Blockchain Infrastructure in the Digital Contents Field
（Sharing of Geth research results）

SingulaNet Ltd.　Koji Machi

Speaker：Koji Machi

2017：Media × Blockchain R&D

2019：Implementation of OEM Service

Summary of Issues

What is the required Blockchain Processing Performance in the Digital Contents Domain?

If a distributed network is built for common apps, 2000 processes per second is sufficient.
(The figures are based on data from our company as well as several other companies.)

Required processing performance for NFTs

Prerequisites

Active users	**10,000,000**
No of actively paying users	1,000,000
No of users using the service simultaneously	100,000
No of users executing Mint/Transfer simultaneously	10,000

If the processing power is…

- 500 processes per second - the process completion waiting time is 20 seconds
- **2000 processes per second - the waiting time is 5 seconds**

What is the required Blockchain Processing Performance in the Digital Contents Domain?

In the case of an extremely large scale network, it may be necessary to process around 10,000 cases per second.
(Based on data from public networks)

Necessary processing performance for NFTs

Prerequisites	
No of active users	**<u>100,000,000</u>**
No of actively paying users.	10,000,000
No of user utilizing the service simultaneously	100,000
No of users executing Mint/Transfer simultaneously	100,000

If the processing power is…

- 500 processes per second - the process completion waiting time is 200 seconds
- 2000 processes per second - the process completion waiting time is 50 seconds
- **<u>10000 processes per second - the process completion waiting time is 10 seconds</u>**

Main Points

Purpose-built consortium network figure aim :

2000 processes per second

Public network figure aim :

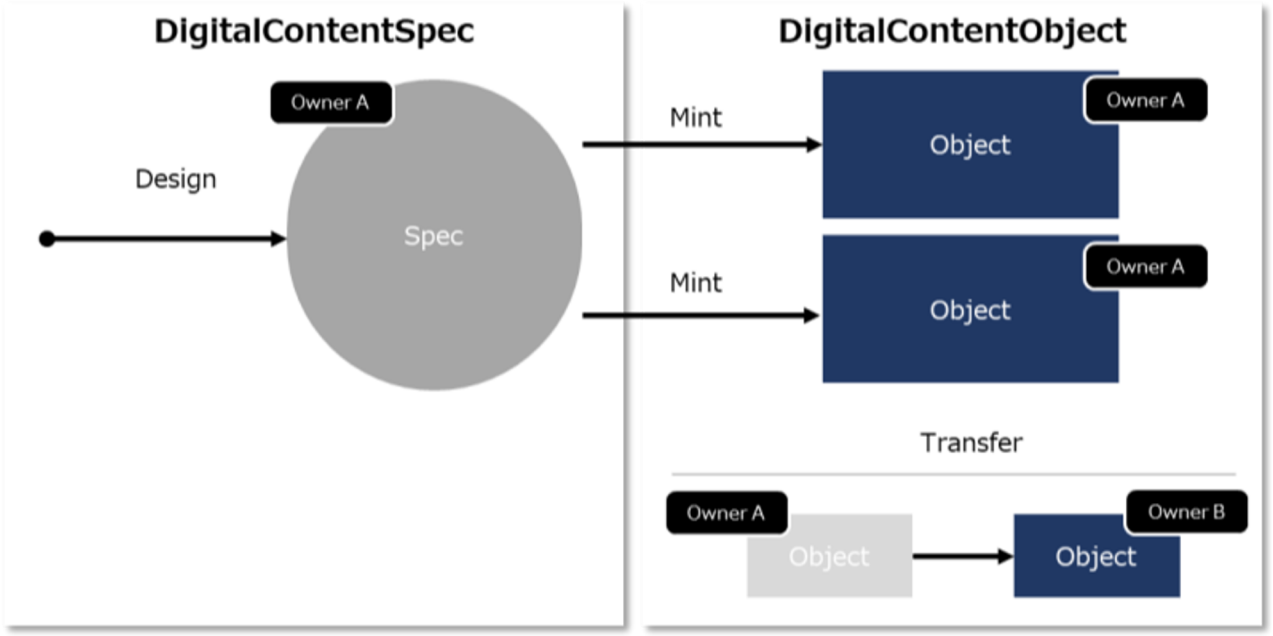
10,000 processes per second

Blockchain aimed at the Japanese Digital Contents Industry

We are starting a new initiative in a consortium with the aim of carrying out research and development in the area of blockchain focusing specifically on solving issues in the contents domain such as copyright distribution.
We are currently conducting research and development using Ethereum, and we hope to make a big announcement in the near future.

