

## Dokumentasi Reproduksi Eksperimen

# EFFICIENT TOPIC PARTITIONING OF APACHE KAFKA FOR HIGH-RELIABILITY REAL-TIME DATA STREAMING APPLICATIONS

Catatan: Panduan ini mengasumsikan Anda menggunakan sistem Ubuntu atau distribusi Linux lain yang menggunakan *apt* package manager.

### 1. Lakukan clone terhadap Kafka Partitioning Experiment Github Repository

Langkah pertama adalah mengkloning GitHub repository yang berisi kode untuk menjalankan eksperimen Kafka Topic Partitioning. Pastikan bahwa pada sistem Anda telah terpasang **git**. Jika belum, Anda dapat menginstalnya dengan perintah berikut:

```
sudo apt update
sudo apt install -y git
```

Kemudian, kloning repository dengan perintah berikut:

```
git clone https://github.com/bryan-ilman-on-github/kafka-part-exp.git
```

### 2. Navigasi ke Repo Eksperimen

Selanjutnya, kita navigasi ke repo github yang baru saja didownload dan pastikan bahwa script yang dibutuhkan mempunyai permission yang tepat untuk dijalankan. Jalankan perintah berikut:

```
cd kafka-part-exp
chmod +x run-experiment.sh
chmod +x collect-experiment.sh
```

### 3. Persiapan Docker pada Host

Pastikan pada host yang akan digunakan sudah terpasang **Docker** dan **Docker Compose**. Jika belum terpasang, ikuti langkah berikut untuk menginstalnya:

```
sudo apt update
sudo apt install -y docker.io docker-compose
```

Setelah Docker diinstal, aktifkan dan jalankan Docker *daemon* dengan perintah berikut:

```
sudo systemctl enable docker
sudo systemctl start docker
```

#### 4. Build Docker Images

Jalankan perintah berikut untuk membangun beberapa Docker *images* yang diperlukan:

```
docker build -t kafka-cluster-image -f kafka-cluster-dockerfile .  
docker build -t kafka-clients-image -f kafka-clients-dockerfile .
```

#### 5. Jalankan Eksperimen

Sebelum menjalankan eksperimen, pastikan sistem Anda memiliki minimal **128 GB RAM** dan **180 GB disk space** yang tersedia. Kebutuhan ini penting untuk memastikan eksperimen berjalan lancar tanpa hambatan. Jalankan eksperimen dengan perintah berikut:

```
./run-experiment.sh -n <number>
```

atau

```
./run-experiment.sh --number <number>
```

Isi **<number>** dengan nomor eksperimen yang ingin dijalankan (1-4). Opsi **-n** atau **--number** bersifat wajib dan digunakan untuk menentukan nomor eksperimen.

#### 6. Ambil Hasil Eksperimen

Setelah eksperimen selesai, ambil hasilnya dengan perintah berikut:

```
./collect-results.sh -n <number>
```

atau

```
./collect-results.sh --number <number>
```

Isi **<number>** dengan nomor eksperimen yang hasilnya ingin diambil (1-4). Opsi **-n** atau **--number** bersifat opsional. Jika tidak diisi, maka hasil dari semua nomor eksperimen akan diambil. Folder bernama **experiment-<number>** akan dibuat untuk menyimpan hasil eksperimen.