DE GRUYTER OPEN
RIGA TECHNICAL UNIVERSITY
Information Technology and Management Science
ISSN 2255-9094 (online)
ISSN 2255-9086 (print)
December 2016, vol. 19, pp. 78–84
doi: 10.1515/itms-2016-0015
https://www.degruyter.com/view/j/itms

Internet of Things: Structure, Features and Management

Vladislavs Aleksandrovičs[1], Eduards Filičevs[2], Jānis Kampars[3]
[1–3] *Riga Technical University*

***Abstract* – Internet of Things (IoT) – a rapidly developing technology today and most likely everyday thing in the future. Numerous devices, computing machines and build-in sensors connected in a single dynamic network continuously receive and exchange information from the outer environment. Huge data clusters are collected and put to use in handmade applications that scrupulously manage and control given objectives. In this way, an interactive technical infrastructure is created, which can oversee and infiltrate any person's vital processes. Though separately every device and technological solution in the IoT can be known for many years, each architecture is unique and provides new challenges for the network owner. This research aims to investigate IoT general structure and management aspects with the knowledge of which the authors will try to answer a trivial question whether it is possible to comprehensively control such a tremendous structure with the current level of technology.**

***Keywords* – Data management, data storage, data transmission, Internet of Things, Web of Things.**

I. Introduction

The term "Internet of Things" (IoT herein) for the first time was mentioned by Kevin Ashton in 1999 representing supply chain management aspects to the public [1]. The idea was too intriguing to develop in such a narrow scope, so through the past decade it spread out covering a wide range of applications such as healthcare, utilities, finances, traffic, etc. Since then the word "things" has changed its meaning, but the main idea of the IoT remains the same – to organise a related communication environment where computing technologies will be able to communicate with one another, that users can adjust for their needs, but at the same time will work autonomously, gathering information from external sources.

The IoT comes in different sizes and shapes – ranging from a few sensors in a room to global structures covering entire countries. Respectively, some ideas, such as intelligent houses, become more and more popular allowing their owners not only to save funds in the future, but also to significantly simplify house management. Independent light intensity, air conditioning, household tool, door management and other control systems are only part of our own little IoT world [2]. Even a person that has never been interested in such technologies can encounter them on a daily basis, for example, by using payment cards while making a purchase, registering a trip in the public transport or other type of personal identification.

This type of dynamic communication in social surrounding is possible thanks to new technologies, such as radio-frequency identification (RFID), Bluetooth, Near Field Communication (NFC), Wireless networks (Wi-Fi), telephone services and local networking. Data exchange between two or more systems becomes a fully functioning Internet of future as the current radical Internet development trends lead us even further to the mutually connected object environments. The necessity of dull information systems that just collect information from external sources and interact with physical world (initialisation/ command/execution) gradually negates. Modern industries require powerful tools that can use existing Internet standards for continuous data transfer, analytical activities, related application planning and internal communication [1].

There is no doubt that technologies are moving forward and day after day handle complex tasks faster than ever before, but at a certain cost. One of the major and urgent issues of today's IoT is the difficulty of collecting and storing data generated by computing devices and sensors within the IoT architecture. Properly speaking, data diversity makes it difficult to organise a unified IoT management system and, taking in account a massive amount of generated information, burdens processes of computing machines.

The goal of the present research is to gain overall information about technologies related to the IoT and approve the possibility of creating centralised management and computing unit for IoT based on the aggregate data. Several topics have been investigated while moving towards the defined objective:

1) Overview of a typical IoT architecture.
2) Management and networking of IoT devices.
3) Implementation of data management among IoT devices.
4) Data storage in the IoT environment.
5) Main advantages and disadvantages of the IoT.
6) Future perspectives of IoT application and development.

II. IoT Architecture Overview

The IoT represents a hybrid architecture, which means it can contain different subsystem architectures. In most cases, the IoT systems are formed by two management architectures: event-driven and time-based. Event-driven architecture sensors transmit data when they sense activity in the external environment, for example, an alarm is triggered if the door is opened at night time. In the time-based architecture, its components continuously transmit data within a certain interval (e.g., climate control system sensors once per second read room temperature). The latter usually work repeatedly after a pause, which can be adjusted separately for each device or set up within a central management system that will send queries to endpoint devices and sensors after a period of time [3]. One such system solution is being offered by Intel Corporation (see Fig. 1) [4].

