

# Architecture and basic conditions for organizing video hosting based on a p2p-network

*Irina Trubchik*<sup>1,\*</sup>, *Ilya Kalashnikov*<sup>1</sup>, *Kirill Ralko*<sup>1</sup>, *Nikilay Kovalev*<sup>1</sup>, and *Mikhail Arhipov*<sup>1</sup>

<sup>1</sup>Don State Technical University, 1, Gagarin ave., 344003, Rostov-on-Don, Russia

**Abstract.** The paper discusses the principles of developing a software package based on a distributed registry (proprietary blockchain) for performing applied tasks within a decentralized video portal. Such structure is concerned to distribute of user-generated content, implying the principles of "reverse economy" that suggest to reward proactive users of the service via sharing advertising revenue of the video portal amongst all participants in the advertising chain.

## 1 Introduction

The paper proposes the principle of an architectural solution for a software package based on a proprietary blockchain registry, designed to encourage user activity. This approach allows to implement one of the applied functions of the blockchain – monitoring the chronology of the consensus execution.

Unlike popular blockchain platforms, the proprietary system operates on the server side, rather than the client, what eliminates the burden on the computing power of users.

The innovativeness of the proposed approach is presented in the consensus of the distributed registry system, which allows to distribute advertising revenues in real monetary units amongst all participants in the content distribution chain (advertising aggregator, web portal, author, audience). The creation of a video portal in this project is a secondary mandatory task that is concerned to demonstrate the system's performance.

The basis for the software architecture is complex, according to the fundamental conditions of the distributed ledger system that is functioning as proprietary cryptographic system. Hereinafter, we will refer to it as "blockchain". Such title is referring to its architectural features, established through peer-to-peer network (P2P network) and based on the principle of distributing tasks and responsibilities between its nodes. The nodes of such a network are equal [1-3].

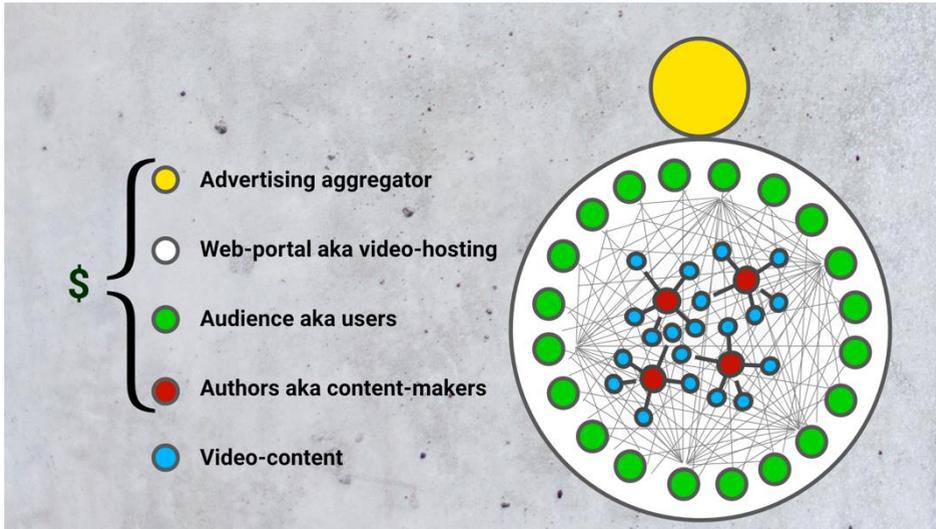
There is no hierarchy in a peer-to-peer network, as well as all its nodes are interconnected simultaneously providing and consuming data on equal terms. Such point makes the P2P network potentially resilient to individual node failures, since there is no single point of failure in such architectural construct.

---

\* Corresponding author: [trubchik@mail.ru](mailto:trubchik@mail.ru)

Importantly, all nodes in a peer-to-peer network are equal in rights, but their certain roles may differ. In particular, in the Bitcoin peer-to-peer network, clients vary greatly in functionality - from “*full nodes*” that store the entire history of the blockchain and give any block on request, to “*light nodes*” that store only headers and the last few blocks. This principle of distribution of roles was taken as a basis in this decision.

