How It Works: High-Speed Super Large Ledger Technology

A report based on information supplied by

Wei-Tek Tsai
Beijing Tiande Technologies

A blockchain framework with Concurrent Byzantine Fault Tolerance (CBFT)

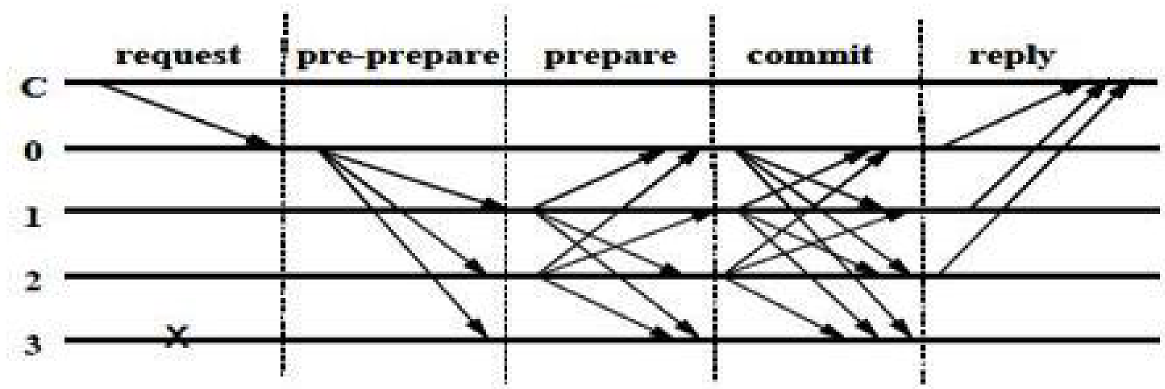
Public blockchains are great. The most famous ones like Bitcoin and Ethereum have sparked a global movement of furious innovation that is revolutionizing the way our data is stored, forever.

The challenge today is how do we make blockchain technology work when dealing with super-large amounts of data (like retail transactions), and at commercially viable transaction speeds (like Visa/Mastercard).

What I want to do in this article is explain an approach developed by Prof Wei-Tek Tsai of Arizona State University. His blockchain was stress-tested earlier this year with a live demo presented to IBM, SAP, AWS, and China's MIIT (Ministry of Industry and Information Technology). The system processed 3.33 billion historical business transactions from a clearinghouse in China. The trading volume is about , to give you some idea, this is about 15.5 times the all historical trading volume of bitcoin since 2008, about 14 years of trading volume of London Stock Exchange (LSE), or about 16 months of U.S. NASDAQ trading volume.

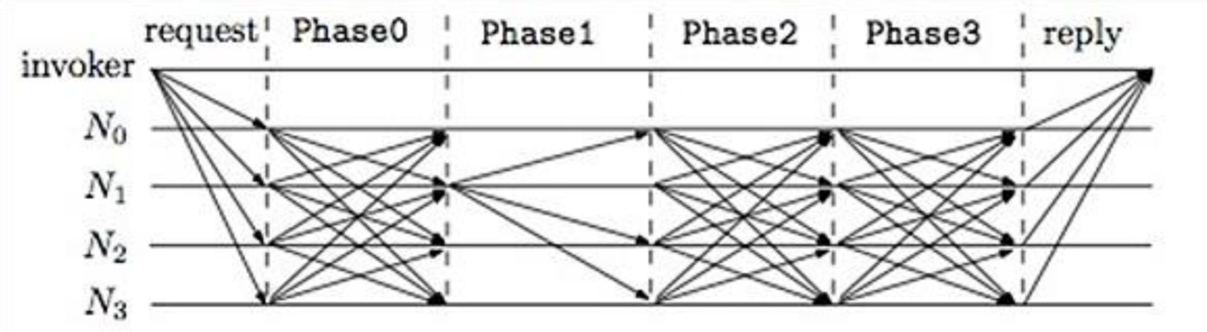
Concurrent Byzantine Fault Tolerance

The core algorithm for reaching consensus with blockchain protocols permissioned blockchains is Practical Byzantine Fault Tolerance (or PBFT). PBFT has three phases as shown by the middle three phases in the diagram below:

