IoT Network Management: Content and Analysis

Jonathan de Carvalho Silva, Joel J. P. C. Rodrigues, and Mario Lemes Proença Jr.

***Abstract*—Currently, physical objects are integrated and connected in the networks to provide services for people performing the well-known Internet of Things (IoT) paradigm. IoT environments are characterized by a high degree of devices heterogeneity and network protocols, where each object may have different processing capabilities or different communication patterns, making a lack of standardization in IoT. This demands a dynamic and context-aware configuration management system. Currently, in the literature, there are no management platforms that address IoT issues, such as scalability, heterogeneity, and context-aware. For this purpose, the study elaborates on available policies and solutions for network management and devices in IoT, highlighting and discussing the features of each studied project, identifying open research issues on the topic.**

***Index Terms*—Network Management; Device Management; Internet of Things; IoT**

I. Introduction

Internet of Things (IoT) is one of the leading emerging technologies contributing to the realization of new information and communication technologies (ICTs) applications. IoT is considered an intelligent domain model without qualification of advanced communication technologies and sensitivity added to administrative bodies of cities (or communities) and their citizens [1].

The services offered by IoT can benefit a plethora of application areas. Among them, health services, infrastructures, and public sector services are very promising domains for IoT applications. Remote health monitoring, for example, makes a big difference in people lives (with chronic illnesses, for instance), while at the same time decreases costs of health care for these patients and improves their quality of life. Residential automation using IoT enables devices such as smart thermostats to adjust the ambient temperature, and lamps that are remotely controlled for safety and energy saving. However, IoT presents many challenges for managing until providing intelligent and integrated services anytime and anywhere.

One of the most important challenges in IoT managing is the devices heterogeneity that belongs the network, using many different technologies at each layer, as shown in Figure 1. Finally, factors such as the imprecision of the collected data (e.g., RFID systems can generate between 60 and 70% of incorrect data), high data in real time, and the implicit semantics impose challenges in the configuration of IoT environments [2].

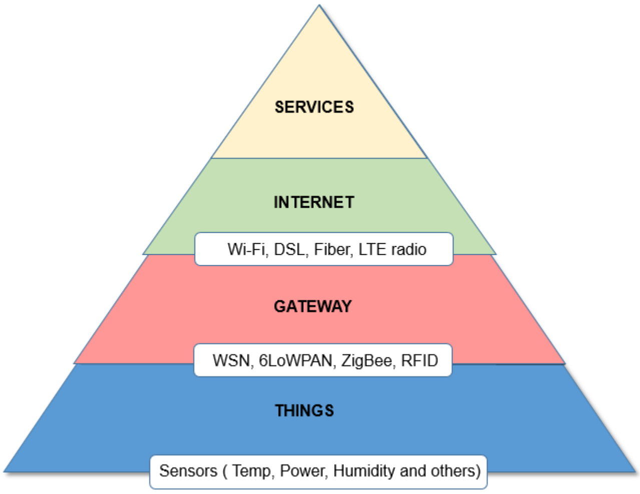
SERVICES
INTERNET
Wi-Fi, DSL, Fiber, LTE radio
GATEWAY
WSN, 6LoWPAN, ZigBee, RFID
THINGS
Sensors (Temp, Power, Humidity and others)

Fig. 1. Illustration of a layered IoT Structure.

IoT management is more complex than network management both in wireless sensors networks (WSNs) and IP networks. Moreover, in IoT networks, it is necessary to support network applications and services that involve *i*) the use of devices with different features; *ii*) the interaction between IoT networks, requiring local management (i.e., homeowner), and global management (i.e., companies). Both of them are context aware. The available IoT management architectures partially meet these requirements [3]. Then, this paper focuses on available policies, approaches, and solutions, including tools from networks management to devices management for IoT. Among the available technologies for these networks, it describes and performs a comparison study considering their heterogeneity, scalability, supported technologies, security, among others. Based this research study, the most promising technology was chosen for simulation and deployment in real environments, such as the living Lab of the Inatel smart campus project.

