

## Eksperimen Konfigurasi *Data Center* dan *Virtual Machines* Pada Simulator *CloudSim* Terhadap Kinerja *Cloudlet* (*Data Center and Virtual Machines Configuration Experiments in the CloudSim Simulator on Cloudlet Performance*)

Diva Septiawan, Giri Wahyu Wiriasto\*

Jurusan Teknik Elektro, Fakultas Teknik Universitas Mataram, Jalan Majapahit no.62 Mataram 83125, Indonesia

\*Corresponding author: [giriwahyuwiriasto@unram.ac.id](mailto:giriwahyuwiriasto@unram.ac.id)

(Diterima: 21 Februari 2024; Diterima: 14 Juni 2024; Tersedia online : 29 June 2024)

### ABSTRACT

Cloud computing has become a crucial paradigm in the management and delivery of computing services. One vital aspect of cloud computing is the performance of cloudlet execution, referring to the processing or execution of tasks within the cloud computing environment. Hence, a study was conducted to investigate the influence of various data center (DC) and virtual machine (VM) configurations on cloudlet execution time. The objective of this research is to analyze the impact of Data Center and virtual machine configurations on cloudlet performance in a simulation environment using CloudSim, an open-source simulation toolkit commonly used to evaluate and analyze cloud computing systems. The primary focus of this study is on the performance of cloudlets, which are the smallest units of work in the cloud computing model. The research involves variations in parameters related to the Data Center and virtual machines, including the number of physical machines, resource allocation, and task management schemes. Experiments were conducted to measure the impact of these configuration changes on cloudlet response time, throughput, and resource usage. The performance measurements are expected to provide in-depth insights into how Data Center and virtual machine configurations can affect the efficiency and effectiveness of cloud computing.

### ABSTRAK

Komputasi awan telah menjadi paradigma penting dalam pengelolaan dan penyediaan layanan komputasi. Salah satu aspek krusial dalam komputasi awan adalah kinerja eksekusi cloudlet, yang mengacu pada proses pemrosesan atau eksekusi tugas di dalam lingkungan komputasi awan. Oleh karena itu, dilakukan penelitian untuk menginvestigasi pengaruh variasi konfigurasi data center (DC) dan mesin virtual (VM) terhadap waktu eksekusi cloudlet. Tujuan penelitian ini adalah untuk menganalisis dampak konfigurasi Data Center dan mesin virtual terhadap kinerja cloudlet dalam lingkungan simulasi menggunakan CloudSim, sebuah toolkit simulasi sumber terbuka yang umum digunakan untuk mengevaluasi dan menganalisis sistem komputasi awan. Fokus utama penelitian ini adalah kinerja cloudlet, yang merupakan unit pekerjaan terkecil dalam model komputasi awan. Penelitian melibatkan variasi parameter pada Data Center dan mesin virtual, termasuk jumlah mesin fisik, alokasi sumber daya, dan skema manajemen tugas. Eksperimen dilakukan untuk mengukur dampak perubahan konfigurasi tersebut terhadap waktu respon cloudlet, throughput, dan penggunaan sumber daya. Hasil pengukuran kinerja diharapkan dapat memberikan wawasan mendalam tentang bagaimana konfigurasi Data Center dan mesin virtual dapat memengaruhi efisiensi dan efektivitas komputasi awan.

### Keywords:

Cloudsim cloudlet;  
Cloud computing simulator;  
Data center and virtual machine;

### Kata kunci:

Cloudsim cloudlet;  
Cloud computing simulator;  
Data center and virtual machine;

## 1. Pendahuluan

Komputasi awan telah memainkan peran sentral dalam evolusi teknologi informasi, memberikan solusi yang skalabel dan fleksibel untuk pemrosesan dan penyimpanan data.

