EasyChair Preprint

№ 56

Development of the universal data transfer protocol: mobile solution

Anna Krepchenko, Anastasiia Strielkina, Artem Tetskyi and Dmytro Uzun

April 10, 2018

Development of the universal data transfer protocol: mobile solution.

Anna Krepchenko[1], Anastasiia Strielkina[2], Artem Tetskyi[2] and Dmytro Uzun[2]

Kharkiv, Ukraine
[1] anna.krepchenko@student.csn.khai.edu
[2] {a.strielkina, a.tetskiy, d.uzun}@csn.khai.edu

Abstract. The paper deals with the process of development of the universal data transfer protocol for Internet of Things projects. It describes the reasons for the mobile solution is needed. All main ways of communication for Internet of Things systems are described. It is spoken in details about different ways to transfer data on the mobile side. WIFI, Bluetooth and Bluetooth Low Energy are noted as main existed protocols. Much attention is given to Bluetooth Low Energy as the main protocol for the universal solution to base on. The method proposed explained with the example of the common healthcare project's system. The main data buses are defined.

Keywords: Internet of Things, Protocol, Mobile, Bluetooth Low Energy.

1 Introduction

1.1 Motivation

Internet of Things (IoT) is a socio-technical phenomenon with the power to disrupt our society such as the Internet before. IoT promises the (inter-) connection of myriad of things providing services to humans and machines. It is expected that by 2020 tens of billions of things will be deployed worldwide (see Fig.1). It became evident that the traditional centralized computing and analytic approach does not provide a sustainable model this new type of data. A new kind of architecture is needed as a scalable and trusted platform underpinning the expansion of IoT. The data gathered by the things will be often noisy, unstructured and real-time requiring a decentralized structure storing and analyzing the vast amount of data. Due to limited roles, energy constraints, etc., however, IoT devices may use mission-tailored or proprietary wireless protocols that smartphones do not speak natively. The number of mobile users grows every year. The number of mobile phones and wearable or connected devices grows accordingly (see Fig.2).

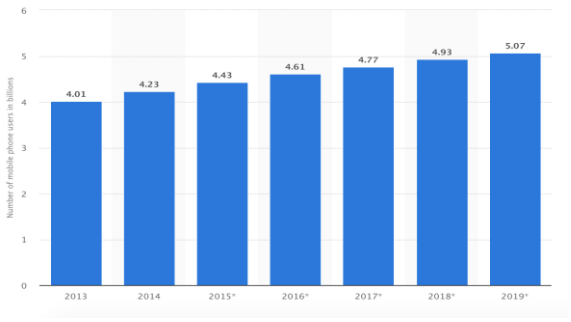

Fig. 1. Number of mobile phone users worldwide from 2013 to 2019 [1]

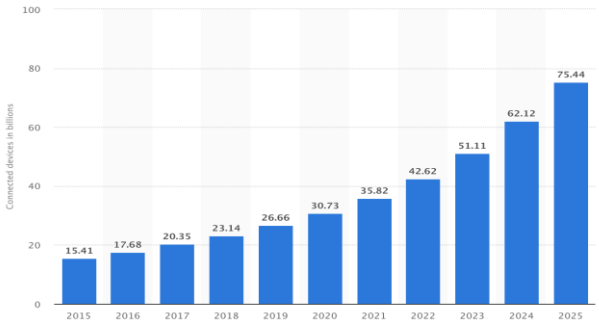

Fig. 2. Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 [2]

Every new device built later 2015 implements own realization of wireless connection. This way of development makes slower the development process and restricts communication between different devices. That is why devices cannot be connected to big systems without pain. Universal communication protocol of the IoT devices will resolve this problem.

1.2 State of the art

There are a lot of protocols and standards for the IoT system [3]. The diversity of wireless networks and protocols used by IoT makes solutions non-interoperable. Moreover, some Things are not always able to communicate due to resource constraints or environmental factors [4]. However, there is an apparent fact that no IoT standard or protocol has been specified to the date; companies, large or small, are working with their own platforms or frameworks. That is why all protocols introduced before are developed to support specific system. Any company makes some improvements to give some advantages for their solution, if it uses common solutions and implements protocol.

