

# Simulation and Results

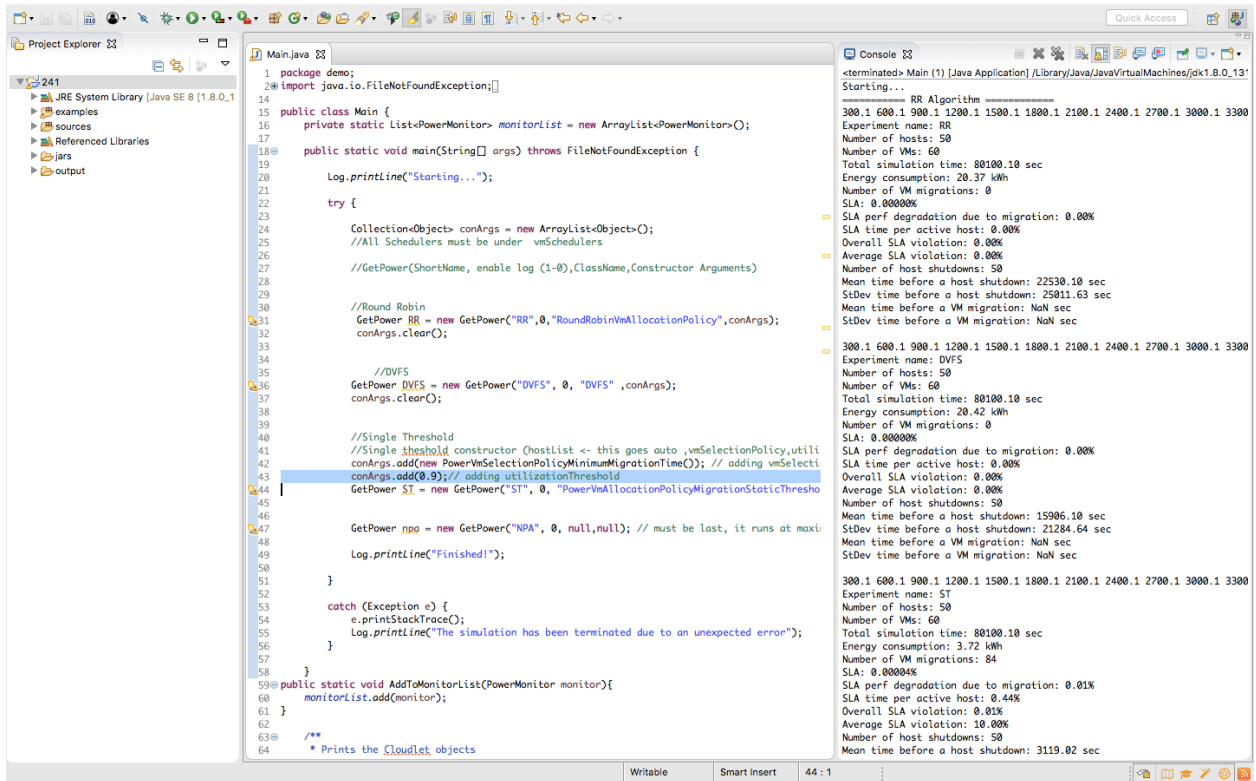
## 1. Run 60 VMs, get the energy consumption from each allocation policy.

The screenshot displays an IDE with a Java project named 'demo'. The 'Main.java' file is open, showing a class with a 'main' method that runs a simulation. The code includes imports for 'java.io.FileNotFoundException' and 'java.util.ArrayList'. It defines a 'Main' class with a 'main' method that takes 'String[] args' and throws 'FileNotFoundException'. The 'main' method contains a 'try' block with several 'GetPower' calls for different allocation policies: 'RR', 'DVFS', 'Single Threshold', and 'NPA'. Each call is followed by a 'Log.println' statement. The 'catch' block handles 'Exception e' and prints the stack trace. The 'AddToMonitorList' method is also defined, which adds a 'PowerMonitor' to the 'monitorList'.

The 'Console' window shows the output of the simulation. It displays the results for three different allocation policies: 'RR', 'DVFS', and 'ST'. Each policy's results are summarized in a table with columns for 'Experiment name', 'Number of hosts', 'Number of VMs', 'Total simulation time', 'Energy consumption', 'Number of VM migrations', 'SLA', 'SLA perf degradation due to migration', 'SLA time per active host', 'Overall SLA violation', 'Average SLA violation', 'Number of host shutdowns', 'Mean time before a host shutdown', 'StDev time before a host shutdown', 'Mean time before a VM migration', and 'StDev time before a VM migration'.