The remainder of this paper is organized as follows. The related work to networks and devices management for IoT is presented in Section II. Section III describes the network management for IoT and its main features. In the sequence, the network management protocols for IoT are addressed in Section IV while Section V considers devices management for IoT and its features. Section VI focuses on the devices management platforms for IoT. Finally, Section VII discusses the available solutions and identifies open research issues for future works and the paper is concluded in Section VIII.

Jonathan de Carvalho Silva and Joel J. P. C. Rodrigues¸ National Institute of Telecommunications (INATEL), Santa Rita do Sapucaí - MG, Brazil, E-mails: jonathancs@inatel.br, joeljr@ieee.org.

Joel J. P. C. Rodrigues¸ Instituto de Telecomunicações, Universidade da Beira Interior, Portugal; University of Fortaleza (UNIFOR), Brazil.

Mario Lemes Proença Jr., State University of Londrina, Brazil, E-mail: proenca@uel.br.

II. Related Work

The management of an IoT platform is proposed and classified into two main categories: 1) Network Management and 2) Device Management, which is elaborated in this section.

A. Network Management

Network management has as a requirement that includes handling large volumes of data for an IoT platform, allowing it with a demand for data collection and analysis and, consequently, to provide answers, decisions, and/or actions in a fast and efficient way. The most relevant approaches are described in this section.

NETCONF – This type of network management adopts the Manager-Agent model, which defines a protocol used for devices in IoT [4]. Management Information Base (MIB) is a set of Managed Objects (MOs) where each MO represents a Managed Resource. New solutions about network management should be adapted to development requirements of IoT. Interconnecting smart devices with IP is a prospective direction and IP standardizations can be utilized for IoT integrated management. The NETCONF is the new standard network management, that overcomes the weaknesses of the Simple Network Management Protocol (SNMP), and provides a better configuration of IP network devices due to the effective use of XML (eXtended Markup Language) and related technologies.

COMAN – There are several candidates for COMAN technologies, but in this work it is limited to OMA-LwM2M (Open Mobile Alliance for Lightweight M2M or IoT device management). OMA Lightweight M2M aims to provide an protocol sub-layer adjacent to enable management of M2M services between the LwM2M server and LwM2M client. Nowadays, the first version provided a protocol for devices, that can meet various management requirements. [5]. There are several OMA-LwM2M with COMAN requirements, such as energy states and appliances monitoring, logging, system authentication, and peripheral management and access controls to the system and peripheral management.

B. Device Management

Device Management corresponds to the ability to provide device location and status information, allowing, among other features, to disconnect some stolen or unrecognized devices, update embedded software, modify security settings, modify hardware configurations remotely, locate a lost device, delete sensitive data from devices, and even enable interaction between devices.

RestThings – Similar the EcoDiF, RestThing [6] is a REST-based Web services infrastructure that aims to hide device heterogeneity and provide a way to integrate devices into Web applications. The platform aims to enable developers to build applications using REST principles, combining physical and Web resources. Devices and Web information are both represented as resources and handled by a uniform interface in the REST style. The RESTful API allows transmitting data between sensors that use IP, gateways, Web server, and Web applications. RestThing works with three types of data formats, namely JSON, XML, and CSV.

Xively – Xively platform [7] uses cloud services to manage data provided by devices. The platform provides an API for collecting data from sensors, thus allowing the visualization of historical data and providing mechanisms to trigger events based on the data generated by the sensors (triggers). Similar to EcoDiF, Xively is based on REST principles and Web standards, such as HTTP and URIs (Uniform Resource Identifiers). In this way, the platform provides well-defined and standardized interfaces, minimizing incompatibility problems between different devices. It is a commercial and a closed source solution created in 2016.

III. Network Management for IoT

IoT devices management needs to adapt the dynamic and often unknown topology of these networks. For example, providing devices location and status information. For Delicato *et al.* [8], management should also consider the possibility of devices being integrated into an environment, used opportunistically, and not previously planned. Thus, it is important that a management platform enables the devices discovery in the considered environment, in a dynamic way, to meet the requirements of the applications.