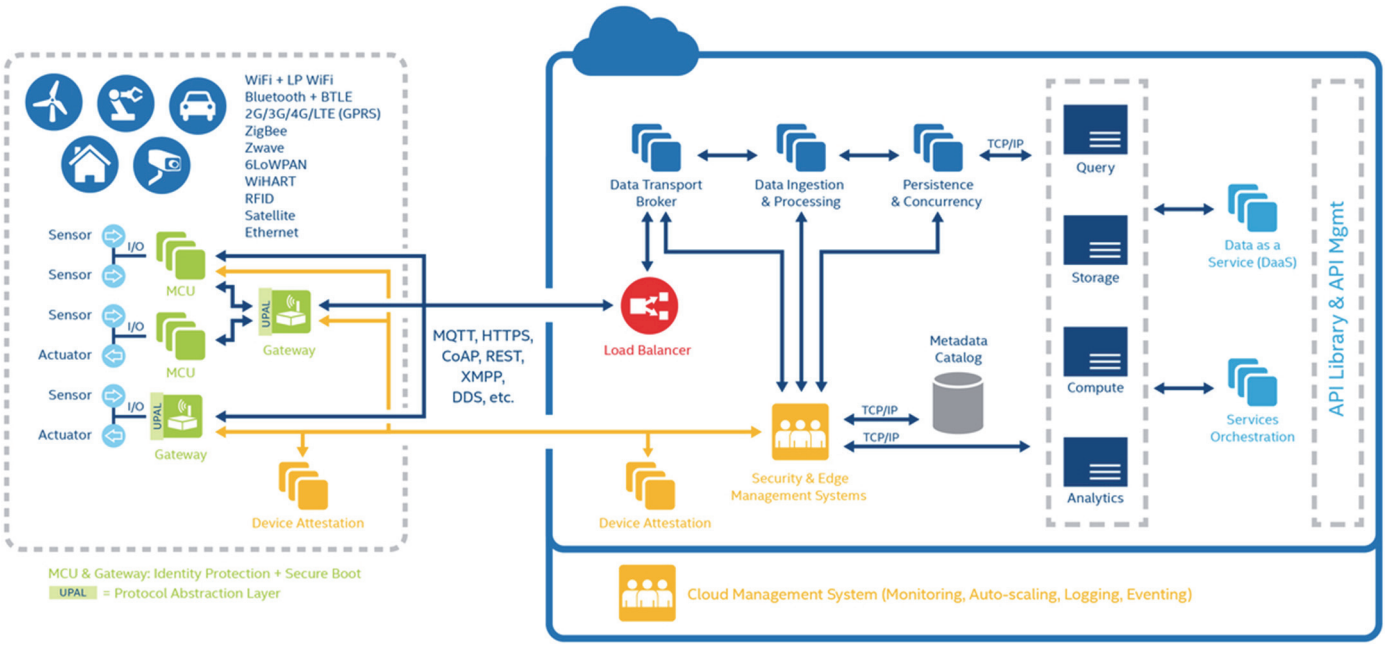

Fig. 1. Architecture of IoT central management system [4].

Network architecture can be divided into three topologies: point-to-point connection, star and mesh. Point-to-point network topology establishes connection for data transfer between two stations [5]. Simplicity is the advantage and, at the same time, limitation of current technology, guaranteeing low cost, but also depriving a possibility to reach devices outside the network. Star topology consists of multiple terminal nodes and only one central hub. All nodes can communicate with one another, but only by transmitting or receiving data through the central hub. Such a structure allows reaching low latency, high throughput and in a certain way protects the system from crashes when one of the nodes stops working. In comparison with a point-to-point connection range, a star-type network is still limited by the central hub, which is cut off from the global environment if it goes down. Mesh is a network topology that employs one of two decentralised connection arrangements: full mesh topology or partial mesh topology. In a full mesh topology, each network node (workstation or other devices) is connected to each of the others. In a partial mesh topology, some nodes are connected to all, then others are only connected to those nodes, with which they exchange the most data [6]. Mesh network is used for many applications requiring a long range and broad area coverage allowing one to build a network of nearly unlimited size. The disadvantage of this topology is its complexity comparing to point-to-point and star type networks, which can lead to high latency, more expenses and technical problems within a network.

