ABOUT SOME OF THE BLOCKCHAIN PROBLEMS

A.V. Bogdanov [1], A.B. Degtyarev [1], V.V. Korkhov [1], M. Kamande [1], O.O. Iakushkin [1], V. Khvatov [2]

[1] *Saint Petersburg State University, 7/9 Universitetskaya nab., St. Petersburg, 199034, Russia*

[2] *BGX, Toronto, Canada*

E-mail: a.v.bogdanov@spbu.ru

This year the Blockchain technology celebrates ten years since its inception in 2008. The technology is in its third generation now, however many issues still exist and the fourth generation is already anticipated. In this paper we consider some of the problems of Blockchain 3.0 and discuss possible approaches to their resolution on the way to the next generation of the technology Blockchain 4.0.

Keywords: blockchain, consensus, distributed ledger

1. Introduction

Blockchain is a special data structure usually implemented as a linked list (block chain), in form of a distributed system, where a copy of the list is stored on many computers (nodes) and synchronized using a special protocol (consensus).

The Blockchain solution was proposed by the legendary founder of the first successful Bitcoin cryptocurrency network, Satoshi Nakamoto, in the form of a practical implementation of bitcoin as its public translation ledger. Although many of the technological solutions were laid in the first protocol implementation (block encryption, the double-spending problem solution, currency mining), for some time the approach remained an application to Bitcoin and an integral part of this solution only. This time, which can be called the Blockchain 1.0 generation, for several years, the crypto community tested the system for various types of attacks and searched for vulnerabilities in the protocol, and around the original system developed its own infrastructure of access to Bitcoin nodes: wallets, mining programs, etc.

In 2013, the young founder of Bitcoin Magazine, Vitalik Buterin, proposed a new concept for the blockchain network called Ethereum. The main difference of this system was the ability to execute small unchangeable programs, smart contracts. Ethereum was launched on July 30, 2015 and revealed a new generation of blockchain systems - Blockchain 2.0. It was Ethereum that brought blockchain technology to a new level, allowing for distributed computing as part of a new economy. This gave a start to many projects by implementing their own virtual currency, including secondary tokens.

The growth of the crypto economy over the following years brought into use a whole pool of technological solutions, some of which were aimed at solving applied solutions, others were looking for new and more efficient algorithms.

2. Consensus

In addition to the blockchain structure, the idea of Bitcoin was supplemented with a number of other possibilities aimed at supporting the structure itself and its operability with a number of cryptographic functions. In essence, this system stores the flow of transactions within itself, in the process of implementing which, the issue of additional currency (Bitcoin) takes place in favor of network-supporting enthusiasts – miners. Information is stored as a chain of blocks, a block contains transactions, each block header contains hashes of each transaction, as well as a hash of the previous block header. The result is an immutable chain of blocks, which allows you to see all the transactions while maintaining the anonymity of those behind these transfers. Such a system leads to a high stability of the transaction formation process, the emergence of a method for decentralized processing of transactions, data exchange between nodes (node consensus), and resistance to attempts to substitute information.

The way to achieve consensus through computational work is called Proof-of-Work (PoW). Thanks to PoW, Bitcoin is able to function on millions of computers, but it has several disadvantages: it leads to a significant slowdown in synchronization operations (more than 10 minutes to form a block), and is also accompanied by a waste of energy.

In 1982, a group of scientists (Leslie Lamport, Marshall Pease, Robert Shostak) published a document that outlined the problem of reliability in a decentralized system [1]. In the “The Byzantine Generals Problem” the authors suggested a mental experiment: “Imagine that a group of generals, each command part of the Byzantine army, surrounds the enemy city. Generals can only communicate as messengers, but in order to conquer the city, they must agree on a battle plan. The problem is that one or more generals can be a traitor who will try to distort messages and sabotage the plan. The question is, how many treacherous generals can there be in the army, so that it can still act as a single force?”

The solution appeared in 1999, when Miguel Castro and Barbara Liskov introduced a practical byzantine fault tolerance algorithm (PBFT) [2]. PBFT can handle a huge amount of direct peer-to-peer (or distributed) messages with minimal delay. This means that programmers can create secure and fault-tolerant private distributed networks. Since 1999, PBFT has been implemented in different ways, and it has been refined in several technological approximations. The previously mentioned Proof-of-

Work method is a revision of PBFT – users must repeatedly run the corresponding algorithms to verify the transactions of other system participants.

