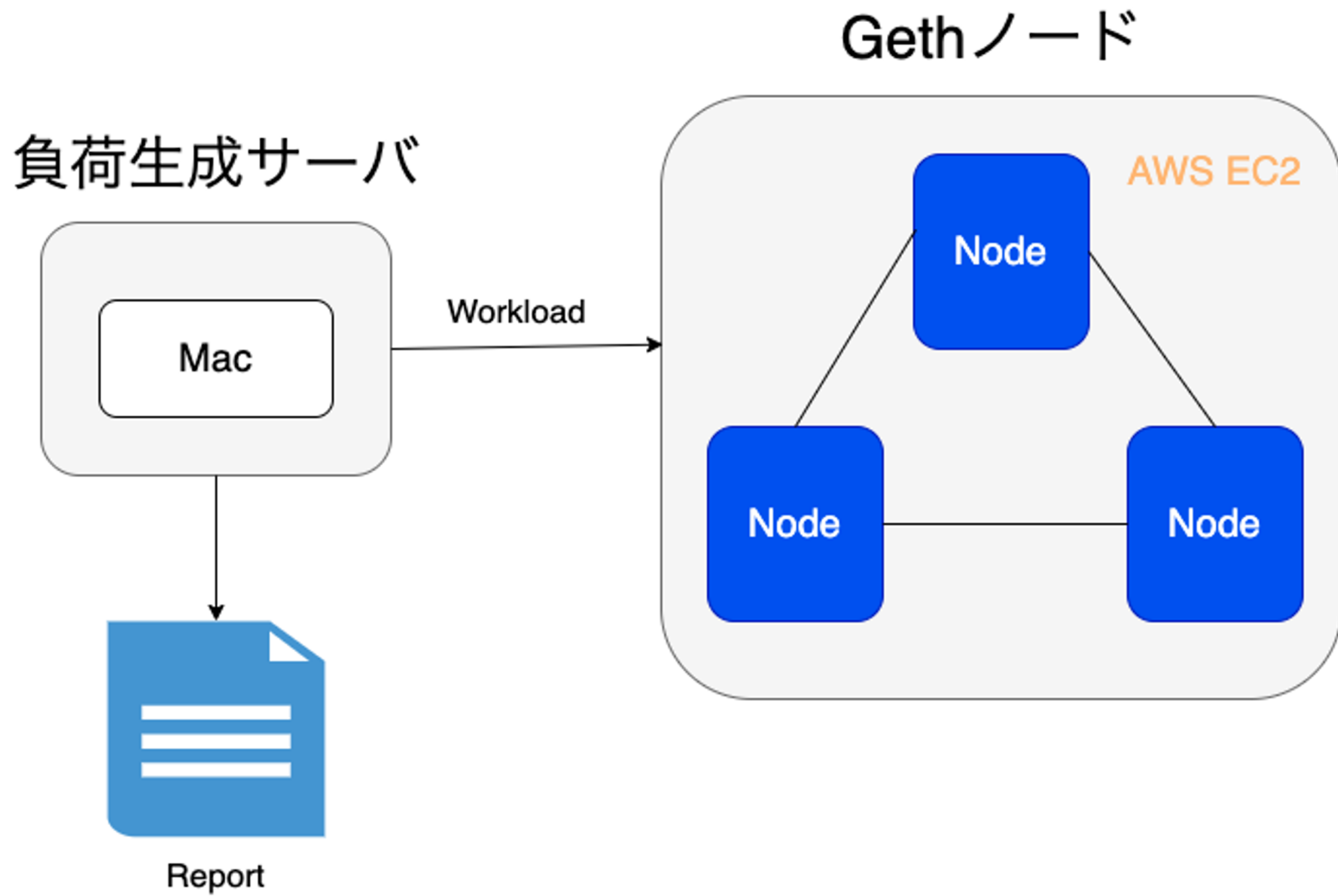


## ■システム構成概要図



# 前頁の構成を準備し、gethのTransferを対象にテストを実施しました

## テスト実施条件 - チェーン情報

- Xx
- Xx

種別	項目	定義	値
ブロックチェーンに関する情報	Blockchain Client Name	ブロックチェーンのクライアント名	geth
	Consensus Algorithm	ブロックチェーンにおけるネットワーク全体の合意を行うための方式。	Proof of Authority (Clique)
	Transaction Method	ブロックチェーンのデータをある値から別の値に変更する状態遷移法。 性能測定の対象となる処理の内容。	Transfer balanceOf
	Network Size (Node数)	合意形成に参加しているバリデータノードの数。	3台

