Using Blockchain-based approach for building the system events logging service

G Silkina\textsuperscript{1}, M Barabanova\textsuperscript{2}, S Gazul\textsuperscript{2} and V Kiyaev\textsuperscript{2}

\textsuperscript{1}Peter the Great St. Petersburg Polytechnic University, Polytechnicheskaya str. 29, St. Petersburg, 195251, Russia
\textsuperscript{2}St. Petersburg State University of Economics, Sadovaya str., St. Petersburg, 191023, Russia

\textsuperscript{*}E-mail: stanislav@gazul.ru

Abstract. The article deals with Blockchain technology, namely with its aspect of forming distributed registries. It serves as the platform for creating and storing corporate information resources which need higher protection. A lot of marketing uproar on cryptocurrency is feeding the interest in employing this technology for to enable cryptocurrency operation. But not only this factor explains the technology success. Blockchain technology also favorably compares to classic centralized registries and databases. The article describes the journaling service of data computing processes and events system, both based on the Blockchain technology. We display all the advantages of the existing Blockchain types for developing corporate journaling systems. And we also show the use of such a system for medium and large organizations, allowing for their computational capabilities and potential for scaling and developing such a system.

1. Introduction

The digitalization process in the economy various sectors is rapidly increasing the corporate data volumes [1; 2]. To ensure competitiveness, organizations have to increasingly use not only classic information systems, but also to apply various modern methodologies, models and services [5]. Due to this, supporting and developing modern corporate IT infrastructure becomes more and more difficult, with time [6; 11].

One of the most important tools used for administrating and supporting information systems is the subsystem of journaling system events and collisions. In case of technical failures or incidents - system errors, failures, unauthorized access to information system resources, data leaks, virus or DDOS attacks - the IT infrastructure administrators or the devops can use a system log to quickly identify the fact, time and essence of the incident. Such information in most cases allows to identify the cause of the events, to eliminate the resulting problem or to assess the level of adverse external influence (risk level) with minimal damage.

However, the journaling systems are mostly functioning on specific service levels and in various formats, e.g. server solutions from Canonical, Docker, Microsoft, IBM, etc.

These important problems, in our opinion stimulate the development of a specialized service able to log all events simultaneously in all key nodes of an organization’s information infrastructure and would be an additional tool for generating efficient network monitoring.
While analyzing the platforms and tools for creating the above-described journaling service, such as storing log files in file storage, databases, etc., the authors have identified the following problems complicating the generation of unified logging system - the different formats system logs, storing system logs in different locations (this may be relevant for some Linux server systems and applications, but especially for various Windows systems and applications). Besides, with the classic approach and use of standard tools for maintaining system logs, there is a possibility of unauthorized modification, replacement or deletion at various levels of administration.

In this regard, the authors focused on the capabilities and features of the Blockchain technology. In our opinion, this technology has not yet completely unlocked its potential against the background of marketing noise and speculative hype around cryptocurrencies. Thus, in accordance with the well-known Gartner Hype Cycle trend, the Blockchain technology will transfer to the “Platoeu of Productivity” stage only a few years after its hype peak, as in case with cloud computing after 2009–2011.

For confirmation of this hypothesis, we consider two graphs (figure 1 and 2). The first graph shows the worldwide dynamics of the user requests’ number for “Blockchain” keyword in the Google search system (figure 1).

![Figure 1](https://example.com/figure1.png)

**Figure 1.** Number of requests for the keyword "Blockchain".

The second figure shows the rate dynamics of the most worldwide popular (by the number of users) cryptocurrency— Bitcoin (figure 2).

![Figure 2](https://example.com/figure2.png)

**Figure 2.** Dynamics of Bitcoin cryptocurrency rate for 5 years.

In comparing the presented graphs, it becomes clear that the marketing hype around the most popular cryptocurrency Bitcoin aroused interest in Blockchain and vice versa. The correlation coefficient R on the segment 2017–2019 strives for 1.

Currently, both the hype and the interest in technology have markedly decreased. However, we can successfully use them in a large number of applications. The key advantages of the Blockchain technology are well known: the potential of decentralized data processing in distributed registries, the mechanisms ensuring the integrity and confidentiality of data, the high reliability and fault tolerance of the system, the ease of developing and generating the systems based on Blockchain technology.

2. **Features of the Blockchain-systems’ types**

All these advantages are well suited for problem solving when generating a journaling service for key nodes of the organization’s information infrastructure. So, for large organizations, it is possible to
support such a service by using the hardware of the organization’s various subdivisions (identified as nodes), that can be geographically distributed.

It important to maintain the strict reporting mechanism for providing the data security level. The logging system should record all the events and incidents related to the efficiency and security of the systems’ functioning. Record keeping should be complemented by a thorough analysis of registration data. The specialists on data infrastructure support should receive up-to-date information on the system status and arising incidents, with the above information not to be changed by any other division employees or not to be lost due to local software or hardware failure or external virus or DDOS attack. Developments for implementing such approach to generating a journaling system are already underway [3, 4].

