CLOUDSIM

Introduction

- Open-source framework
- Used to simulate cloud computing infrastructure and services
- Developed : CLOUDS Lab organization
- Written : Java

Example

If you were to deploy an application or a website on the cloud and wanted to test the services and load that your product can handle and also tune its performance to overcome bottlenecks before risking deployment, then such evaluations could be performed by simply coding a simulation of that environment with the help of various flexible and scalable classes provided by the CloudSim package, free of cost.

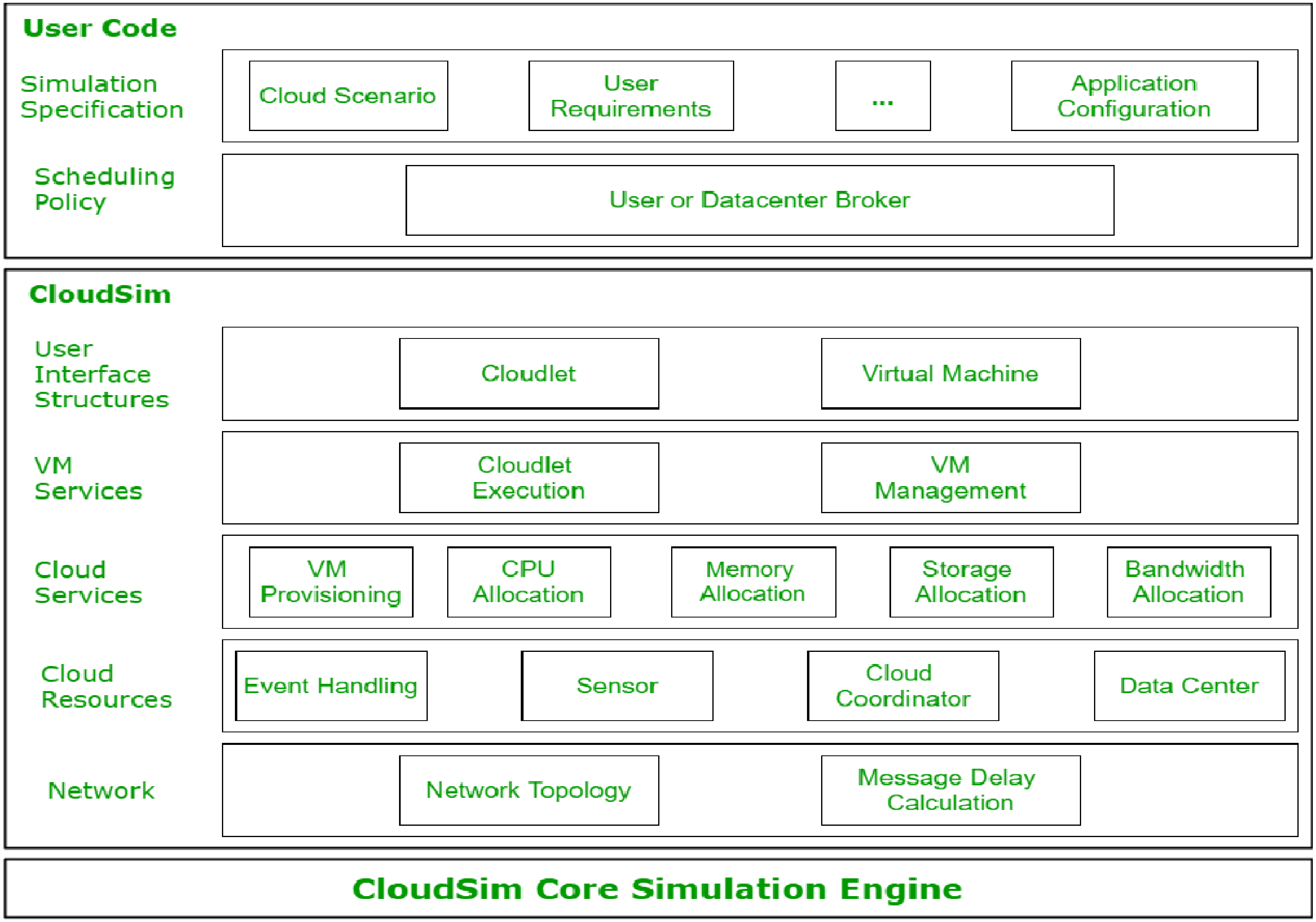
Benefits

- No capital investment involved
- Easy to use and Scalable
- Risks can be evaluated at an earlier stage
- No need for try-and-error approaches

Why use CloudSim?