This implies that there no well-known universal data transfer protocol developed for IoT systems.

In maximum coverage, there are few ways of communication in IoT system:

- remote device—smartphone;
- smartphone—remote device;
- smartphone—server side;
- server side—smartphone;
- remote device— server side [5].

1.3 Goals and structure

The main goal is a building a concept of protocol, suitable for IOT devices, communicated with mobile devices. Besides, it is also necessary to investigate already existed protocols, find differences between them according mobile specific. Universal data transfer protocol should be explained with the example of the common IOT system.

The universal protocol should support data transfers in both directions between remote device and smartphone and should be easily integrated with other usual protocols, used by another way of communication. The remainder of the paper is structured as follows. Section 2 presents a brief description of existed solutions and protocols to base on. The universal data protocol is shown in section 3 and an example of realization for the virtual system is shown in section 4. Sector 5 contains conclusions and future steps of research.

2 Analysis of existed solutions

A data transfer protocol is a standardized format for transmitting data between two devices [6]. Despite their numbers, networking protocols show little variety, because all data transfer protocols use the same embedded principles and concepts for communication. The rules can be expressed by algorithms and data structures, raising the opportunity for hardware independence in digital computing systems.

2.1 Methods of research

The notion of a universal protocol provides a rationale for standardization of protocols; assuming the existence of a universal protocol, development of protocol standards using a consensus model might be a viable way to coordinate protocol design efforts [7]. To get the less weight of the device and the most portability at the same time, there no chance to use WIFI as the main and long-term communication way. It can be used to configure remote device at the first time. However, long time usage will be the main reason for small battery life duration.

Bluetooth—is safer than WIFI for human health, but at the same time, the distance between conjugated devices cannot exceed theoretical maximum under ideal conditions of 10 meters [8]. In addition, this method transmission constantly uses the device's

Bluetooth adapter in the active mode, which significantly reduces the duration of work from the built-in battery. However, there are no difficulties with the connection and transfer data to your smartphone [9].

BLE (Bluetooth Low Energy)—to have permanent communication with a smartphone and keep battery power as long as you can. This solution gives the most productive portability [10]. The Bluetooth Low Energy technology keeps all advantages of Bluetooth connection, and expand the theoretical maximum of the distance between the devices to the 100 meters in the perfect conditions. This way of communication significantly reduces consumed energy, which follows from the name of the technology. The only disadvantage of this protocol is the support from Android version 4.3 and IOS version 8.0 and iPhone 4s. BLE allows the short bursts of the long-range radio connection, making it ideal for Internet of Things applications, that don't require the continuous connection but depend on long battery life [11].

The latest version of Bluetooth Low Energy 5.0 has a longer characteristic, but there are not enough mobile devices supported this revision on BLE.

Bluetooth mesh networking – is a protocol based upon Bluetooth Low Energy that allows for many-to-many communication over a Bluetooth radio [12].

Foundation models have been defined in the core specification. Two of them are mandatory for all mesh nodes: Configuration Server (mandatory), Configuration Client, Health Server (mandatory), Health Client.

Not all IoT projects are health projects. Communication is carried in the messages that may be up to 384 bytes long. This is not suitable value for the mobile side. So the best supported by the mobile devices revision of BLE is 4.0.

Key BLE 4.0 terms and concepts:

1. Generic Attribute Profile (GATT)—The GATT profile is a general specification for sending and receiving short pieces of data known as "attributes" over a BLE link. All current Low Energy application profiles are based on GATT.
2. Attribute Protocol (ATT)—GATT is built on top of the Attribute Protocol (ATT). ATT is optimized to run on BLE devices. Each attribute is uniquely identified by a Universally Unique Identifier (UUID), which is a standardized 128-bit format for a string ID used to uniquely identify information. The attributes transported by ATT are formatted as characteristics and services [13].
3. Service—a service is a collection of characteristics. For example, you could have a service called "Heart Rate Monitor" that includes characteristics such as "heart rate measurement." A list of existing GATT-based profiles and services can be found on bluetooth.org.
4. Characteristic—a characteristic contains a single value and 0-n descriptors that describe the characteristic's value. A characteristic can be thought of as a type, analogous to a class.
5. Descriptor—Descriptors are defined attributes that describe a characteristic value.