Content-NFT

We have announced information regarding the implementation of the next-generation NFT smart contract reference that supports copyright distribution.



Research：Investigation of the Causes

Research Results：research carried out in 2020

White Paper
- **Function-Level Bottleneck Analysis of Private Proof-of-Authority Ethereum Blockchain**

Members
- KENTAROH TOYODA (Member, IEEE)
- KOJI MACHI
- YUTAKA OHTAKE
- ALLAN N. ZHANG(Associate Member, IEEE)

Objectives

Private Ethereum Blockchain base systems are highly requested in the industrial sector. However, the throughput performance the system is insufficient.

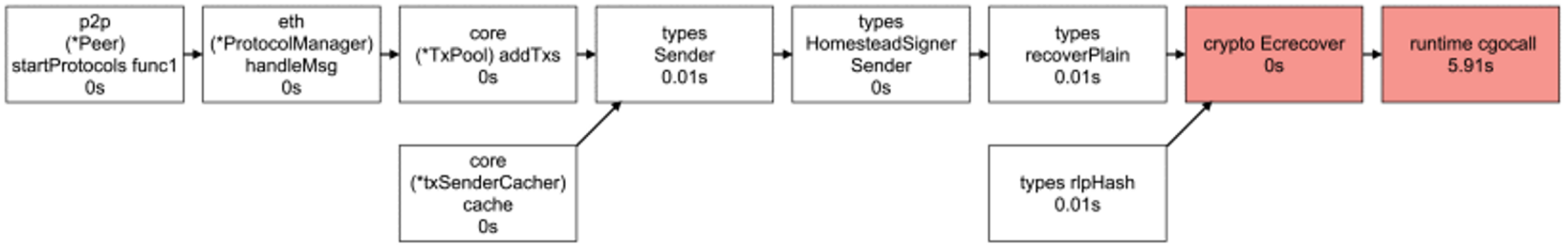
Many researchers have analyzed the private Blockchain performance, but the root cause of the problem has not been analyzed during these studies.

- In this study, a bottleneck analysis has been carried out regarding the function level details of a private Ethereum Blockchain.
- As the Ethereum client application Geth was developed using the go language, the function time was calculated using a customized golang function with the resource profiling tool pprof.
- In order to configure the parameters simply and conduct the tests, a Shell Script was created to automatically carry out the private Ethereum Blockchain creation process using a Docker container.
- A series of tests were carried out, and specified ① the bottleneck function that was called each time the transaction reached the Ethereum node.
- Additionally, it was discovered that the ② multi-thread was not being utilized sufficiently.

① Bottleneck function specification

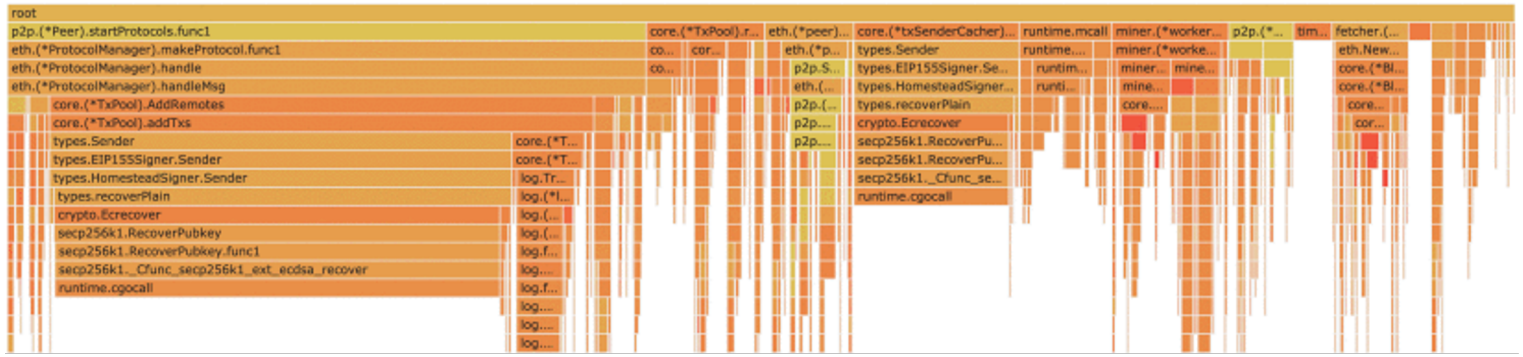
Bottleneck processing of the PoA algorithm was identified in this research
(In fact, it was not an issue with the PoA algorithm, but a problem with Geth overall.)

