POLYTECHNIQUE
MONTRÉAL

UNIVERSITÉ
D'INGÉNIERIE

Comparison of Ethereum and Hyperledger Fabric's Performances

By: Alice Gong, Yuhan Li & Yanis Toubal **(Team Yankee)**

Plan

Background 01

Experiment 02

03 Results

04 Discussion

05 Conclusion

01

Background

Blockchain

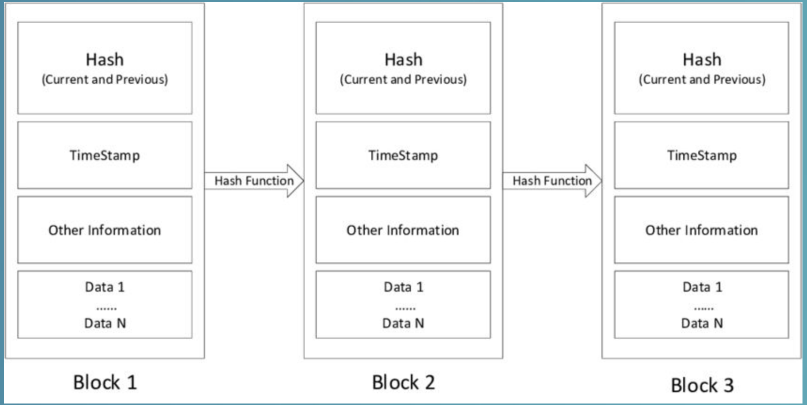
Cryptocurrency Ledger: Wallet of hardware cryptocurrency. Keeps records of users' anonymous identity, cryptocurrency balance, and transactions.

Distributed Ledger: Decentralized database that eliminates the need of central authority.

Decentralized

Secure

Consensus-driven



Hash
(Current and Previous)

TimeStamp

Other Information

Data 1
......
Data N

Block 1

Hash Function

Hash
(Current and Previous)

TimeStamp

Other Information

Data 1
......
Data N

Block 2

Hash Function

Hash
(Current and Previous)

TimeStamp

Other Information

Data 1
......
Data N

Block 3

Immutable

Chronological

Irreversible

Hyperledger Fabric

- Is open source under the Linux Foundation
- Is a permissioned blockchain platform
- Doesn't require a cryptocurrency
- Uses a crash-fault tolerant (CFT) algorithm

HYPERLEDGER
FABRIC

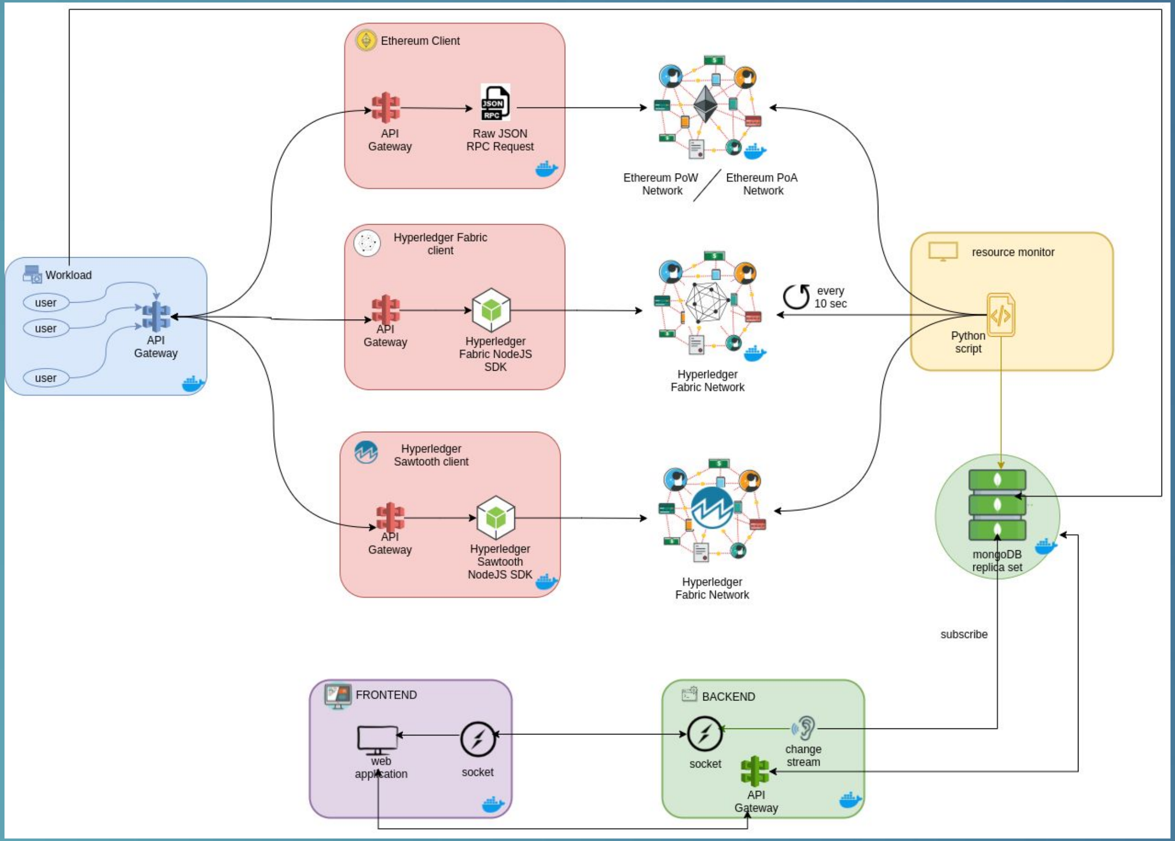
Ethereum

- Blockchain platform for dApps
- Has its own cryptocurrency, Ether
- Initially, Proof-of-Work (PoW)