In a broad sense, any distributed system operating in a non-trusted environment should provide Byzantine Fault Tolerance, resistance to various types of attacks – actions of "bad" network nodes. A distributed system, in contrast to a centralized system, must select "correct" transactions, avoiding such vulnerabilities as double-spending.

3. Decentralization and distribution

A system can be built as decentralized and distributed in varying degrees. At the same time, its architecture and topology will be different in terms of decisions made on the number of nodes, the order of decisions made, tasks to be solved, load profiles. According to the well-known DCS theorem, decentralization, consensus and scalability form a triangle that limits the chosen solution: the harder the consensus and the greater the decentralization, the harder it is to achieve the desired performance of a distributed system (see Figure 1).

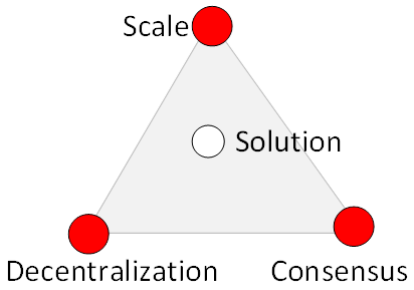

Figure 1. DSC theorem

The implementations of various synchronization mechanisms between nodes in a distributed system (consensus mechanism) are described in detail in [3].

The implementation of the synchronization mechanism between nodes in a distributed system (consensus mechanism) in a more general approach is determined by the following parameters:

- **Decentralized governance:** a single central authority cannot ensure the completion of a transaction.
- **Quorum structure:** Nodes exchange messages in predestination (paths that may include steps or levels).
- **Authentication:** this process provides the means to verify the identity of participants.
- **Integrity:** it provides verification of the integrity of a transaction (for example, mathematically by means of cryptography).
- **Non-repudiation:** provides a means to verify that the intended sender actually sent the message.
- **Privacy:** this helps ensure that only the intended recipient can read the message.
- **Fault tolerance:** The network works efficiently and quickly, even if some nodes or servers do not work or are slow.
- **Performance:** takes into account bandwidth, survivability, scalability and latency.

4. Scalability

The problem of transaction processing performance directly follows from the architecture of the classic blockchain. Indeed, the solution in which each node must be ready to process any transaction becomes weaker with the growth of nodes and the number of processed transactions due to the delay in the exchange between nodes. At the same time, the mechanism for achieving consensus is the most vulnerable link, a critical component responsible for the performance of a distributed system.

The solution to this problem may lie in additions to the existing architecture, as well as a more radical change in the very principle of the algorithm at the expense of all the compromise in terms of decentralization or the complexity of the mechanism.
Here are the most common ways to solve the problem:

- **Optimization of the data structure** leads to reduction in the volume of processed and transmitted data. In particular, SegWit, which optimizes the structure of data storage in blocks of Bitcoin and some other currencies (Litecoin, Vertcoin), follows this path;
- **Optimization of block sizes**, e.g. increasing the size of a block in Bitcoin to 2 MB: increasing the block size will increase the speed of processing transactions, since more transactions will be included at a time;
- **Off-chain state channels**: the formation of secure computing outside the blockchain network and then saving the results to the network. In fact, it works in three stages: a) State fixing through some mechanism (for example, smart contract and multi-signature), b) Data exchange between two participants without the involvement of the network, c) Providing status back to the blockchain network, closing the channel and unlocking the state. Implementations of this method are Lightning Network and Raiden;
- **Sharding**: processing segmentation, like in distributed databases. Transactions are clustered according to different shards and, depending on the cluster, are sent to a different server. This approach has found application, in particular, with federative consensus. However, in general, this approach is not applicable to all practical problems;
- **Plasma**: an approach in which payment channels (sidechains) are formed according to embedded smart contracts. Although it looks similar to the off-chain state channels, its fundamental difference is in the use of mechanisms built into the blockchain. The Proof of Fraud mechanism is integrated into the Plasma project implemented over Ethereum: the smart contract logic that allows users to quickly withdraw funds from the side chain to the main blockchain protected by full-fledged mining in case of an attack;
- **Off-chain computation** is another approach that takes transaction calculations out of the main network. An example of such a solution is the TrueBit network. At the expense of additional nodes, Solvers, smart contracts are processed outside the main network, a deposit is made to the Solver's account. In case of successful resolution, Solver is rewarded. Otherwise, the deposit is withdrawn, and the conflict is resolved in the main network by mining, the so-called Verification Games;
- **Changing Consensus Algorithm**: as part of this approach, the very architecture of data synchronization between nodes is changing. Recently, new models have spread, significantly raising the speed of data processing.

