

Blockchain as a Platform for Fog Computing

Ivan Podsevalov(✉), Oleg Iakushkin, Ruslan Kurbangaliev, and Vladimir Korkhov

Saint-Petersburg State University, St. Petersburg, Russia
podsevalov.ivan@yandex.ru, o.yakushkin@spbu.ru

Abstract. In this article, we observe the prospective approaches and techniques in the modern system of automatic control of a group of robots based on distributed ledger technology, using modern methods and development tools. The architecture of this solution is based on the junction of two rapidly developing technologies, such as the Internet of Things (IoT) and distributed ledgers technologies (DLT). We focused on the advantages of the built-in architecture, chose the platforms on which each step will work properly, and looked up at the problems that may arise when implementing individual module of this architecture.

Keywords: Corda · Distributed ledger · Blockchain · Fog

1 Introduction

Modern people, surrounded by an abundance of technology, strive to assign more and more tasks to machines or smart devices, due to their cost and increasing capabilities. The financial advantage is that people do not need to pay to the machine for the work done; we only purchase and periodically maintain it. If we consider the issue in perspective, then over time the costs pay off. Machines do not have a human factor, which would affect the result of the work. We can be sure that the work will be completed properly within the time frame we have specified.

No system can boast about the stable uninterrupted operation so this is why we would like to have some kind of trusted person who is able to guarantee the authenticity of a particular event. The approach to solving this problem can be implemented by means of distributed ledger technology integrated into the system.

It should be noted the observed forms of interaction are built up in a "person-person" or "person-device" manner. But thanks to the concept of the Internet of Things (IoT) we get another type of device-to-device communication. Based on this, we can create some kind of infrastructure that will allow us not only to manage the devices but also to provide information about their condition.

To this day, the cleaning of large premises is a routine and labor-intensive process that requires the involvement of a group of people to achieve a faster

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11620, pp. 596–605, 2019.
https://doi.org/10.1007/978-3-030-24296-1_48

accomplishment of the task. Similarly with robots: the larger the area, the more machines are needed for fairly fast work. But vacuum cleaning robots are more economically advantageous if we consider this project with a future perspective. What if we could to create a system that would optimally and logically work in a group of vacuum cleaning robots and between groups?

In this article, we will look at and talk about the synergy of the two current technologies - the Internet of Things and distributed ledgers - with a projection on the problem described above. In the process, we will form a methodology, which will be a key component in managing a swarm of vacuum cleaners. We set the task as follows: to solve the problem of consistency, security, and optimal operation of groups of robots on the basis of distributed ledgers and using a simple and intuitive interface to enable the user to control this system. Of course, using the provided client interface, the user will be able to monitor the status and status of the machines. The architecture of the solution and the implementation of the prototype will be covered in the following paragraphs.

We emphasize that our project aims to create a software package for managing a swarm of robots using the blockchain. As a supporting task, we dwelt on the problem of cleaning large facilities with the help of robotic vacuum cleaners, where it is advisable to use a group or even several groups. The number of groups can vary from the area of the room, and the user-defined restrictions. Thus, we would like to implement a system that could automatically manage groups of robots and at the same time maintain some kind of real-time report on past events and current status.

2 Architecture

In this chapter, we observe the architecture of our system (see Fig. 1). The central component of our implementation is the cloud, which provides communication to nodes with the end user. This node is the point of contact of the user with the

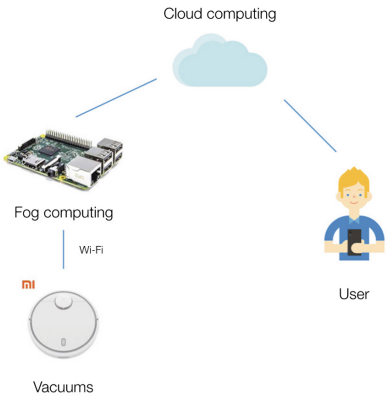

Fig. 1. System architecture

system. In addition, the cloud provides a reliable and secure connection to the microcomputer(s) of the system - Raspberry Pi (RPi). Further, by means of the Wi-Fi wireless LAN technology, secure transfer of data from the RPi to robot vacuum cleaners of a separate group and in the opposite direction is carried out.

