# CS 441 HW - 1

## ### Description: Simulation of network datacenters with a particular configuration and varying of policies. Also using the observations to try simulate 3 DC cluster with different cloud models

Name: Santhanagopalan Krishnamoorthy UIN: 675432388 --- ### Objective

1. Explore the different components of Cloud simulation.
2. Develop 3 datacenters with different configurations.
3. Run different services - SaaS, IaaS, PaaS, FaaS on these datacenters.
4. Document the output you obtain through when you run these services and reason out the measurements obtained in these simulations.

### Instructions

##### Development Environment

* **OS:** MacOS
* **IDE** IntelliJ IDEA
* **Java Version** 16.0.2
* **Scala Version** 3.0.2

##### Running the application

* Clone the project
* On the console

sbt clean compile test

sbt clean compile run

#### Details

Two types of Services are discussed in this simulations: SaaS and IaaS.

SaaS service uses datacenters with different autoscaling/no scaling feature to demonstrate the effect of scaling on the performance. IaaS service was defined to evaluate the impact of VmAllocation policy on the cloudlet execution.

1. SaaSDataCenter[1 - 3] - Three different datacenters created and hosted as a SaaS for applications to be executed. The three configurations are No Scaling, Horizontal autoscaling, Vertical autoscaling.
2. SaaSApplication - This objects acts like a broker; It recieves the type of data center to be run on, and executes the cloudlet on it.
3. IaaSSimulation1 - This is a custom datacenter designed by the user which has a specific VmAllocationPolicy that Migrates/Allocates VM based on custom Host Utilisation percentage.
4. IaaSSimulation2 - This is also a datacenter designed by the user but doesn't have any custom VmAllocationPolicy but

#### Basic Program Flow

1. The main driver program is Simulation.Scala in the root directory. This then calls all the 5 simulations via SaaSApplication and IaaSSimulation classes and logs the results. Each of these simulations use their own respective configuration files namely, datacenter1.conf , datacenter2.conf , datacenter3.conf, iaasconfig.conf
2. The SaaSApplication and IaaSSimulation classes use utils package to create DataCenters, Vms, Cloudlets etc.
3. The Utils classes/objects consists of the following:
   1. SaaSDataCenter - Super class for all the SaaS data centers and generate the common configuration.
   2. SaaSDataCenter(X) - Class that contains data center specific configuration and components.
   3. ClouldetHelper - Generates Clouldets(both at the start and dynamically)
   4. Measurements - Calculates the cost for running the cloudlets

#### Assumptions made with regards to Provider and Consumer

|  |  |  |
| --- | --- | --- |
| Model | Provider | Consumer |
| SaaS | The Provider has total control of the VMs, Hosts, DataCenter Characteristics(hardware), Cloudlet CHaractersitics | Consumer can only specify the number of Cloudlets (Which simulates him connecting to get the service) |
| PaaS | The Provider has control over the VMs, Hosts, Datacenter Characteristics(hardware) | Consumer can specify the Cloudlet characteristics (Data) and Number of VMs (Middleware to some extent) |
| IaaS | The Provider has control over only the hardware | Consumer has control over the VMs, Cloudlets |

These various levels of abstractions have been simulated in these examples by assuming that:

1. The three SaaS datacenters' configs were defined by the Cloud Service providers and only the Cloudlet information(Applications to be executed) were provided by the user through a config file - SmallApplicationConfig.conf
2. The full control of the IaaS platforms are given to the user and is configured using iaasconfig.conf

## SaaS Implementation

Same clouldet load is given to three datacenters with scaling implemented to measure the effect in terms of Execution time and Cost.