Sebagai paradigma penting dalam lingkungan komputasi awan, pengoptimalan kinerja sistem menjadi krusial untuk memastikan efisiensi dan efektivitas penggunaan sumber daya komputasi (Buyya, 2009). Simulasi komputasi awan menggunakan aplikasi CloudSim (Cloud Simulator) memiliki peran penting dalam mengembangkan dan memahami sistem awan. *CloudSim* memungkinkan para peneliti, pengembang, dan profesional IT untuk mensimulasikan lingkungan awan dengan cara yang realistis, tanpa perlu melibatkan infrastruktur fisik yang sesungguhnya. Dengan *CloudSim*, kita dapat mengevaluasi kinerja dan efisiensi sistem awan dalam berbagai skenario, memperhitungkan faktor-faktor seperti jumlah host, spesifikasi mesin virtual, dan beban kerja (*cloudlet*). Simulasi ini memberikan wawasan tentang bagaimana sistem akan berperilaku di dunia nyata, memungkinkan pengoptimalan desain sebelum implementasi actual (Foster, 2008; Vaquero, 2011). Salah satu aspek kunci dalam mencapai tujuan ini adalah memahami dan mengelola waktu eksekusi *cloudlet*, yaitu proses pemrosesan tugas terkecil dalam model komputasi awan. *Cloudlet* pada *cloudSim* adalah unit kerja yang merepresentasikan beban kerja atau tugas yang dapat dijalankan di dalam lingkungan komputasi awan. Secara khusus, *cloudlet* adalah representasi dari pekerjaan yang dapat dijalankan pada mesin virtual (VM) di pusat data awan. Dalam *cloudSim*, pengguna dapat mendefinisikan *cloudlet* untuk menggambarkan pekerjaan atau tugas yang perlu dilakukan. *Cloudlet* memiliki atribut seperti panjang pekerjaan (jumlah instruksi), ukuran data, dan tingkat prioritas. Saat simulasi dijalankan, *cloudSim* memproses *cloudlet* ini pada mesin virtual yang sesuai di pusat data awan (Calheiros, 2011).

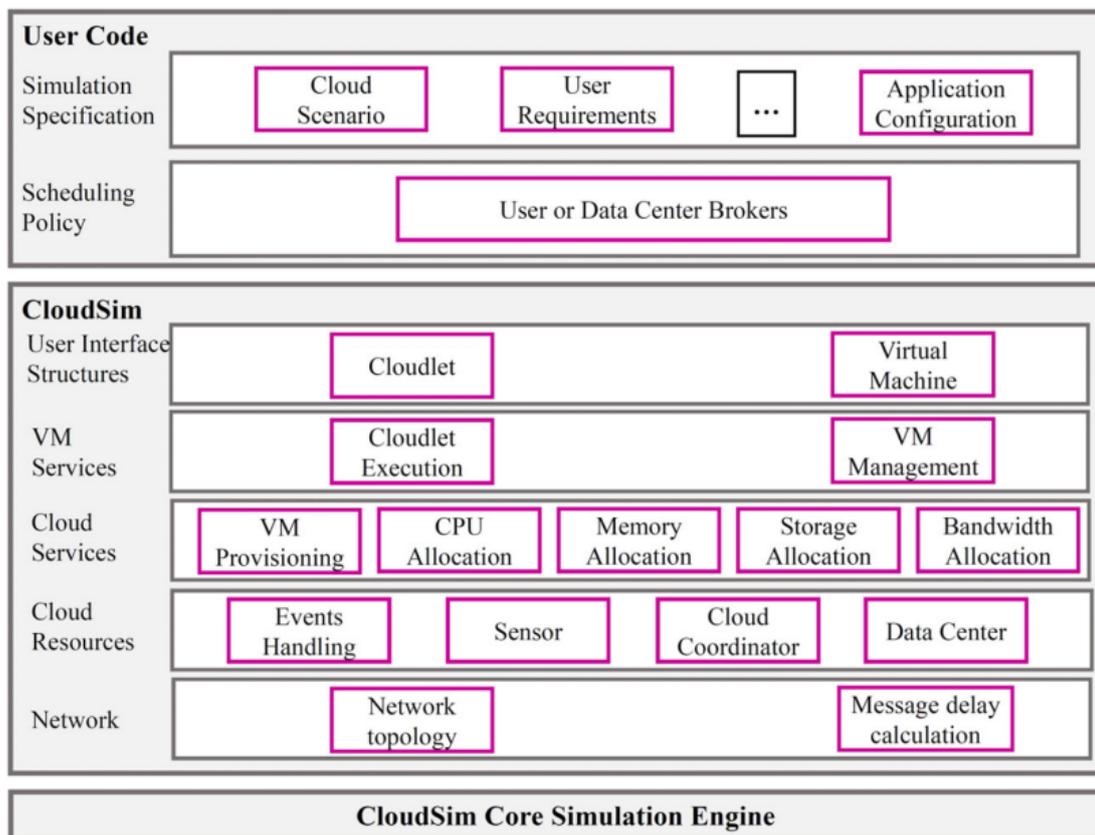
Pengguna dapat menggunakan *cloudlet* untuk menguji dan menganalisis kinerja sistem awan, memahami bagaimana pekerjaan didistribusikan di antara VM, dan mengevaluasi berbagai parameter sistem seperti waktu eksekusi, penggunaan sumber daya, dan efisiensi sistem secara keseluruhan. Sejalan dengan kompleksitas infrastruktur komputasi awan, konfigurasi *Data Center* (DC) dan *Virtual Machines* (VMs) memiliki peran utama dalam membentuk kinerja sistem secara keseluruhan. Dalam konteks ini, penelitian sebelumnya telah dilakukan untuk menganalisis pengaruh variasi konfigurasi DC dan VMs terhadap waktu eksekusi *cloudlet* (Beloglazov, 2010). Penelitian tersebut mendalam pada parameter-parameter kritis seperti jumlah mesin fisik, alokasi sumber daya, dan skema manajemen tugas, dengan menggunakan *cloudSim* sebagai alat simulasi yang umum digunakan. Dalam *cloudSim*, MIPS (*Million Instructions Per Second*) merujuk pada kapasitas pemrosesan suatu mesin virtual (VM) atau host dalam simulasi lingkungan awan. Konsep ini membahas seberapa cepat VM atau host dapat mengeksekusi instruksi pemrosesan. Parameter konfigurasi VM atau host dalam simulasi, memungkinkan simulasi efisiensi dalam menangani pekerjaan (*cloudlet*) dalam konteks lingkungan awan. Pengaturan nilai MIPS memainkan peran kunci, di mana nilai yang tinggi mencerminkan kapasitas pemrosesan yang tinggi dan kemampuan eksekusi instruksi yang cepat. Sebaliknya, nilai MIPS yang rendah menunjukkan kapasitas pemrosesan yang lebih rendah. Analisis nilai MIPS ini mendukung pemahaman dan evaluasi kinerja sistem awan yang sedang disimulasikan dalam *cloudSim* (Calheiros, 2011).

