An improved P2P File System Scheme based on IPFS and Blockchain

Yongle Chen, Hui Li* , Kejiao Li and Jiyang Zhang

Shenzhen Key Lab of Information Theory & Future Network Arch. Future Network PKU Lab of National Major Research Infrastructure
Shenzhen Engineering Lab of Converged Networking Technology PKU Inst. of Big Data Technology,
Huawei & PKU Jointly Engineering Lab of Future Network Based on SDN,

Shenzhen Graduate School, Peking University, P. R. China. Email: lih64@pkusz.edu.cn

2017 IEEE International Conference on Big Data (BIGDATA)

중앙대학교 소프트웨어학과
20150249 김용재

Introduction

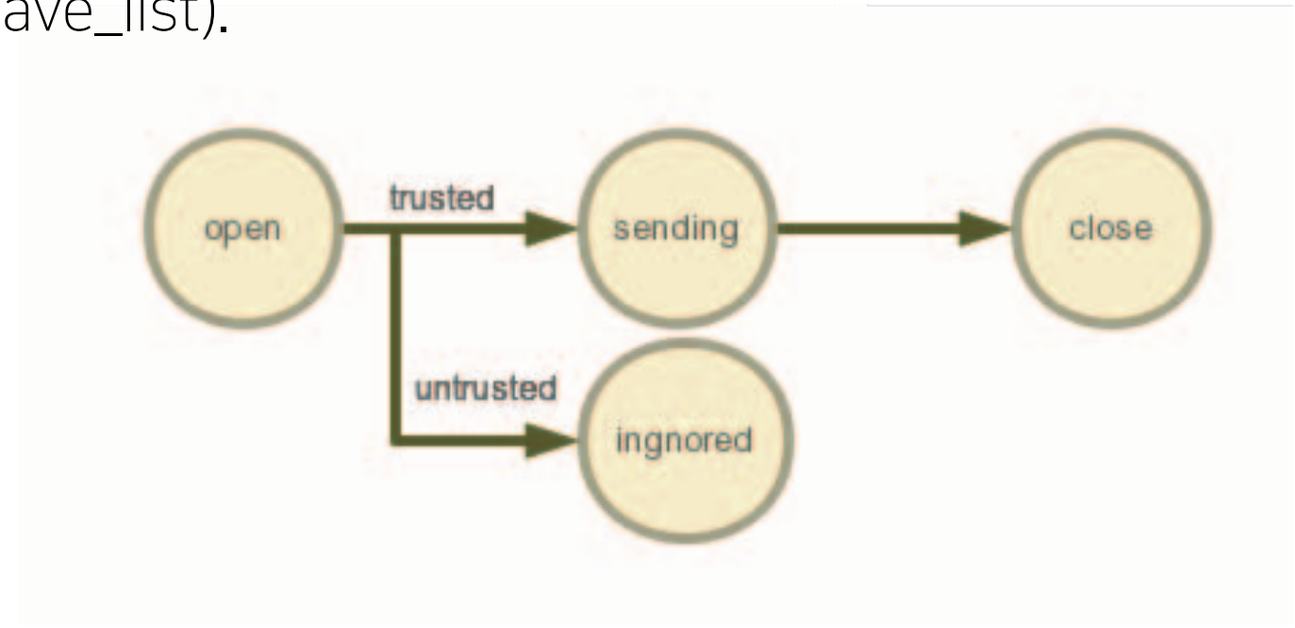
- IPFS is a peer-to-peer version controlled filesystem that synthesizes learnings from many previous successful systems.
- However, lots of data transfer everywhere and it is quite difficult to make version control over these data. IPFS does not take into account the special circumstances of large content service providers.
- Thus, We propose a scheme that combines three replication scheme and erasure codes storage scheme.

IPFS

- IPFS represents the InterPlanetary File System, which is a peer-to-peer distributed file system, aims to replace HTTP.
- IPFS synthesizes many of the best ideas from the most successful systems to date.
- BitSwap Protocol is one of the best ideas they think makes IPFS different from other block storage distributing system.

BitSwap

- BitSwap Credit is a simple credit-like system which solves the problem of free-loading but never sharing.
- They introduce debt ratio in BitSwap Strategy and the debt ratio becomes a measure of trust which incentivize nodes to exchange a lots of data.
- BitSwap Ledger is very important to a connection between BitSwap peers.
- In the lifetime of a peer connection, BitSwap peers are looking to acquire a set of blocks (want_list), and have another set of blocks to offer in exchange (have_list).