### Cloudet Information

* Length = 11 (6 - At the start, 5 - During the simulation)
* Pes = 2
* MIs = 2000000
* fileSize = 300
* outputSize = 300
* UtilizationModeCpu = Full

## Configuration 1 - No scaling

### DataCenter

* Hostnumber = 5

### Host

* PEs = 4
* mipsCapacity = 20000
* RAMInMBs = 500000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### VM

* vmNumbers = 3
* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* BandwidthInMBps = 1000

### Cost

* costPerSecond = 0.01
* costPerMem = 0.02
* costPerStorage = 0.001
* costPerBw = 0.005

#### Results

Below are some of the results

SIMULATION RESULTS  
  
Cloudlet|Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPEs|StartTime|FinishTime|ExecTime  
 ID| |ID| ID|CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds  
-----------------------------------------------------------------------------------------------------  
 0|SUCCESS| 2| 0| 4| 0| 2| 2000000| 2| 0| 7900| 7900  
 1|SUCCESS| 2| 0| 4| 1| 2| 2000000| 2| 0| 7900| 7900  
 2|SUCCESS| 2| 0| 4| 2| 2| 2000000| 2| 0| 7900| 7900  
 3|SUCCESS| 2| 0| 4| 0| 2| 2000000| 2| 0| 7900| 7900  
 4|SUCCESS| 2| 0| 4| 1| 2| 2000000| 2| 0| 7900| 7900  
 5|SUCCESS| 2| 0| 4| 2| 2| 2000000| 2| 0| 7900| 7900  
 6|SUCCESS| 2| 0| 4| 0| 2| 2000000| 2| 101| 8001| 7899  
 7|SUCCESS| 2| 0| 4| 1| 2| 2000000| 2| 101| 8001| 7899  
 8|SUCCESS| 2| 0| 4| 2| 2| 2000000| 2| 101| 8001| 7899  
 9|SUCCESS| 2| 0| 4| 0| 2| 2000000| 2| 102| 8001| 7899  
 10|SUCCESS| 2| 0| 4| 1| 2| 2000000| 2| 102| 8001| 7899  
 11|SUCCESS| 2| 0| 4| 2| 2| 2000000| 2| 102| 8001| 7899  
-----------------------------------------------------------------------------------------------------  
  
The total cost is 642.074796

### Observation

* Since there is no scaling in this simulation and the cloudlet scheduler is TimeShared, the time taken to execute is sequential. Thus, the total time of exection of the set of clouldlets is 8001 seconds. However, since there is no scaling implemented, the cost incurred is $642.07

## Configuration 2 - Auto Scaling - Horizontal

### DataCenter

* Hostnumber = 5

### Host

* PEs = 20 (Higher than normal because Host should have enough PEs to adapt when VM horizontally scales when overloaded; No of VMs are same as sim #1)
* mipsCapacity = 20000
* RAMInMBs = 500000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### VM

* vmNumbers = 3
* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* BandwidthInMBps = 1000

### Cost

* costPerSecond = 0.20
* costPerMem = 0.02
* costPerStorage = 0.001
* costPerBw = 0.005

#### Results

Below are some of the results

SIMULATION RESULTS  
  
Cloudlet|Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPEs|StartTime|FinishTime|ExecTime  
 ID| |ID| ID|CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds  
-----------------------------------------------------------------------------------------------------  
 0|SUCCESS| 2| 0| 20| 0| 2| 2000000| 2| 0| 2000| 2000  
 1|SUCCESS| 2| 0| 20| 1| 2| 2000000| 2| 0| 2000| 2000  
 2|SUCCESS| 2| 0| 20| 2| 2| 2000000| 2| 0| 2000| 2000  
 9|SUCCESS| 2| 0| 20| 3| 2| 2000000| 2| 101| 2101| 2000  
 10|SUCCESS| 2| 0| 20| 4| 2| 2000000| 2| 101| 2101| 2000  
 11|SUCCESS| 2| 0| 20| 5| 2| 2000000| 2| 101| 2101| 2000  
 3|SUCCESS| 2| 0| 20| 0| 2| 2000000| 2| 2000| 4000| 2000  
 4|SUCCESS| 2| 0| 20| 1| 2| 2000000| 2| 2000| 4000| 2000  
 5|SUCCESS| 2| 0| 20| 2| 2| 2000000| 2| 2000| 4000| 2000  
 6|SUCCESS| 2| 0| 20| 0| 2| 2000000| 2| 4001| 6000| 2000  
 7|SUCCESS| 2| 0| 20| 1| 2| 2000000| 2| 4001| 6000| 2000  
 8|SUCCESS| 2| 0| 20| 2| 2| 2000000| 2| 4001| 6000| 2000  
-----------------------------------------------------------------------------------------------------  
   