The main nodes of the system are: service users; promotional videos; comments in discussions; blocks with stored results of quantitative and qualitative analysis of user activity; interests of users (change over time); advertisement income from commercial views that is distributed amongst all stakeholders of the service (Figure 1).



**Fig. 1.** Income distribution structure amongst stakeholders of the portal VideoStake / architectural construct of the video-hosting.

The implementation of such architecture is shown in the startup VideoStake, which is now being developed on the basis of DSTU-students (headed by I.S. Kalashnikov). The interaction scheme of the nodes is shown in Figure 2.

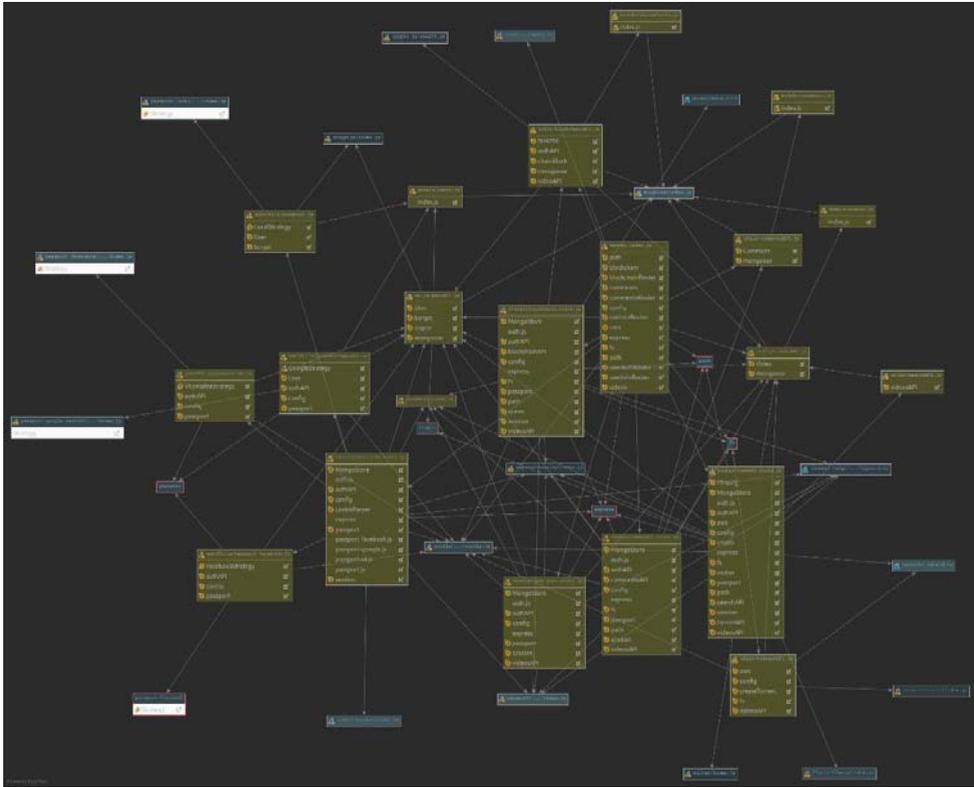
## 2 Backend components

The peculiarities of the organization of the peer-to-peer structure are closely related to the problem of finding information in it [4-6]. The search begins with the fact that the node, concerned to find a file, is regarded to generate a request containing the signs of this file. The task of the search algorithm is to ensure that this request is delivered in the most efficient way to the node(s) that actually has/have the desired file. The main stages of the search process, regardless of the specific algorithm, are:

- 1) formation of a search query;
- 2) sending a search query;
- 3) local execution of the request;
- 4) sending the search result;
- 5) processing of search results;

Different search algorithms differ mainly in the mechanism of forwarding and delivering the search query to the node with the desired information. Based on how the nodes of the structure are interconnected to each other and how information is searched in such network

(Figure 2), the approach of the structured P2P network was chosen for the implementation of the project.