Melalui penelitian ini, diharapkan dapat diperoleh wawasan mendalam tentang bagaimana konfigurasi DC dan VMs dapat memengaruhi kinerja eksekusi *cloudlet*. Dengan pemahaman yang lebih baik terkait faktor-faktor yang mempengaruhi waktu eksekusi *cloudlet*, administrator sistem dan peneliti dapat merancang strategi yang lebih efektif dalam mengelola sumber daya komputasi awan. Implikasi praktis dari penelitian ini diharapkan dapat memberikan kontribusi signifikan terhadap pengembangan dan pemeliharaan infrastruktur komputasi awan yang adaptif dan efisien. Oleh karena itu, penelitian ini memperluas wawasan kita tentang dinamika kompleks yang memengaruhi kinerja komputasi awan melalui analisis mendalam terhadap waktu eksekusi *cloudlet*.

## 2. Metodologi

### 2.1. Arsitektur CloudSim

Arsitektur *CloudSim* melibatkan beberapa elemen utama yang membentuk kerangka kerja simulasi komputasi awan. Ini mencakup struktur antarmuka pengguna (*User Interface Structures*) untuk interaksi pengguna dengan sistem, layanan mesin virtual (*Virtual Machines Services*) untuk manajemen sumber daya VM, layanan awan (*Cloud Services*) yang menyediakan layanan umum, sumber daya awan (*Cloud Resources*) yang mencakup host fisik dan virtual, serta struktur jaringan (*Network*) yang memodelkan komunikasi dan pengaruh jaringan. Saat ini, penelitian difokuskan pada lapisan *Virtual Machines Services*, khususnya pada eksekusi tugas (*Cloudlet Execution*) di dalam mesin virtual. Ini mencerminkan upaya untuk mendalami pada kinerja dan efisiensi eksekusi tugas dalam lingkungan simulasi *CloudSim*, memberikan kontribusi penting dalam pemahaman dan pengembangan sistem komputasi awan.



**Gambar 1.** Arsitektur Cloudsim (Calheiros, 2011)

Gambar 1 adalah sebuah gambaran tentang arsitektur dasar cloudsims. Dapat dilihat bahwa cloudsims terdiri dari *User Interface Structures*, *Virtual Machines Services*, *Cloud Services*, *Cloud Resources*, dan *Network*. Penelitian yang dilakukan saat ini berada pada layer *Virtual Machines Services* yaitu *Cloudlet Execution*.

## 3. Hasil dan Pembahasan

Dilakukan 3 percobaan dengan program yang berbeda untuk mengetahui pengaruh variasi konfigurasi *Data Center* dan *Virtual Machines* terhadap waktu eksekusi *cloudlet*. Dimana ketiga program tersebut memiliki jumlah data center dan virtual machines yang berbeda. Untuk ketiga program tersebut dapat dilihat pada sub bab 3.1.