But the documentation of GATT protocol does not allow to have some general characteristics for all the devices. That's why every new IOT device has its own implementation of the GATT [14].

3 Universal data protocol

The main flow of hierarchy used as in GATT was designed.

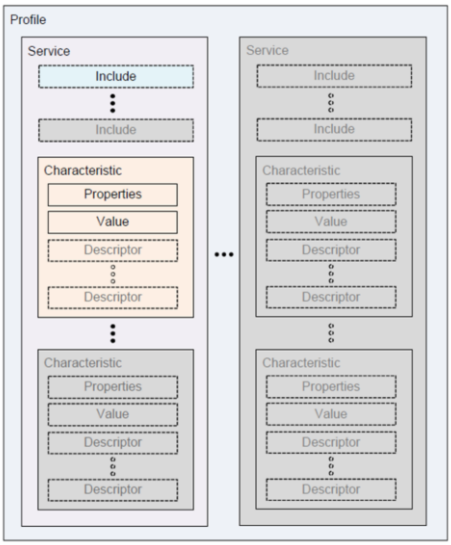

Fig. 3. GATT Profile [15]

The main operands of the protocol are buses and characteristics. Buses should be created by the logic meaning. Characteristics should be grouped by the belonging to each bus. Each bus should contain at least one characteristic. Each characteristic has 20 bytes for data.

Protocol consists of:

- Data buses;
- Data characteristics.

Analyzing the predication of the characteristics was found a different kind of characteristics. The universal characteristic should just contain 20 bytes of the data [16]. The most common way to sync a remote device with any receiver is to share some data and add the specific time, it has happened. Therefore, characteristic should have some bytes to be filled with “time”. Integer has 4 bytes to save some value. It can be held the Unix time in the integer value. Unix time is a system for describing a point in time, defined as the number of seconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970 [17], minus the number of leap seconds that have taken place since then. That’s why some characteristics will have 4 bytes for time and 16 bytes for data. Any remote device should have a possibility to be reconfigured after some usage time. Configuration can be done with a range of allowable values, for example, describe the minimum and the maximum lines. Also, it can be done with 1 needed parameter such as index or rate. Both of these ways can be covered with characteristic consist of the 2 parts, 10 bytes each of them.

Types of characteristics:

- Simple characteristic(Fig. 4)
- Time characteristic(Fig. 5)
- Configuration characteristic(Fig. 6)

Fig. 4. Simple characteristic

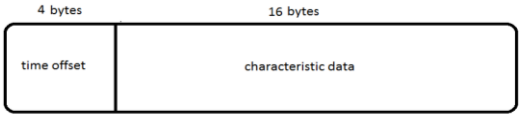

Fig. 5. Time characteristic

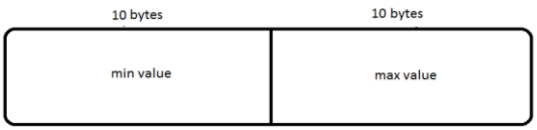

Fig. 6. Configuration characteristic

Characteristics of the protocol have the base:
BASE = 0000xxxx-0000-1000-8000-00805f9b34fb
This allows to provide up to 65535 different characteristics using the protocol.

4 Implementation example

As an example, can be examined the IOT healthcare project with few buses and characteristics. System consists of simple remote device with specific sensors and phone. The ranges of measurements can be configured from the phone. Alerts are sent by remote device if measurements are out of range. Remote device has LED controlled from the phone (see Fig.7).

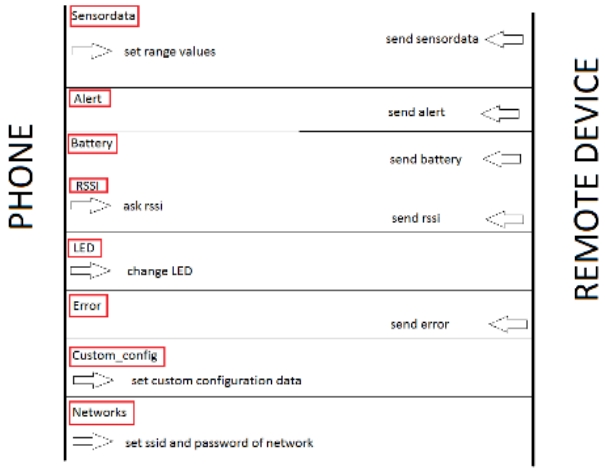

Fig. 7. Buses of the system

Sides of communication protocol:

- remote device – bracelet or other wearable device;
- phone.