Reliability and security at appropriate levels are achieved through the use of distributed ledger technology integrated into our project. Due to its flexibility and technical features, Corda was chosen as the distributed ledger platform. The key factors that have influenced the choice of this platform are limited access, scalability and the presence of more declarative smart contracts, for example, unlike the Ethereum. In addition, not every node can see most of the transaction graph, hence it will not waste resources on processing them. That is, the nodes are obtained only those transactions in which they are involved. Secondly, since there is no single blockchain network, it becomes possible to connect two independent networks through two-way communication between special nodes. Thirdly, an arbitrary ZIP containing geo-information can be included in the transaction. Which can be used in smart-contract code, therefore simplify the development of this system.

3 System Components

As a smart vacuum cleaner, we use Xiaomi Mi Robot Vacuum Cleaner with the Ubuntu 14.04.3 LTS operating system over which the Java Runtime Environment (JRE) is installed and the Corda node is deployed. Note that the robot vacuum cleaner has limited computing power, so the system needs to store and process the minimum necessary data for coordinating the operation of the device.

Raspberry Pi runs on the Raspbian operating system on which the Corda node is deployed using a virtual machine. After receiving data from each member of the group - vacuum cleaning robots, the computer sticks together maps of the area using the software package developed by us and sends further instructions to the robots. The main task the RPi is responsible is work coordination and optimization of vacuum cleaners.

Group of robots led by Raspberry Pi are controlled and coordinated by the cloud server application, based on Corda node software as well. Control of the entire system and monitoring its current state are available through the web API, based on Spring web-server software.

Now we focus on the distributed ledger technology that we use in the project. In a nutshell, we have three types of nodes in our system: for the cloud, for the RPi, and for the robot vacuum cleaner. Let's imagine a network with fifteen nodes - Fig. 2. This diagram clearly shows the interaction areas of individual pairs. For each device, the circle (s) of communication is strictly and clearly defined. For example, a robot from one circle does not have the ability to communicate with the cloud or RPi of another circle and vice versa. In turn, the RPi cannot communicate with the robot cleaner or the RPi of another circle. And, of course, the central node is the Cloud, which does not have access to vacuum cleaners at all.

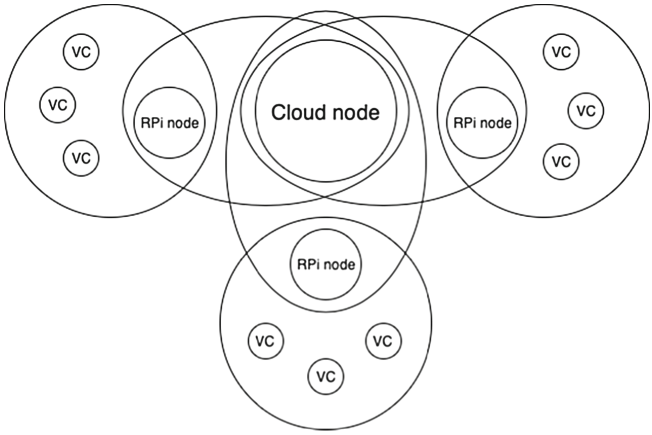

Fig. 2. Interconnection circles

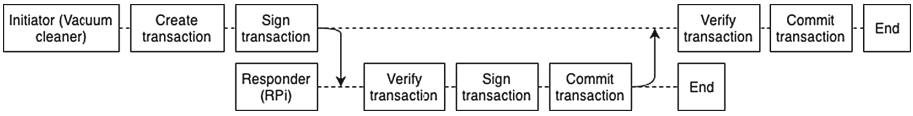

Fig. 3. Flows

The nodes of our system communicate with each other by means of a strictly defined format - Fig. 3. The diagram shows a block of the chain - the State, that is, the data set that is transmitted from one node to another is stored and clearly detected by the system. The state contains a link to the contract, participants (nodes that exchanged information) and some transaction metadata. Thanks to the Corda distributed ledger platform, we can transfer ZIP files from one node to another hashing an attachment, which ensures that we can reliably and securely transfer a map of the area with accompanying data.