III. Particular Qualities of IoT

The IoT application development is not part of this research as it may vary due to each developer's individual approach and style. Data transmission and storage in a dynamic network are important and challenging problems. The IoT can consist of unlimited number of devices that are integrated into local or global, physical and wireless networks. The pool of these automated devices and sensors generates and transmits large amounts of data in real time, which has no use without appropriate filtering and data processing.

A. Network Protocols

Data transmission is possible thanks to different network protocols – a semantic and syntactic rule set that determines computer network functional block activity at the process of data transmission. In computerized networks that are built according to open system architecture requirements, a protocol determines behaviour of one layer entity during data transmission [7].

Creating an IoT network is certainly not an easy task, especially if there are sensors that cannot be included in the global address schema interfering with the ability to make a fully-fledged sensor node. Therefore, traditional IP protocols are not suitable for data exchange. Moreover, IoT nodes are closely dependent upon constant energy sources, network channel throughput capacity and storage parameters, which require sophisticated resource management [7]. In case of wireless sensors, a need of adding data sink to the network arises. All gathered information at first will be stored in the sink and reach other nodes of the network afterwards. Correctly selected data transfer strategy between endpoint sensors and sinks, their disposition and configuration can improve IoT network bandwidth, significantly reduce energy costs and prevent nearly located sensors from sending the same information to data analysis devices [9], [10].

B. Data Transmission

Data transmission in the IoT is a complicated process that can consume a large amount of network resources for its purposes. Information may differ depending on a device type and transmission protocols. For example, ISO 8583 standard, on the basis of which payment terminal messages are made, generates

a data string that represents customer's payment transaction (Fig. 2).

```
020042000400000000021612345678901234560609173030123456789ABC1000123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
```

Fig. 2. Example of ISO 8583 message body.

A short message contains all the necessary information about the payment card, terminal and financial part of the purchase. Unfortunately, neither human nor computer can make use of this information without a decoding instruction. System holders should choose between compact messages in the IoT environment, spending more computing resources on data decryption or just sending full-size messages that can overburden network traffic. The encoded message mentioned earlier may look firm, but the information it contains can never be stored or used for data analysis in its original form. It is worth thinking of an effective data transmission policy that works with the ready-to-use information, like the new payment terminal standard ISO 20022 based on Extensible Markup Language (XML). Designed for network documents it makes information look more convenient and facilitates data parsing. Figure 3 contains an example of ISO 20022 message.

```
<CdtTrfTxInf>
  <IntrBkSttlmAmt Ccy='USD'>12500</IntrBkSttlmAmt>
  <IntrBkSttlmDt>2009-10-29</IntrBkSttlmDt>
  <Dbtr>
    <Nm>ACME NV.</Nm>
    <PstlAdr>
      <StrtNm>Amstel</StrtNm>
      <BldgNb>344</BldgNb>
      <TwnNm>Amsterdam</TwnNm>
      <Ctry>NL</Ctry>
    </PstlAdr>
  </Dbtr>
  <DbtrAcct>
    <Id>
      <Othr>
        <Id>8754219990</Id>
      </Othr>
    </Id>
  </DbtrAcct>
  <DbtrAgt>
    <FinInstnId>
      <BIC>EXABNL2U</BIC>
    </FinInstnId>
  </DbtrAgt>
</CdtTrfTxInf>
```

Fig. 3. Example of ISO 20022 message body.

C. Heterogeneity

IoT device heterogeneity is one of the IoT distinctive characteristics and likewise its weak spot. Depending on the complexity, the IoT architecture may include multiple levels of devices, each of which is focused on a specific type of function execution. The difference is not only in transmitted protocols, but mostly in device intricacy, which directly reflects their computing resource usage rates and the amount of data being passed through them. It has divided the IoT architecture into multiple levels, where the lowest vertical heterogeneity level contains endpoint devices/sensors that work with external environment and by moving upwards the top more complicated routing and computing devices can be met (see Fig. 4). In the optimised system architectures, each next level should contain fewer devices.

