**Analyzing Differences in Real-life Performance for Decentralized Applications based on Holochain and Blockchain Models**

**Mid-point Report**

Harshit Hajela ([haha4350](mailto:haha4350@colorado.edu)), Pramod Kulkarni ([prku8035](mailto:prku8035@colorado.edu)), Nikhil Jain ([nija5462](mailto:nija5462@colorado.edu))

**Problem Statement**

Blockchain has transcended its initial scope for only developing currencies and is now increasingly being used to develop a variety of decentralized services and applications like storage, payload based decentralized apps, smart and secure IOT devices and smart contracts that ensure inviolability. Unfortunately, many of the intrinsic characteristics of the Blockchain model make it unsuitable as the backbone of choice for certain types of decentralized applications due to inherent scaling and performance limitations and high cost per transaction.

Holochain is an alternative decentralized architecture model that can similarly be used to build dApps that aims to solve these problems of scalability while retaining the benefits of decentralization. Its primary improvement is to do away with the idea of immediate global consensus to validate transactions and replaces it with a peer-based accountability system instead. Additionally, it also implements optimizations to how transactions are propagated and how the overall state is stored and replicated to improve speed and reduce storage requirements.

The objective of our project is to analyse the magnitude of the architectural differences between the two models by comparing the effect they have on the real-life performance of distributed applications at scale. We plan to do this by implementing a decentralized application using both the Holochain and Blockchain models and evaluating its performance on various metrics.

**Related Work**

The problems inherent in scaling a distributed ledger system with strong and immediate consistency like the one proposed in the original bitcoin paper [1] have been known for quite some time and are explored in “On Scaling Decentralized Blockchains” [2]. It investigates how designing for an acceptable level of throughput imposes strict theoretical limitations on the block size and the block interval, the former of which has shown an increasing trend for Bitcoin and thus predicts poor transaction throughput in the future. It proposes models for reparameterization of these while also accepting that only limited benefits can be achieved through this process.

It also discusses how there are a lot of redundancies in the blockchain storage model such as the lack of agent specific views and requiring complete copies of the entire ledger on all nodes, something that takes a lot of time (~4 days) to assemble from scratch.

Finally, it points out issues with the transaction propagation model that ends up requiring transmitting every transaction twice and the computation overhead with every node validating every single transaction it transmits, greatly limiting throughput.

Similar issues were also pointed out in [5]. The Holochain whitepaper [3] both expands upon these problems by showcasing how Blockchain based computation has extremely high resource footprint and is therefore extremely harmful to the environment. Specifically, the electricity consumption due to Bitcoin and Ethereum mining exceeds that of many countries [4]. In addition, the inefficient nature of Blockchain means that the cost incurred per transaction is extremely high.

It then suggests a model that improves upon Blockchain by incorporating many of the improvements suggested in [2] to increase performance and eliminate redundancy, such as implementing a different consensus mechanism in which every node can act as a notary and for every transaction, a random set of validator devices is recruited to perform validation. It also utilizes replicated DHTs to avoid storing the entire distributed ledger while also ensuring that all the information is preserved in the event of a network partition. With these improvements the paper claims that highly increased efficiency and that the same computation can be performed at 1/10000th of the cost as compared to Ethereum.

**Design and Implementation**

We plan to implement a large-scale decentralized messaging system where each message can have a very large number of recipients using both models. The reason behind choosing such a system is that it best captures the scale at which large scale decentralized applications run and where high throughput and availability are very important concerns.

One of the characteristics of the Holochain model is that the framework itself does not specify how each app should address potential collisions in the DHT and therefore app developers need to decide that for themselves, and this would also be one of the considerations that go into designing the application behaviour.