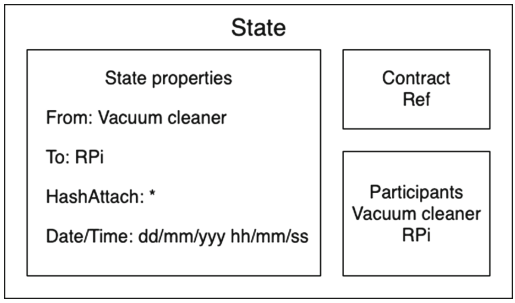

Fig. 4. State

Instead of global translation, Corda uses the node-to-node communication format. This means that in order to coordinate the updates of our ledger, it is necessary to specify the contractors and the necessary information to be transmitted. Corda automates a similar process with the help of flows - Fig. 4.

4 Implementation

One of the important issues in the development of the project is the algorithm (see Fig. 5) for connecting or gluing and synchronizing maps. It is fundamental - without it, the work of the entire system is impossible, since it will not be possible to distribute the task of cleaning between all vacuum cleaners and each robot will start to complete cleaning of the same large area. The process of combining maps can be represented as follows: before starting work, all vacuum cleaners in the same room are set sequentially in one line with a given initial distance of one meter. The first one to go is the cleaner with RPi, then from left to right, the others are turned on. When vacuum cleaners are turned on, each of them draws up a primary map using a LIDAR sensor. Possible variation of this solution - the user is given the opportunity through a special web form to put vacuum cleaners in the order he needs with the distance set by the user.

For further work of groups of vacuum cleaners, these maps must be synchronized both in groups and between groups. Inside the groups, each vacuum cleaner transmits its map and on-time to the RPi. At RPi, data is processed

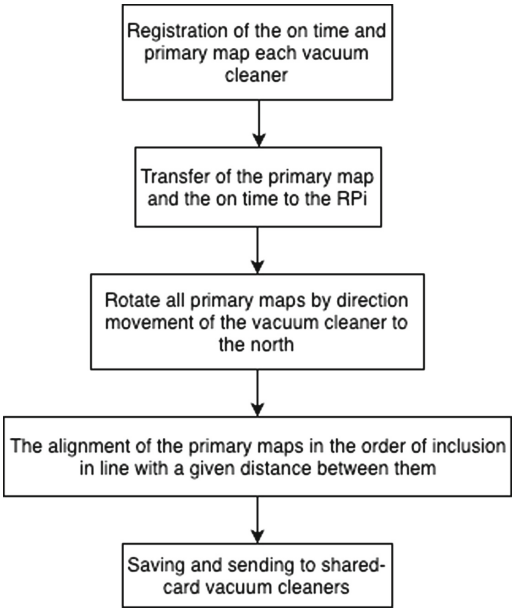

Fig. 5. Algorithm

— the maps from all vacuum cleaners are turned northward by the direction of movement of the robot, then overlapped with each other with a given initial distance in the order that the vacuum cleaners are turned on.