5. Proposed approach

The technologies listed above solve processing and consensus problems in different ways. Currently, there is a rapid development of this area, and there is no single settled solution. In our opinion, the most serious problem is a significant increase in the speed of transactions and clearing. Acceptance of cryptocurrency can seriously slow down the turnover of funds and, in practice, can take up to a week at a price up to 30% of the entire transaction. The p2p payment channels used to accelerate seriously increase the speed of transactions, however, the basic principle of the technology is violated: it is impossible to simultaneously change the state of the system in all its nodes. There are some discrete time intervals in which the system is not synchronized. At the same time, there are such opportunities for significant security breaches as:

- Carrying out third-party transactions through already established chains of reliable users
- The possibility of including third parties as an additional node of the Blockchain system with the participation of an unscrupulous partner of a streamlined chain.

It is clear that without solving security problems, it is impossible to talk about the practical use of distributed registry technologies. It seems to us that solving these problems requires an integrated approach to technology based on the creation of hybrid systems with several mechanisms for ensuring an acceptable transaction rate and a given level of security based on a rigorous mathematical model.

- To solve these issues we propose to use of the following concept [4]:
- At the node level, the use of virtual servers that make up the node complex (a.k.a. virtual supercomputers) [5,6];
- Construction of the topological structure of nodes, when the transaction processing logic itself is carried out in a specific cluster and only then spreads to the entire network (by analogy with FBA Stellar);
- As a base layer for storing information, taking advantage of DAG structures;
- Division of processing into intranet and extranet (interacting with channels of other currencies)

6. Conclusion

In this paper, we focused on the analysis of integration of distributed ledger systems with the practical tasks of the business environment and related problems of the technology. While today there is no silver bullet, such an algorithm for building distributed ledgers that would satisfy all the needs of abstract business problems does not yet exist, we identified the actual architecture problems in solving which such an algorithm could become a reality, and suggested an approach based on which we plan to form such an algorithm in future works. We plan to concentrate on the development of a new generation of consensus algorithms that will address the current blockchain problems based on the emerging BGX platform [4].

Acknowledgement

This work was partially supported by the grant of Saint Petersburg State University no. 26520170.

References

[1] Leslie Lamport, Robert Shostak, and Marshall Pease. 1982. The Byzantine Generals Problem. *ACM Trans. Program. Lang. Syst.* 4, 3 (July 1982), 382-401. DOI: 10.1145/357172.357176

[2] Miguel Castro and Barbara Liskov. 1999. Practical Byzantine fault tolerance. In *Proceedings of the third symposium on Operating systems design and implementation* (OSDI '99). USENIX Association, Berkeley, CA, USA, 173-186.

[3] Consensus: Immutable agreement for the Internet of Value, KPMG, 2016, URL: https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmg-blockchain-consensus-mechanism.pdf

[4] BGX Blue paper, 2018, URL: https://bgx.ai/documents/en/BGX_BLUEPAPER_1.0.pdf

[5] A. Bogdanov, A. Degtyarev, V. Korkhov. Desktop supercomputer: what can it do? Physics of Particles and Nuclei Letters, 2017, Volume 14, Issue 7, pp 985–992 DOI: 10.1134/S1547477117070032

[6] Alexander Bogdanov, Alexander Degtyarev, Vladimir Korkhov, Vladimir Gaiduchok, Ivan Gankevich. Virtual Supercomputer as basis of Scientific Computing, in series: Horizons in Computer Science Research, vol. 11, eds.: Thomas S. Clary, pp. 159-198, Nova Science Publishers, 2015, ISBN: 978-1-63482-499-6