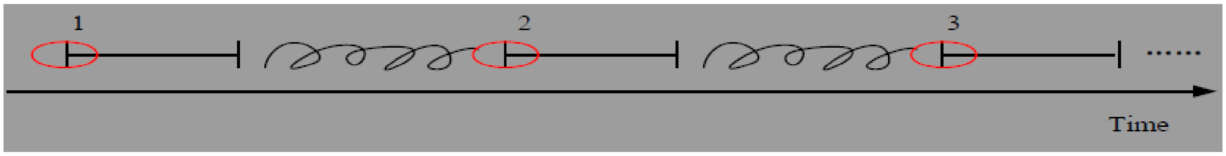


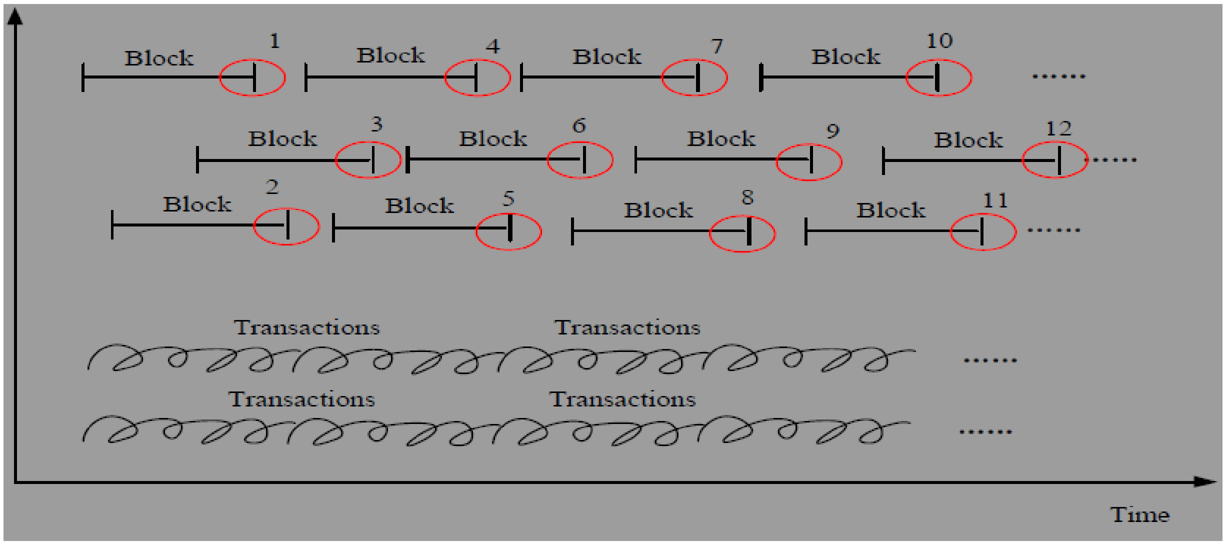
The problem with PBFT is that it performs consensus sequentially, meaning they vote on one block at a time. And this is a critical performance bottleneck.

Prof. Tsai's system uses concurrent byzantine fault tolerance (CBFT). CBFT allows multiple blocks to be voted concurrently to speed up the voting process. With CBFT, the traditional three phases of PBFT remain the same, but a new phase is added to the front of the process.



While that may seem like adding an extra layer of operation, the front phase is crucial as it performs a pre-processing action to specifically determine (i.e. fix) the contents of blocks, once contents of each is known, the block can be voted on by all the nodes. This saves time because voting on pre-processed blocks can then be conducted concurrently. The process can be illustrated as follows:





The curly lines represent the collection of transaction data prepared to be voted on. The straight line represents the voting operation.

The diagram shows three concurrent threads of processes executing, while transaction data is being collected. Two streams of data mean transaction data may come from multiple sources (due to parallel execution), and three or more concurrent threads of PBFT voting, greatly enhancing TPS (transactions per second).

In short, the key innovation is to first reach consensus on the contents of the block(s) to be voted on. Once the contents of each block have been fixed, the process of voting on those fixed blocks can then be processed concurrently, rather than sequentially.

Dual-Chain Structure

Prof. Tsai's system also divides the blockchain into two separate kinds of chains. Each chain is responsible for one kind of actions. This makes it easier to manage each kind of chains with load balancing. For instance, in a single system, the blockchain could be divided into an A-Blockchain (ABC) to manage account information, and a T-BlockChain (TBC) to manage trading information.

By splitting blockchains into two separate chains, institutions on the blockchain can share data to the public chain selectively (by public, I mean all operators on the TBC). For instance, each institution can maintain its own ABC, and when it wants to share a specific subset of ABC data, it can share that data to the TBC. A good example, for instance, would be an ABC operator to issue a request to conduct a trade with another ABC operator on a TBC . Once the trade is completed, the data on the TBC is sent back to the two ABC, and the data on the TBC and ABC are all immutable.

An example of the key operations rules is:

- An ABC performs account maintenance only. It sends messages to TBCs for trading.
- A TBC links multiple ABCs together and performs trading between different ABC operations. The TBC keeps track of the complete trading record. Every change in any ABC can be traced to a trading record at a TBC.