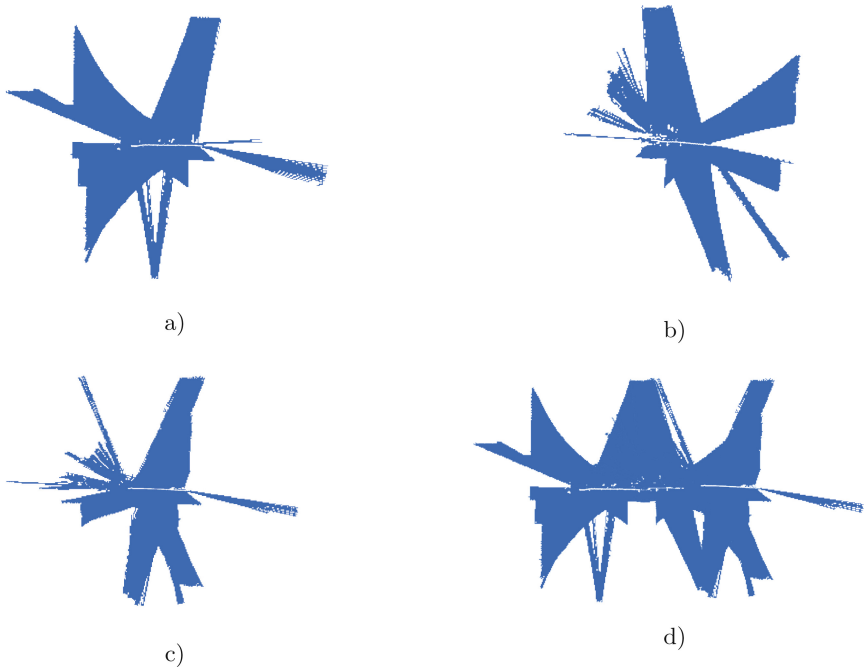

Fig. 6. Map pooling

Thus, knowing the turn-on time and, accordingly, the order, using the data from the compass, you can uniquely glue the maps from different vacuum cleaners into one (see Fig. 6).

Synchronization of maps from different vacuum cleaners within a group occurs through RPi - each vacuum cleaner sends a map to RPi every few seconds from which it is sent to the rest of the robots. In turn, RPi sends the group map to the cloud server, which sends it for synchronization to other RPi, and they to the robots of their group (see Fig. 7). All operations for storing and synchronizing maps are implemented via the blockchain.

There may be situations when the cleaning area is large and requires more than one group. In this case, with known limits of the cleaning area in the cloud, the area is divided into an amount equal to the number of necessary groups and then its cleaning zone is sent to each group and then RPi divides it into equal zones for each vacuum cleaner. The process of separating the harvesting area can be described as follows: the harvesting selection area is a rectangle, it is divided into equal parts on the larger side.

