



CeADAR - The National Centre for Applied Data Analytics and Machine Intelligence

---

## Blockchain Scalability Testing (WP-3) Proposal

Document Type:	Test Plan (Proposal)
Project Title:	Blockchain Scalability Testing (WP-3) Proposal
Work Package:	3
Authors:	Prabhakaran AK, Saad Shahid, Oisín Boydell
Document Version:	0.1
Date:	March 2020
Number of pages:	??

### ABSTRACT

Proposed plan for the Blockchain Scalability Testing (WP-3).

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Tool</b>	<b>3</b>
2.1	Requirements . . . . .	3
2.2	Assumptions . . . . .	3
2.3	About Hyperledger Caliper . . . . .	3
2.4	Why Hyperledger Caliper . . . . .	3
2.5	Performance & Scalability Working Group - metrics . . . . .	4
2.5.1	Success Rate . . . . .	4
2.5.2	Transaction & Read Latency . . . . .	4
2.5.3	Transaction & Read Throughput . . . . .	4
2.5.4	Dependency Latency vs Throughput . . . . .	4
2.6	Resource Consumption . . . . .	5
<b>3</b>	<b>Experiment</b>	<b>5</b>
3.1	Experiment runs . . . . .	5
3.2	Plotting the results . . . . .	5
3.3	Hyperledger Caliper version . . . . .	5
3.4	Blockchain network . . . . .	6

# 1 Introduction

Blockchain scalability testing is proposed to predict the performance of the blockchain network while the consortium grows over the period of time.

As the number of nodes in the consortium grows, volume of the transactions going through the network is expected to grow as well.

This document proposes a test plan to analyze the impact of a growing blockchain network.

## 2 Tool

Following are the various blockchain bench-marking tools available in the market:

- Gremlin - Bench-marking tool to test the network components of the distributed ledger system.
- Hyperledger Caliper<sup>1</sup> - Tool to measures various performance metrics of smart contract. Supports multiple version of hyperledger fabric and few other blockchain platforms.
- MixBytes Tank<sup>2</sup> - Cloud-based blockchain bench-marking tool which does not have features to support consensus and underlying blockchain network component testing.

### 2.1 Requirements

The requirement is to analyze the scalability performance of a smart contract in a blockchain network. The scalability check will determine the latency variation against the variation in transaction volume and throughput.

### 2.2 Assumptions

The methodology outlined in this document will be used as a reference point for all the scalability tests on the smart contracts, that will be part of work package six (WP-6)

The scalability tests part of WP-3 will be conducted in phases, when each smart contract(s) in WP -6 becomes available.

For two different smart contracts having same processing overhead and computational complexity, the scalability performance check will be done against any one of them.

### 2.3 About Hyperledger Caliper

We are going to use Hyperledger Caliper [?], a blockchain bench-mark tool to measure the performance of the blockchain implementation with a set of predefined use cases.

Hyperledger caliper is a blockchain performance bench-marking tool from the Linux foundation.

### 2.4 Why Hyperledger Caliper

Caliper has a defined Performance & Scalability Working Group (PSWG) [?] which contains definitions and metrics for the blockchain network bench-marking.

It has support for multiple versions of hyperledger fabric which is our blockchain platform

---

<sup>1</sup><https://www.hyperledger.org/projects/caliper>

<sup>2</sup><https://mixbytes.io/tank>

for the present use case

Caliper helps us to determine the various-metrics for given volume of transactions at defined throughput.

## **2.5 Performance & Scalability Working Group - metrics**

The metrics that are defined by the PSWG of Hyperledger caliper are as follows:

### **2.5.1 Success Rate**

Measures all successful and failed transactions for a test cycle.

- This metrics is based on the volume of the transactions.
- Caliper allows users to configure the transaction volume for the testing.
- Caliper final report includes the success and failure rate for the given volume of transactions.

### **2.5.2 Transaction & Read Latency**

Measures the time for an issued transaction to be completed and a response being available to the application that issued the transaction.

- This metrics is based on the time taken for one single transaction in seconds.
- Latency is the output by hyperledger caliper for given volume of transactions at defined throughput.
- Maximum, minimum and average latency in seconds for the test cycle is provided.

### **2.5.3 Transaction & Read Throughput**

Measures the flow rate of all transactions through the system, in transactions per second, during the a cycle.

- This metrics defines the flow rate of transaction into the blockchain system.
- Throughput can be configured at caliper.
- Unit of measurement for throughput is Transactions per second (TPS).
- Various types of throughput feeding is possible in caliper.
  - fixed-rate : Fixed rate of throughput is maintained from start to end of the transaction volume.
  - linear-rate : Variable rate between beginning and towards end of the transaction volume.
  - composite-rate: Composite rate allows both fixed rate and linear rate throughput to be used on given transaction volume.

### **2.5.4 Dependency Latency vs Throughput**

Hyperledger caliper provides sample smart contracts to run the scalability check, and based on the results of running it against different transaction volume and throughput we arrived at the following assumption:

- Latency increases linearly with number of nodes in the network.
- Throughput is configurable for each test run.
- Latency increases when the volume of the transaction and throughput is increased.

The same will be verified against each of our smart contract across WP-6 in phases.

## 2.6 Resource Consumption

Measures the following resource parameters of the blockchain network

- Max. and Min, memory - Memory used in MegaBytes
- CPU resource consumption - CPU usage in percentage
- I/O traffic - KiloBytes/ MegaBytes

## 3 Experiment

Hyperledger caliper allows us to test the blockchain scalability performance.

We intend to evaluate impact of variuos metrics on blockchain network while scaling up on the following scenarios:

### 3.1 Experiment runs

For this experimental purpose we will keep the total volume of transaction fixed at 15000 and perform the following.

A series of tests will be run against the number of nodes vs throughput to find the latency

# of nodes / TPS	100	500	1500	2500
10				
25				
50				
100				
5000				

- For a given number of nodes the tests are run at various throughput.
- The above process is repeated for increased node counts.

### 3.2 Plotting the results

The results from the above experiments are plotted on charts e.g. latency for a given number of nodes vs throughput in transactions per seconds (TPS).

These charts will help us understand the impact of the scalability of the network physically (increasing the nodes) or by scaling up the volume of transactions and throughput.

### 3.3 Hyperledger Caliper version

Hyperledger Caliper 0.2 supports multiple versions of Hyperledger Fabric, starting from fabric V1.0 and above.

The configuration file contains the following details regarding the fabric network

- Docker commands (if not preexisting network).
- Fabric client node identity details.
- Channel and chaincode details.
- Organization identity details.
- Peers and Certificate authority server address details.

### 3.4 Blockchain network

Hyperledger caliper comes with default blockchain network containing two nodes. The network comes in following different variations:

- Solo orderer
- Kafka orderer
- Go level db
- Couch db

The network can be bootstrapped on any unix machine for testing purpose.

We can also use our own blockchain networks or networks that are bootstrapped in any of the cloud storage providing blockchain-as-a -service.