Communication Chain in the Internet of Things with Spread-out Electronic Device System Abstraction.

1st Andrei Bragarenco
Department of Microelectronics and Biomedical Engineering
Technical University of Moldova
Chisinau, Republic of Moldova
andrei.bragarenco@mib.utm.md

2nd Galina Marusic
Department of Computer Science and Systems Engineering
Technical University of Moldova
Chisinau, Republic of Moldova
galina.marusic@adm.utm.md

3rd Calin Ciufudean
Department of Computers, Automatics and Electronics
University Stefan cel Mare of Suceava
Suceava, Romania
calin@eed.usv.ro

***Abstract*—Communication is the act of transferring information between entities. Existing architectural communication approaches provide solutions for data transfer from the sender to the receiver. While those solutions are targeting different domains, we can find a lot of similarities in their architectures. In this paper, we follow the interpersonal communication process principles with its communication components for defining a whole communication chain that combines the sensor actuator and data transfer items in one concept, pointing to the person to person and device to device communication similarities. We come with an approach of a methodology for the Internet of Things abstracted by the Spread-out Electronic Device concept. We consider the whole IoT as a single electronic device with cut-out and broken wires and replaced with communication chains.**

Keywords—Internet of Things, MQTT, Twitter, ROS, communication chain, spread-out electronic device, layered architecture

I. Introduction

Nowadays, the new trend in technology, the Internet of Things (IoT), connects a massive number of smart objects, products, smart devices, and humans. In this trend, all mentioned things could be organized in groups or systems and cooperate in solving some specific tasks providing information and services to each other involving a typical communication process. Communication is the act of data exchange between things as an extensive system entity and plays an essential role in the system operation. There are a lot of technologies that implement the communication process for information transfer between entities. Curiously, almost all communication technologies are inspired by interpersonal communication, meaning communication between people.

One such example of communication is the online social networking tool Twitter, where users send and receive short posts called tweets. People who want to share information publish and those interested in someone's posts are subscribing to his tweets [1]. This social network platform's system is complex, but its communication relies mainly on the publish-subscribe message architecture.

There are many similarities between Twitter and one of the most popular protocols used for IoT applications - Message Queuing Telemetry Transport (MQTT). MQTT is a protocol specifically designed for "machine to machine" communication. MQTT protocol runs over TCP / IP stack and has a data packet size with a low overhead minimum (> 2 bytes) so that consumption of the power supply is also reduced [2]. MQTT is the perfect solution for Internet of Things applications. It is designed for devices with low bandwidth. Additionally, it is a simple messaging protocol that allows sending, reading, and publishing data from sensor nodes and much more. Comparing with Twitter, it is the same thing but for devices. MQTT protocol uses a publish/subscribe architecture. Its event-driven approach enables messages to be pushed to the client [3]. Devices are like users. Like Twitter, where users tweet a message to all the followers, a device publishes a message to all subscribers. And just like a Twitter user receives tweets from all people he follows, a device receives messages from all the devices it is subscribed to [4].

Robot Operating System (ROS) proposes another publish/subscribe architecture consisting of a collection of programs. That allows a user to control a mobile robot's operations efficiently. At its core, ROS is an anonymous publish/subscribe message-passing middleware with asynchronous communications. Some modules will issue a set of topics, while others subscribe to that topic. When new data is published, the subscribers can learn about the updates and can act on them. ROS provides the communication using a message-passing approach that forces developers to focus on pure interface logic. [5] The ROScore is a service that provides connection information to nodes so that they can transmit messages to one another. Every node connects to ROScore at startup to register details of the message streams it publishes and the streams it wants to subscribe to. When a new node appears, ROScore provides it with the information needed to form a direct peer-to-peer connection with other nodes publishing and subscribing to the same message topics. Every ROS system needs a running ROScore service so nodes can find other nodes [6].

Automotive is one of the most safety-critical domains. A typical automotive system consists of several Electronic Component Units (ECU) communicating on Controller Area Net (CAN) buses. Software stacks provide support for the computations and communication, including application tasks, the middleware, drivers, and bus peripherals. For communication purposes, tasks read input signals at their activation and write their outputs in shared variables at the end. Some application task reads the input data from a sensor, computes intermediate results sent over the network to other tasks, and, finally, another task, executing on a remote node, generates the outputs as the result of the computation [6]. The read/write operations with shared variables are similar to the publish/subscribe architecture for communication in the vehicle network.

All the aforementioned architectural approaches implement the communication processes that are most convenient in the domain those are operating, a worldwide social network for Twitter, a collection of devices for IoT with MQTT, a complex ECU with ROS, or an ECU interconnection for Automotive with CAN. Even those who

978-1-6654-4067-7/21/$31.00 ©2021 IEEE