., Choe, E.K., Shepherd, J., Greenstein, B., Watson, N., Consolvo, S. and Kientz, J.A. (2012) 'Lullaby: A capture & access system for understanding the sleep environment', *The ACM Conference on Ubiquitous Computing (UbiComp)*, pp.226–234.

Keoh, S.L., Kumar, S.S. and Tschofenig, H. (2014) 'Securing the internet of things: a standardization perspective', *IEEE internet of Things Journal (IoT-J)*, Vol. 1, No. 3, pp.265–275.

Knierim, P., Nickels, J., Musiol, S., Könings, B., Schaub, F., Wiedersheim, B. and Weber, M. (2012) 'Find my stuff: a search engine for everyday objects', *The 11th International Conference on Mobile and Ubiquitous Multimedia (MUM).*

Kocher, I.S., Chow, C-O., Ishii, H. and Zia, T.A. (2013) 'Threat models and security issues in wireless sensor networks', *International Journal of Computer Theory and Engineering (IJCTE)*, Vol. 5, No. 5, pp.830–835.

Köhler, M., Wörner, D. and Wortmann, F. (2014) 'Platforms for the internet of things – an analysis of existing solutions', *The 5th Bosch Conference on Systems and Software Engineering (BoCSE).*

Koskela, M., Rahikainen, M. and Wan, T. (2007) *Software Development Methods: SOA vs. CBD, OO and AOP*, pp.1–16, Technical Report, Aalto University, Finland.

Kotis, K. and Katasonov, A. (2012) 'Semantic interoperability on the web of things: the semantic smart gateway framework', *The 6th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*, pp.630–635.

Krawczyk, H., Bellare, M. and Canetti, R. (1997) 'HMAC: keyed-hashing for message authentication', *IETF RFC 2104.*

Kulkarni, R.V., Förster, A. and Venayagamoorthy, G.K. (2011) 'Computational intelligence in wireless sensor networks: a survey', *IEEE Communications Surveys and Tutorials*, Vol. 13, No. 1, pp.68–96.

Kumar, R., Kohler, E. and Srivastava, M. (2007) 'Harbor: software-based memory protection for sensor nodes', *The 6th International Symposium on Information Processing in Sensor Networks (IPSN)*, pp.340–349.

Kushalnagar, N., Montenegro, G. and Schumacher, C. (2007) 'IPv6 over low-power wireless personal area networks (6LoWPANs): overview', assumptions, problem statement, and goals, *IETF RFC 4919.*

Lasassmeh, S.M. and Conrad, J.M. (2010) 'Time synchronization in wireless sensor networks: a survey', *The IEEE SoutheastCon Conference*, pp.242–245.

Lathia, N., Rachuri, K.K., Mascolo, C. and Rentfrow, P.J. (2013) 'Contextual dissonance: design bias in sensor-based experience sampling methods', *The ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, pp.183–192.

Lee, S., Kim, H., Hong, D.K. and Ju, H. (2013) 'Correlation analysis of MQTT loss and delay according to QoS level', *The International Conference on Information Networking (ICOIN)*, pp.714–717.

Lenzen, C., Sommer, P. and Wattenhofer, R. (2009) 'Optimal clock synchronization in networks', *The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pp.225–238.

Lerche, C., Hartke, K. and Kovatsch, M. (2012) 'Industry adoption of the internet of things: a constrained application protocol survey', *The 17th Conference on Emerging Technologies and Factory Automation (ETFA)*, pp.1–6.

Locke, D. (2010) *MQ Telemetry Transport (MqTT) V3.1 Protocol Specification*, IBM Developer Works Technical Library [online] http://www.Ibm.Com/Developerworks/Webservices/Library/Ws-Mqtt/Index.Html (accessed 1 April 2016).

Lv, J., Yuan, X. and Li, H. (2011) 'A new clock synchronization architecture of network for internet of things', *The International Conference on Information Science and Technology (ICIST)*, pp.685–688.

Macedo, D., Guedes, L.A. and Silva, I. (2014) 'A dependability evaluation for internet of things incorporating redundancy aspects', *The 11th International Conference on Networking, Sensing and Control (ICNSC)*, pp.417–422.

Mayer-Schönberger, V. (2009) *Delete: The Virtue of Forgetting in the Digital Age*, Princeton University Press, New Jersey, USA.

Mills, D. (1991) 'internet time synchronization: the network time protocol', *IEEE Transactions on Communications*, Vol. 39, No. 10, pp.1482–1493.

Miorandi, D., Sicari, S., De Pellegrini, F. and Chlamtac, I. (2012) 'internet of things: vision, applications and research challenges', *Ad Hoc Networks*, Vol. 10, No. 7, pp.1497–1516.

Mukherjee, A., Paul, H.S., Dey, S. and Banerjee, A. (2014) 'ANGELS for distributed analytics in IoT', *IEEE World Forum on internet of Things (WF-IoT)*, pp.565–570.