The network management is based on the following five functional components: configuration, failure, member, report, and state:

- The configuration (self-configuration) is responsible for performing system configuration initialization functions, such as collecting and storing the configurations of the other functional components and devices.

- The failure (self-aware) aims to identify, isolate, correct, and record failures that occur in the IoT system.

- The member is responsible for handling member associations of the IoT system and important information from any relevant entity (IoT service, device, applications, user).

- The report allows the information refining provided by other management functions, generating reports or retrieving reports from a history.

- The state (self-monitoring) aims to monitor and provide the past, present, and future states of the IoT system that are required by the function *Fail*. It has a function to change or apply for a particular state in the system.

The IoT network management solutions should have several requirements that are reported in [9]. Interoperability should be performed between the different devices and platforms available in the environment. Discovery and management of devices available in the considered environment are performed dynamically to meet the application requirements. Context-aware where information, such as location and state of network objects, is used to perform actions or react to an incentive. Scalability to support the network enlargement and working properly even in situations of intense use. Security and dynamic adaptation to maintain the data integrity and privacy, turn them available, and guarantee the availability and quality of the applications during their execution.

IV. Network Management Protocols

A. Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) [10] is a TCP/IP protocol and the most used since it is simple and easy to deploy. This protocol was designed in the early 1980s with the purpose to provisionally solving the communication problems between heterogeneous networks.

The success of SNMP comes from the fact that it was the first non-proprietary, public management protocol that is easy to deploy and enables effective heterogeneous environments. There are three main versions of this protocol. The second version offers support for an efficient transfer of large blocks of data and centralized network management guidance, problems that were not addressed in the first version. Another issue that was not supported in the SNMP version 1 is security, which is the main goal of the SNMP version 3 [11].

An SNMP device can be anyone that it is connected with others devices, performing a machine-to-machine communication. The SNMP Agent is a software for network management that is installed in a connected device. It responds to queries from SNMP managers and send a trap message to the manager when occurred specified events. The MIB is a virtual database organized through a tree structure where there are object identifiers (OIDs) with the objective of keep information about devices management in a communication network.

Figure 2 illustrates the SNMP manager and an agent uses a MIB and five basic commands (GET, GET-RESPONSE, SET, SET-RESPONSE, and TRAP) to exchange information among them.

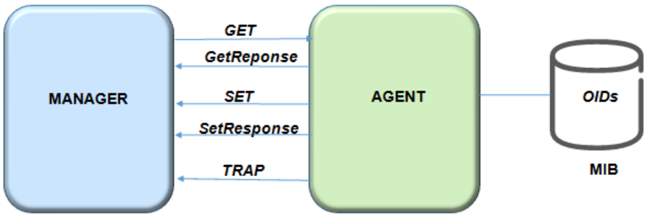

Fig. 2. Communication Scheme of the SNMP protocol.

B. Internet of Things Platform's Infrastructure for Configurations (IoT-PIC).

The Internet of Things Platform's Infrastructure for Configurations (IoT-PIC) have rule to provide a unique and general way to perform the commissioning of the platform. The architecture of the IoT-PIC is shown in Figure 3. This architecture is inspired by the SNMP.v1 (described in the previous section) and aims to perform the configuration and composition of hardware and software resources.

The IoT-PIC architecture considers the global and local levels composed of two components:

- An IoT-PIC Manager (PIC-M) responsible for the management of the composition stage at a global level.

- An IoT-PIC Agent (PIC-A) responsible for handling the interconnection and register the component at local level.

This work aims to evaluate the Extensible Messaging and Presence Protocol (XMPP) protocol as a valid alternative for network management in context IoT. XMPP – an Internet Engineering Task Force (IETF) standard also known as Jabber – is a protocol based on XML for real-time messaging, for exchange presence information and request-response services. The XMPP have high performance in terms of latency, scalability, and robustness [12]. It is free and open-source protocol.