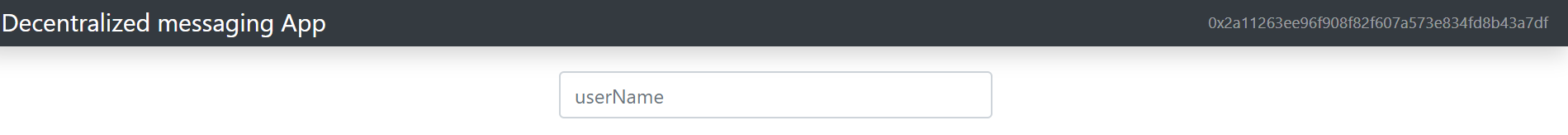
We shall be evaluating the application performance on the following metrics:-

1. Scalability – how well the system performs in response to increase in number of users and data
2. Efficiency – calculate latency of responses and memory and compute utilization
3. Reliability – judge the fault tolerance and recovery capabilities of the system
4. Availability – compare the uptime of the services at scale and with partial failures involved

For Blockchain implementation, we have used Ethereum. The wide community support and the faster processing unlike bitcoin makes it appropriate option to our problem statement. For initial implementation, we have used Ganache for creating personal blockchain for Ethereum Development and Truffle Framework for implementing the smart Contract.

Smart Contracts are the building blocks of blockchain applications. All the code on the blockchain is contained in smart contracts- data Structure, functions to manipulate data structure. Smart contracts are written in language called Solidity. Truffle framework provide compiler for executing the code.

For our implementation, we have used a message structure and a mapping list of messages. For participants we have used mapping of user. Other than this, we have two function defined in our contract-one to add new message and other to add new user. The frontend is written in html, bootstrap and for client-side code, we have used jquery. We have used web3 library to connect our client-side code to our blockchain. This library give access to the smart contracts. To connect our browser so that, we can accept the transaction on the blockchain, we are using Metamisk Browser extension.



This is our home page. You can see that on the right side we have the public key for the blockchain.

A screenshot of a cell phone

Description automatically generatedThis is metamisk showing no transaction has processed so far. We will add some user here now.

A screenshot of a cell phone

Description automatically generatedWhen we add a new user , we are manipulating our smart contract data. For any processing, we need to pay some fees to the node which is executing the code for us.

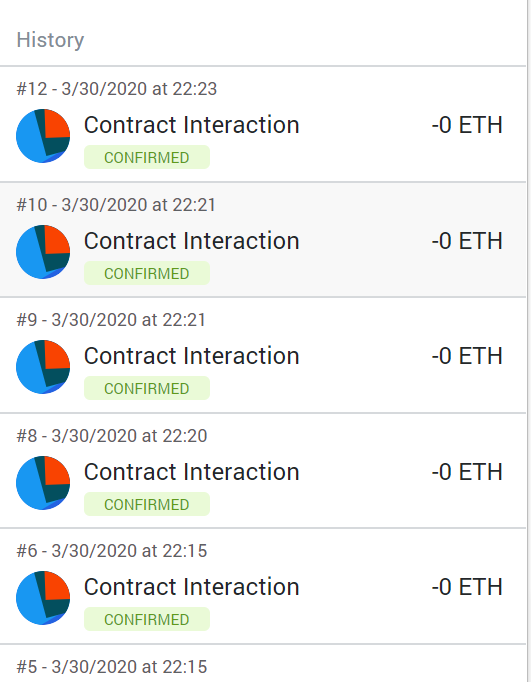
A screenshot of a cell phone

Description automatically generatedChat window with users joined the conversation.

A screenshot of a cell phone

Description automatically generated

Screenshot with the conversation.

 Each new message or new participant is a new transaction for decentralized Messaging App.

Giithub: <https://github.com/nikhiljain217/BlockChain-MessagingApp>

Preliminary results and Next Step

In this phase of the project, we have evaluated the architectural differences between Holochain and blockchain while implementing the messaging App.

One of the major differences between Blockchain and Holochain is the involvement of Users. In case of Blockchain, user does not have to be part of chain or processing node for the computation. The user is entity who is using the resources and paying a fee for doing that. The actual data is stored and manipulated by the blockchain nodes. But in case of Holochain, each user is using the resources as well as contributing to it. This explains why it is permissioned chain. As the nodes which are processing the request are also the nodes which are using the application.

Another major difference is how data is stored. In case of Blockchain, the whole smart contract is stored at every node. But in case of Holochain, not every node has all the data, and each node can have partial data which could be decided by the Application code. This also explains why Blockchain is Data-centric and Holochain is Agent-centric.

In the final phase of our project, we will compare the performance of Blockchain with Holochain based on the metrics mentioned above. Also, we will compare the metrics in terms of both number of users and number of messages.