### 3.1. Bagian kode skrip pada simulasi *cloudSim*

Eksperimen ke-1 berikut ini yakni merubah parameter pada objek '*datacenter*' pada potongan kode skrip dalam bahasa java. Pada simulasi *cloudSim* merupakan contoh sederhana

yang menggunakan *cloudSim Toolkit* untuk mensimulasikan lingkungan *cloud computing*. Bagian yang ditampilkan adalah sebagian dari keseluruhan program. Bagian ini adalah bagian pembuatan objek 'datacenter' dan sebuah parameter virtual machines (vm) dengan spesifikasinya. Berikut ini pada eksperimen dua dibuat sebuah variabel 'datacenter' dengan 1 host dan sebuah virtual machines.

Pertama, sebuah objek Datacenter dengan nama "Datacenter\_0" dibuat untuk mewakili pusat data dalam simulasi. Selanjutnya, objek DatacenterBroker dibuat untuk bertindak sebagai broker dalam sistem, dan ID broker diambil untuk keperluan identifikasi. Kemudian, sebuah mesin virtual (VM) diinisialisasi dengan nilai-nilai seperti ID VM, kapasitas pemrosesan (MIPS), ukuran, kapasitas RAM, lebar pita, jumlah unit pemrosesan (PE), tipe hipervisor (dalam hal ini "Xen"), dan skema jadwal penggunaan pemrosesan untuk cloudlet pada VM, yang ditentukan sebagai CloudletSchedulerTimeShared. Inisialisasi ini menjadi langkah awal penting dalam konfigurasi elemen-elemen simulasi, membentuk dasar untuk mengevaluasi kinerja dan efisiensi sistem komputasi awan dalam konteks simulasi CloudSim.

```
Datacenter datacenter0 = createDatacenter("Datacenter_0");

    DatacenterBroker broker = createBroker();
    int brokerId = broker.getId();

    vmlist = new ArrayList<Vm>();

    int vmid = 0;
    int mips = 1000;
    long size = 10000;
    int ram = 512;
    long bw = 1000;
    int pesNumber = 1;
    String vmm = "Xen";
    Vm vm = new Vm(vmid, brokerId, mips, pesNumber,
ram, bw, size, vmm, new CloudletSchedulerTimeShared());
```

#### Eksperimen ke-2

Pada eksperimen ke-2, dibuat satu buah data center dengan 1 host dan dua buah virtual machines dengan spesifikasi yang sama. Untuk spesifikasi lengkapnya dapat dilihat pada tabel program eksperimen ke-2 tersebut.

```
Datacenter datacenter0 = createDatacenter("Datacenter_0");
    DatacenterBroker broker = createBroker();
    int brokerId = broker.getId();
    vmlist = new ArrayList<Vm>();
    int vmid = 0;
    int mips = 250;
    long size = 10000; //image size (MB)
    int ram = 512; //vm memory (MB)
    long bw = 1000;
    int pesNumber = 1; //number of cpus
    String vmm = "Xen"; //VMM name

    Vm vm1 = new Vm(vmid, brokerId, mips, pesNumber, ram,
bw, size, vmm, new CloudletSchedulerTimeShared());
    vmid++;
```

```
Vm vm2 = new Vm(vmid, brokerId, mips, pesNumber, ram,  
bw, size, vmm, new CloudletSchedulerTimeShared());
```

Objek Datacenter dengan nama "Datacenter\_0" dibuat dengan memanggil fungsi *createDatacenter*, yang mungkin telah terdefinisi sebelumnya untuk mengonfigurasi pusat data dalam simulasi. Selanjutnya, objek *DatacenterBroker* diciptakan dengan memanggil fungsi *createBroker*, dan ID broker diambil untuk tujuan identifikasi. Sebuah *ArrayList* dengan nama *vmList* dideklarasikan untuk menyimpan daftar VM.

Kemudian, dua VM dibuat dengan nilai-nilai inisialisasi tertentu seperti ID VM, kapasitas pemrosesan (MIPS), ukuran, kapasitas RAM, lebar pita, jumlah unit pemrosesan (PE), tipe hipervisor (dalam hal ini "Xen"), dan skema jadwal penggunaan pemrosesan untuk *cloudlet* pada VM, yang ditetapkan sebagai *CloudletSchedulerTimeShared*. Setiap VM diberikan ID yang berbeda dengan menaikkan nilai *vmid*. Proses ini membentuk langkah awal dalam menentukan karakteristik dan konfigurasi VM untuk keperluan simulasi *cloudSim*.