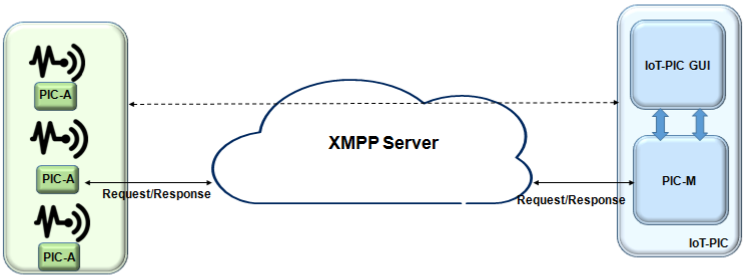

Fig. 3. Illustration of the IoT-PIC architecture.

V. IoT Devices Management

Devices management are addressed by two components: Device Manager and Device Agent. The Device Manager is a server-side that communicates through protocols with devices and provides both individual or bulk device control. It also manages the software and application deployment remotely on the device and can lock and/or clean the device, if needed.

The Device Manager works in conjunction with the Device Agent and there are different agents for different types of platforms and devices. The Device Manager also needs to maintain the identity list of the devices and map them to their owners. It must also work with the access and identity management layer to manage access control over the devices.

The Device Manager agent supports managing the installed software, enabling/disabling device functions, managing security controls and identifiers, monitoring device availability, keeping track of device location, if available, locking or cleaning the device remotely if it is compromised, among others. Unmanaged devices can communicate with the rest of the network, but there is no agent involved. Semi-managed devices are those that implement some parts of the Device Manager, for example, such as feature control, but not software management.

VI. Device Management Platforms

A. Management for Internet of Things (ManIoT)

This section describes the ManIoT (Management for Internet of Things) platform. It manages devices that make-up the IoT environments. An environment corresponds to a domain of applications and sensors installed physically in this environment. The ManIoT platform also takes into account the heterogeneity of the devices or ”things”. So ManIoT does not require major modifications or installation of additional software on network devices or on user devices. Application access to the ManIoT platform is performed through a Web user interface.

The ManIoT specifies a data model and information to standardize the data format used in the communication between applications, services, and devices. The status of the devices (on/off) and the *id* (identification device) are examples of features used in the information model. It also aims to be scalable and supports the integration with other systems since the platform makes use of popular protocols and standards for industry data models, such as XML and RestFul.

ManIoT establishes two management scopes, Local and Global/Remote, illustrated in Figure 4.

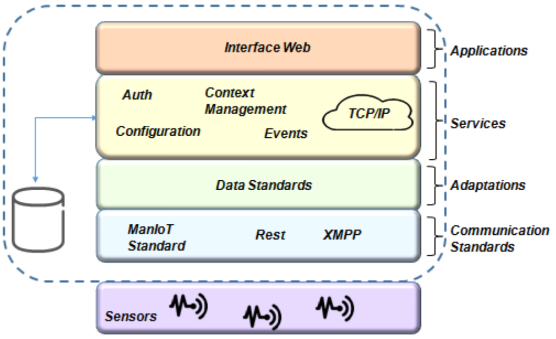

Fig. 4. ManIoT: Local/Global Management layered architecture.

The Local manager acts within a scenario, managing the devices that make up this scenario from information about the context. In this way, for example, the local manager can control the events that an application or user can perform, such as turning on or off a lamp. The Global/Remote manager seeks to standardize the actions performed in different scenarios from high-level directives. Thus, for instance, an energy concessionaire could define maximum consumption rates by area or residence in periods of potential blackouts using the global manager.

B. *SmartThings*

The open-source platform SmartThings [14] allows users to build applications and connect them to devices, actions and services offered by the platform. SmartThings also enables the integration of new devices and provides support for applications (SmartApps) communicating with external Web services by sending Push notifications, SMS, and the presentation of your REST terminal. Figure 5 illustrates the SmartThings architecture.

Figure 5 shows the infrastructure blocks of the devices illustrating the SmartThings architecture. The Hub provides communication between the "things" (sensors and actuators, for instance) and the applications. When the SmartThings system receives the messages and, these are analyzed, identifies the type of used device based on the Device Handler. Its output is performed through SmartThings events. The Subscription Management has the purpose to perform corresponding events of the Device Handlers with the SmartApp that is using them.