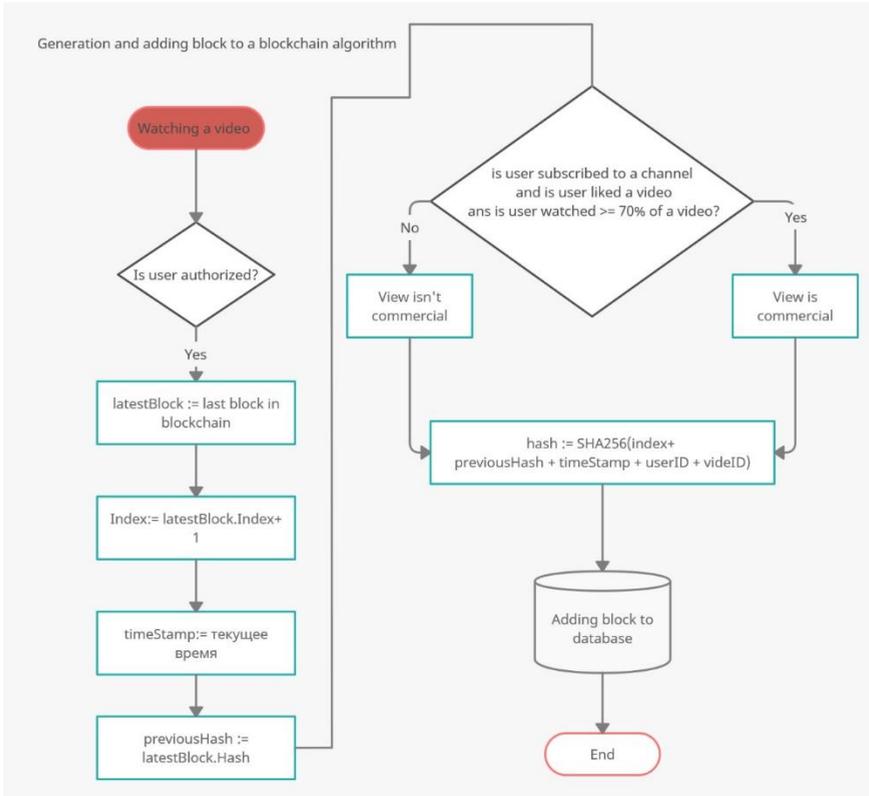


**Fig. 2.** Scheme of interaction of the nodes of the VideoStake project.

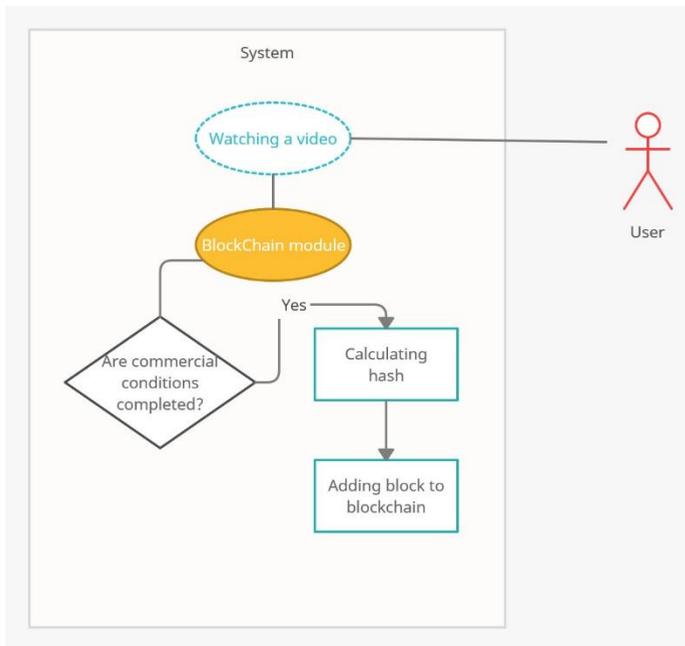
Structured peer-to-peer systems use routing algorithms to find nodes. The routing algorithm determines exactly where the target node should be located in the overlay structure. This process is closely related to the geometry of the P2P network and the connectivity or information stored at each node (Fig.3).