### Eksperimen ke-3

Pada kode program eksperimen ke-3 dibuat sebuah data center dengan 2 *host* dan 2 buah *virtual machines* yang spesifikasinya berbeda.

```
Datacenter datacenter0 = createDatacenter("Datacenter_0");  
DatacenterBroker broker = createBroker();  
int brokerId = broker.getId();  
vmList = new ArrayList<Vm>();  
int vmid = 0;  
int mips = 250;  
long size = 10000; //image size (MB)  
int ram = 2048; //vm memory (MB)  
long bw = 1000;  
int pesNumber = 1; //number of cpus  
String vmm = "Xen"; //VMM name  
Vm vm1 = new Vm(vmid, brokerId, mips, pesNumber, ram,  
bw, size, vmm, new CloudletSchedulerTimeShared());  
vmid++;  
Vm vm2 = new Vm(vmid, brokerId, mips * 2, pesNumber,  
ram, bw, size, vmm, new CloudletSchedulerTimeShared());
```

Sebuah objek Datacenter dengan nama "Datacenter\_0" dibuat melalui panggilan fungsi *createDatacenter*, yang mungkin telah sebelumnya didefinisikan untuk menentukan konfigurasi pusat data dalam simulasi. Selanjutnya, objek *DatacenterBroker* dibuat melalui fungsi *createBroker*, dan ID broker diambil untuk keperluan identifikasi. Daftar VM, disebut *vmList*, dideklarasikan sebagai *ArrayList* untuk menyimpan VM yang akan dibuat. Kemudian, dua VM, *vm1* dan *vm2*, diinisialisasi dengan parameter seperti ID VM, kapasitas pemrosesan (MIPS), ukuran, kapasitas RAM, lebar pita, jumlah unit pemrosesan (PE), tipe hipervisor ("Xen"), dan skema jadwal penggunaan pemrosesan untuk *cloudlet* pada VM, yang ditentukan sebagai *CloudletSchedulerTimeShared*. VM pertama, *vm1*, memiliki konfigurasi standar dengan nilai MIPS, RAM, dan bandwidth tertentu. ID VM dinaikkan untuk persiapan pembuatan VM berikutnya. VM kedua, *vm2*, memiliki nilai MIPS dua kali lipat dari *vm1* untuk menciptakan variasi dalam kapasitas pemrosesan. Langkah-langkah ini menciptakan dua VM yang berbeda dalam simulasi, dengan konfigurasi yang berbeda untuk memungkinkan analisis kinerja.

### 3.2. Hasil Simulasi

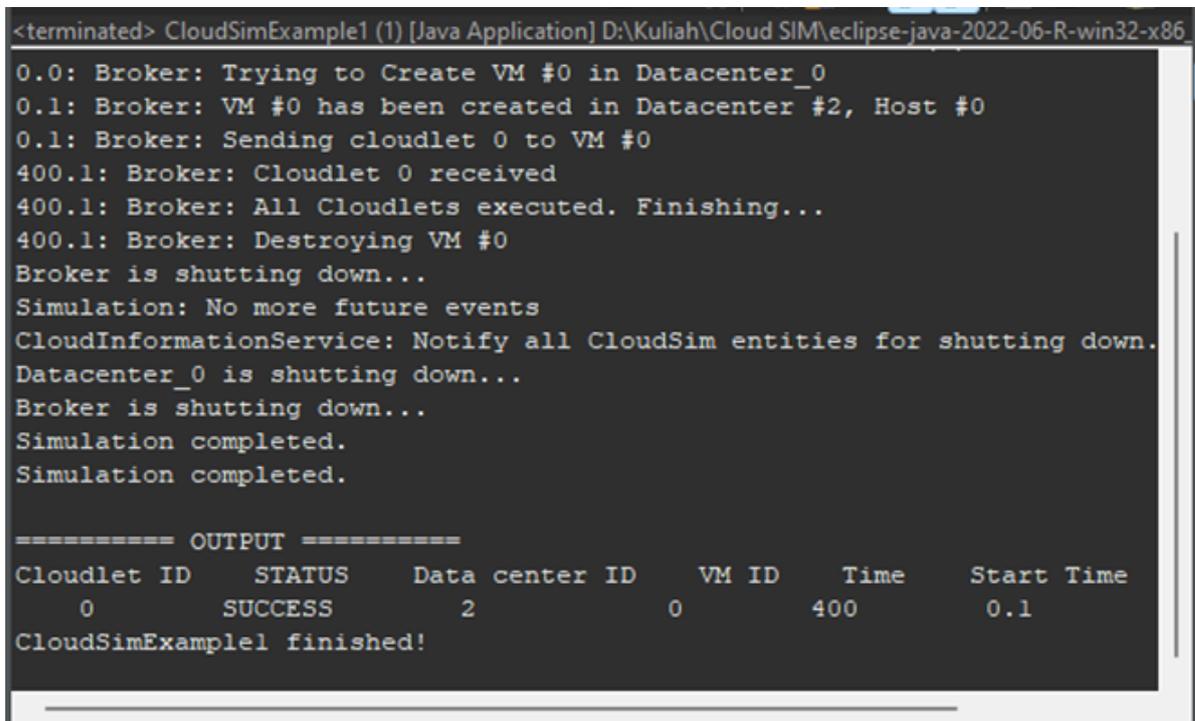
Untuk simulasi *cloudsim* yang dilakukan akan diukur dari beberapa parameter yaitu mips, RAM dan jumlah CPU terdapat pada bagian program (Wirayudha, 2024). Sedangkan untuk spesifikasi

