

*French version follows.*

**Purpose of the study:** We are looking for volunteer participants for the evaluation of the usability and efficiency of Blockchain benchmarking tools by conducting a comprehensive comparison between two Blockchain platforms, namely Hyperledger Fabric and Ethereum (Clique). Depending on your choice you can use either blockbench <https://github.com/ooibc88/blockbench> or BlockCompass <https://github.com/yorku-ease/blockcompass.git> using at least three workloads in your experiments. One “small”, one “medium”, one “large”.

**Eligibility:**

- Basic understanding of how to use command line, edit software configuration files and perform software experiments is necessary.

**OR**

- Having completed 2 or 3 years in an undergraduate program on Computer Science, Computer Engineering, Software Engineering or any equivalent studies would suffice.

- Knowledge about benchmarking tools and Blockchain are considered a plus.

**Resources for Blockcompass:**

1. **The user manual:** [BlockCompass.pdf](#)
2. **Github Repository:** <https://github.com/yorku-ease/blockcompass.git>
3. **Videos Tutorials for the tool:**
  - a. **Ethereum:** <https://www.youtube.com/watch?v=R91RybpXbVA>
  - b. **Hyperledger Fabric:** <https://youtu.be/Jgzrq-yXcj4>

**Resources for Blockbench:**

4. **The user manual:** [Blockbench.pdf](#)
5. **Github Repository:** <https://github.com/polytechnique-ease/blockbench>
6. **Videos Tutorials for the tool:**
  - a. **Ethereum:** <https://youtu.be/arFfhtueiO4>
  - b. **Hyperledger Fabric:** <https://youtu.be/HeeOY-7YeZA>

**In our study you will have to conduct a small comparative study using a benchmark tool. Choose one of the two tools recommended and perform the instructions below.**

**Task to complete if using “BlockCompass”:**

1. Install dependencies as explained in the manual page 17. (please make sure to use node.js version 10.22.0 and Golang version 1.17)
2. Configuration:
  - a. Network configuration: First you need to configure the network setting such as the target blockchain details, IP address to use .. The configuration steps can be found in details in page 20 of the manual.
  - b. Workload configuration: In order to set up the workload schedule ( set the number of users and duration those users will be run) you need to update the [schedule.list](#) file. Refer to the manual page 9 for more details.  
Use at least three workloads in your experiments. One “small”, one “medium”, one “large”. The three experiments would differ with respect to how much load they produce for the tested platform, in other words, how much data is sent to the platform:  
“small” : maximum number of users 50.  
“Medium”: maximum number of users 200.  
“Large”: maximum number of users 1000.
3. Launch the experiment by running [launcher.py](#) script.
4. Generate report at the end of the experiment by going to the frontend (running on port 4200) and selecting the generate report option.
5. Stop the experiment by running clear.sh script.
6. Complete survey:

<https://docs.google.com/forms/d/e/1FAIpQLSc5Osz-KfURDOhrFvAzDuBn2zVQV0rsYQy4dB7gF9RN5IXFew/viewform>

**Task to complete if using “Blockbench”:**

1. Install dependencies as explained in the blockbench manual page 3. (please make sure to use node.js version 10.22.0 and Golang version 1.17)
2. Configuration:
  - a. Start target network as explained in the manual page 7.
  - b. Start the experiment as explained in the manual page 7.

Use at least three workloads in your experiments. One “small”, one “medium”, one “large”. The three experiments would differ with respect to how much load they produce for the tested platform, in other words, how much data is sent to the platform:

“small” : Set transaction rate to 50 tx/s.

“Medium”: Set transaction rate to 200 tx/s.

“Large”: Set transaction rate to 1000 tx/s.

3. Format output as explained in page 8 of the manual after each experiment
4. Stop the experiment by running clear.sh script.
5. Complete survey:

<https://docs.google.com/forms/d/e/1FAIpQLSc5Osz-KfURDOhrFvAzDuBn2zVQV0rsYQy4dB7gF9RN5IXFew/viewform>

### **Consent form:**

Should you agree to participate in this voluntary survey, you will give us your consent and be asked to gather some data with respect to your experience using the benchmarking tools. Consent and data will be collected by a member of the research team that is not involved in the project.

Here is the link for the consent form:

[https://drive.google.com/file/d/1qgjm7\\_2AihE3EBaFx9gUoKsz6Rs4RIEx/view?usp=sharing](https://drive.google.com/file/d/1qgjm7_2AihE3EBaFx9gUoKsz6Rs4RIEx/view?usp=sharing)

Please email your response and the complete form to Wejdene Haouari via email at [haouari.wejdene1@gmail.com](mailto:haouari.wejdene1@gmail.com)

**Objectif de l'étude :** Nous recherchons des participants volontaires pour l'évaluation de la convivialité et de l'efficacité des outils de benchmarking Blockchain en effectuant une comparaison entre deux plateformes Blockchain, à savoir Hyperledger Fabric et Ethereum (Clique). Selon votre choix, vous pouvez utiliser soit blockbench <https://github.com/ooibc88/blockbench> soit BlockCompass <https://github.com/yorku-ease/blockcompass.git> en utilisant au moins trois charges de travail (workload) dans vos expériences. Un "petit", un "moyen", un "grand".