## テスト実施条件 - ハード情報

- Xx
- Xx

種別	項目	定義	値
ハードウェア情報	負荷生成サーバ: 実マシンの場合		
	CPU 種別	マシンのCPU種別	Apple M1
	GPU種別 ※ PoW に GPU を使う場合	マシンのGPU種別	Apple独自設計 8コア
	RAM容量	マシンのRAM容量	16 GB
	ブロックチェーンの各ノード: クラウドの場合		
	使用したクラウドサービス	クラウドサービス名	AWS
	インスタンスタイプ	クラウドサービスのインスタンス名	t2.xlarge
	RAM容量	クラウドサービスのインスタンスのRAM容量	16 GB

# 負荷の総リクエスト数は1000トランザクションとし、検証を実施 Read Latency / Throughputを除く項目について結果を得ました

- 送信速度は50 TPS固定
- block生成時間 10秒ごと
- 成功とみなすまでの承認数は2ブロック
- iostat, vmstatは5秒ごとの結果をリアルタイムで取得

## テスト実施条件 - システム負荷条件

- Xx
- Xx

種別	項目	定義	値
システム負荷条件	負荷生成クライアント数	システム負荷をかけたクライアントの数	1
	負荷総リクエスト数	クライアントから送信した総リクエスト数	1000トランザクション
	負荷継続時間	クライアントからかけた負荷の継続時間	35秒間
	負荷ツール単体での限界リクエスト数	負荷ツール単体でかけられるリクエスト数の最大値	17,541トランザクション



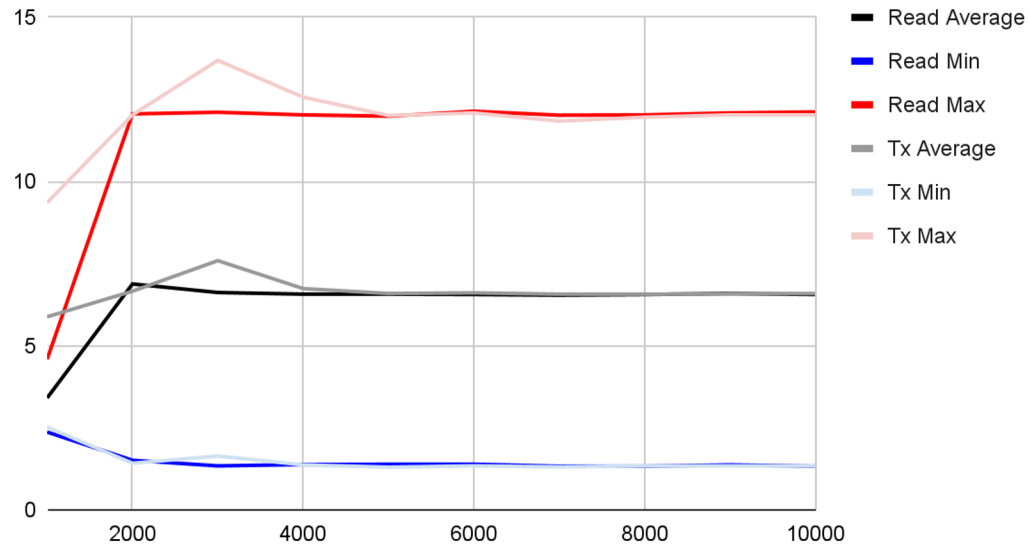
## テスト実施結果

種別	項目	定義	値
ブロックチェーンに関する情報	Read Latency	読み取り要求を送信し、その応答を受信するまでの合計時間	最小: 2.37秒 最大: 4.59秒 平均: 3.41秒
	Read Throughput	1秒あたりの読み取り数	41.8 TPS
	Transaction Latency	ネットワーク全体がトランザクションを検証するのにかかる時間	最小 2.52秒 最大 9.36秒 平均 5.88秒
	Transaction Throughput	定義された期間にブロックチェーンによって有効なトランザクションがコミットされる割合	35.6 TPS
システム負荷に関する情報	CPU負荷	CPUに対する負荷	CPU使用率 最小: 1% 最大: 5% 平均: 3%
	ディスク容量負荷	ディスク容量に対する負荷	1秒当たりのHDDへのwriteの平均 最小: 0.80 KB/s 最大: 146.80 KB/s 平均: 49.47 KB/s