Object Merkle DAG

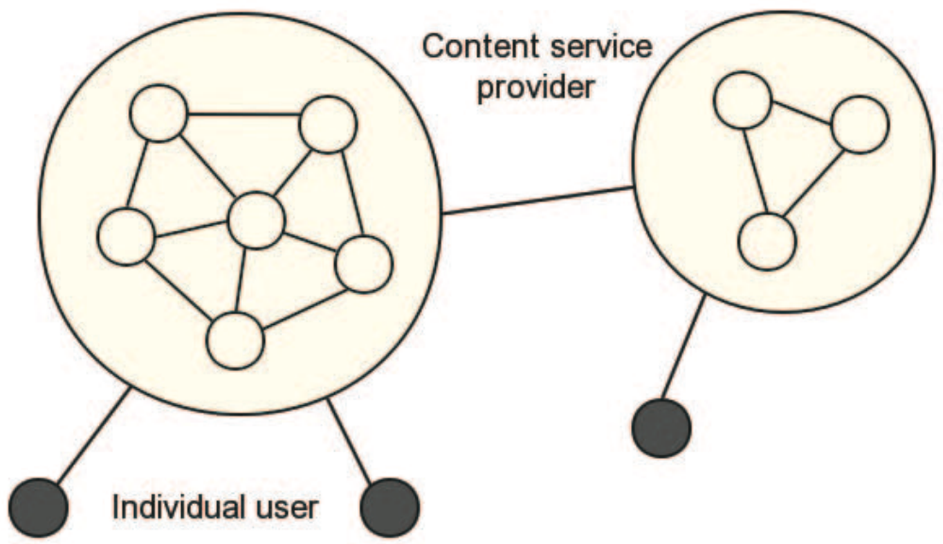
- Object Merkle DAG is almost the best part of IPFS because Content Addressing, Tamper resistance and Deduplication are all useful properties from that design.
- The Git object model fit on top of the Merkle DAG.
- IPFS defines many objects to model a versioned filesystem above the Merkle DAG.

Motivation

- IPFS was originally designed to rely on high throughput to deliver data, but for personal computers, this is a bit inappropriate.
- First of all, service providers cannot rely solely on other people to provide content to their customers, which is very unwise for business.
- Once they lose the data, they lose the customers.
- Service providers need to store large amounts of data because of their role in the service market and data storage scheme becomes important consequently.
- IPFS provides a block storage mode, which is very vulnerable to lose data reliability and availability when servers of IPFS break down.

System Model

- Service providers need to maintain one or more nodes to protect the availability of their services, but individual users do not bother to do that.
- They can easily choose to join a service provider's network and take the service provider as a proxy node.
- Individual users only need a client, which can be a simple browser, to possess a series of data exchange functions, such uploading and downloading.