In this case, the most convenient is the DHT (Distributed Hash Table) - based routing algorithm, which uses the hash of the node identifier to form a uniformly distributed identifier space [7, 8].

The file identifier is generated using the same hash function. Hence, node IDs and data IDs fall into the same ID space. Files are usually stored at the closest node with a node ID greater than or equal to the file ID. This approach allows any node to find a specific file by its name. If the file you are looking for is not found on the node with the closest identifier, then such a file does not exist on the network (Fig.4). In our case, the SHA256 algorithm is used. An example of a class that implements this stage is shown in Figure 5.



**Fig. 3.** Algorithm for searching and collecting information for the VideoStake database.



**Fig. 4.** Blockchain selection algorithm.

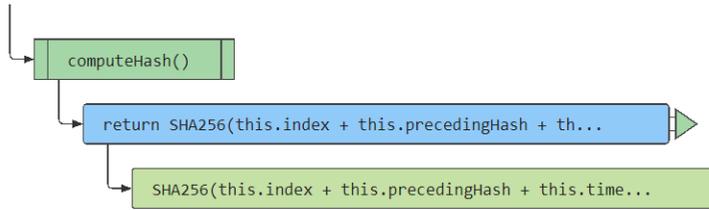


Fig. 5. Hashing a structure node.

### 3 Frontend components

For the web portal and related user functions of the VideoStake service, a web interface was developed that meets all modern requirements and principles of video hosting. Figure 6 shows a screenshot of the user authorization panel.

After logging into his account, the user can watch the videos of interest to him (Figure 7). After that, information about the funds collected through views is accumulated in his personal account.

In the user's personal account, it is possible to track the dynamics of the accumulation of funds collected by users, who is watching advertising videos (Figure 8).

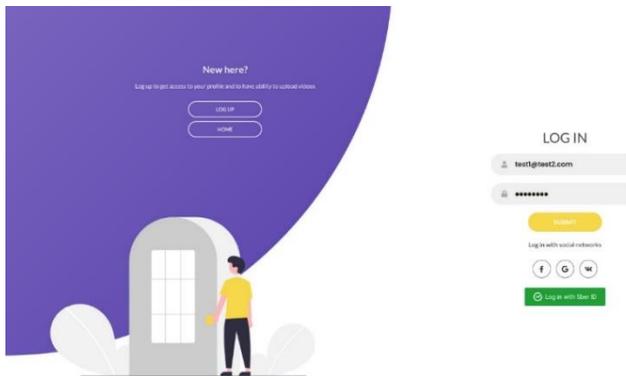


Fig. 6. Panel for user authorization in the service.

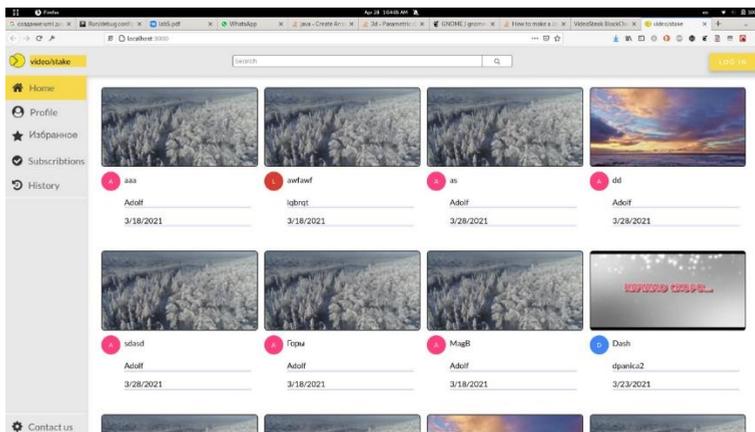
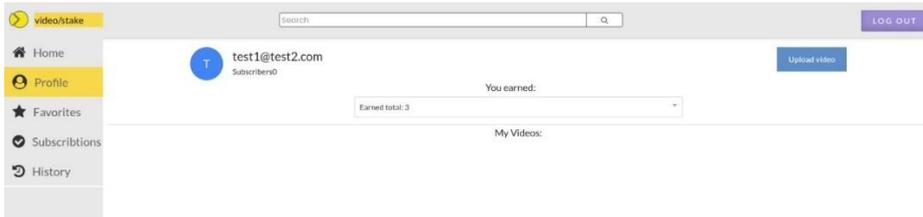


Fig. 7. Video selection page.