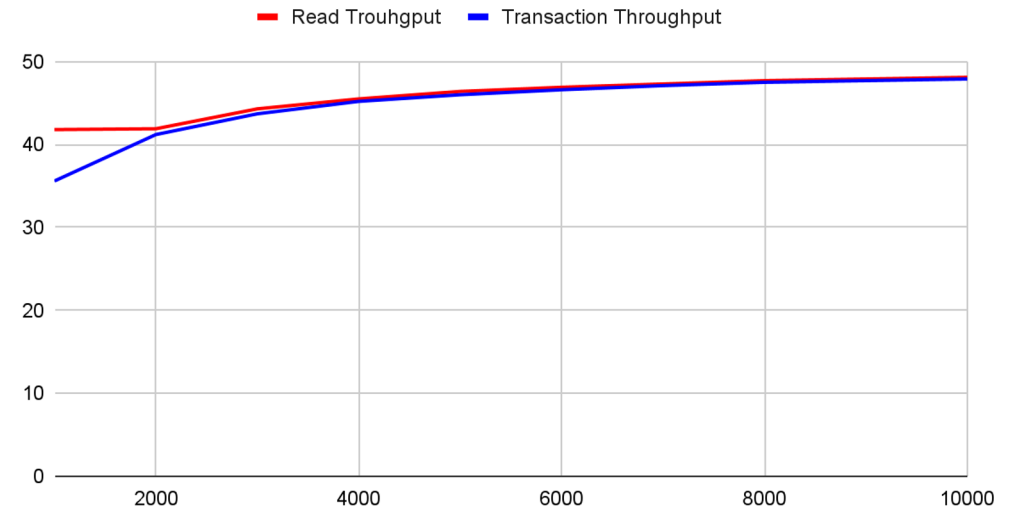
- 送信速度は50 TPS固定

負荷総リクエスト数	Read Latency	Read Throughput	Transaction Latency	Transaction Throughput	CPU使用率	ディスク容量負荷
1000	最小: 2.37秒 最大: 4.59秒 平均: 3.41秒	41.8 TPS	最小 2.52秒 最大 9.36秒 平均 5.88秒	35.6 TPS	最小: 1% 最大: 5% 平均: 3%	最小: 0.80 KB/s 最大: 146.80 KB/s 平均: 49.47 KB/s
2000	最小: 1.51秒 最大: 12.06秒 平均: 6.88秒	41.9 TPS	最小 1.42秒 最大 12.03秒 平均 6.66秒	41.2 TPS	最小: 2% 最大: 11% 平均: 5.29%	最小: 0.80 KB/s 最大: 408.80 KB/s 平均: 73.55 KB/s
3000	最小: 1.34秒 最大: 12.11秒 平均: 6.62秒	44.3 TPS	最小 1.64秒 最大 13.69秒 平均 7.59秒	43.7 TPS	最小: 1% 最大: 12% 平均: 4.78%	最小: 0.30 KB/s 最大: 496.00 KB/s 平均: 69.37 KB/s
4000	最小: 1.38秒 最大: 12.03秒 平均: 6.57秒	45.5 TPS	最小 1.37秒 最大 12.57秒 平均 6.74秒	45.2 TPS	最小: 1% 最大: 8% 平均: 5.80%	最小: 0.80 KB/s 最大: 650.40 KB/s 平均: 70.35 KB/s
5000	最小: 1.39秒 最大: 11.99秒 平均: 6.57秒	46.4 TPS	最小 1.30秒 最大 12.01秒 平均 6.59秒	46.0 TPS	最小: 1% 最大: 13% 平均: 5.21%	最小: 0.80 KB/s 最大: 598.90 KB/s 平均: 91.134 KB/s
6000	最小: 1.39秒 最大: 12.14秒 平均: 6.56秒	46.9 TPS	最小 1.34秒 最大 12.09秒 平均 6.61秒	46.6 TPS	最小: 2% 最大: 13% 平均: 6.02%	最小: 0.80 KB/s 最大: 454.40 KB/s 平均: 109.624 KB/s
7000	最小: 1.33秒 最大: 12.02秒 平均: 6.54秒	47.3 TPS	最小 1.31秒 最大 11.84秒 平均 6.55秒	47.1 TPS	最小: 2% 最大: 13% 平均: 5.95%	最小: 0.80 KB/s 最大: 801.60 KB/s 平均: 85.45 KB/s

