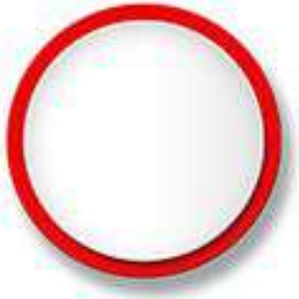
RanchiMall

Blockchain based Data Cloud

Technology

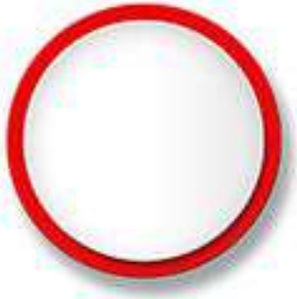


RANCHIMALL

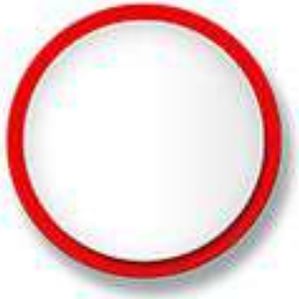
INTRODUCTION

Storing data directly in blockchain
at three big costs

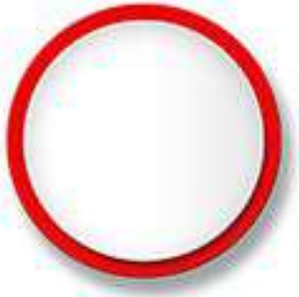
1. Token cost to get your data written in blockchain
2. Sacrifice of speeds needed in order to achieve consensus
3. Sacrifice of total data storage capacity as the same data is stored many times in each blockchain node



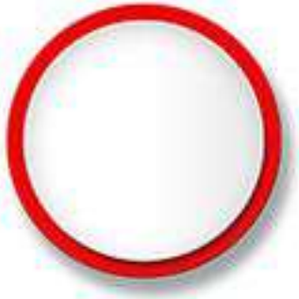
So for a practical decentralized application system, just the blockchain based data storage is not enough.



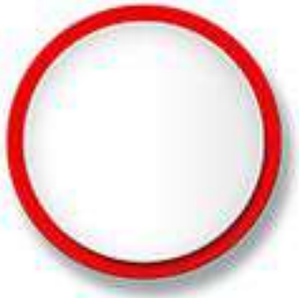
It needs to be supplemented by another system which is also decentralized, but can store data without the costs associated with direct data storage in blockchain.



Thats why we designed a
blockchain based data cloud.



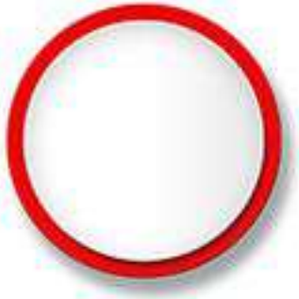
Blockchain based data cloud is decentralized, accessible directly by web, and uses FLO Blockchain ID to digitally sign every piece of data stored.



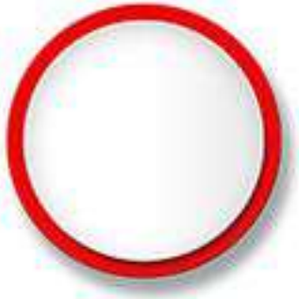
RANCHIMALL
INTRODUCTION

Usage of FLO ID to store digitally signed data is conceptually similar to blockhain data storage where every piece of data stored in the blockchain must also be digitally signed by blockchain ID.

In order to achieve the goals of decentralization, the cloud must be able to split all data to be stored among themselves, and also back enough copies among each other.



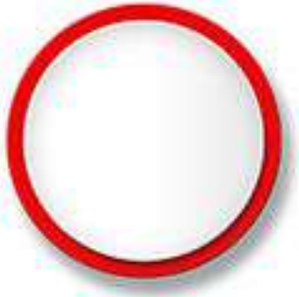
In RanchiMall data cloud every piece of data needs to be sent from a blockchain ID, and be addressed to a recipient of blockchain ID.



The data cloud consists of equivalent individual units called supernodes.

Each supernode runs all the logic needed to operate the data cloud.

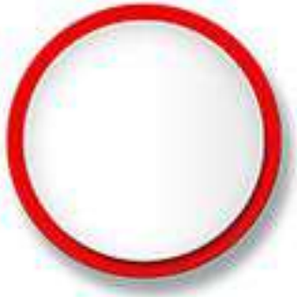
Each supernode is also responsible for storage of data allocated to it by system logic



Each supernode additionally has backup responsibilities for some other supernodes.