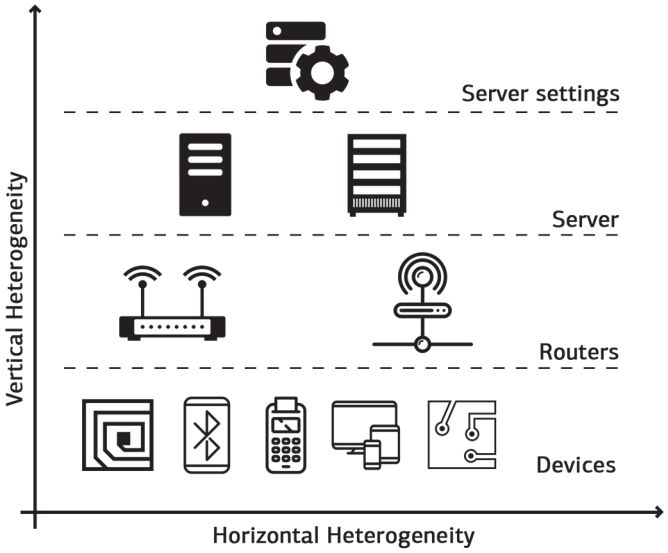

Fig. 4. Heterogeneity axis of IoT.

IoT heterogeneity implies different devices: user personal hardware tools, sensors, routers, switches, hubs, databases, computing servers, etc. In the IoT, each device performs a specific role and executes only necessary functions in order not to overload the system. On the other hand, many devices perform more than one function that can sometimes be similar or even overlap with the functionality of other devices allowing them to replace one another in case of emergency. Omitting details, all devices can be divided into three categories: endpoint devices, IO tools and computing machines. Endpoint devices listen for execution commands from external irritants (like message from users' smartphone) or the main unit (command from Cron) and generate data depending on parameters they receive. In most cases, endpoint devices are difficult to configure, to say nothing of intervention on the programming level. IO devices are restricted in terms of computing resources so they mainly fulfil the role of a mediator between endpoint devices and higher level machines. The last category of devices is responsible for resource-intensive data filtering and processing. It is worth mentioning that all three types can participate in the IoT network as separate physical devices, both appear as inner logical nodes of a single device, but it depends on the complexity of the IoT system or the complication of the device itself.

D. Scalability

Any IoT structure consists of several different devices and sensors. Quantity of these components in a single system is limited by factors, such as a number of input and output channels for a device, Internet or electrical power network limit load, etc. However, any inconveniences can be bypassed with

the help of third party technologies, such as switches and routers, which allow for data exchange between larger numbers of IoT devices. Lots of systems nowadays rely on cloud computing to ease the infrastructure management logic and data analysis. Sending all the generated data to a central cloud resource is not only ineffective, but also fraught with system hang ups. For health-monitoring, emergency-response, and other latency-sensitive applications, delay caused by transferring data to the cloud and back is unacceptable. These situations require a more flexible tool such as Fog computing, involving components of data processing or analytics applications running in multiple distributed clouds and peripheral devices of the network to gain balance in terms of resource allocation [11]. Still, it is very doubtful for all devices in the IoT network to use the same data transmission protocols meaning that a central management system will need to complete additional data unification processes and by that create a supplementary load on the system. It may seem to work in its usual regime with just a few devices connected to the IoT, but a larger number of devices would certainly decrease the quality of service. In order to minimise network latency, system network holders need to organise data management sequence queries and carefully modify them according to their IoT network structure. Unfortunately, queries are not suitable for every system, especially ones that rely on actual information. Pauses between sending and receiving may be too long to use data in real-time activities leaving only two options: to store data for historical analysis and statistics or just dispose them for being needless. Successful IoT system management requires devices-participants that can complete rudimentary procedures, functions and data conversion offline, without using cloud computing units or any other system resources. Eventually, it will result in significant computing process acceleration [7].