- Ethereum Clique uses Proof-of-Authority (PoA)

BlockCompass

- Workload
- Ethereum client
- Hyperledger Fabric client
- Resource monitor
- Back-end
- Front-end



02

Experiment

Purpose

- Quantify and compare Ethereum Clique and Hyperledger Fabric
- Three main questions:
  1. How does the performance of the two implementations compare against **similar** workloads?
  2. How do the two implementations compare against **different** workloads?
  3. Which one is best in write latency, throughput?

Infrastructure

Amazon Virtual Machine

- T2.large
- 8GB RAM
- 2 vCPUs
- 50 GB of hard-disk space

Hyperledger Fabric

- Hyperledger Fabric release v.2.3.2

Ethereum

- Go Ethereum v.1.10.3

Workload

- Simulates users that are submitting data of varied size concurrently

- For this experiment
  - Small: 2
  - Medium: 5
  - Large: 8

03

Results

Performance Metrics

Emit rate Total transactions per second

Throughput Successful transactions per second

Latency Response time per transaction

Error Rate Rate of failed transactions

Ethereum

	Size 2	Size 5	Size 8
Emit rate (tps)	405	337	162
Throughput (tps)	405	337	148
Latency (ms)	1361	1731	4036
Error rate	0	0	0.064

Table 1: Performance Metrics for Ethereum Clique

Hyperledger Fabric

	Size 2	Size 5	Size 8
Emit rate (tps)	739	583	277
Throughput (tps)	739	583	277
Latency (ms)	186	319	1517
Error rate	0	0	0

Table 2: Performance Metrics for Hyperledger Fabric

Resource Consumption Metrics

Ethereum

	Size 2	Size 5	Size 8
CPU	12.8	12.6	7.5
Memory %	4.1	4.3	4.2
Input/ Output MB	759.2	759.2	533.1
Users	64.4	61.2	60.7

Table 3: Resource Consumption Metrics for Ethereum

Hyperledger Fabric

	Size 2	Size 5	Size 8
CPU	5.8	5.4	3.7
Memory %	5.8	0.68	0.66
Input/ Output MB	951.6	951.6	420.2
Users	63.6	62.7	62

Table 4: Resource Consumption Metrics for Hyperledger Fabric

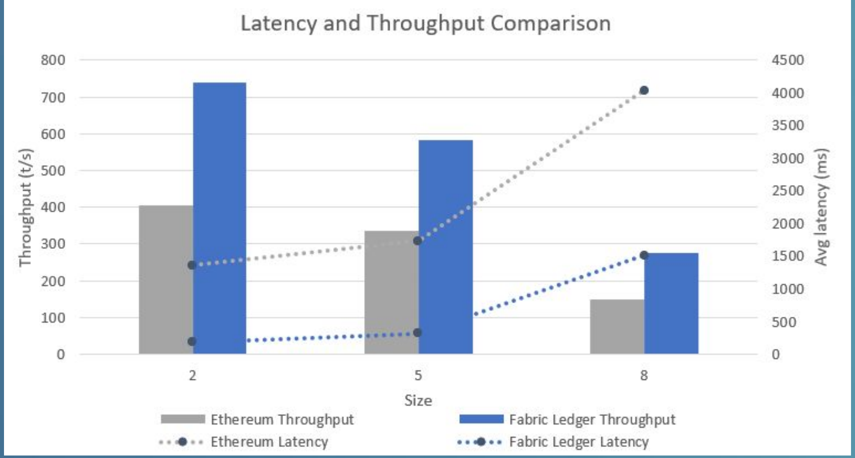
04

Discussion

Recap of purpose

1. How does the performance of the two implementations compare against **similar** workloads?
2. How do the two implementations compare against **different** workloads?
3. Which one is best in write latency, throughput?

Performance Comparison



	Ethereum			Hyperledger Fabric		
	Size 2	Size 5	Size 8	Size 2	Size 5	Size 8
Emit rate (tps)	405	337	162	739	583	277
Throughput (tps)	405	337	148	739	583	277
Error rate	0	0	0.064	0	0	0

Table 5: Combined Performance Metrics

Resource Consumption Comparison

Ethereum

	Size 2	Size 5	Size 8
CPU	12.8	12.6	7.5
Memory (%)	4.1	4.3	4.2
Network I/O (MB)	759.2	759.2	533.1

Table 6: Resource Consumption Metrics for Ethereum

>

>

<

Hyperledger Fabric

	Size 2	Size 5	Size 8
CPU	5.8	5.4	3.7
Memory (%)	5.8	0.68	0.66
Network I/O (MB)	951.6	951.6	420.2

Table 7: Resource Consumption Metrics for Hyperledger Fabric

05

Conclusion