perangkat keras yang digunakan tidak disebutkan secara lengkap pada referensi utama (Calheiros, 2011), tetapi dapat kami asumsikan spesifikasi mengikuti spek perangkat keras saat simulator ini dijalankan. Berikut informasi lain mengenai virtual machine yang disimulasikan.

```
//deskripsi VM
int vmid = 0;
int mips = 100;
long size = 10000; //ukuran gambar (MB)
int ram = 512; //vm memory (MB)
long bw = 1000;
int pesNumber = 1; //jumlah CPU
String vmm = "Xen"; //nama VMM
```

Simulasi cloudsims yang dilakukan akan diukur dari beberapa parameter spesifikasi virtual machines dan jumlah host pada data center. Untuk mengetahui pengaruh dari parameter tersebut maka dapat dilihat hasil running atau eksekusi dari ketiga program tersebut terlebih dahulu yaitu pada gambar 2, 3, dan 4.

Pada gambar 2, dapat dilihat bahwa kecepatan eksekusi dari eksperimen ke-1 adalah 400 ms.



```
<terminated> CloudSimExample1 (1) [Java Application] D:\Kuliah\Cloud SIM\eclipse-java-2022-06-R-win32-x86_
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
400.1: Broker: Cloudlet 0 received
400.1: Broker: All Cloudlets executed. Finishing...
400.1: Broker: Destroying VM #0
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time
0            SUCCESS   2                0       400    0.1
CloudSimExample1 finished!
```

**Gambar 2.** Hasil eksekusi eksperimen ke-1

Pada gambar 3 dapat dilihat bahwa hasil kecepatan eksekusi cloudlet adalah 1000 ms.

```
<terminated> CloudSimExample2 [Java Application] D:\Kuliah\Cloud SIM\eclipse-java-2022-06-R-win32-x86_64\
1000.1: Broker: Cloudlet 0 received
1000.1: Broker: Cloudlet 1 received
1000.1: Broker: All Cloudlets executed. Finishing...
1000.1: Broker: Destroying VM #0
1000.1: Broker: Destroying VM #1
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time
    0         SUCCESS     2             0       1000     0.1
    1         SUCCESS     2             1       1000     0.1
CloudSimExample2 finished!
```

Gambar 3. Hasil eksekusi eksperimen ke-2

Pada Gambar 4, dilakukan eksekusi 2 *cloudlet* di dua virtual machines dengan spesifikasi yang berbeda, Dimana hasil eksekusi dari vm pertama adalah 80 ms dan vm kedua adalah 160 ms.

```
<terminated> CloudSimExample1 [Java Application] D:\Kuliah\Cloud SIM\eclipse-java-2022-06-R-win32-x86_64\
80.1: Broker: Cloudlet 1 received
160.1: Broker: Cloudlet 0 received
160.1: Broker: All Cloudlets executed. Finishing...
160.1: Broker: Destroying VM #0
160.1: Broker: Destroying VM #1
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time
    1         SUCCESS     2             1        80     0.1
    0         SUCCESS     2             0       160     0.1
CloudSimExample3 finished!
```