4.1 Main global data buses

All data can be transferred from remote device to phone and back via data buses. Data buses can be separated by business logic:

- Sensordata;
- Alerts;
- Battery;
- Rssi;
- LED;
- Errors;
- Custom_config;
- Networks.

Sensordata data bus used to transfer specific measurements of each sensor and to set ranges of normal values. Min and max values will be set to byte array with different offsets.

Alerts data bus used to transfer alerts from the remote device to phone immediately.

Battery data bus used to transfer battery level values with the low frequency.

Rssi bus used to get data about signal strength. Provided by Bluetooth as a default feature.

LED bus used to set some data according the led (turn on/ turn off, etc.).

Error bus used to transfer errors of the remote device to the phone.

Custom_config bus used to set specific configurations (such as algorithms settings, sleep time, proximity setup, etc.)

Networks bus used to read and set network's data from the phone to the remote device. The networks will be used by priority to connect.

4.2 Services and characteristics

Approach mentioned in section 4.3 can be build using main operands of protocol for realization main data buses. All data buses will have their own Services and Characteristics (except the rssi). Final realization of the approach results described in Table 1.

Table 1. Characteristics' values

SERVICE_NAME	SERVICE_VALUE	CHARACTERISTIC_NAME	CHARACTERISTIC_VALUE	Updating frequency (Hz)
SENSORDATA_SERVICE	0x3100	"sensor_name" VALUE_CHARACTERISTIC	0x2A37	1
		"sensor_name" RANGE_CHARACTERISTIC	0x2A52	1
ALERT_SERVICE	0x3215	ALERT_CHARACTERISTIC	0x2A3F	1
BATTERY_SERVICE	0x3330	BATTERY_CHARACTERISTIC	0x2A19	1
LED_SERVICE	0x3445	LED_CHARACTERISTIC	0x2AC7	1
ERROR_SERVICE	0x3560	ERROR_CHARACTERISTIC	0x2A54	1
CUSTOM_CONFIG_SERVICE	0x3675	"custom_config_name" CHARACTERISTIC	0x2C33	1
NETWORKS_SERVICE	0x3790	"number" SSID_CHARACTERISTIC	0x2E01	1
		"number" PASS_CHARACTERISTIC	0x2E39	1

4.3 Faced issues

Security and safety of developed protocol depends on protocol it based on. While universal data transfer protocol used BLE 4.0 as base it has the same issues and problems. This issues can be resolved with the following recommendations:

- Activate the communication encryption whenever possible. The use of LTK allows communication to be encrypted between the master and the slave from the first moment. All devices from a control network that uses Bluetooth should make use of the encryption.
- Do not accept connections from unknown devices. Activate the white list option in the master and require pairing with a key of at least 5 characters, thus avoiding malicious devices connecting without permission.
- Continuously revise the list of registered trusted devices in order to avoid malicious devices appearing.
- Assign a name to the devices that does not reflect extra information such as the brand, the device model, the location or service. With these measures it is difficult for possible attackers to benefit from vulnerabilities associated with specific devices and carry out targeted attacks.
- Maintain device configuration in invisible mode to make it difficult to detect from other devices.

Following recommendation can be implemented "as it is", but it will decrease the speed of data transfer.

5 Conclusions and Future Work

This paper describes process development a universal data transfer protocol for IOT project. The reasons why the mobile solution is needed were examined. All main ways of communication are every IOT system were described. WIFI, Bluetooth and Bluetooth Low Energy were highlighted as main existed protocols. Much attention was given to Bluetooth Low Energy as the main protocol for the universal solution to base on. The method proposed explained with the example of the common healthcare project's system. As result data transfer protocol, defined main data buses and protocols to base on, was created. Next steps of research will be defining terms of use for data transfer and implementing the universal protocol in multiple IOT systems and resolving security issue without speed lost.