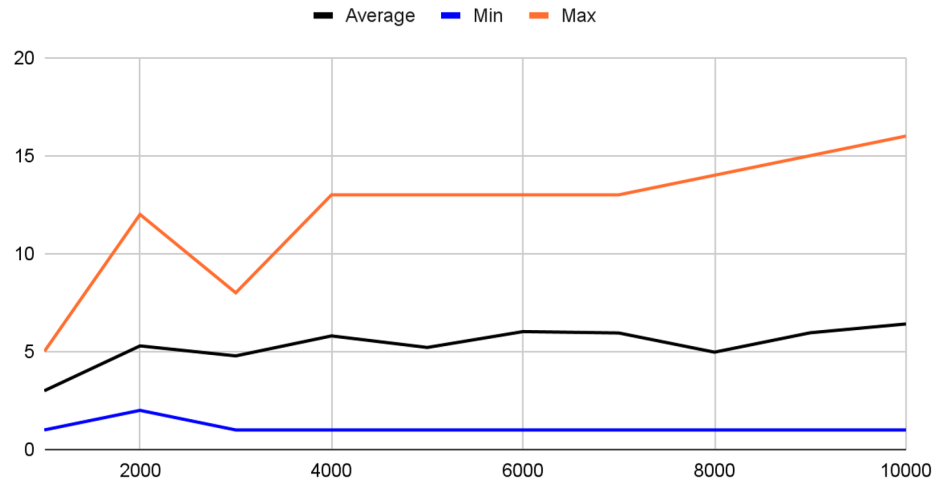
### Read Latency / Transaction Latency (秒)



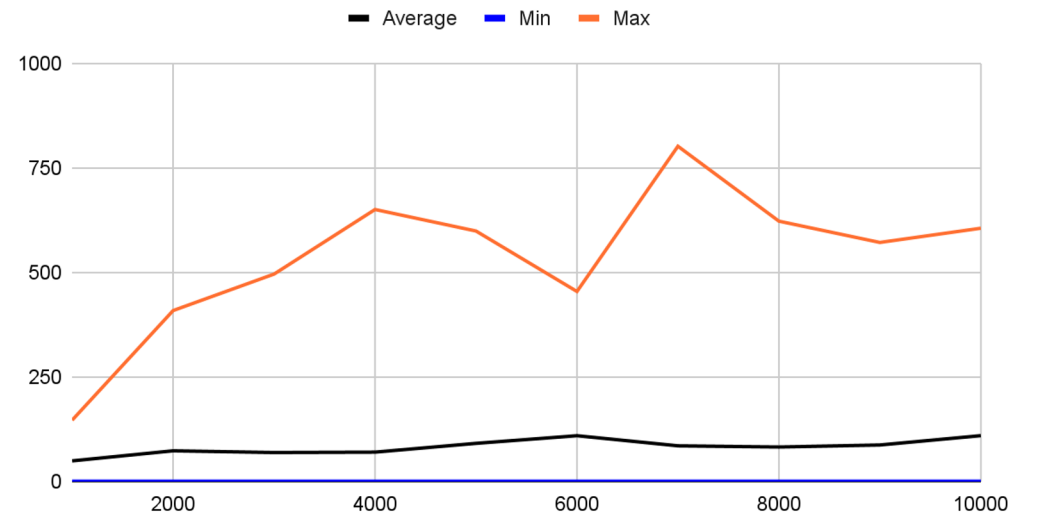
### Read Throughput / Transaction Throughput (TPS)



### CPU使用率 (%)



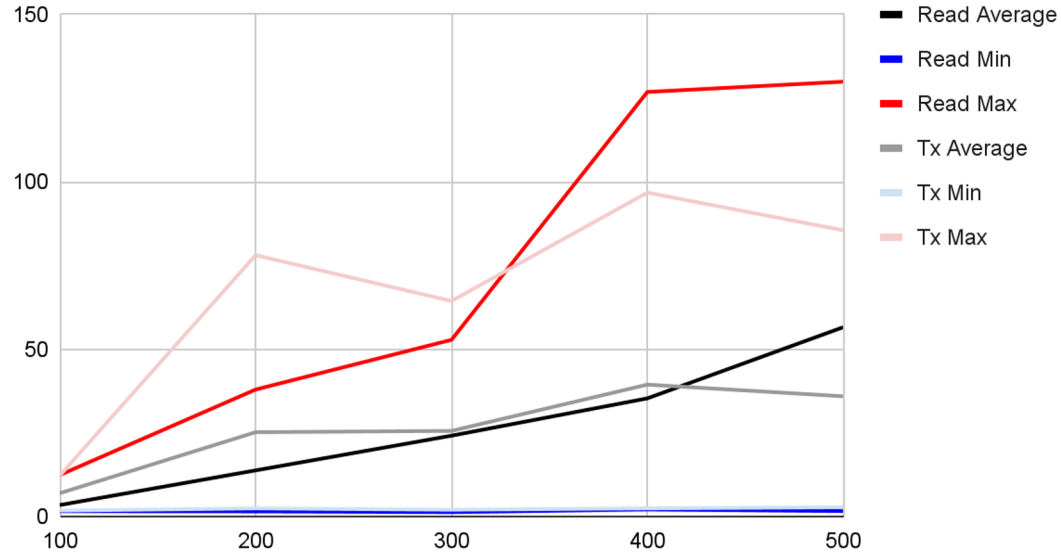
### ディスク負荷 (KB/s)