Experiment name	Number of hosts	Number of VMs	Total simulation time	Energy consumption	Number of VM migrations	SLA	SLA perf degradation due to migration	SLA time per active host	Overall SLA violation	Average SLA violation	Number of host shutdowns	Mean time before a host shutdown	StDev time before a host shutdown	Mean time before a VM migration	StDev time before a VM migration
RR	50	60	80100.10 sec	20.37 kWh	0	0.00000%	0.00%	0.00%	0.00%	0.00%	50	22530.10 sec	25011.63 sec	NaN sec	NaN sec
DVFS	50	60	80100.10 sec	20.42 kWh	0	0.00000%	0.00%	0.00%	0.00%	0.00%	50	15906.10 sec	21284.64 sec	NaN sec	NaN sec
ST	50	60	80100.10 sec	3.87 kWh	82	0.00003%	0.01%	0.37%	0.01%	10.00%	50	3460.85 sec			

2. Change the utilization of STH from 0.5 to 0.9, get the different energy consumption/SLA violation/VM migration number for each allocation policy.



The screenshot shows an IDE with a Java file named `Main.java` and a console window. The code defines a `main` method that runs a simulation for different allocation policies: `RR`, `DVFS`, `Single Threshold`, and `ST`. The `ST` policy is highlighted in the code, showing a utilization threshold of 0.9. The console output displays simulation results for each policy, including metrics like total simulation time, energy consumption, SLA violations, and VM migrations.

```
1 package demo;
2 import java.io.FileNotFoundException;
3
14 public class Main {
15     private static List<PowerMonitor> monitorList = new ArrayList<PowerMonitor>();
16
17     public static void main(String[] args) throws FileNotFoundException {
18         Log.println("Starting...");
19
20         try {
21             Collection<Object> conArgs = new ArrayList<Object>();
22             //All Schedulers must be under vmSchedulers
23             //GetPower(ShortName, enable log (1-0),ClassName,Constructor Arguments)
24
25             //Round Robin
26             GetPower RR = new GetPower("RR", 0, "RoundRobinVmAllocationPolicy", conArgs);
27             conArgs.clear();
28
29             //DVFS
30             GetPower DVFS = new GetPower("DVFS", 0, "DVFS", conArgs);
31             conArgs.clear();
32
33             //Single Threshold
34             //Single threshold constructor (hostList <- this goes auto ,vmSelectionPolicy,utili
35             conArgs.add(new PowerVmSelectionPolicyMinimumMigrationTime()); // adding vmSelecti
36             conArgs.add(0.9); // adding utilizationThreshold
37             GetPower ST = new GetPower("ST", 0, "PowerVmAllocationPolicyMigrationStaticThresho
38
39             GetPower npa = new GetPower("NPA", 0, null, null); // must be last, it runs at maxi
40
41             Log.println("Finished!");
42
43         } catch (Exception e) {
44             e.printStackTrace();
45             Log.println("The simulation has been terminated due to an unexpected error");
46         }
47     }
48
49     public static void AddToMonitorList(PowerMonitor monitor){
50         monitorList.add(monitor);
51     }
52
53     /**
54      * Prints the Cloudlet objects
55      */
56 }
```

Console Output:

```
<terminated> Main (1) [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_131
Starting...
===== RR Algorithm =====
300.1 600.1 900.1 1200.1 1500.1 1800.1 2100.1 2400.1 2700.1 3000.1 3300
Experiment name: RR
Number of hosts: 50
Number of VMs: 60
Total simulation time: 80100.10 sec
Energy consumption: 20.37 kWh
Number of VM migrations: 0
SLA: 0.000000%
SLA perf degradation due to migration: 0.00%
SLA time per active host: 0.00%
Overall SLA violation: 0.00%
Average SLA violations: 0.00%
Number of host shutdowns: 50
Mean time before a host shutdown: 22530.10 sec
StDev time before a host shutdown: 25011.63 sec
Mean time before a VM migration: NaN sec
StDev time before a VM migration: NaN sec
300.1 600.1 900.1 1200.1 1500.1 1800.1 2100.1 2400.1 2700.1 3000.1 3300
Experiment name: DVFS
Number of hosts: 50
Number of VMs: 60
Total simulation time: 80100.10 sec
Energy consumption: 20.42 kWh
Number of VM migrations: 0
SLA: 0.000000%
SLA perf degradation due to migration: 0.00%
SLA time per active host: 0.00%
Overall SLA violation: 0.00%
Average SLA violations: 0.00%
Number of host shutdowns: 50
Mean time before a host shutdown: 15906.10 sec
StDev time before a host shutdown: 21284.64 sec
Mean time before a VM migration: NaN sec
StDev time before a VM migration: NaN sec
300.1 600.1 900.1 1200.1 1500.1 1800.1 2100.1 2400.1 2700.1 3000.1 3300
Experiment name: ST
Number of hosts: 50
Number of VMs: 60
Total simulation time: 80100.10 sec
Energy consumption: 3.72 kWh
Number of VM migrations: 84
SLA: 0.00004%
SLA perf degradation due to migration: 0.01%
SLA time per active host: 0.44%
Overall SLA violation: 0.01%
Average SLA violations: 10.00%
Number of host shutdowns: 50
Mean time before a host shutdown: 3119.02 sec
```