

Wajahat Khan
CS-441
PROF.-Mark Grechanik
TA- Abhijeet Mohanty

IDE: IntelliJ
Gradle: 6.0.1

To run this simulation you first have to clone the repo and after cloning add the gradle dependencies for cloudsims.

To run the simulation from IntelliJ open the src/main/java/hw I have a class called hw1441.java where I implemented all the things. So, running that file (I used jdk 11) will start the simulation.

Also, If you want to run from command line you can use commands like "Gradle build test" and to run the simulation use "Gradle build run".

In the same folder you will find typesafe config files with all the needed constants. So, I have two .conf file one of them contains all the data.

Also, with .conf file we created respective .java file to access the data. I also created one abstract class to create the instance of different policies for instance VmAllocationPolicy.

Unit test our test folder src/test I only wrote 1 test class with 5 test cases init which are simply checking the creation of vms and cloudlets. Mapper and reducer are being used in this simulation and their allocation is based on the ratio of 1:1 so 1 reducer for 1 mapper although we can change it to anyway but we have

to make sure that number of mappers are greater you can find the allocation function in hw1441.java class under the name Mappers and reducer allocation().

VmAllocationPolicyBestFit and VmAllocationPolicySimple is used with both CloudletScheduler (TimeShared and Space Shared).

Due to less number of hosts in my DC some cloudlets won't be assigned task. I left it intentionally to see how the cloudlet manage load.

Fourth Part

Here is the output of my two Datacenters both using different configuration such as VmAllocation policy and Different UtilizationModel (full and stochastic)

```
INFO CloudInformationService0: Notify all CloudSim Plus entities to shutdown.
INFO 2748.13: NetworkDatacenter1 is shutting down...
INFO 2748.13: NetworkDatacenter2 is shutting down...
INFO
===== Simulation finished at time 2748.13 =====

INFO Total Cost of 0 cloudlets: 38.022000000000006ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 1 cloudlets: 38.022000000000006ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 2 cloudlets: 38.042ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 3 cloudlets: 38.042ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 4 cloudlets: 56.011ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 5 cloudlets: 56.011ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 6 cloudlets: 56.011ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 7 cloudlets: 56.011ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 8 cloudlets: 38.044000000000004ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 9 cloudlets: 38.044000000000004ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 10 cloudlets: 38.044000000000004ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 11 cloudlets: 38.044000000000004ACTUAL RAM UTILIZATION: 1.0
INFO Total Cost of 12 cloudlets: 65.47754248583739ACTUAL RAM UTILIZATION: 0.7164539004694559
INFO Total Cost of 13 cloudlets: 84.97899769071037ACTUAL RAM UTILIZATION: 0.4532150718274175
INFO Total Cost of 14 cloudlets: 41.54056250534353ACTUAL RAM UTILIZATION: 0.5505799811756450
INFO Total Cost of 15 cloudlets: 341.5146300336651ACTUAL RAM UTILIZATION: 0.0991161014225328

SIMULATION RESULTS

Cloudlet[Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPES|StartTime|FinishTime|ExecTime
ID| ID| CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds
-----
1|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 22| 101| 159
0|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 22| 101| 159

SIMULATION RESULTS

Cloudlet[Status |DC|Host|Host PEs |VM|VM PEs |CloudletLen|CloudletPES|StartTime|FinishTime|ExecTime
ID| ID| CPU cores|ID|CPU cores| MI| CPU cores| Seconds| Seconds| Seconds
-----
1|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 22| 101| 159
0|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 22| 101| 159
3|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 182| 342| 160
2|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 182| 342| 160
4|SUCCESS| 2| 0| 2| 2| 1| 40000| 1| 23| 343| 320
10|SUCCESS| 2| 0| 2| 2| 1| 40000| 1| 23| 343| 320
5|SUCCESS| 2| 1| 2| 3| 1| 40000| 1| 23| 343| 320
11|SUCCESS| 2| 1| 2| 3| 1| 40000| 1| 23| 343| 320
7|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 342| 502| 160
6|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 342| 502| 160
9|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 502| 662| 160
8|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 502| 662| 160
13|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 663| 959| 297
12|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 663| 1057| 394
15|SUCCESS| 1| 0| 2| 1| 1| 40000| 1| 940| 1130| 178
14|SUCCESS| 1| 1| 2| 0| 1| 40000| 1| 1050| 2735| 1677

hw1441 finished!

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.3/userguide/command_line_interface.html#sec:command_line_warnings

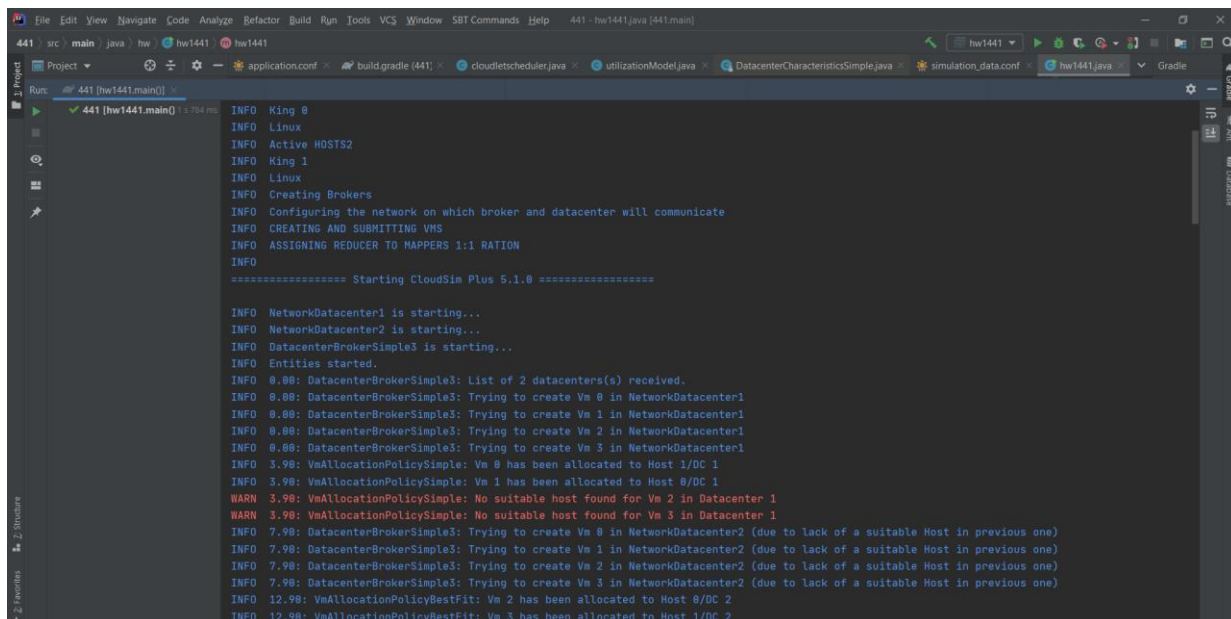
BUILD SUCCESSFUL in 1s
3 actionable tasks: 2 executed, 1 up-to-date
1:12:28 AM: Task execution finished 'hw1441.main()'.
```