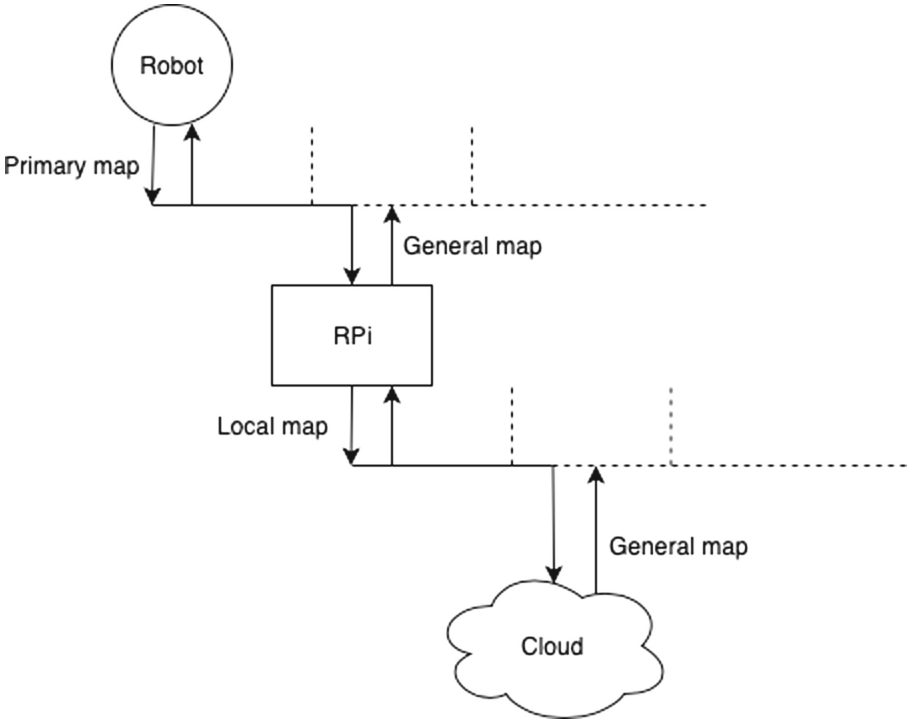

Fig. 7. Communication map

5 Corda Performance Evaluation

For our task the main issue is the speed of data transmission, so we estimated the time cost for data transmission between nodes within the circle. Figures 8 and 10 shows the architectures of the tests produced. For the first test, the ZIP file was uploaded to the central node of the group, after that the data was transferred sequentially from node to node, referring to the hash of the downloaded file. For the second test, the ZIP file was uploaded to the central node of the main group (group 3) and whistles to the nodes within its group, after which similar actions occurred within groups 1 and 2. A series of tests were produced: the file size and the number of nodes in the group changed (see Figs. 9 and 11).

In Fig. 10, the ZIP is uploaded to the Cloud node, after that this node whistled to the nodes in his group (group 3), that is, both RPi. RPi's in turn whistle to the nodes in their groups (groups 1 and 2), that is, all devices. The message will be considered to be distributed when all nodes in the groups have received it. The number of nodes in groups 1 and 2 ranged from 2 to 5, for which tests were performed (see Fig. 11).