 The total cost is 708.1146

### Observation

* Since there is Horizontal Scaling in this simulation, VMs are increased (In this case, the number of VMs increased to 6 due to heavy load). Thus, the total time of exection of the set of clouldlets is 6000 seconds (Net difference is 2000 seconds) . However, since there is horizontal scaling implemented, the cost incurred is $708.11

## Configuration 3 - Auto Scaling - Horizontal

### DataCenter

* Hostnumber = 5

### Host

* PEs = 4
* mipsCapacity = 20000
* RAMInMBs = 500000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### VM

* vmNumbers = 3
* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* BandwidthInMBps = 1000

### Cost

* costPerSecond = 0.30
* costPerMem = 0.02
* costPerStorage = 0.001
* costPerBw = 0.005

#### Results

Below are some of the results

SIMULATION RESULTS  
  
Cloudlet|Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPEs|StartTime|FinishTime|ExecTime  
 ID| |ID| ID|CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds  
-----------------------------------------------------------------------------------------------------  
 0|SUCCESS| 2| 0| 4| 1| 3| 2000000| 2| 0| 5234| 5234  
 1|SUCCESS| 2| 0| 4| 2| 3| 2000000| 2| 0| 5234| 5234  
 2|SUCCESS| 2| 0| 4| 3| 3| 2000000| 2| 0| 5234| 5234  
 3|SUCCESS| 2| 0| 4| 1| 3| 2000000| 2| 0| 5234| 5234  
 4|SUCCESS| 2| 0| 4| 2| 3| 2000000| 2| 0| 5234| 5234  
 5|SUCCESS| 2| 0| 4| 3| 3| 2000000| 2| 0| 5234| 5234  
 6|SUCCESS| 2| 0| 4| 1| 3| 2000000| 2| 101| 5334| 5233  
 7|SUCCESS| 2| 0| 4| 2| 3| 2000000| 2| 101| 5334| 5233  
 8|SUCCESS| 2| 0| 4| 3| 3| 2000000| 2| 101| 5334| 5233  
 9|SUCCESS| 2| 0| 4| 1| 3| 2000000| 2| 102| 5335| 5232  
 10|SUCCESS| 2| 0| 4| 2| 3| 2000000| 2| 102| 5335| 5232  
 11|SUCCESS| 2| 0| 4| 3| 3| 2000000| 2| 102| 5335| 5232  
-----------------------------------------------------------------------------------------------------  
   
 The total cost is 786.200144

### Observation

* Since there is Vertical Scaling in this simulation, No of PE cores are increased in the VMs, thus, managing to reduce the load of the cloudlets. Thus, the total time of exection of the set of clouldlets is 5335 seconds (Net difference is 2665 seconds). However, since there is vertical scaling implemented, the cost incurred is $786.20

## Inference

* These simulations demonstrate the execution of same load of cloudlets on different data centers with Auto Scaling implemented. The effect is measured by reduction of execution time for cloudlets and in turn, the cost incurred for these varied services.

## IaaS Implementation

Since this is IaaS, we have control over how VMs are scheduled on the hosts and we can leverage this for optimised performances. Same cloudlets are implemented to two datacenters where one datacenter implements a different VMAllocationPolicy in host which allocates based on the utilisation of the VMs.

### Cloudet Information

* Length = 4
* Pes = 2
* MIs = 2000000
* fileSize = 300
* outputSize = 300
* UtilizationModeCpu = Full

## Configuration 1 - VmAllocationPolicyMigrationBestFitStaticThreshold

### DataCenter

* Hostnumber = 3
* vmNumbers = 4
* HOST\_UNDER\_UTILIZATION\_THRESHOLD\_FOR\_VM\_MIGRATION = 0.1
* HOST\_OVER\_UTILIZATION\_THRESHOLD\_FOR\_VM\_MIGRATION = 0.7
* HOST\_SEARCH\_RETRY\_DELAY = 60

### Host 1

* PEs = 4
* mipsCapacity = 1000
* RAMInMBs = 10000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### Host 2

