TUGAS AKHIR SBD

WORDCOUNT ON HADOOP VS JAVA

KELOMPOK 1

SPESIFIKASI PERANGKAT

Processor 12th Gen Intel(R) Core(TM) i7-1260P   2.10 GHz
Installed RAM 16.0 GB (15.7 GB usable)

Wordcount Java Code

```java
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.util.HashMap;
import java.util.Map;

public class wordcount {

    Run | Debug
    public static void main(String args[]) {
        System.out.println(x:"Counting Words");
        Map<String, Integer> wordCountMap = new HashMap<>();

        try (BufferedReader br = new BufferedReader(new FileReader(fileName:"gutenberg-1G.txt"))) {
            String line;
            while ((line = br.readLine()) != null) {
                StringBuilder sb = new StringBuilder();
                String[] words = line.split(regex:"\\s+");
                for (String word : words) {
                    sb.append(word).append(str:" ");
                    wordCountMap.put(word, wordCountMap.getOrDefault(word, defaultValue:0) + 1);
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        try (BufferedWriter bw = new BufferedWriter(new FileWriter(fileName:"1G_outputJava.txt"))) {
            for (Map.Entry<String, Integer> entry : wordCountMap.entrySet()) {
                String wordCount = entry.getKey() + " " + entry.getValue();
                bw.write(wordCount);
                bw.newLine();
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

HADOOP VS JAVA (10MB)

21.658 s

```
Execution time: 21.658 s
PS C:\hadoop\sbin>
```

0.334 s

```
ptime 1.0 for Win32, Freeware - http://www.pc-tools.net/
Copyright(C) 2002, Jem Berkes <jberkes@pc-tools.net>

=== java wordcount ===
Counting Words

Execution time: 0.334 s
PS C:\Users\Rafi2\00_TEMPLATE_JAVA>
```

HADOOP VS JAVA (100MB)

39.324 s

```
Execution time: 39.324 s
PS C:\hadoop\sbin>
```

4.989 s

```
ptime 1.0 for Win32, Freeware - http://www.pc-tools.net/
Copyright(C) 2002, Jem Berkes <jberkes@pc-tools.net>

=== java wordcount ===
Counting Words

Execution time: 4.989 s
PS C:\Users\Rafi2\00_TEMPLATE_JAVA>
```

HADOOP VS JAVA (1GB)

114.251 s

```
Execution time: 114.251 s
PS C:\hadoop\sbin>
```

49.462 s

```
ptime 1.0 for Win32, Freeware - http://www.pc-tools.net/
Copyright(C) 2002, Jem Berkes <jberkes@pc-tools.net>