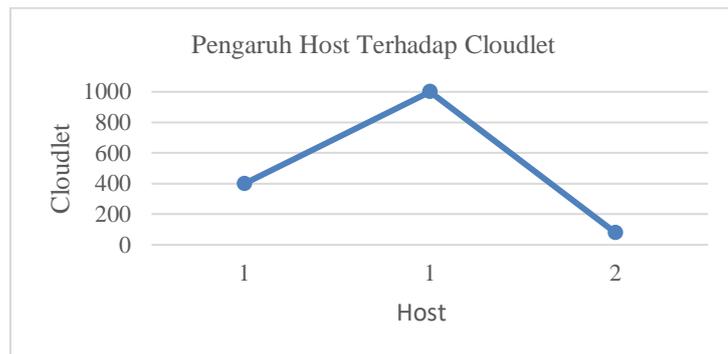
Gambar 4. Hasil eksekusi eksperimen ke-3

### ***Pengaruh Jumlah Host Terhadap Kecepatan Cloudlet***

Berdasarkan hasil program tersebut, dapat dikatakan bahwa jumlah host mempengaruhi kecepatan eksekusi *cloudlet*. Perhatikan gambar 5 berikut:

Perlu diingat bahwa program 1 terdiri dari 1 host dan 1 *cloudlet*, program kedua terdiri dari 1 host dan 2 *cloudlet* dan program 3 terdiri dari 2 host dan 2 *cloudlet*. Berdasarkan gambar 5, dapat diketahui bahwa jika jumlah host semakin banyak, maka kecepatan eksekusi *cloudlet*

juga semakin cepat. Hal ini disebabkan karena dengan semakin banyak host akan memudahkan dalam melakukan scheduling dan dengan dengan penambahan jumlah host, terjadi peningkatan potensi paralelisme (Wang, 2010).

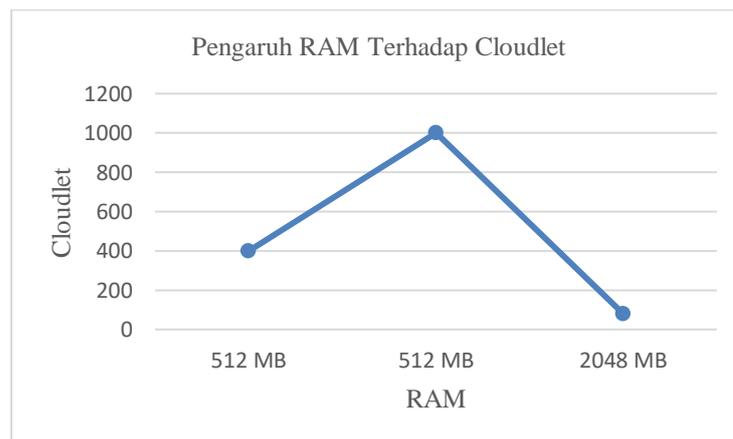


**Gambar 5.** Pengaruh Jumlah Host Terhadap *Cloudlet* dalam MIPS (*Million Instruction per second*)

*Cloudlet* dapat dijadwalkan pada host yang berbeda secara bersamaan, yang dapat mempercepat waktu eksekusi secara signifikan. Hal ini dapat dilihat pada kasus eksperimen ke-2 yang mana memiliki 1 host namun menjalankan 2 cloudlet sehingga kecepatannya lebih lambat dibandingkan dengan eksperimen ke-1 yang menjalankan 1 *cloudlet* dengan 1 host.

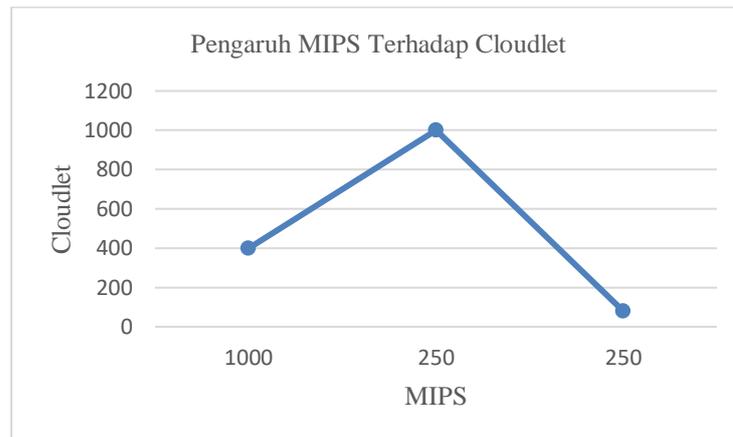
#### ***Pengaruh Spesifikasi Virtual Machines***