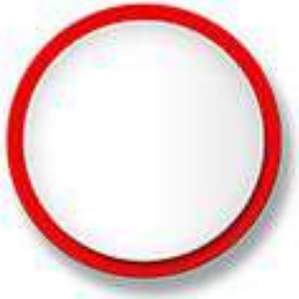
In case any supernode goes down for any reason, other supernodes need to detect it, and provide data services on behalf of the unavailable supernode.

When the unavailable supernode comes back, other supernodes have to restore the additional data that were received on behalf of it.



Now we need to address the question, which supernode stores what data.

So we need a mechanism to partition the entire data in such a way that any neutral entity like a browser client should be able to figure that out.



We already know every piece of stored data has sender blockchain ID, and receiver blockchain ID.

So we can use either of those blockchain IDs to divide the entire data among different supernodes in the cloud.

In RanchiMall Data cloud we have used receiverID to partition the data.

RanchiMall

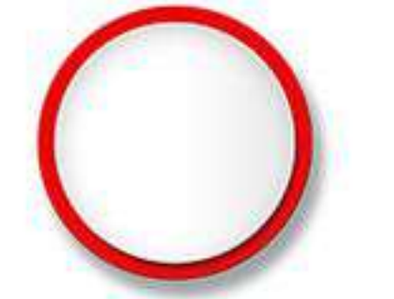
So the scheme we will use is this:

We will first determine the receiverID of data being stored.

Then we will allocate a FLO ID to every supernode as well.

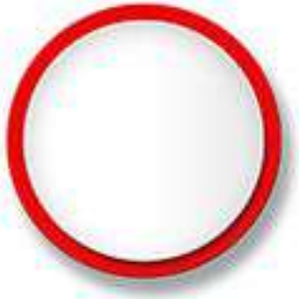
And then we will find an artificial way to measure distance between receiverID and every supernodeID.

The supernodeID which is closest to receiverID gets to store that data.



Similarly for backups, any supernode can compute distances from other supernodes.

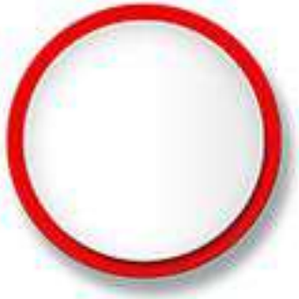
And when a supernode receives a piece of data to store, the closest supernodes can store the backups.



If we want data to be backed up more than once, then the system can be instructed from the blockchain to store backup for a supernode into two or three or four closet supernodes.

In case the backup data has to be retrieved by a supernode, then it can first ask the closest supernode to give it the data.

If closest is not available, then it can ask the next closest for the same backup.



In summary, the browser that needs to store the data in the cloud selects the closest supernode to data receiverID, and sends data to it.

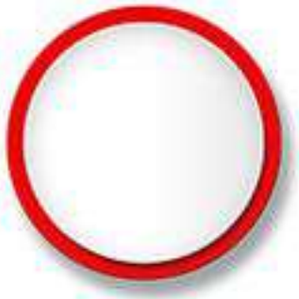
Any supernode selects the closest backup supernodes and stores its backup.

A supernode retrieves the backup data by progressively asking the closest supernodes one by one.

Role of Blockchain

There is a main application FLO ID.
Lets name it cloud ID.

The cloud ID can write the FLO IDs of all the authorized supernodes in the FLO Blockchain.

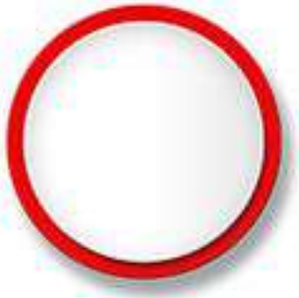


The browser clients can read the blockchain directly at cloud ID address, and retrieve the list of authorized supernodes.

Then the clients can start sending the data piece to be stored in the cloud directly to the correct supernode by selecting the closest one.

In order to read the data back from the cloud, the browser simply has to ask a given supernode:

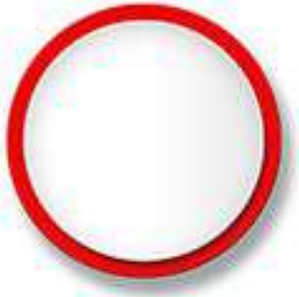
"Give me all data you have for a given receiverID."



This data cloud design can grow automatically and scale up to infinite data.

All the data is available on the browser.

The system is censor resistant.



For technical details, please refer to RanchiMall GitHub page for blockchain based data cloud.

RanchiMall uses blockchain based data cloud to store running transactional data for it's FLO based distributed applications.