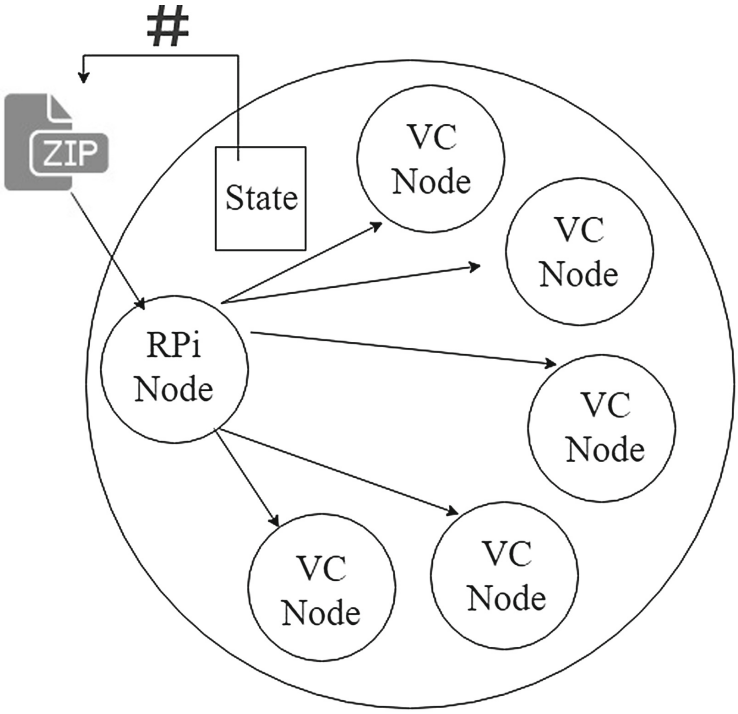

Fig. 8. Group architecture

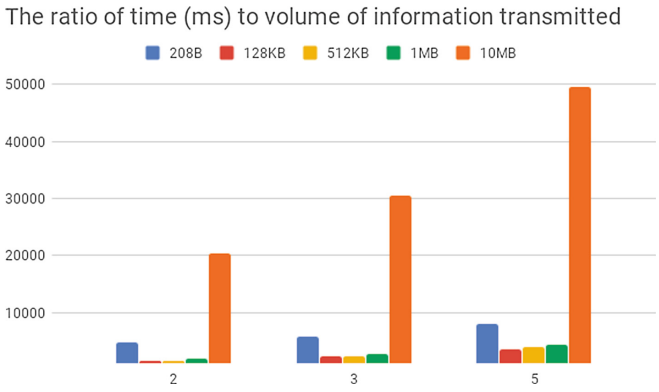

Fig. 9. Test results

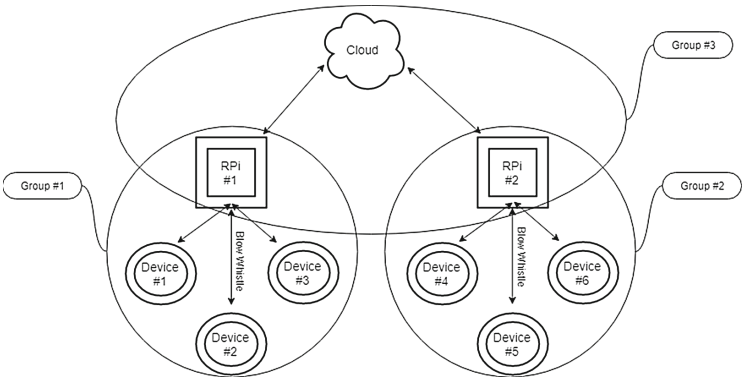

Fig. 10. Groups architecture

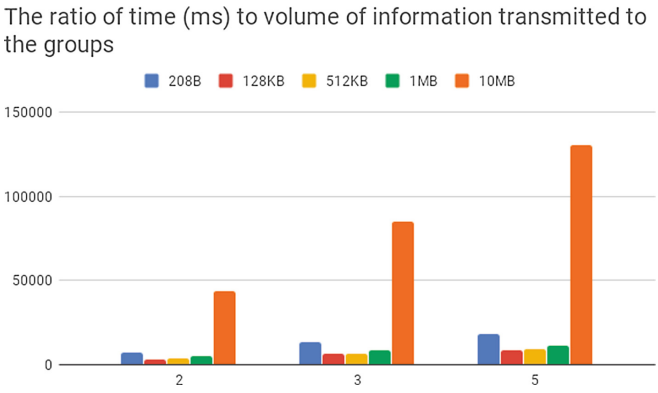

Fig. 11. Test results

6 Conclusion

In the course of creating a control and monitoring system for groups of vacuum cleaning robots, an architecture was created and prototyping of its main components was made. An algorithm for combining maps of several vacuum cleaners is presented and tested, an algorithm for the interaction of system nodes using distributed ledgers to solve key problems of security, stability, and scalability has been compiled. In the future, it is planned to optimize the work of the existing algorithms and will aim at providing user convenience.

References

1. Giese, D.: Having fun with IoT: reverse engineering and hacking of xiaomi IoT devices. DEFCON 26 (2018)
2. Atlam, H.F., Wills, G.B.: Intersections between IoT and distributed ledger. In: Advances in Computers, Chap. 3. Elsevier (2019)

3. Zhang, K., Vitenberg, R., Jacobsen, H.: Deconstructing blockchains: concepts, systems, and insights. In: Proceedings of the 12th ACM International Conference. ACM (2018)
4. Kshetri, N.: Can blockchain strengthen the Internet of Things? IEEE IT Prof. **19**(4), 68–72 (2017)
5. Zyskind, G., Nathan, O., Pentland, A.S.: Decentralizing privacy: using blockchain to protect personal data. In: 2015 Proceedings of the IEEE Security and Privacy Workshops. SPW 2015, pp. 180–184. IEEE (2015)
6. Lyu, X., et al.: Optimal schedule of mobile edge computing for Internet of Things using partial information. IEEE J. Sel. Areas Commun. **35**(11), 2606–2615 (2017)
7. Corda. https://github.com/corda/corda. Accessed 21 Mar 2019
8. Corda Documentation. https://docs.corda.net. Accessed 21 Mar 2019
9. Dustcloud. https://github.com/dgiese/dustcloud. Accessed 21 Mar 2019
10. Dustcloud. https://github.com/Hypfer/Valetudo. Accessed 21 Mar 2019