Parameter yang paling berpengaruh dalam waktu kecepatan eksekusi virtual machines adalah RAM dan kecepatan mips. Pertama mari bahas tentang RAM, perhatikan gambar 6 berikut:



**Gambar 6.** Pengaruh RAM Terhadap *Cloudlet*

Berdasarkan gambar 6, dapat dilihat bahwa yang memiliki waktu eksekusi paling cepat adalah eksperimen ke-3 dengan ram 2048. Oleh karena itu dapat disimpulkan bahwa semakin besar RAM yang dimiliki oleh suatu virtual machines, maka semakin cepat pula waktu eksekusi dari cloudlet. Perlu diingat untuk kasus eksperimen ke-2 yang memiliki kapasitas yang sama dengan program 1, namun kecepatan eksekusinya lebih lama dari program 1. Hal ini disebabkan karena eksperimen ke-2 menjalankan 2 cloudlet dengan 1 host saja sementara program 1 hanya menjalankan 1 cloudlet saja. Itulah yang menyebabkan mengapa program 2 waktu eksekusinya lebih lambat. Kemudian selanjutnya adalah pengaruh mips terhadap kecepatan eksekusi *cloudlet*. Perhatikan gambar 7 berikut:



**Gambar 7.** Pengaruh MIPS Terhadap Cloudlet

Pada eksperimen ke-1 memiliki mips 1000, eksperimen ke-2 dan ke-3 memiliki mips 250. Berdasarkan gambar 7 dapat dilihat bahwa program yang memiliki waktu eksekusi *cloudlet* paling cepat adalah pada eksperimen ke-3. Jadi, semakin rendah nilai mips maka semakin cepat waktu eksekusi *cloudlet* berlangsung. Namun pada kasus eksperimen ke-2 yang memiliki mips yang sama dengan eksperimen ke-3 lebih lambat dari eksperimen ke-3 dikarenakan jumlah host pada program 2 hanya satu dan menjalankan 2 *cloudlet*, sedangkan pada eksperimen ke-3 memiliki 2 host dan memiliki RAM yang lebih unggul.

#### 4. Kesimpulan

Berdasarkan percobaan yang telah dilakukan, dapat disimpulkan bahwa terdapat banyak perhitungan dan faktor yang mempengaruhi waktu eksekusi *cloudlet*. Faktor pertama adalah jumlah host, di mana semakin banyak jumlah host, kecepatan eksekusi *cloudlet* juga semakin cepat. Dengan adanya banyak host, tingkat efisiensi kerja pun meningkat. Selanjutnya, dalam hal spesifikasi mesin virtual, pengaruh terbesar pada kecepatan eksekusi *cloudlet* adalah RAM dan MIPS. Semakin besar RAM yang dimiliki, waktu eksekusi *cloudlet* cenderung semakin cepat. Namun, untuk MIPS, semakin kecil nilai MIPS, kecepatan eksekusi *cloudlet* juga meningkat. Selain itu, jumlah *cloudlet* yang dieksekusi juga dapat mempengaruhi kecepatan eksekusi *cloudlet*. Untuk menghindari kasus seperti pada eksperimen ke-2, perlu dilakukan perhitungan yang cermat dalam menentukan jumlah host dan spesifikasi mesin virtual agar kinerja yang diperoleh optimal.

#### Referensi

- Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 25(6), 599-616.
- Beloglazov, A., & Buyya, R. 2010. Energy efficient resource management in virtualized cloud data centers. In 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (pp. 826-831). IEEE.
- Foster, I., Zhao, Y., Raicu, I., & Lu, S. 2008. Cloud computing and grid computing 360-degree compared. In *Grid Computing Environments Workshop*, 2008. GCE'08 (pp. 1-10). IEEE.
- Calheiros, R.N., Ranjan, R., Beloglazov, A. De Rose, C.A.F., Buyya, R. 2011. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithm, *Softw. Pract. Exp.* 41 (1), 23–50.
- Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. 2011. A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1), 50-55.

- Wang, L., Tao, J., Kunze, M., Castellanos, A. C., Kramer, D., Karl, W., & Ranjan, R. 2010. Scientific cloud computing: Early definition and experience. In 2010 IEEE *Second International Conference on Cloud Computing Technology and Science* (pp. 97-104). IEEE.
- Wirayudha, M. F., Satriadi, A., & Wiriasto, G. W. 2024. Performance Analysis of Cloud Computing Simulator CloudSim with Virtual Machine Configuration Changes. *JEITECH (Journal of Electrical Engineering and Information Technology)*, 2(1).