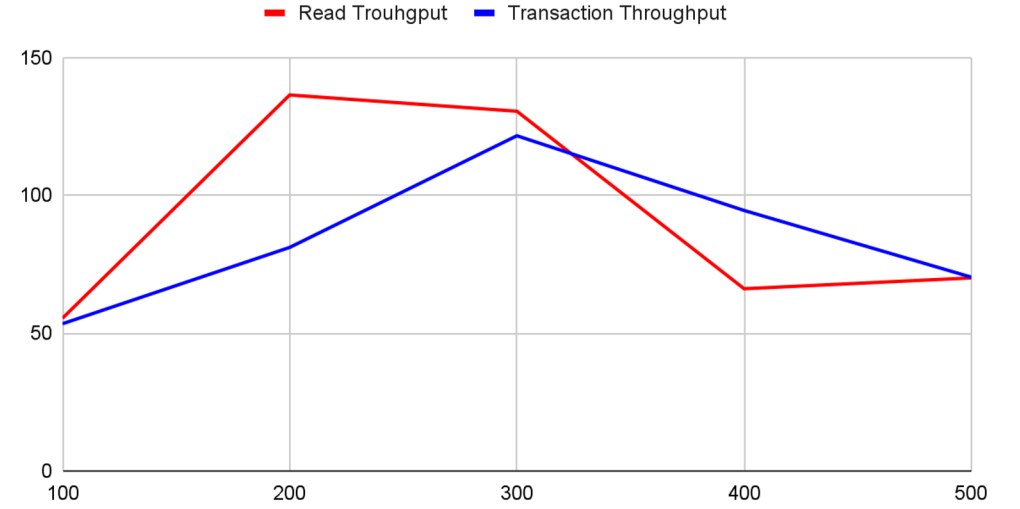
- 総トランザクション数を10000で固定して目標TPSを増加

目標TPS	Read Latency	Read Throughput	Transaction Latency	Transaction Throughput	CPU使用率	ディスク容量負荷
100	最小: 1.57秒 最大: 12.39秒 平均: 3.41秒	55.4 TPS	最小 1.75秒 最大 12.23秒 平均 6.95秒	53.4 TPS	最小: 1% 最大: 39% 平均: 17%	最小: 0.8 KB/s 最大: 1093 KB/s 平均: 147.58 KB/s
200	最小: 1.51秒 最大: 37.93秒 平均: 13.77秒	136.4 TPS	最小 2.42秒 最大 78.08秒 平均 25.18秒	81.1 TPS	最小: 1% 最大: 83% 平均: 53.65%	最小: 0.8 KB/s 最大: 1072 KB/s 平均: 125.6 KB/s
300	最小: 1.29秒 最大: 52.78秒 平均: 24.14秒	130.5 TPS	最小 2.01秒 最大 64.39秒 平均 25.57秒	121.6 TPS	最小: 1% 最大: 82% 平均: 56.78%	最小: 0.7 KB/s 最大: 2708 KB/s 平均: 192.75 KB/s
400	最小: 2.09秒 最大: 126.85秒 平均: 35.29秒	66.1 TPS	最小 2.43秒 最大 96.77秒 平均 39.40秒	94.5 TPS	最小: 1% 最大: 82% 平均: 75.42%	最小: 0.7 KB/s 最大: 1894 KB/s 平均: 108.02 KB/s
500	最小: 1.61秒 最大: 129.93秒 平均: 56.55秒	70.3 TPS	最小 2.76秒 最大 85.53秒 平均 35.91秒	70.3 TPS	最小: 1% 最大: 82% 平均: 63.717%	最小: 0.60 KB/s 最大: 2947.2 KB/s 平均: 129.093 KB/s