IV. Data Storage

Depending on the IoT infrastructure, devices can use different data storage and transmission mechanisms. There are IoT tools that store information received from sensors directly in their internal built-in memory. The latter ones above all work autonomously and accumulate only necessary amount of information to perform real-time activities or to execute preset conditions with the help of aggregated data. The internal memory of these tools is usually very limited and it is meant only for sensor originated data storage.

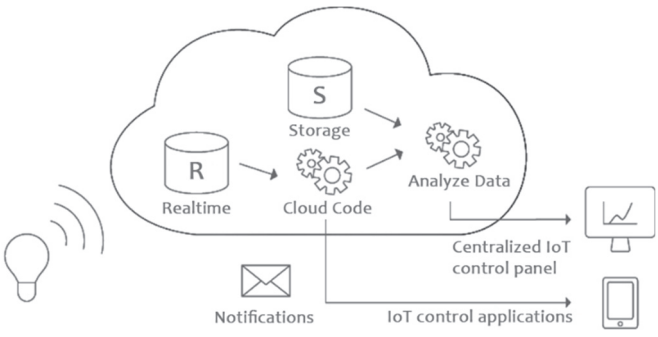

Fig. 5. Structure of centralised data computing and storage.

Nowadays a centralised data storage standard is used more extensively (Fig. 5). It allows the IoT devices to transmit data to a centralised server where it can be stored, analysed or managed [12]. In theory, the number of connected devices and stored data can be infinite within such a system.

V. Device Management

The key feature of IoT devices is that unlike conventional accessories, which can be controlled and managed using mechanical switches or buttons, the former ones can also be handled electronically via programs and frameworks. Simply put, IoT devices can be controlled remotely and the network owner will always be able to check their autonomous operating condition [13].

IoT devices can be managed along with other IoT tools connected to the same network, with or without human involvement. By exchanging information among themselves and performing built-in functions, IoT devices are capable of sending commands to one another, thereby managing mutual activity [13] (for example, in the morning bed and room sensors capture the moment when the person wakes up and pass a command to a coffee machine to make some fresh hot coffee).

Today, the role of smartphones in the IoT device management process also becomes widely documented. Mobile applications can openly send operation initialisation commands or even directly connect to the device to make certain setting adjustments (for example, on the way home a person sends a message command from a smartphone to prepare a bath upon arrival).

It is necessary for each IoT network to have unified management capabilities provided by a central control panel that can identify a device activity based on the aggregated data or user commands. For sure it is easier to maintain small networks, such as a previously-mentioned smart house concept, where even if the network holder does not have a central point to control all devices at once, a possibility to maintain and work with each of them individually remains. Hereto small-scale networks generally consist of developer tools that previously have special management API [3] and a framework, for example, scriptr.io provides a JavaScript code fragment that an application would use to establish a connection with the IoT device and receive its current state (see Fig. 6) [14]:

```
var http = require("http");
var ip = request.headers["x-forwarded-for"];
var iplookupUrl = "http://ip-api.com/json/" + ip;
var response = http.request({url: iplookupUrl});
if(response.status == "200"){
    var result = JSON.parse(response.body);
...
```

Fig. 6. Scriptr.io JavaScript code fragment [14].

IoT architectures with a large number of connected devices are quite another matter. Such a structure can contain all kinds of tools that cannot be managed jointly. It is worth trying to use a hierarchical structure, which provides a system that is responsible only for its own subtype devices, but similar

systems are controlled by higher level management units. Eventually, such a system will require many and various third party solutions to integrate all system parts together.

IoT frameworks simplify the process of connection to IoT devices that are based on data transmission and network protocols (IPv6, 6LoWPAN, MQTT) [15].

The latest operation systems seek to support IoT features, such as device connection and management by default, for example, Windows 10 IoT Core or Apple HomeKit. If a ready IoT platform does not meet system specific requirements, Windows, Linux and Mac OSX operation system users can still use SSH client terminals to refer to IoT devices and get sensor reading results or change the settings.

VI. Advantages and Disadvantages of IoT

The major advantage of the IoT is the possibility to integrate the system into the surrounding environment where it can adjust and perform work based on aggregated data, making day-to-day life more convenient, easier and more productive. IoT devices can be compact, ergonomic and may not differ from regular social life objects.