VII. Discussion and Open Issues

Based on the main features of the studied management solutions, a qualitative study is performed. Two tables were created in order to characterize the most relevant management protocols and platforms. Table I summarizes the main features of the protocols considering their standardization process, resources, data, encoding, and transport stack.

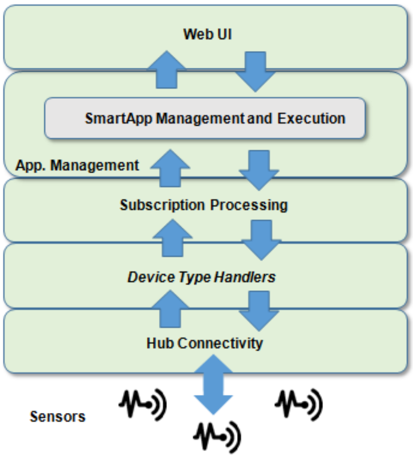

Fig. 5. Illustration of the SmartThings architecture.

TABLE I
Main features comparison between the most relevant network management protocols for IoT.

	SNMP/LNMP	NETCONF	XMPP
Standard	IETF	IETF	IETF
Resource	OIDs	Paths	URLs
Data Modeling	SMI	YANG	WSDL
Encoding	BER	XML	XML
Transport Stack	UDP	SSH/TCP	HTTP, WebSocket

Several solutions are based on SNMP protocol as may be seen in [15] [16], and new solutions are using SNMP architecture as a base to develop new solutions like the use of XMPP protocol shown in IoT-PIC. A study presented in [13] concluded that SNMP makes more efficient use of resources, responding to a processing requests up to ten times faster than NETCONF.

Table II summarizes a comparison among the most relevant management platforms for IoT, considering RestThing, SmartThings, and ManIoT, regarding several important network technologies and protocols, and types of management approaches.

It can be observed that no IoT device management platform was able to meet all the raised requirements. Such platforms only treat subsets of requirements, addressing them in different ways. Interoperability is an example. Despite being attended by all, SmartThings believes that the use of widely used protocols and Web technologies are sufficient to mitigate the problems of heterogeneity among devices, while platforms like Carriots and Xively believe that support for other protocols is important. Other requirements, such as contextual science and dynamic adaptation are little discussed. Contextual science is approached by many platforms regarding the inclusion of semantic data, such as location, collection time and others, together with the collected data.

TABLE II
MAIN FEATURES COMPARISON BETWEEN AVAILABLE MANAGEMENT PLATFORMS FOR IOT.

	RestThing	SmartThings	ManIoT
Heterogeneity	X	X	X
Security and Privacy		X	X
Scalability	X	X	X
Interoperability	X	X	X
SNMP			
NETCONF			X
6LoWPAN			X
Device Management	X	X	X
Local Management		X	X
Global Management	X	X	X
Remote Management	X	X	X

It can be concluded that the state of the art on Devices Management for IoT stills in an early stage, where the most relevant requirements have not been completely explored. Nevertheless, ManIoT platform can be considered the best solution according to the mapped features. There are some open research issues since technologies and available approaches still diverge, and there is no solution capable to cover all the necessary requirements for a reference architecture.

VIII. CONCLUSION

The large number of devices resulting from IoT, naturally, demand management and control solutions for the various services and, therefore, there is a need for platforms that integrate these services. Through management, the capabilities offered by IoT devices can be used to provide services that will serve a myriad of applications. However, the available IoT management platforms partially meet the requirements defined in the literature.

This work addressed the available policies, approaches and solutions (including tools), from network management to device management for IoT. For network management, many IoT network solutions continue based on SNMP protocol, providing management support for any platform always looking to improve the latency, scalability, and robustness. For IoT devices management, ManIoT technology was compared with other concurrent technologies and it was concluded that ManIoT predicts scalability and promotes the integration of multiple devices featuring local/remote management, heterogeneity and security. ManIoT is generic and can be used in a plethora of scenarios.