Basically we are using BestFit and Simple Vmallocation Policy to assign our VMs to host and to assign cloudlets to VMs I'm using cloudletSpaceShared policy for datacenter 1 and CloudletTimeShared for dataCenter 2. Here we have 4 hosts and 1 broker. We are using mapper and reducer approach to split the work, we are implementing it using 1:1 ration so 1 reducer for each mapper. If we notice cloudlet 12-15 have high cost one of the main reasons here is that we are using different CloudScheduler policies and UtilizationFullModel. So, we calculate cost of a cloudlet based on the length and their completion of time. There are other few factor as well. However, in our situation CloudschedulerSpaceShared and UtilizationStochastic policy turn out to be

more efficient as it gives us more cost effective cloudlets. Also, we can always test it by executing more tasks.

Explain part 5:

Broadcast storming shows us the bouncing of data between two datacenters. Here we are using topology.brite file from cloudsim examples and we assigned briteNode=0 and briteNode=3 to datacenter to communicate and broker is on node = 2.



```
INFO King 0
INFO Linux
INFO Active HOSTS2
INFO King 1
INFO Linux
INFO Creating Brokers
INFO Configuring the network on which broker and datacenter will communicate
INFO CREATING AND SUBMITTING VMS
INFO ASSIGNING REDUCER TO MAPERS 1:1 RATION
INFO
===== Starting CloudSim Plus 5.1.0 =====
INFO NetworkDatacenter1 is starting...
INFO NetworkDatacenter2 is starting...
INFO DatacenterBrokerSimple3 is starting...
INFO Entities started.
INFO 0.00: DatacenterBrokerSimple3: List of 2 datacenters(s) received.
INFO 0.00: DatacenterBrokerSimple3: Trying to create Vm 0 in NetworkDatacenter1
INFO 0.00: DatacenterBrokerSimple3: Trying to create Vm 1 in NetworkDatacenter1
INFO 0.00: DatacenterBrokerSimple3: Trying to create Vm 2 in NetworkDatacenter1
INFO 0.00: DatacenterBrokerSimple3: Trying to create Vm 3 in NetworkDatacenter1
INFO 3.98: VmAllocationPolicySimple: Vm 0 has been allocated to Host 1/DC 1
INFO 3.98: VmAllocationPolicySimple: Vm 1 has been allocated to Host 0/DC 1
WARN 3.98: VmAllocationPolicySimple: No suitable host found for Vm 2 in Datacenter 1
WARN 3.98: VmAllocationPolicySimple: No suitable host found for Vm 3 in Datacenter 1
INFO 7.98: DatacenterBrokerSimple3: Trying to create Vm 0 in NetworkDatacenter2 (due to lack of a suitable Host in previous one)
INFO 7.98: DatacenterBrokerSimple3: Trying to create Vm 1 in NetworkDatacenter2 (due to lack of a suitable Host in previous one)
INFO 7.98: DatacenterBrokerSimple3: Trying to create Vm 2 in NetworkDatacenter2 (due to lack of a suitable Host in previous one)
INFO 7.98: DatacenterBrokerSimple3: Trying to create Vm 3 in NetworkDatacenter2 (due to lack of a suitable Host in previous one)
INFO 12.98: VmAllocationPolicyBestFit: Vm 2 has been allocated to Host 0/DC 2
INFO 12.98: VmAllocationPolicyBestFit: Vm 3 has been allocated to Host 1/DC 2
```

So, as we can see in our simulation datacenters are send vms to different datacenters because of no suitable hosts in them. This is one of the example of broadcast storm where our datacenter send Vms and the cloudlets attach to them to the next available dataCenter. In my simulation one of the reason to keep number of host less is to show the idea of broadcast storm. So, basically we have less hosts and more cloudlets this way the datacenter have no choice other than communicating over the nodes to each other and sending tasks or to be specific VMs and Cloudlets