cgocall …In encryption / decryption processing, the general function of Go language (process to call C language) is utilized.
The processing involved in calling C language from Go language is much heavier compared to Go language IF processing or C language IF processing.



In the case of cgocall stack overflow public QA…
"As you've discovered, there is fairly high overhead in calling C/C++ code via CGo. So in general, you are best off trying to minimize the number of CGo calls you make. "

Source : https://stackoverflow.com/questions/28272285/why-cgos-performance-is-so-slow-is-there-something-wrong-with-my-testing-code



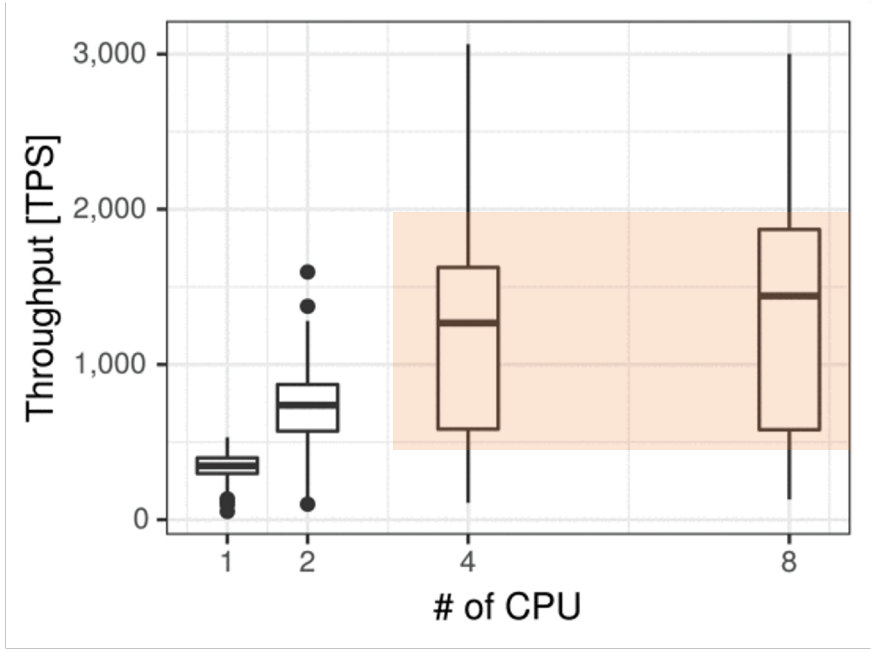
Should everything be written in Go？

Should everything be written in C?

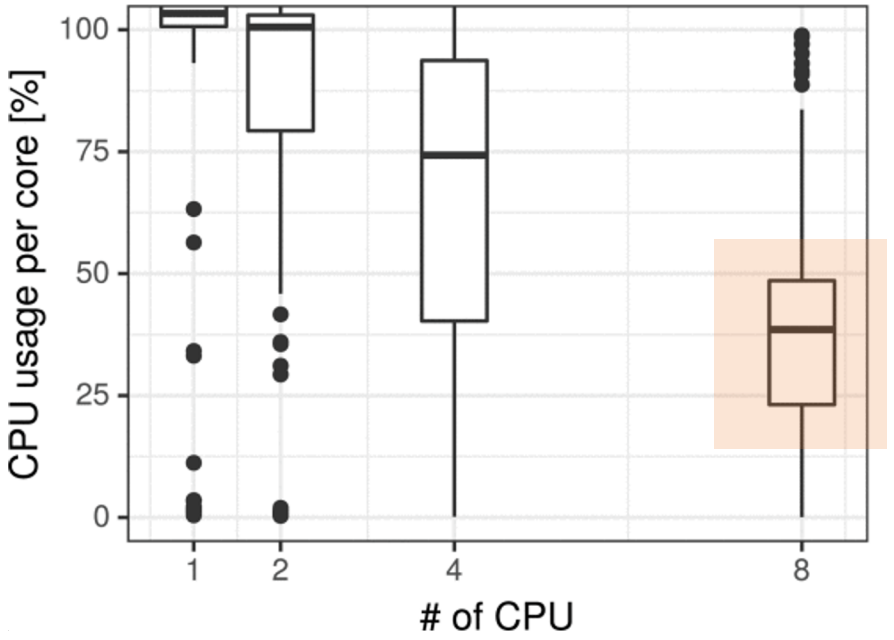
② Multi-threads are not being Utilized Suffiently