ACKNOWLEDGMENTS

This work has been partially supported by National Funding from the FCT - Fundação para a Ciência e a Tecnologia through the UID/EEA/500008/2013 Project, by Finatel through the Inatel Smart Campus project, and by Finep, with resources from Funttel, grant no. 01.14.0231.00, under the Radiocommunication Reference Center (Centro de Referência em Radiocomunicações — CRR) project of the National Institute of Telecommunications (Instituto Nacional de Telecomunicações — Inatel), Brazil.

REFERENCES

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista & M. Zorzi, "Internet of Things for smart cities", *IEEE Internet of Things Journal*, vol. 1, February 14, 2014; pp. 22-3, DOI: 10.1109/JIOT.2014.2306328.

[2] M. Meng, W. Ping & C. Chao-Hsien, "Data Management for Internet of Things: Challenges, Approaches and Opportunities", *IEEE and Internet of Things Conference (iThings/CPSCom 2013)*, Aug. 20-23, 2013, DOI: 10.1109/GreenCom-iThings-CPSCom.2013.199.

[3] C. MacGillivray, "The Platform of Platforms in the Internet of Things". *IBM: White Paper*, February 2016.

[4] H. Xu, C. Wang, W. Liu, & H. Chen, "Applying the Extension Model to Management of Smart Objects Hui", *International Journal of Hybrid Information Tecnology*, vol. 7, March 2014; pp.113-122 DOI: 10.14257/ijhit.2014.7.2.12.

[5] B. Greevenbosch & P. Van der Stok, "Internet-Draft, COMAN - Candidate Technologies.", *IETF*, 2013. Available online: https://tools.ietf.org/html/draft-greevenbosch-coman-candidate-tech-03 (accessed April 17, 2017).

[6] Qin, W.; Li, Q.; Sun, L.; Zhu, H. and Liu, Y., "RestThing: A restful Web service infrastructure for mash-up physical and web resources", *IFIP 9th International Conference on Embedded and Ubiquitous Computing (EUC 2011)*, Melbourne, Australia, October 24-26, 2011, pp. 197–204.

[7] LogMeIn, "IoT Platform for Connected Devices - Xively," 2015. Available online: https://www.xively.com/ (accessed April 17, 2017).

[8] F. C. Delicato, P. F. Pires, T. Batista, "Middleware solutions for the Internet of Things", *Springer Briefs in Computer Science*, 2013, DOI: 10.1007/978-1-4471-5481-5-1.

[9] P. F. Pires, E. Cavalcante, T. Barros, F. C. Delicato, T. Batista & B. Costa, "A platform for integrating physical devices in the Internet of Things", *12th IEEE International Conference on Embedded and Ubiquitous Computing* (EUC 2014), Milan, Italy, August 26-28, 2014; pp. 234-241 DOI: 10.1109/EUC.2014.42.

[10] Simple Network Management Protocol, *SNMP Research International*, 2017. Available online: http://www.snmp.com/protocol/ (accessed April 17, 2017).

[11] A. B. Ericsson, "Simple Network Management Protocol 5.2.5 (SNMP)", *Erlang*, March 14, 2017.

[12] XMPP performances. Available online: https://xmpp.org/ (accessed March 23, 2017).

[13] M. Santos, T. O. Castro, D. F. Macedo, B. Horizonte, "ManIoT : Uma Plataforma para Gerenciamento de Dispositivos da Internet das Coisas." *SBRC 2016*, Salvador-BA, Brasil, May 30 - June 03, 2016.

[14] SmartThings Developer Documentation, "Developer Documentation: Release 1.0", 2015. Available online: https://media.readthedocs.org/pdf/smartthings/latest/smartthings.pdf (accessed February 10, 2017).

[15] Neha, M. S. Meena & Rajbir "Implementation of SNMP (Simple Network Management Protocol) on Sensor Network", *International Journal of Advanced Research in Computer Engineering & Technology* (IJARCET), Vol. 5, no. 5, May 2016.

[16] H. Hui-ping, X. Shi-de, M. Xiang-yin, "Applying SNMP Technology to Manage the Sensors in Internet of Things", *The Open Cybernetics & Systemics Journal*, August 15, 2015, pp. 1019–1024.