

## INTISARI

### PENGARUH GARBAGE COLLECTOR TERHADAP PERFORMA APACHE SPARK PADA PENGAPLIKASIAN BIG DATA

Oleh  
Raffi Febriandika Utama

Pemrosesan big data bergantung terhadap resource komputasi terutama memori yang besar. Implementasi aplikasi big data umum dijumpai berbasis *Java Virtual Machine (JVM)*. Tantangan optimasi dalam big data dapat difokuskan dengan menguji komponen internal yang bekerja langsung dengan memori. Garbage collector merupakan komponen di *Java Virtual Machine (JVM)* untuk melakukan manajemen memori secara otomatis.

Penelitian ini menganalisis performa garbage collector G1GC, ZGC dan ShenandoahGC dengan pengujian performa. Pengujian performa dilakukan sebanyak 3 kali untuk masing-masing parameter garbage collector dan parameter memori. Pengujian akan mencatat data total garbage collection, waktu runtime aplikasi, penggunaan memori, pause time, durasi dan beban CPU.

Hasil penelitian menunjukkan garbage collector memiliki karakteristik masing-masing dan keunggulan performa yang bervariasi. Runtime pengujian paling cepat dengan 1 menit 27 detik berjalan dengan garbage collector G1GC mengungguli ZGC dan ShenandoahGC. Pause time paling cepat dengan 0.001 detik berjalan dengan ZGC mengungguli ShenandoahGC dan G1GC. Throughput paling tinggi 96.20% berjalan dengan ShenandoahGC mengungguli ZGC dan G1GC.

**Kata kunci:** *Java Virtual Machine (JVM)*, garbage collector, Apache Spark, pengujian performa

## ABSTRACT

### THE EFFECT OF GARBAGE COLLECTOR ON THE PERFORMANCE OF APACHE SPARK WITH BIG DATA STUDY CASE

By  
Raffi Febriandika Utama

Big data processing depends on computational resources, especially large memory. Implementation of big data applications is commonly found based on *Java Virtual Machine (JVM)*. The challenge of optimization in big data can be focused with testing the internal component that works directly on memory. Garbage collector is a component in *Java Virtual Machine (JVM)* that automatically manages memory.

This research is analyzing the performance of garbage collector G1GC, ZGC and ShenandoahGC with performance testing. The performance testing will run 3 times for garbage collector parameter and memory parameter. The performance testing will collect total garbage collection, application runtime, memory used, pause time, duration and load CPU.

The result concludes each garbage collector has unique characteristics and varying performance strength. The fastest runtime with 1 minute 27 seconds run in G1GC outperforms ZGC and ShenandoahGC. The fastest pause time with 0.01 seconds run in ZGC outperforms ShenandoahGC and G1GC. The highest throughput with 96.20% run in ShenandoahGC outperforms ZGC and G1GC.

**Keywords:** *Java Virtual Machine (JVM)*, garbage collector, Apache Spark, pengujian performa