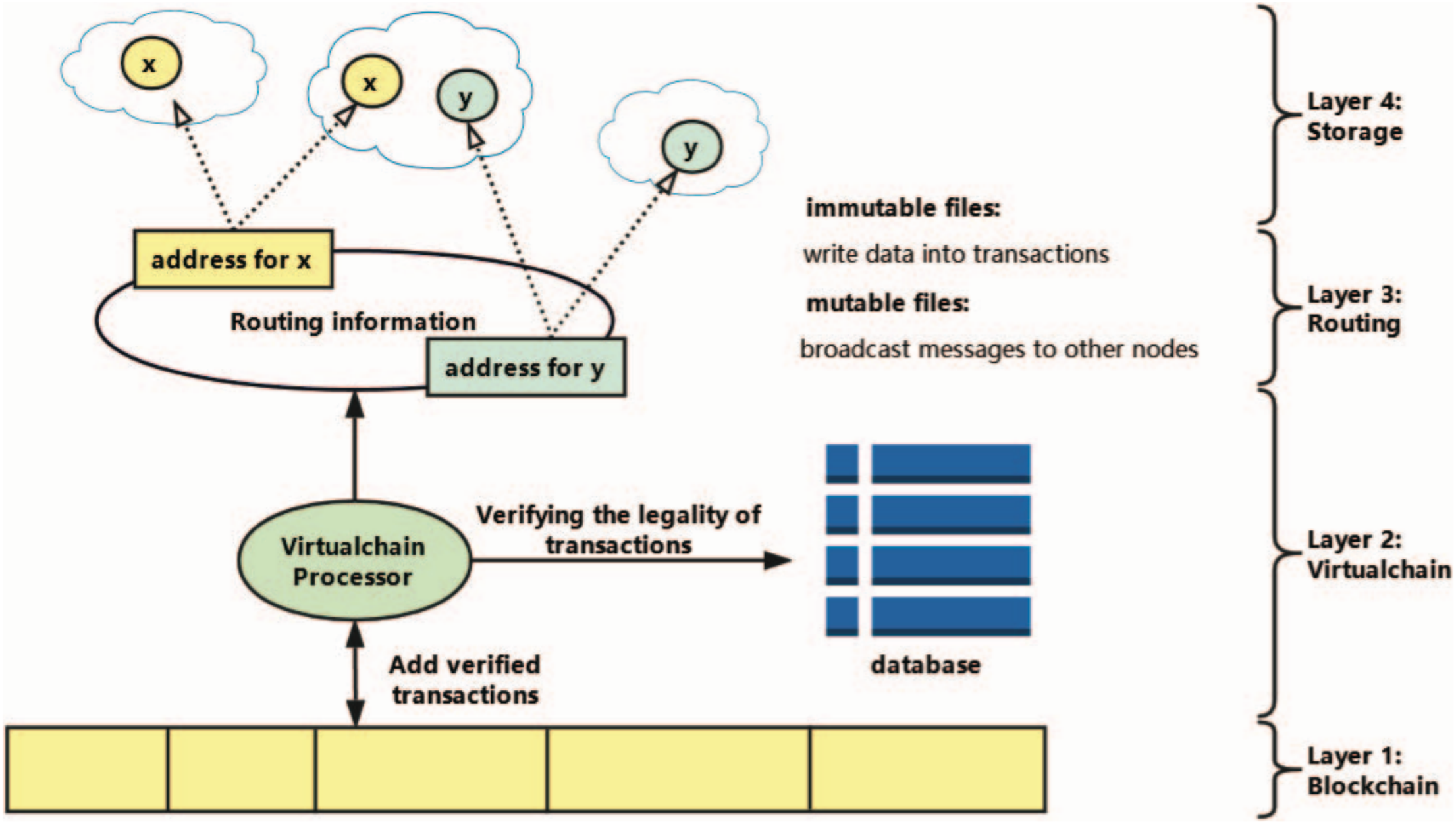


Figure 3. Overview of novel scheme architecture

Data Storage Scheme

- BitSwap peers are looking to exchange blocks in two lists and these blocks can be any part of any files.

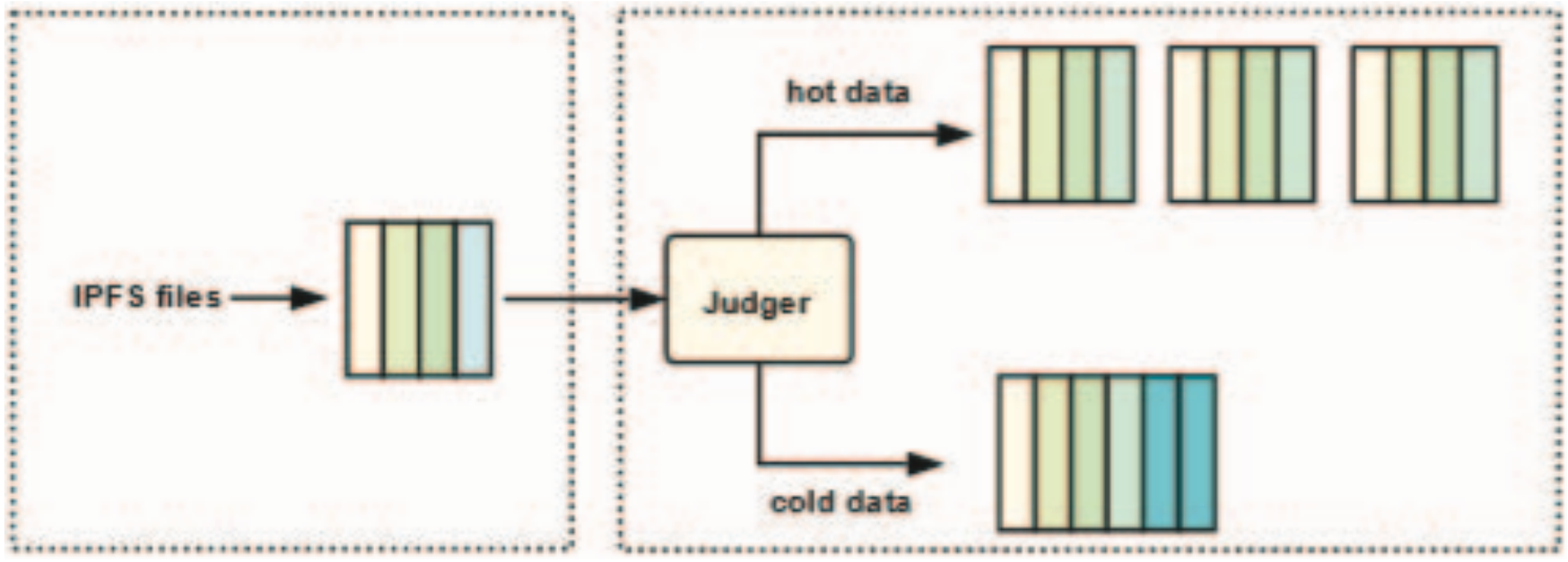


Figure 4. Flow chart of Data Storage Scheme

Future

- The ideas behind IPFS are the product of decades of successful distributed systems research in academia and open source.
- IPFS is an ambitious vision of new decentralized Internet infrastructure, upon which many different kinds of applications can be built.
- At its best, it could push the web to new horizons, where publishing valuable information does not impose hosting it on the publisher but upon those interested, where users can trust the content they receive without trusting the peers they receive it from, and where old but important files do not go missing.