* PEs = 5
* mipsCapacity = 1000
* RAMInMBs = 500000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### Host 3

* PEs = 5
* mipsCapacity = 1000
* RAMInMBs = 25000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### VM 1

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 2

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 3

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 4

* mipsCapacity = 1000
* PEs = 1
* RAMInMBs = 10000
* Size = 1000

### Cost

* costPerSecond = 0.01
* costPerMem = 0.02
* costPerStorage = 0.001
* costPerBw = 0.005

#### Results

Below are some of the results

SIMULATION RESULTS  
  
Cloudlet|Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPEs|StartTime|FinishTime|ExecTime  
 ID| |ID| ID|CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds  
-----------------------------------------------------------------------------------------------------  
 0|SUCCESS| 2| 0| 4| 0| 2| 20000| 2| 0| 20| 20  
 1|SUCCESS| 2| 1| 5| 1| 2| 20000| 2| 0| 20| 20  
 2|SUCCESS| 2| 1| 5| 2| 2| 20000| 2| 0| 20| 20  
 3|SUCCESS| 2| 2| 5| 3| 1| 20000| 1| 0| 20| 20  
-----------------------------------------------------------------------------------------------------

### Observation

* In this case, after VM2 is allocated to HostID 1, VM3 is allocated to HostID 2 instead of 1, even though Host ID 1 has 1 PE remaining which is VM3's requirement. This is because Host ID1 surpasses the threshold when VMs 1 and 2 are allocated. Thus, VM ID 3 is allocated to Host ID 2.

## Configuration 2 - VmAllocationFirstFit

### DataCenter

* Hostnumber = 3
* vmNumbers = 4

### Host 1

* PEs = 4
* mipsCapacity = 1000
* RAMInMBs = 10000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### Host 2

* PEs = 5
* mipsCapacity = 1000
* RAMInMBs = 500000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### Host 3

* PEs = 5
* mipsCapacity = 1000
* RAMInMBs = 25000
* StorageInMBs = 1000000
* BandwidthInMBps = 16000

### VM 1

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 2

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 3

* mipsCapacity = 1000
* PEs = 2
* RAMInMBs = 10000
* Size = 1000

### VM 4

* mipsCapacity = 1000
* PEs = 1
* RAMInMBs = 10000
* Size = 1000

### Cost

* costPerSecond = 0.01
* costPerMem = 0.02
* costPerStorage = 0.001
* costPerBw = 0.005

#### Results

Below are some of the results

SIMULATION RESULTS  
  
Cloudlet|Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPEs|StartTime|FinishTime|ExecTime  
 ID| |ID| ID|CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds  
-----------------------------------------------------------------------------------------------------  
 0|SUCCESS| 2| 0| 4| 0| 2| 20000| 2| 0| 20| 20  
 1|SUCCESS| 2| 1| 5| 1| 2| 20000| 2| 0| 20| 20  
 2|SUCCESS| 2| 1| 5| 2| 2| 20000| 2| 0| 20| 20  
 3|SUCCESS| 2| 1| 5| 3| 1| 20000| 1| 0| 20| 20  
-----------------------------------------------------------------------------------------------------

### Observation

* In this case, after VM2 is allocated to HostID 1, VM3 is allocated to Host ID 1 because Host ID 1 still has capacity for one more PE (Unlike Configuration #1 where it was allocated to Host ID 1)

## Inference

Through these simulations, we can understand that VmMigrationPolicyOnThreshold can be useful when Host load increases beyond a limit and if not implemented, the host might crash/Under perform.

#### Discussion

1. From SaaS service, we observed the impact of scaling on the performance of the cloudlets and the VMs
2. From the IaaS service, we observed the VmAllocationPolicy impact on the host performance,

#### Future Improvements

* Understanding the impact of using both Horizontal scaling and Vertical scaling, and further understanding when it is optimal to implement them.
* IaaS services offer high customisation. Thus, many other factors can be looked at to optimise cloud computing, like Networking Hosts, VmSchedulers, CloudletSchedulers.
* Combining various factors to get an even more depth understanding of cloud architecture.