The IoT can be used in medicine performing a continuous medicine check, in everyday life – developing the smart house paradigm, wide application in all kinds of industries, utilities, logistics, education, entertainment sectors, etc. [16].

Undoubtedly, the IoT world of future can be exciting and carefree, although at the moment this technology is too young and requires huge investment of funds to realise the conceived potential. One of the most salient weaknesses of the IoT is its implementation and upkeep costs.

The production rates of devices with inherent IoT modules are too slow and the advertisement campaigns are not generating enough interest to let this technology gain widespread popularity among people. It is hard to control and manage a large number of IoT devices, for every IoT device manufacturer develops software that supports only his products, creating certain limitation of choice for the potential customer and the need to use individual applications for IoT device administration.

The lack of systems with unified management possibilities that could have control over the tools owned by the host is a significant disadvantage at the point of IoT device management [3]. All this reflects in the fact that the system administrator is not able to see the information summary generated from all sensor readings, because different manufacturer applications do not support information exchange between one another. Thus, the full potential of the IoT cannot be achieved.

The security issue is open as well, for it is quite important not to allow malicious users to gain access over the IoT devices or the provided data. Of course, security measures such as certificates, data encoding and access key generation are used to actively protect IoT devices, but it is too soon to talk about full protection, as IoT open public network security still evolves, indicated by new investigations and development in the field of network protocols [15].

Implementation of the IoT technology in vital life processes requires a serious consideration of further actions in case of system failure, for example, breakdown of electrical power supply, damage caused to sensors or invalid data upload to the system under the influence of external exposures. Thereby both IoT software and technical security solutions need to be designed [16], [17].

VII. Technology Application and Development Perspectives

The IoT has drawn strong interest from electronic device manufacturers and investors. This niche market as well as the technology itself expects rapid growth and connection of 33 billion devices (Fig. 7) to the Internet by the year 2020 that will increase continuously [17], [18]. Thus, in turn, it will increase the IoT market turnover value that by the year 2020 could reach the mark of 1.7 trillion United States dollars [19].

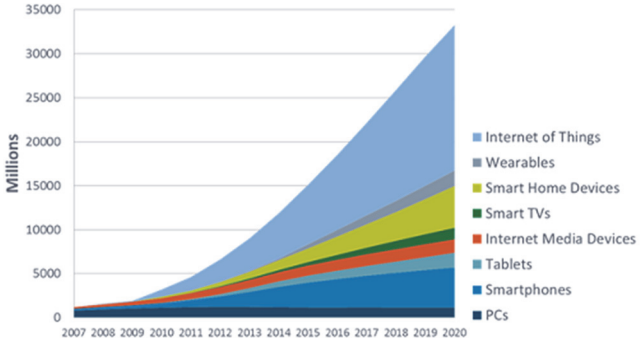

Fig. 7. The estimated Internet-connected devices count till 2020 [18].

Analytical data indicate that this industry can reach its full potential in the near future with the development of IoT management architecture and the emergence of new management layer – Web of Things (WoT herein). By its nature, the WoT requires the development of an individual IoT device management layer (Fig. 8) that carries out a unified web-based device management environment [20].

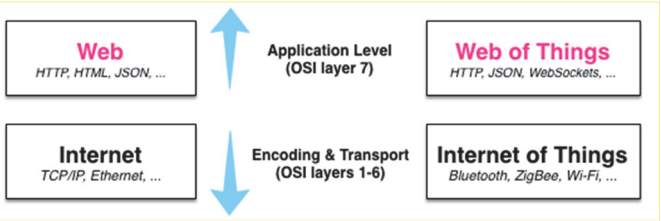

Fig 8. Management layers of IoT (OSI – Open Systems Interconnection) [20].

Such a solution allows monitoring data provided from all sources and performing it regardless of manufacturer and data transmission protocols, which makes it possible to better organise the IoT infrastructure and interaction among devices in the IoT network. For large IoT networks (on a city or country level, for example, real-time global traffic management system) it will be quite important to have a unified management system and a central data processing node that can aggregate all information and data flows from active devices.

