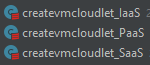
**Flow of the code:**













Main Simulation file Individual Simulations Reading conf. files Policies

There are in total 5 (+1) types of Simulations that can be run.

**Individual Simulations:**

* ScalingSim offers users a As a Service model which is scalable. i.e., if the CPU Utilization of the VMs offered to the user cross a certain threshold, then the VMs scale out (scale horizontally/increase in the number of VMs)
* IaasSim/PaasSim/SaasSim all simulate the different types of as a Service models that cloud offers.
* DatacenterNetworking simulates 3 Datacenters, all offering different as a Service models, and connects them together using the Brite topology.

**Reading conf. files:**

* There are 5 configuration files that can be found under the resources folder.
  + Application.conf : Config file used by the pre-defined example, BasicCloudSimPlusExample.
  + DataCenter.conf: Config file containing all the configurations of the Datacenter.
  + user\_IaaS/ \_PaaS / \_SaaS .conf: Config files used by the different Service modes. (users can modify)
* The VM & Cloudlets config. (user input) are read by the createvmcloudlet\_IaaS/ \_PaaS / \_SaaS classes.
* The createHost function present inside commonutils reads the Host configurations and creates a list of these hosts.
* createDatacenter function creates a Datacenter based on the given type (Simple/Network), and sets VM allocation Policy. The hosts created using createHost function are part of the datacenter.

**Policies:**

* Cost: The cost class is used to find the total cost of a particular execution. The Datacenter characteristics that affect the cost of the execution are, the time duration of the execution, memory, storage and the bandwidth used during the execution.
* scalingutil: Adds horizontal scaling over the created VMs. The scaling takes place once the CPU utilization percentage of a certain VM crosses the threshold set.

**As a service model Implementation:**

**SaaS:** The Service Provider has total control over the VMs, Hosts, DataCenter characteristics and the Cloudlet characteristics. The Consumer can only specify the number of Cloudlets, which connect them to the service.

**PaaS:** The Service Provider has control over the VMs, Hosts and Datacenter Characteristics. The Consumer can specify the Cloudlet characteristics and Number of VMs.

**IaaS:** The Service Provider has control over only the hardware (the hosts in the Datacenter). The Consumer has control over the VMs and Cloudlets specifications.

*Based on the model type, the config files of the respective files and the Datacenter.conf file are read.*

**Testing:**

Scalatest framework is utilized to write some basic test cases.

* testDatacenter - Tests the creation of a Datacenter, given a list of hosts.
* testHost – Tests the creation of a single host.
* testVMAllocationPolicy – Tests if a given policy is a type of VM Allocation Policy or not.