However, many interesting questions and opportunities remain, to be pointed out in that article. To do this, let’s review the main types of Blockchain-systems.

The first type is a centralized blockchain-system with a trusted center. If there is a trusted center within the Blockchain-system, then after a certain period of time (or after a certain number of transactions) it forms a new block, supplying it not only with a hash sum, but also with its electronic signature. Each client of the system will be able to verify that all the blocks in the chain have been generated and confirmed by a trusted center and no one else. If such a trusted center is relevant and not compromised, then the attacker will be unable to modify the system log. However, in our opinion, using Blockchain technology in this case is redundant, since with a trusted center available, one can simply refer to it by requesting to sign each transaction, with adding time and a serial number to it. The number provides the order and the impossibility of adding (deleting) transactions from the chain, the electronic signature of the trusted center - the impossibility of modifying specific transactions.

Due to this, we think that such Blockchain type will not be suitable for generating logging systems. The second type is a centralized Blockchain-system without a guaranteed trusted center. In case of no trusted center allocated among the Blockchain-system centers, the guarantee is required that any user of the system will be unable to recreate the whole chain of blocks, deleting data from any block or adding another one. To provide that guarantee, we can use the following two methods [3]. The first method requires using the additional trusted repository. After creating the next block, the center sends the hash code from the new block to the trusted (and independent from this center) storage. Trusted storage should not accept any changes to the hash codes of already created blocks. The stored data volume may be small as compared to the total journal volume. In our opinion, such option may be quite suitable for generating journaling systems of system processes logging for medium and large distributed organizations.

The second method is to add a timestamp generated by a trusted time center to each block. Such timestamp should contain the generation time and the electronic signature of the center. The signature should be calculated on the basis of the hash code of the block and the timestamp. In case the “non-trusted” center wants to re-create part of the block chain, a gap in the timestamps will be fixed. It should be noted that this method does not guarantee such a center from simultaneously generating two chains of blocks, supplementing them with correct timestamps, and then replacing one chain by another.

The third type is decentralized blockchain system. Currently, decentralized blockchain systems are of great interest. Such systems have no dedicated centers for generating blocks. Every participant of such system can take a set of transactions (records) expecting to be included into the system log and create a new block for recording it in the general blockchain. Since this issue has been widely covered and actively discussed, there is no need to discuss in detail the potentials and advantages of the basic type of Blockchain-systems.

The fourth type is blockchain-similar systems. The current Blockchain technology uses peer-to-peer decentralized networks without dedicated center with unlimited number of nodes. In our case, this is not so. In this regard, the authors present, below, the conceptual description of the proposed
Blockchain-similar system for journaling events, which the authors named Blockchain-like Event Journaling System (BEJS).

3. BEJS levels description

The system architecture is shown in figure 3 and includes three levels, with their description presented below. Let’s review them in detail.

The first level is Event Collectors Level (ECL). The event collector (EC) creates a common pool of all unrecorded events of system logs throughout all system nodes. Each record is provided with the department code, title and ID of the source subsystem, as well as with the creation timestamp. The required event level is also specified, along with their classifier enabling the events to reach the relevant pool. For example: ordinary events, incidents, failures, critical errors, etc.

The second level is the Journaling Level (JL). Operating agents of that level are named journaling agents (JA). Such agents “collect” new blocks from a pool of unregistered system log events generated at the ECL level, and sign them with the electronic digital signature (figure 3).

![Figure 3. BEJS levels.](image)

The operating algorithm of the JA is described below:

- collect from the pool of unregistered system log events the ones fitting into 1 block (1-3 MB) and having the earliest timestamp (showed up earlier). In the case of the added event not fully fitting in the current block it should be rejected for fitting in the forming block and remain in reserve for the next block;
- add the previous block data to the block;
- add your own data (JA) to the block (for accountability).

The block structure for the BEJS can be selected based on the requirements of a specific IT infrastructure. We propose the following typical block structure for such systems (table 1).

| Field name | Description | Volume (bytes) |
|------------|-------------|----------------|
| ADDRESS    | Address is a public key generated by an | 4              |
asymmetric encryption algorithm (for example, RSA), based on the private key specified for a specific JA.

**BINDING**

The hash is calculated by using SHA256 (or a more recent version of SHA-512) from the previous block address and from the sum of the hashes of all the current block transactions. This calculation requires the previous block address.

**TIME**

Block timestamp.

**JOURNALLER_ID**

ID JA that created the block.

**EVENTS**

The events of the system logs from the pool, signed by digital signature (GOST R 34.10-2012).

In case the next block is simultaneously generated by two or more JA, and subject to the complete identity of the blocks, any of the generated blocks should be selected. If the blocks are not identical, the larger block should be selected.

Such cases can occur quite rarely during the normal system operation mode, since each of the JA follows the same algorithm for including events in the block.