Scaling

With the separation of ABCs and TBCs a number of optimizations are now possible. Specifically, load balancing can be achieved, as an ABC can be split into multiple sub-ABCs, each responsible for a specific set of accounts. These sub-ABCs can run on top of different processors for parallel and concurrent processing. A traditional blockchain cannot easily be split because it carries out both account and trading activities.

I should make a footnote here and mention that it is certainly possible to split a traditional blockchain into multiple sub-chains in a process called "sharding". The problem with traditional sharding proposals is that, "shard-ed" sub-chains are still inter-connected, and they also process trading transactions amongst shards.

However, by separating account and trading functions, when an An ABC is split into, say, 3 sub ABCs, each account will reside in just one sub ABC, and none of the ABCs actually handle trading activities. Thus, sub- ABCs do not interfere with each other. In this case, each sub ABC can run on different processors to speed up operations without any interactions between any sub ABC. This is traditional load balancing in cloud computing. But after the separation into ABCs and TBCs, blockchains can be further divided, combined (if so desired later), and scaled. As the workload increases, one can add additional servers so that the overall system can maintain high performance.

Mass Data

Prof Tsai's system can be fully integrated with bigdata platforms. Data is captured into a blockchain, and the data converted into bigData platforms such as HBase. The data in the HBase can be analyzed using tools such as R, MLlib, SPSS, and SAS. Furthermore, big data platforms can be incorporated into each node of a blockchain, if necessary.

Multi-Level Redundancy

Fault tolerance is an important factor in any mass consumer facing system. To ensure system reliability, the Prof Tsai's system has been designed with four levels of redundancy:

- **Multiple Blockchains**: The system consists of multiple blockchains (ABCs and TBCs) interconnecting to each other, rather than a single blockchain linking every participant. This ensures data are saved in different chains for redundancy. For example, Euroclear has proposed a set of blockchains: Asset, Cash, Derivative, Fund, and Collateral ledgers;
- **Multiple nodes per Blockchain**: Each blockchain will have multiple nodes participating in voting and saving system states. This ensures that all the data stored in a blockchain are saved in different nodes for reliability;
- **Multiple processors for a node**: Each node can run on top of a cluster of processors for reliability, availability, and high performance;
- **Distributed storage**: Data stored in each node can be saved in distributed storages such as RAID for reliability and performance.