Munir, S., Stankovic, J., Liang, C. and Lin, S. (2013) 'New cyber physical system challenges for human-in-the-loop control', *The 8th International Workshop on Feedback Computing*.

National Intelligence Council (NIC) (2008) *Disruptive Civil Technologies: Six Technologies with Potential Impacts on US Interests Out to 2025*, Conference Report CR 2008-07 [online] https://www.fas.org/irp/nic/ (accessed 17 May 2016).

Oleshchuk, V. (2009) 'internet of things and privacy preserving technologies', *Proceeding of the 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (VITAE)*, pp.336–340.

Olurin, M., Adams, C. and Logrippo, L. (2012) 'Platform for privacy preferences (P3P): current status and future directions', *The Tenth Annual International Conference on Privacy, Security and Trust (PST)*, pp.217–220.

Palattella, M.R., Accettura, N., Vilajosana, X., Watteyne, T., Grieco, L.A., Boggia, G. and Dohler, M. (2013) 'Standardized protocol stack for the internet of (important) things', *IEEE Communications Surveys and Tutorials*, Vol. 15, No. 3, pp.1389–1406.

Pokorný, J. (2004) 'Web searching and information retrieval', *Computing in Science and Engineering (CiSE)*, Vol. 6, No. 4, pp.43–48.

Radomirovic, S. (2010) 'Towards a model for security and privacy in the internet of things', *The 1st International Workshop Security of the internet of Things (SecIoT)*, Network Information and Computer Security Laboratory, pp.1–6.

Ravi, D., Wong, C., Lo, B. and Yang, G-Z. (2017) 'A deep learning approach to on-node sensor data analytics for mobile or wearable devices', *IEEE Journal of Biomedical and Health Informatics*, Vol. 21, No. 1, pp.56–64.

Rawat, P., Singh, K., Chaouchi, H. and Bonnin, J. (2013) 'Wireless sensor networks: a survey on recent developments and potential synergies', *The Journal of Supercomputing*, Vol. 66, No. 1, pp.1–48.

RELYonIT (2012–2015) *Research by Experimentation for Dependability on the internet of Things* [online] http://www.relyonit.eu/ (accessed 10 August 2016).

Reynolds, V., Cahill, V. and Senart, A. (2006) 'Requirements for an ubiquitous computing simulation and emulation environment', *The First International Conference on Integrated internet Ad Hoc and Sensor Networks*.

Roman, R., Najera, P. and Lopez, J. (2011) 'Securing the internet of things', *IEEE Computer*, Vol. 44, No. 9, pp.51–58.

Said, O. and Masud, M. (2013) 'Towards internet of things: survey and future vision', *International Journal of Computer Networks (IJCN)*, Vol. 5, No. 1, pp.1–17.

Saint-Andre, P. (2011) 'Extensible messaging and presence protocol (XMPP): core', *internet Engineering Task Force (IETF), Request for Comments: 6120* [online] https://tools.ietf.org/html/rfc6120 (accessed 1 February 2016).

Sakamura, K. (2006) 'Challenges in the age of ubiquitous computing: a case study of t-engine – an open development platform for embedded systems', *The 28th International Conference on Software Engineering (ICSE)*, pp.713–720.

Sandhu, R., Coyne, E., Feinstein, H. and Youman, C. (1996) 'Role-based access control models', *IEEE Computer*, Vol. 29, No. 2, pp.38–47.

Santanche, A., Nath, S., Liu, J., Priyantha, B. and Zhao, F. (2006) 'Senseweb: browsing the physical world in real time', *The 5th International Conference on Information Processing in Sensor Networks (IPSN)*.

Santucci, G. (2009) 'internet of the future and internet of things: what is at stake and how are we getting prepared for them?', *Future internet Workshop*, Oslo, Norway.

Sarhan, Q.I. (2013) 'Security attacks and countermeasures for wireless sensor networks: survey', *International Journal of Current Engineering and Technology (IJCET)*, Vol. 3, No. 2, pp.628–635.

Schirner, G., Erdogmus, D., Chowdhury, K. and Padir, T. (2013) 'The future of human-in-the-loop cyber-physical systems', *Computer*, Vol. 46, No. 1, pp.36–45.

Serral, E., Valderas, P., and Pelechano, V. (2010) 'Towards the model driven development of context-aware pervasive systems', *Pervasive and Mobile Computing*, Vol. 6, No. 2, pp.254–280.

Shelby, Z. (2012) *Constrained RESTful Environments (CoRE) Link Format CoRE*, IETF Standard, RFC 6690 [online] https://tools.ietf.org/html/rfc6690 (accessed 10 January 2016).

Singh, V.K., Kushwaha, D.S., Singh, S. and Sharma, S. (2015) 'The next evolution of the internet … internet of things', *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)*, Vol. 2, No. 1, pp.31–35.

Sousa Nunes, D.S., Zhang, P. and Sa Silva, J. (2015) 'A survey on human-in-the-loop applications towards an internet of all', *IEEE Communications Surveys and Tutorials*, Vol. 17, No. 2, pp.944–965.