6 Acknowledgements

This paper results from the Erasmus+ programme educational project ALIOT «Internet of Things: Emerging Curriculum for Industry and Human Applications» (reference number 573818-EPP-1-2016-1-UK-EPPKA2-CBHE-JP, website http://alIoT.eu.org) in which the appropriate course is developed (ITM4 - IoT for health systems) within its framework, we have developed modules related to IoT systems modelling. The authors would like to thank colleagues on this project, within the framework of which the results of this paper were discussed. The authors also would like to show their deep gratitude to colleagues from Computer systems, networks and cybersecurity, National Aerospace University «KhAI» for their patient guidance, enthusiastic encouragement and useful critiques of this paper.

References

1. The Statistics Portal, Number of mobile phone users worldwide from 2013 to 2019. https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/ [accessed January 8, 2018]
2. The Statistics Portal, Internet of Things connected devices installed base worldwide from 2015 to 2025 https://www.statista.com/statistics/471264/IoT-number-of-connected-devices-worldwide/ [accessed January 8, 2018]
3. Postscapes, IoT Standards and Protocols https://www.postscapes.com/internet-of-things-protocols/ [accessed February 10, 2018].
4. Andrejs Bondarevs, Patrik Huss, Qinzhong Ye, Shaofang Gong, Universal Internet of Things solution: Protocol independent, Industrial Technology (ICIT), 2017 IEEE International Conference doi: 10.1109/ICIT.2017.7915553
5. Anna Krepchenko, Put your IOT back on rails https://medium.com/@anna.krepchenko/put-your-IoT-back-on-rails-8b6f14399918 [accessed January 5, 2017].
6. Microsoft, About data transfer protocols. https://technet.microsoft.com/en-us/library/cc731604(v=ws.11).aspx [accessed December 17, 2017].
7. Real-time performance management of assisted living services for Bluetooth low energy sensor communication, Integrated Network and Service Management (IM), 2017 IFIP/IEEE Symposium on, doi:10.1109/IEEESTD.2016.7542098
8. World Health Organization (WHO), Compendium of innovative health technologies for low-resource settings http://apps.who.int/iris/bitstream/handle/10665/108781/-9789241564731_eng.pdf;jsessionid=D3F94C4FCC73E32E0003251C8432FA7B [accessed April 1, 2017].
9. Basagni S., Bruno R., Petrioli C. NETWORKING 2002: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications. Springer; Berlin/Hidelberg, Germany: 2002. Device discovery in Bluetooth networks: A scatternet perspective; pp. 1087–1092.
10. Survey of Vehicle IoT Bluetooth Devices, Service-Oriented Computing and Applications (SOCA), 2014 IEEE 7th International Conference on, 17-19 Nov. 2014 doi:10.1109/SOCA.2014.20
11. Duflot M., Kwiatkowska M., Norman G., Parker D. A formal analysis of bluetooth device discovery. Int. J. Softw. Tools Technol. Transf. 2006; 8:621–632.
12. Hyun-Soo Kim, Jungyub Lee, Ju Wook Jang, BLEmesh: A Wireless Mesh Network Protocol for Bluetooth Low Energy Devices doi:10.1109/FiCloud.2015.21
13. Elvis Pfützenreuter, Bluetooth: ATT and GATT https://epxx.co/artigos/bluetooth_gatt.html [accessed January 9, 2018]
14. Bluetooth SIG, GATT Services https://www.bluetooth.com/specifications/gatt/services [accessed January 8, 2018]
15. Bluetooth SIG, Bluetooth Core Specification Version 4.0. Specification of the Bluetooth System. https://www.bluetooth.com/specifications/gatt/generic-attributes-overview [accessed January 8, 2018]
16. BLE-Stack User's Guide, Logical Link Control and Adaptation Layer Protocol (L2CAP) http://dev.ti.com/tirex/content/simplelink_cc2640r2_sdk_1_35_00_33/docs/ble5stack/ble_user_guide/html/ble-stack/l2cap.html [accessed January 9, 2018]
17. The Open Group Base Specifications Issue 7, section 4.16 Seconds Since the Epoch. doi:10.1109/IEEESTD.2016.7542098