=== java wordcount ===
Counting Words

Execution time: 49.462 s
PS C:\Users\Rafi2\00_TEMPLATE_JAVA>
```

ANALISIS

File 10 MB

Ketika memproses file berukuran 10 MB, program Word Counter yang diimplementasikan dalam Java cenderung memberikan waktu eksekusi yang lebih cepat daripada menggunakan Hadoop. Hal ini disebabkan oleh ukuran file yang relatif kecil, sehingga Java dapat menangani proses penghitungan kata dengan cepat tanpa memerlukan distribusi dan paralelisasi tugas yang kompleks seperti yang dilakukan oleh Hadoop.

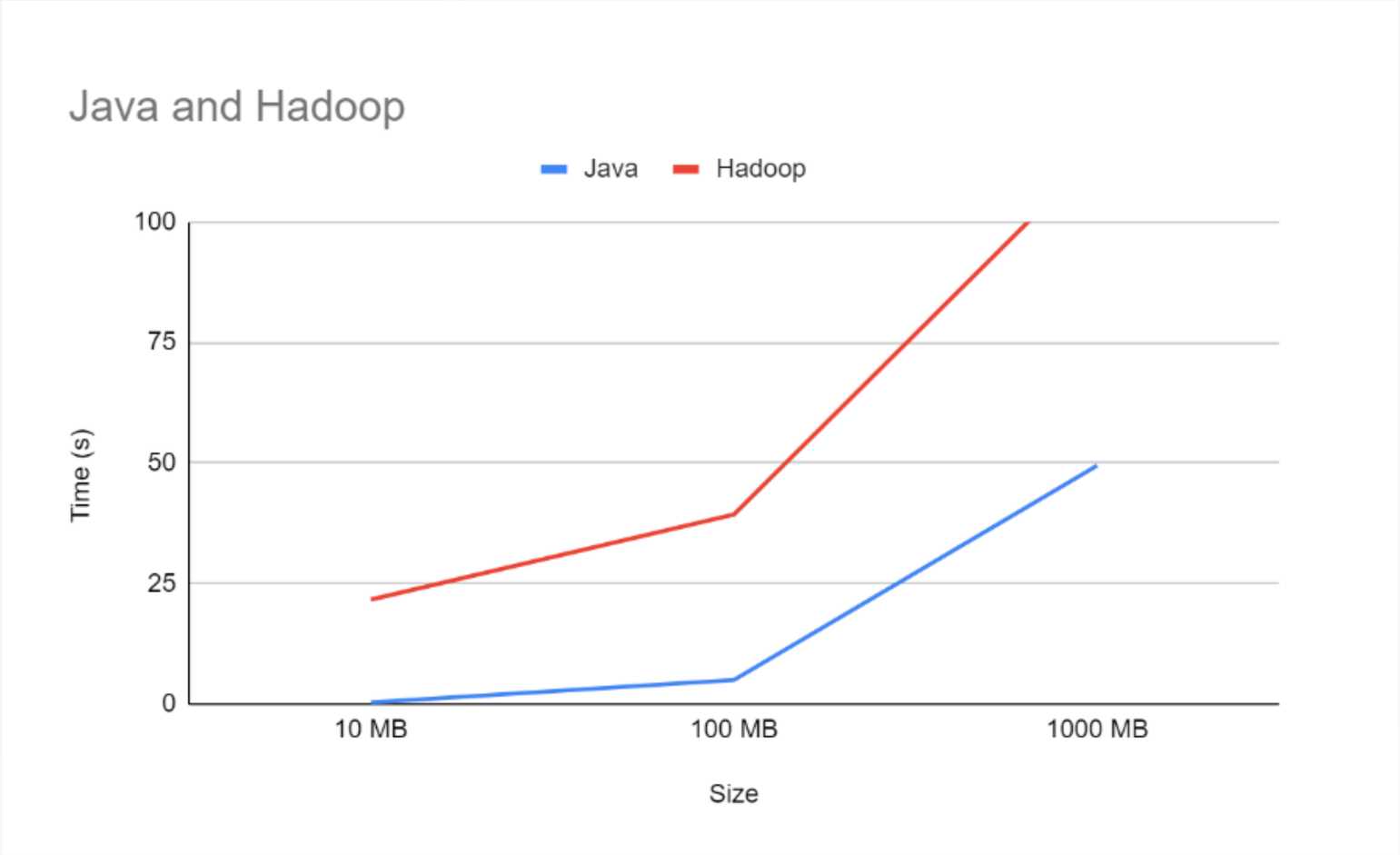
File 100 MB

Pada file berukuran 100 MB, meskipun Hadoop masih mampu menghitung kata dengan baik, Java masih menunjukkan keunggulan dalam hal waktu eksekusi. Hal ini dikarenakan ukuran file masih relatif kecil.

File 1 GB

Ketika kami memperbesar ukuran file menjadi 1 GB, keunggulan Hadoop menjadi lebih jelas. Java mungkin menghadapi keterbatasan daya komputasi dan memori dalam memproses file sebesar itu secara efisien. Di sisi lain, Hadoop menjadi efektif dalam memproses file yang lebih besar karena dapat memanfaatkan distribusi dan paralelisasi tugas secara lebih optimal. Dengan membagi tugas menjadi tugas-tugas kecil yang dikerjakan oleh beberapa node secara paralel, Hadoop dapat mengoptimalkan penggunaan sumber daya dan mempercepat waktu eksekusi secara signifikan.

ANALISIS


Java and Hadoop
Java
Hadoop
100
75
50
25
0
Time (s)
10 MB
100 MB
1000 MB
Size