Stankovic, J. (2014) 'Research directions for the internet of things', *IEEE internet of Things Journal (IoT-J)*, Vol. 1, No. 1, pp.3–9.

Steven, J. (2013) *Google Moves Away from the XMPP Open-Messaging Standard* [online] http://www.zdnet.com/article/google-moves-away-from-the-xmpp-open-messaging-standard/ (accessed 19 Jan 2015).

Sulistyo, S. (2013) 'Software development methods in the internet of things', *The International Conference on Information and Communication Technology (ICT-EurAsia)*, Vol. LNCS 7804, pp.50–59.

Tan, L. and Wang, N. (2010) 'Future internet: the internet of things', *The 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, Vol. 5, No. 15, pp.376–380.

Thangavel, D., Ma, X., Valera, A., Tan, H.X. and Tan, C.K.Y. (2014) 'Performance evaluation of MQTT and CoAP via a common middleware', *The Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pp.1–6.

The GS1 EPCglobal Architecture Framework, GS1 Version 1.6 (2014) [online] http://www.gs1.org/sites/default/files/docs/epc/architecture_1_6-framework-20140414.pdf (accessed 14 April 2014).

The internet Society (ISOC) (2015) *The internet of Things: An Overview – Understanding the Issues and Challenges of a More Connected World*, White Paper, pp.1–53 [online] http://www.internetsociety.org/doc/iot-overview (accessed 20 January 2016).

Thompson, C. (2009) *25 Ideas for 2010: Digital Forgetting*, Wired UK [online] http://www.wired.co.uk/magazine/archive/2009/12/features/25-ideas-for-2010-digital-forgetting (accessed 15 February 2016).

Tokuda, H., Yonezawa, T. and Nakazawa, J. (2014) 'Monitoring dependability of city-scale IoT using D-case', *IEEE World Forum on internet of Things (WF-IoT)*, pp.371–372.

Toma, I., Simperl, E. and Hench, G. (2009) 'A joint roadmap for semantic technologies and the internet of things', *The 3rd STI Roadmapping Workshop*.

Tsai, C., Lai, C., Chiang, M. and Yang, L.T. (2014) 'Data mining for internet of things: a survey', *IEEE Communications Surveys and Tutorials*, Vol. 16, No. 1, pp.77–97.

Van den Heuvel, W.J., Zimmermann, O., Leymann, F., Lago, P., Schieferdecker, I., Zdun, U. and Avgeriou, P. (2009) 'Software service engineering: tenets and challenges', *The ICSE Workshop on Principles of Engineering Service Oriented Systems*, pp.26–33.

Vazquez, I. (2009) 'Social devices: semantic technology for the internet of things', *Week at ESI*.

Wahlster, W. (2008) 'Web 3.0: semantic technologies for the internet of services and of things', *Lecture at the Dresden Future Forum*.

Wang, X., Gu, W., Schosek, K., Chellappan, S. and Xuan, D. (2004) *Sensor Network Configuration under Physical Attacks*, Technical Report (OSU-CISRC-7/04-TR45), Department of Computer Science and Engineering, Ohio State University, Ohio, USA.

Weiser, M. (1999) 'The computer for the 21st century', *ACM Mobile Computing and Communications Review*, Vol. 3, No. 3, pp.3–11.

Wickramasuriya, J., Datt, M., Mehrotra, S. and Venkatasubramanian, N. (2004) 'Privacy protecting data collection in media spaces', *The 12th Annual ACM International Conference on Multimedia*, pp.1–48.

Wood, A.D. and Stankovic, J.A. (2008) 'Human in the loop: distributed data streams for immersive cyber-physical systems', *ACM SIGBED Review*, Vol. 5, No. 1, pp.1–2.

Xu, X., Huang, S., Chen, Y., Browny, K., Halilovicy, I. and Lu, W. (2014) 'TSAaaS: time series analytics as a service on IoT', *International Conference on Web Services (ICWS)*, pp.249–256.

Yang, D-L., Liu, F. and Liang, Y-D. (2010) 'A survey of the internet of things', *The 1st International Conference on E-Business Intelligence (ICEBI)*, pp.358–366.

Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M. (2014) 'internet of things for smart cities', *IEEE internet of Things Journal (IoT-J)*, Vol. 1, No. 1, pp.22–32.

Zeng, D., Guo, S. and Cheng, Z. (2011) 'The web of things: a survey (invited paper)', *Journal of Communications*, Vol. 6, No. 6, pp.424–438.

Zhao, K. and Ge, L. (2013) 'A survey on the internet of things security', *The Ninth International Conference on Computational Intelligence and Security (ICCIS)*, pp.663–667.

Zia, T.A. and Zomaya, A.Y. (2006) 'Security issues in wireless sensor networks', *Proceeding of the International Conference on Systems and Networks (ICSNC)*, Tahiti, French Polynesia.