

1. Pendahuluan

Latar Belakang

Pada zaman ini teknologi berkembang dengan sangat pesat. Sebagai pengembang aplikasi yang berurusan dengan data diperlukan kemampuan *Big Data Analytics*. *Big Data* adalah sebuah teknik untuk menangkap, menganalisa, memproses serta memvisualisasikan *dataset* yang berpotensi sangat besar[1]. *Big Data* mempunyai beberapa karakteristik yang sering dijumpai yaitu *Volume*, *Varieties*, dan *Velocity*. Di dalam karakteristik *Varieties* tipe data yang biasa digunakan yaitu terstruktur, semi-terstruktur, dan tidak terstruktur[2]. *File System* saat ini menggunakan data yang tersusun secara terstruktur salah satu contohnya yaitu *SQL*. Untuk menangani data yang tidak terstruktur diperlukan suatu mekanisme *File System* yang menggunakan *NoSQL*.

Permasalahan yang sering dijumpai dalam *Big Data* yaitu dalam hal *storage* dan *processing*. Dalam *processing* kemampuan prosesing akan menjadi terhambat ketika pembacaan pada storage masih lambat, oleh karena itu dibutuhkan *file system* yang memiliki kemampuan pengaksesan data yang cepat.

Hadoop Distribute File System (HDFS) suatu *File System* yang mampu menyimpan data yang besar, menghubungkan blok data yang besar untuk aplikasi dalam sistem terdistribusi[3].

Cassandra File System (CFS) merupakan *open-source*, distribusi, penyimpanan luas kolom, dan sistem manajemen database *NoSQL* yang dirancang untuk menangani sejumlah besar data di banyak server komoditas, menyediakan ketersediaan tinggi tanpa titik kegagalan tunggal[4]. Untuk menangani sejumlah besar data di banyak server komoditas, menyediakan ketersediaan tinggi tanpa titik kegagalan tunggal, dengan replikasi tanpa master asinkron yang memungkinkan operasi latensi rendah untuk semua klien. CFS menawarkan pos relasional sebagai solusi basis data yang berarti jika melalui pos ini CFS tidak menawarkan semua fitur dari database tradisional tetapi tetap mengikuti keluarga kolom *keyspace* untuk penyimpanan basis data dan memperkenalkan indeks primer dan sekunder untuk data-data yang besar. CFS itu sendiri memiliki skema data yang tidak rasional dalam mendapatkan informasi data yang tidak terstruktur yaitu *NoSQL*.

Pada penelitian terkait[11], peneliti melakukan analisis performansi *file system* menggunakan HDFS dan CFS. Namun, dalam penelitian tersebut, peneliti menggunakan *load-read* dan *load-write*. Sehingga hasil *throughput* yang dibutuhkan belum maksimal. Atas latar belakang tersebut, penulis membahas mengenai analisis HDFS dan CFS pada performansi penyimpanan pada *file system* menggunakan *NoSQL*.

Topik dan Batasannya

Adapun batasan yang terdapat pada tugas akhir ini adalah sebagai berikut :

- Analisis menggunakan *Stressing tools Yahoo! Cloud Serving Benchmarking*
- Konfigurasi *file system* seluruhnya *default*, kecuali *record count* dalam menguji performansi di *file system* tersebut.
- Parameter pengujian hanya menggunakan *run time* dan *throughput*.

Berdasarkan latar belakang yang sudah dipaparkan sebelumnya permasalahan yang ditarik adalah:

- Untuk membaca data yang tidak terstruktur dibutuhkan suatu mekanisme yang disebut *NoSQL* dan *stressing tools yahoo! cloud sering benchmarking*