Surprisingly, but with the increase of IoT network complexity, its quality of service could grow as well. This can be explained with the possibility of devices to overtake identical or similar functions when one tool becomes damaged or is refused by the system for an unknown reason [7] (for example, if a distance tracking device in someone's sport shoes breaks down, sensors in smart shorts start to carry out this function).

VIII. Results

As it has been mentioned earlier in the paper, the IoT represents a highly scalable and heterogenous system architecture, so in theory it is possible to design a multilayer structure that can implement and manage any kind of device or data within it. Unfortunately, implementation of such IoT is directly related to the financial possibilities of system holders. The more diverse the system is, the more third party apparatus or software solutions it requires to guaranty a robust and reliable service level. We also assume that this dependence tends to exponential growth since the probability of connecting a new entity to all existing ones, with the help of only one solution, tends to zero. Global practice shows that computerized tool developers in the field of data management aim for internationally recommended standards and languages such as XML, Json or YAML, which is not always possible due to social, technical, security or other reasons. Despite all limitations, the IoT gains popularity in various aspects of life, from housekeeping to some of the most complex manufacturing and infrastructure management industries. Research in the field of IoT architectures and frameworks leads us to a conclusion that products from a single manufacturer that can work synchronously and provide a special management API from the very beginning are unlikely to be used. Despite the fact that such systems often show high work efficiency, low latency and are much less likely to suffer a critical system failure, it is necessary to consider their designing on a single base. In reality, hosts have no choice but to acquiesce to incomplete system or integrate expansive tools that can unify information in their network. In the latter case, the author group offers a solution – to develop an additional level for a complete IoT system, which will be responsible for data conversion and transformation of the system to the so-called WoT [20]. Architecture provides that a node consisting of several devices or subnetworks is connected to a computer, for example, Raspberry Pi (but may be any other alternative) which participates only in the process of data transfer and conversion to a user-required standard. Each of these machines costs approximately 40€, not to mention the need of developing software for a particular IoT network. The latest Raspberry Pi versions are already equipped with 2.4 GHz WiFi 802.11 bgn module and Bluetooth 4.1, but for older models they should be purchased separately. By implementing the architecture with a web management layer, hosts will be able to control data management processes and remotely administrate IoT devices if they are capable of receiving third party commands. The negative sides of this architecture are large material investments, complicated implementation and instability. For everything to work correctly, both IoT and third party solutions will need to be readjusted and configured in case a new type of device is added to the IoT environment. Nevertheless, some specialists believe that in the nearest future new generation network protocols will be introduced to the public, making it possible to transmit and manage data without any problems.

References

[1] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, issue 7, 2013, pp. 1645–1660. https://doi.org/10.1016/j.future.2013.01.010

[2] S.-P. Tseng, B. R. Li, J. Pan and C. Lin, "An Application of Internet of Things with Motion Sensing on Smart House," in *2014 International Conference on Orange Technologies (ICOT)*, Xian, 2014, pp. 65–68. https://doi.org/10.1109/ICOT.2014.6956600

[3] P. J. Windley, "API Access Control with OAuth: Coordinating interactions with the Internet of Things," *IEEE Consumer Electronics Magazine*, vol. 4, issue 3, pp. 52–58, July 2015.

[4] The Intel IoT Platform. [Online]. Available: https://theiotlearninginitiative.gitbooks.io/internetofthings101/content/documentation/Intel.html [Accessed: Apr. 10, 2016.].

[5] M. Pacelle, "3 topologies driving IoT networking standards," 2014. [Online]. Available: http://radar.oreilly.com/2014/04/3-topologies-driving-iot-networking-standards.html [Accessed: Apr. 10, 2016.].

[6] M. Rouse, "Mesh network topology (mesh network)," 2015. [Online]. Available: http://internetofthingsagenda.techtarget.com/definition/mesh-network-topology-mesh-network [Accessed: Sept. 20, 2016.].