**Fig. 8.** Panel of the user's personal account.

## 4 Practical implementation

Taking into account a dynamic nature of digital technologies, it is important for practitioners and academics to understand why users intend to continue to be its users or not [9-11]. The study of the performance of the software complex was carried out as part of a group experiment of user experience.

Table 1 shows the result of a comparative calculation of income per 1 video channel with 4 pieces of video content uploaded per month. The comparison represents high effectivity of native advertisement towards classical ads that interrupt user experience of video watching. A possible operator for native ads might be “Mirriad”, a UK company that is concerned to add advertising by real-time editing video canvas via CV [12]. Virtual loans are used as real currency.

**Table 1.** Comparative analysis of the effectiveness of the VideoStake model. Figures are given in thousand rubles – Russian currency.

	VideoStake		YouTube	
Content maker	25% (> 50 K rub)	50% (6-24 K rub)	55% (6.6-26 K rub)	25% (3-12 K rub)
audience	45% (108 K rub)	40% (4.8-19 K rub)	0%	0%
video portal	10% (24 K rub)	10% (1.2-4.8 K rub)	45% (5.4-21 K rub)	45% (5.4-21 K rub)
Advertising aggregator	20% (48 K rub) Mirriad	Google, Yandex, Mail.ru	Google	30% (3.6-14 K rub) AIR, GTR, ScaleLab

## References

1. V. Murgul, *Procedia Engineering* **117**, 808-818 (2015) doi:10.1016/j.proeng.2015.08.145
2. *Computer Networking: A Top-Down Approach* — Kurose, Ross (2012)
3. Xuemin Shen, Heather Yu, *Handbook of Peer-to-Peer Networking* (2010)
4. M. Andreas, Antonopoulos: *Mastering Bitcoin: Programming the Open Blockchain, 2nd Edition* (2017)
5. M. Bellare, *New Proofs for NMAC and HMAC: Security without Collision-Resistance*

6. H. Krawczyk, *The Order of Encryption and Authentication for Protecting Communications*
7. M. Groot, *A Primer in Financial Data Management*, Chapter 5 — Data Management Tools and Techniques – Academic Press, 127-177 (2017) doi:10.1016/B978-0-12-809776-2.00005-3
8. A. Pólvara, S. Nascimento, J.S. Lourenço, F. Scapolo, *Technological Forecasting and Social Change* **157**, 120091 (2020) doi:10.1016/j.techfore.2020.120091
9. T. Macaulay, Chapter 13 – RIoT Control, 279-368 (2017) doi:10.1016/B978-0-12-419971-2.00013-3
10. S. Chaudhary, T. Schafeitel-Tähtinen, M. Helenius, E. Berki, *Computer Science Review* **33**, 69-90 (2019) doi:10.1016/j.cosrev.2019.03.002
11. K. Witkowski, *Procedia Engineering* **182**, 763-769 (2017) doi:10.1016/j.proeng.2017.03.197
12. M. Yan, R. Filieri, M. Gorton, *International Journal of Information Management*, **58**, 102315 (2021) doi:10.1016/j.ijinfomgt.2021.102315
13. Tae-Yang Kim, Dong-Hee Shin, *Telematics and Informatics* **34**, 1081-1092 (2017) doi:10.1016/j.tele.2017.04.016
14. Ch. Long, A. Chen, P. Pengcharoen, R. Chi-Wing Wong, *Information Systems* **88**, 101441 (2020) doi:10.1016/j.is.2019.101441
15. J.P. Jung, M.S. Zenati, A. Hamad, et al., *HPB* **21(6)**, 679-686 (2019) doi:10.1016/j.hpb.2018.10.012
16. S. El Massah, M. Mohieldin, *Ecological Economics* **169**, 106490 (2020) doi:10.1016/j.ecolecon.2019.106490