The third level is Verification and Confirmation Level (VCL). The agents of the verification and confirmation level (AVCL) contain all records available in the BEJS, provide backup of such data, confirm, record, and add new blocks to the general blockchain.

Agents of this level are checking up whether the JA sending the new block really has the rights to generate the block with the data it contains (for example, does this JA is really relevant to the data systems from where the events were placed in the current block). After successful confirmation by AVCL agent, events in the block are removed from the pool of unregistered events (ECL level).

Otherwise, the block is not confirmed and not added to the general blockchain. All events stay in the pool of unregistered events. In this case, error notification and detailed incident report are delivered to the administrators of the relevant network segments.

Such AVCL, in distributed organizations, can be virtual machines or compute containers operating in various segments of corporate data center or in cloud. Those AVCL can be deployed at the distributed data centers of the departments across the world in order to support BEJS of the entire organization.

Note that cryptographic hashing algorithms are the fundamental security parts of Blockchain-similar systems, for example, the cryptographic algorithm SHA-256 or newer, SHA-512.

In case of selecting decentralized Blockchain the implementation of BEJS can be very similar to the architecture of Blockchains using cryptocurrencies, for example Bitcoin.

### 4. Development and technical support of BEJS

As noted earlier, the approach called Superchain [3] may well be used for implementing flexibilities of controlling access to data, as well as for general scaling of the above logging system architecture for large, distributed organizations.

In this case, each subdivision (branch) of the organization must have its own BEJS for accumulating data from the local system logs of this subdivision (branch) data systems operating by the above algorithm. Local units entering this system must be synchronized in the BEJS organization.

Such Blockchain-similar system can be called Superchain, and its operation can be maintained by the data center of the organization’s head unit.

In this case, each unit's BEJS becomes a JA agent and operates on the general confirmation and verification level (General Confirmation and Verification Level, GCVL (figure 4)).
In general, the described architecture of journaling system, in our opinion, is relevant and interesting due to the wide range of technical potential for optimal (in terms of data center capacity utilization) generation of such systems. For example, agents from all levels of such system can be automatically launched or created by orchestration system in convenient time (for example, after business hours, with diminished data center load) using Docker containers or virtual machines [8]. And they can be stopped the moment the data center resources are needed for other tasks during business hours [10]. Moreover, it is possible to launch any agents from different levels of BEJS in the cloud [2].

The performance comparison conducted by us [7], as well as by other experts [9], including experts from IBM, shows that a swarm of containers (Docker Swarm) running within a computing cluster can easily cope with the tasks of efficient and secure journaling based on Blockchain-similar system.

5. Conclusion
In conclusion, we would like to emphasize the actuality of viewing Blockchain technology as the platform for generating various corporate IT services. In particular, we were able to show the potential of using this technology for generating corporate IT infrastructure journaling systems. The approach to creating a Blockchain-similar corporate journaling system (BEJS) presented in the article allows us to solve a sufficiently large number of problems arising with administering dynamically configured hybrid distributed information systems.

In addition, Blockchain, as a technology, has undoubtedly promising prospects. By various estimates, the formation of blockchain-using startups will attract investments exceeding $ 5 billion in 2018–2020. Blockchain-based crowdfunding (ICO) has already been actively used to attract investments in startups. This will make this technology an alternative to venture investment in traditional systems of storing and processing Big Data in all areas of economic and social activity.

References
[1] Barabanova M, Lebedeva L, Rastova Y and Uvarov S 2018 Economic Annals-XXI 173 32-7
[2] Lobanov O, Minakov V, Minakova T and Schugoreva V 2017 Communications in Computer and Information Science 745 347-61
[3] Pourmajidi W and Miranskyy A 2018 IEEE International Conference on Cloud Computing, CLOUD 2018-July 978-82
[4] Shekhtman L M and Waisbard E 2018 SYSTOR 2018 – Proc. of the 11th ACM Int. Systems and
Storage Conf. 131

[5] Silkina G 2017 *Proc. of the 29th Int. Business Information Management Association Conf. - Education Excellence and Innovation Management through Vision 2020: From Regional Development Sustainability to Global Economic Growth* 1165-76

[6] Silkina G 2018 *IOP Conference Series: Materials Science and Engineering* **497** 012018

[7] Trofimov V, Kiyaev V and Gazul S 2017 *Proc. of 2017 20th IEEE Int. Conf. on Soft Computing and Measurements, SCM 2017* 788-91

[8] Yusupov R and Musaev A 2017 *SPIIRAS Proceedings* **2** 51 5-34

[9] Taherizadeh S, Stankovski V and Grobelnik M 2018 *Sensors* **18** 9 2938

[10] Taherizadeh S, Jones A C, Taylor I, Zhao Z and Stankovski V 2018 *J. of Systems and Software* **136** 19-38

[11] Vasileva O and Kiyaev V 2018 *Proc. of the 3rd Int. Scientific Conf. on Int. Information Technologies for Industry, IITI 2018, Advances in intelligent systems and computing* **874** 401-9