[7] L. Tan and N. Wang, "Future internet: The Internet of Things," in *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, Chengdu, 2010, pp. 376–380. https://doi.org/10.1109/ICACTE.2010.5579543

[8] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, issue 3, pp. 325–349, May 2005. https://doi.org/10.1016/j.adhoc.2003.09.010

[9] C. Wang, H. Ma, Y. He and S. Xiong, "Adaptive approximate data collection for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems,* vol. 23, no. 6, pp. 1004–1016, June 2012. https://doi.org/10.1109/tpds.2011.265

[10] F. Chen and R. Li, "Sink node placement strategies for wireless sensor networks," *Wireless Personal Communications*, vol. 68, no. 2, pp. 303–319, Jan. 2013. https://doi.org/10.1007/s11277-011-0453-x

[11] A. V. Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," *Computer*, vol. 49, no. 8, pp. 112–116, Aug. 2016. https://doi.org/10.1109/MC.2016.245

[12] C. Perera, A. Zaslavsky, P. Christen and D. Georgakopoulos, "Context Aware Computing forThe Internet of Things: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, First Quarter 214. https://doi.org/10.1109/SURV.2013.042313.00197

[13] M. Elkhodr, S. Shahrestani and H. Cheung, "A Smart Home Application Based on the Internet of Things Management Platform," *2015 IEEE International Conference on Data Science and Data Intensive Systems*, Sydney, 2015, pp. 491–496. https://doi.org/10.1109/DSDIS.2015.23

[14] *Documentation | scriptr.io*. (2016). [Online]. Available: https://www.scriptr.io/documentation [Accessed: Apr. 10, 2016.].

[15] *Internet of Things Protocols & Standards*. [Online]. Available: http://postscapes.com/internet-of-things-protocols [Accessed: Apr. 10, 2016.].

[16] D. Bandyopadhyay and J. Sen, "Internet of Things: Applications and Challenges in Technology and Standardization," *Wireless Personal Communications*, vol. 58, issue 1, pp. 49–69, May 2011. https://doi.org/10.1007/s11277-011-0288-5

[17] R. Khan, S. U. Khan, R. Zaheer and S. Khan, "Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges," in *2012 10th International Conference on Frontiers of Information Technology*, Islamabad, 2012, pp. 257–260. https://doi.org/10.1109/FIT.2012.53

[18] *33 Billion Internet Devices By 2020: Four Connected Devices For Every Person In World*. (2014). [Online]. Available: https://www4.strategyanalytics.com/default.aspx?mod=pressreleaseviewer&a0=5609 [Accessed: Apr. 10, 2016.].

[19] *Internet of Things Market to Reach $1.7 Trillion by 2020: IDC*. (2015). [Online]. Available: http://blogs.wsj.com/cio/2015/06/02/internet-of-things-market-to-reach-1-7-trillion-by-2020-idc/ [Accessed: Apr. 10, 2016.].
[20] D. D. Guinard and V. M. Trifa, "Comparing IoT and WoT," *Building the Web of Things*, 2016, p. 8.

Eduards Filičevs obtained a Bachelor's degree in Computer Science from Riga Technical University in 2014. At present, he is studying at the Master study programme "Information Technology". Since 2014 he has been working as a Programmer of the Department of Bank Information System Development at JSC Latvijas pasta banka. The areas of interest: system architecture for big data management, application development, digital service integration.
E-mail: Eduards.Filicevs@rtu.lv

Vladislavs Aleksandrovičs holds a Bachelor's degree and currently is studying at the Master study programme "Information Technology" of Riga Technical University. At the same time, he works as a Web Developer and Designer at the international web development company. His research interests include the latest web programming (php, html5, css3, js ect.) and design technologies.
E-mail: Vladislavs.Aleksandrovics@rtu.lv

Jānis Kampars obtained his Doctoral degree in Information Technology from Riga Technical University in 2012. His areas of interest are application integration, data integration, web development and cloud computing. Jānis Kampars works as an Assistant Professor and Researcher at the Faculty of Computer Science and Information Technology, RTU, Latvia. He has been working at RTU since 2004. Recently he has participated in FP7 project Capability as a Service. Jānis Kampars also delivers study courses on web programming, data analysis and processing, cloud computing, business system programming at RTU.
E-mail: Janis.Kampars@rtu.lv