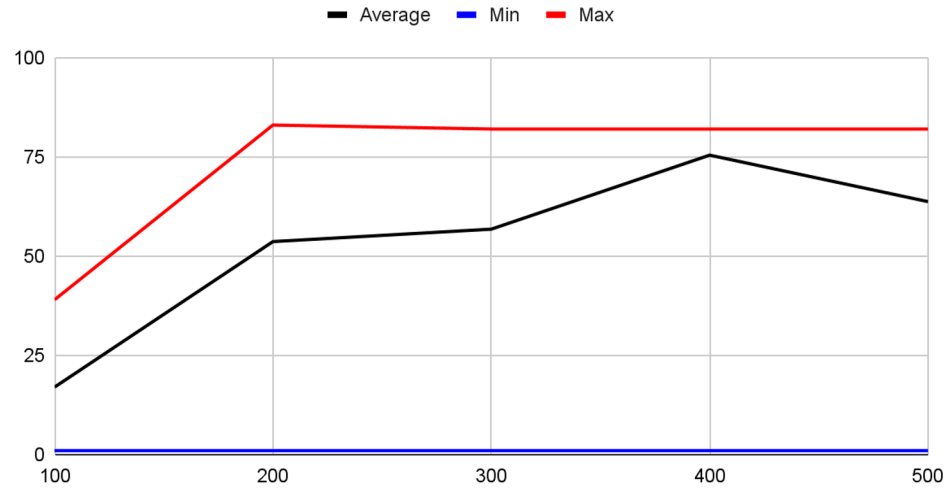
### Read Latency / Transaction Latency (秒)



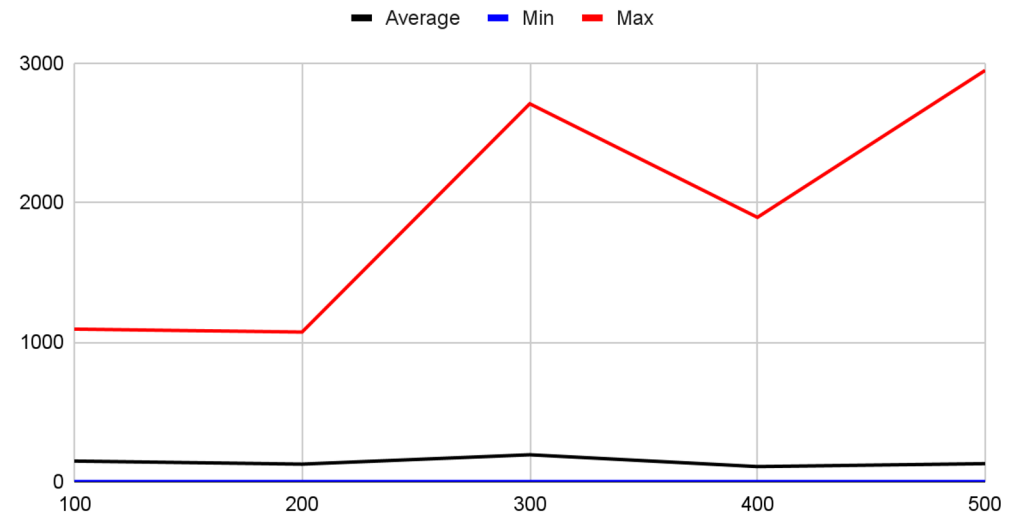
### Read Throughput / Transaction Throughput (TPS)



### CPU使用率 (%)



### ディスク負荷 (KB/s)



## テスト結果サマリ

### 2つのパターンでテスト

- 1. 送信速度は50 TPSで固定してそうトランザクション数を増加させる**
  - スループットは総トランザクション数を増やすにつれて50TPSに近づく
  - レイテンシーやCPU・ディスク負荷のMAX値は大きくなり、ばらついてくるが平均値は微増で、50TPS固定でリクエストを10,000まで増やしても、通常通りリクエストをさばけて負荷はほぼかかかっていないという印象
  - ただし、限界まで増やしていくと、ペンディングトランザクションがどんどん溜まっていき、17,000ぐらいで、処理が追いつかなくなる
  
- 1. 総トランザクション数を10,000で固定して目標TPSを増加させる**
  - スループットは200TPSから300TPSあたりで最大になり、300TPS以降はCPU使用率とディスク負荷は上昇するが、レイテンシーと負荷が大きくなるだけで、スループットは逆に減少していくので、このあたりが限界値と思われる
  - Ethereum mainnetは現在15 TPSほどと言われているので、分散性を犠牲にしてプライベートチェーンにすることによって20倍ほど速度の向上を期待できる