Multi-thread processing peaks at parallel 4 core CPUs & above

CPU distributed processing cannot be utilized effectively due to the waiting time that occurs during basic Geth processing.



No large difference was found between the number of processed items with 4 core CPUs and 8 core CPUs



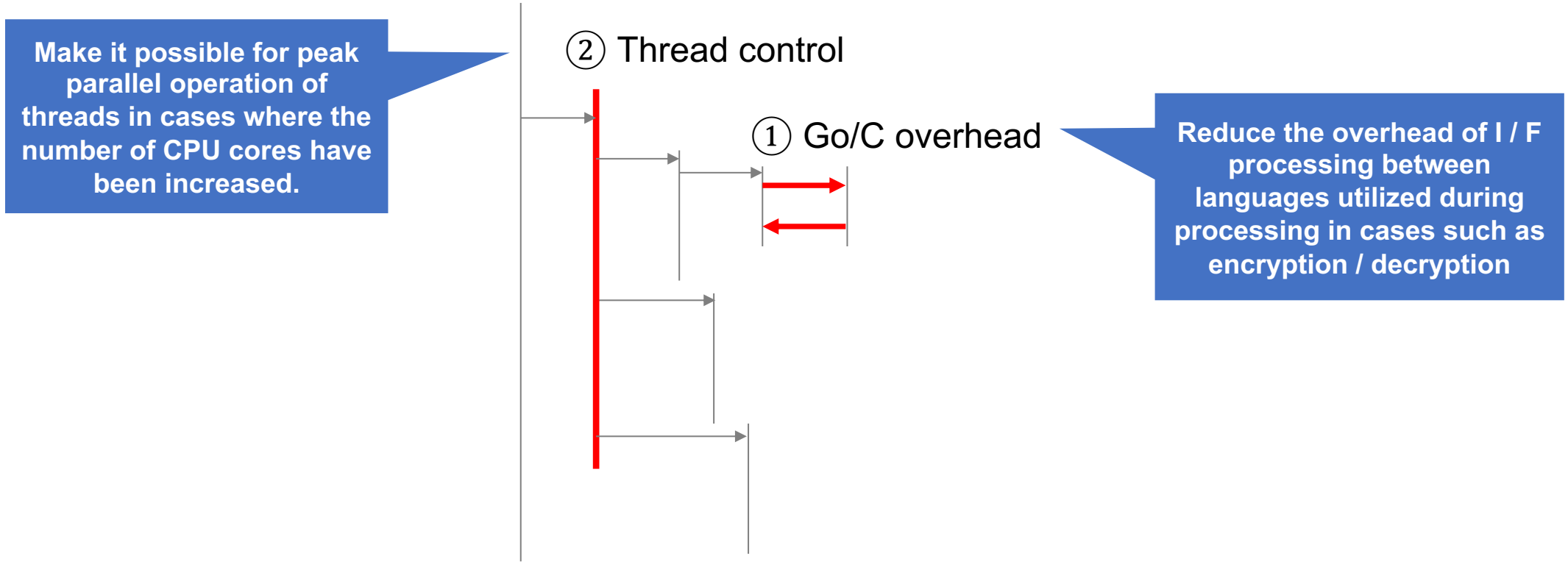
In the case of utilizing 8 core CPUs, the CPU usage ratio dropped significantly

Could an improvement be made in thread control with Go's main processing?

Conclusion

- **The performance of Ethereum Geth has a bottleneck in the thread control mechanism and Go / C language processing interface.**
- **In order to improve the processing performance to be able to handle 2,000 cases per second, the generic processing part (consensus algorithm) needs to be modified.**

A processing performance bottleneck aimed at achieving a public network (handling 10,000 cases per second) in the digital contents domain.



Thank you for listening