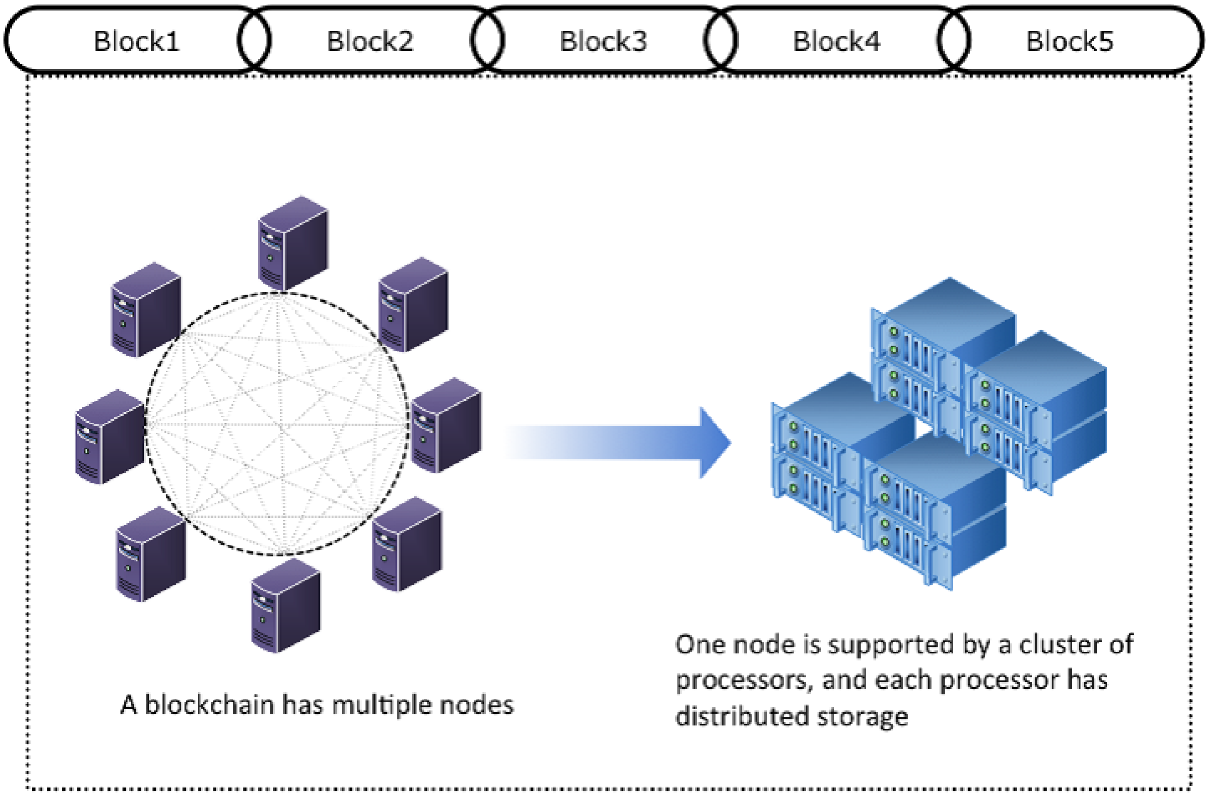
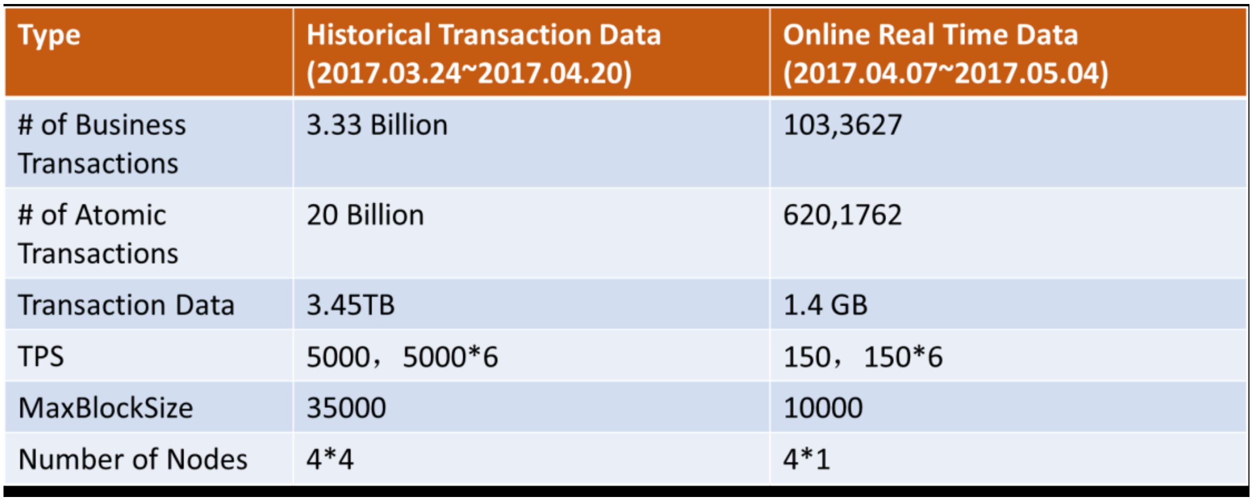

Block1
Block2
Block3
Block4
Block5
A blockchain has multiple nodes
One node is supported by a cluster of processors, and each processor has distributed storage

Figure 5 Blockhains with Multiple Levels of Redundancies

System Test Results

The above super-large ledger system was first deployed for the Guangdong Clearing House (GCH), as an operational test. The test ran for two months. In the first month, the system processed historical data of 3.333B of transactions. This is equivalent to 16 months of trading at NASDAQ, or 14 years of trading at LSE. The system reached an average of 5,000 TPS.
In the second month, the system operated on real-time data. In 20+ days, the system processed approximately 1M transactions in real time.



Type	Historical Transaction Data (2017.03.24~2017.04.20)	Online Real Time Data (2017.04.07~2017.05.04)
# of Business Transactions	3.33 Billion	103,3627
# of Atomic Transactions	20 Billion	620,1762
Transaction Data	3.45TB	1.4 GB
TPS	5000， 5000*6	150， 150*6
MaxBlockSize	35000	10000
Number of Nodes	4*4	4*1

To provide a sense of scale of the test, 3.33 billion transactions is equivalent to:

- 15.5 times the all time historical trading volume of bitcoin since 2008
- 16 months of trading volume for the NASDAQ exchange
- 14 years of trading volume for the London Stock Exchange (LSE)
- 231 hours of trading volume for global Visa transactions

The system used 4 nodes, each with IBM 4 x86 processors. Each processor was interconnected by a high-speed network with high-speed switches.

The system was then subsequently demonstrated to IBM, SAP, AWS, MIIT (Ministry of Industry and Information Technology), numerous universities and research institutions such as Peking University, banks such as PBOC and ICBC, and government agencies such as MIIT and local governments. The system has been in public display since May 2017 at Guiyang Big Data Expo.

The system has also been independently tested by two parties:
(1) MIIT. A one-week test to examine the code used in the system (i.e., white-box testing).
(2) The clearing results were independently tested and evaluated by the hosting institution.