**Admissibilité à participer:**

- Une compréhension de base de l'utilisation de la ligne de commande, de la modification des fichiers de configuration logicielle et de la réalisation d'expériences logicielles est nécessaire.

***OU***

- Avoir terminé 2 ou 3 ans dans un programme de premier cycle en informatique, en génie informatique, en génie logiciel ou toute autre étude équivalente suffirait.

- Des connaissances sur les outils de benchmarking et Blockchain sont considérées comme un plus.

**Ressources pour Blockcompass:**

1. **Le manuel d'utilisation:** [BlockCompass.pdf](#)
2. **Github:** <https://github.com/polytechnique-ease/blockcompass>
3. **Videos demo:**
  - a. **Ethereum:** <https://www.youtube.com/watch?v=R91RybpXbVA>
  - b. **Hyperledger Fabric:** <https://youtu.be/Jgzrq-yXcj4>

### Ressources pour Blockbench:

1. Le manuel d'utilisation: [Blockbench.pdf](#)
2. Github: <https://github.com/polytechnique-ease/blockbench>
3. Videos demo:
  - a. Ethereum: <https://youtu.be/arFfhtueiO4>
  - b. Hyperledger Fabric: <https://youtu.be/HeeOY-7YeZA>

**Dans notre étude vous devrez réaliser une petite étude comparative à l'aide d'un outil de référence. Choisissez l'un des deux outils recommandés et suivez les instructions ci-dessous.**

### Tâche à effectuer si vous choisissez « BlockCompass » :

1. Installez les dépendances comme expliqué dans la page 17 du manuel.  
(assurez-vous d'utiliser node.js version 10.22.0 et Golang version 1.17)
2. Configuration:
  - a. Configuration réseau : vous devez d'abord configurer les paramètres réseau tels que les détails de la blockchain cible, l'adresse IP à utiliser.. Les étapes de configuration se trouvent dans les détails à la page 20 du manuel.
  - b. Configuration de la charge de travail : afin de configurer la planification de la charge de travail (définir le nombre d'utilisateurs et la durée d'exécution de ces utilisateurs), vous devez mettre à jour le fichier [schedule.list](#).

Reportez-vous à la page 9 du manuel pour plus de détails.

Utilisez au moins trois charges de travail dans vos expériences. Un "petit", un "moyen", un "grand". Les trois expériences diffèrent en ce qui concerne la charge qu'elles produisent pour la plate-forme testée, en d'autres termes, la quantité de données envoyées à la plate-forme :

« small » : nombre maximum d'utilisateurs 50.

« Moyen » : nombre maximum d'utilisateurs 200.

« Large » : nombre maximum d'utilisateurs 1000.

3. Lancez l'expérience en exécutant le script [launcher.py](#).
4. Générez un rapport à la fin de l'expérience en vous rendant sur le frontend ( port 4200) et en sélectionnant l'option de génération de rapport.
5. Arrêtez l'expérience en exécutant le script clear.sh.
6. Complétez le questionnaire:

<https://docs.google.com/forms/d/e/1FAIpQLSeomcXvEujrFEhF6iqXx0-spI0tixJyaAb32qToZBCeXcrbGg/viewform>

**Tâche à effectuer si vous choisissez « Blockbench » :**

1. Installez les dépendances comme expliqué dans le manuel de blockbench page 3. (assurez-vous d'utiliser node.js version 10.22.0 et Golang version 1.17)
2. Configuration:
  - a. Démarrez le réseau cible comme expliqué dans le manuel page 7.
  - b. Démarrez l'expérience comme expliqué dans le manuel page 7.

Utilisez au moins trois charges de travail dans vos expériences. Un "petit", un "moyen", un "grand". Les trois expériences diffèrent en ce qui concerne la charge qu'elles produisent pour la plate-forme testée, en d'autres termes, la quantité de données envoyées à la plate-forme :

« small » : définissez le taux de transaction à 50 tx/s.

« Moyen » : définissez le taux de transaction sur 200 tx/s.

« Large » : définissez le taux de transaction sur 1 000 tx/s.

3. Formater la sortie comme expliqué à la page 8 du manuel après chaque expérience

4. Arrêtez l'expérience en exécutant le script clear.sh.

Étude complète:

5. Complétez le questionnaire:

<https://docs.google.com/forms/d/e/1FAIpQLSeomcXvEujrFEhF6iqXx0-spI0tixJyaAb32qToZBCeXcrbGg/viewform>

**Formulaire de consentement:**

Si vous acceptez de participer à cette enquête volontaire, veuillez nous donner votre consentement et votre retour d'expérience sur l'utilisation de benchmarking tool. Le consentement et les données seront recueillis par un membre de l'équipe de recherche qui n'est pas impliqué dans le projet.

Voici le lien pour le formulaire de consentement :

<https://docs.google.com/document/d/1boPCQQsrTZffmLv04ohfXp2nmk1Di9A/edit?usp=drivesdk&oid=108625548253717777925&rtpof=true&sd=true>

Veuillez envoyer votre réponse et le formulaire complet à Wejdene Haouari par courrier électronique à [haouari.wejdene1@gmail.com](mailto:haouari.wejdene1@gmail.com)