- Open source and free of cost, so it favors researchers / developers working in the field.
- Easy to download and set-up.
- It is more generalized and extensible to support modelling and experimentation.
- Does not require any high-specs computer to work on.
- Provides pre-defined allocation policies and utilization models for managing resources, and allows implementation of user-defined algorithms as well.
- The documentation provides pre-coded examples for new developers to get familiar with the basic classes and functions.
- Tackle bottlenecks before deployment to reduce risk, lower costs, increase performance, and raise revenue.

CloudSim Architecture



User Code

Simulation Specification: Cloud Scenario | User Requirements | ... | Application Configuration

Scheduling Policy: User or Datacenter Broker

CloudSim

User Interface Structures: Cloudlet | Virtual Machine

VM Services: Cloudlet Execution | VM Management

Cloud Services: VM Provisioning | CPU Allocation | Memory Allocation | Storage Allocation | Bandwidth Allocation

Cloud Resources: Event Handling | Sensor | Cloud Coordinator | Data Center

Network: Network Topology | Message Delay Calculation

CloudSim Core Simulation Engine

CloudSim Architecture

- **CloudSim Core** Simulation Engine provides interfaces for the management of resources such as VM, memory and bandwidth of virtualized Datacenters.

- **CloudSim layer** manages the creation and execution of core entities such as VMs, Cloudlets, Hosts etc. It also handles network-related execution along with the provisioning of resources and their execution and management.

- **User Code** is the layer controlled by the user. The developer can write the requirements of the hardware specifications in this layer according to the scenario.

CloudSim Architecture

- **Datacenter**: used for modelling the foundational hardware equipment of any cloud environment, that is the Datacenter. This class provides methods to specify the functional requirements of the Datacenter as well as methods to set the allocation policies of the VMs etc.

- **Host**: this class executes actions related to management of virtual machines. It also defines policies for provisioning memory and bandwidth to the virtual machines, as well as allocating CPU cores to the virtual machines.

- **VM**: this class represents a virtual machine by providing data members defining a VM’s bandwidth, RAM, mips (million instructions per second), size while also providing setter and getter methods for these parameters.

CloudSim Architecture

- **Cloudlet**: a cloudlet class represents any task that is run on a VM, like a processing task, or a memory access task, or a file updating task etc. It stores parameters defining the characteristics of a task such as its length, size, mi (million instructions) and provides methods similarly to VM class while also providing methods that define a task's execution time, status, cost and history.

- **DatacenterBroker**: is an entity acting on behalf of the user/customer. It is responsible for functioning of VMs, including VM creation, management, destruction and submission of cloudlets to the VM.

- **CloudSim**: this is the class responsible for initializing and starting the simulation environment after all the necessary cloud entities have been defined and later stopping after all the entities have been destroyed.

Features of CloudSim

- Large scale virtualized Datacenters, servers and hosts.
- Customizable policies for provisioning host to virtual machines.
- Energy-aware computational resources.
- Application containers and federated clouds (joining and management of multiple public clouds).
- Datacenter network topologies and message-passing applications.
- Dynamic insertion of simulation entities with stop and resume of simulation.
- User-defined allocation and provisioning policies.

How to run my first cloudsim simulation scenario

- Initialize the CloudSim with the current clock time and this will also initialize the core CloudInformationService entity.
- Create Datacenter(s) as Datacenters are the resource providers in CloudSim. We need at list one of them to run a CloudSim simulation.
- Create Broker to simulate the user workload scheduling as well as virtual machine allocation and placements.
- Create one/more virtual machine and submit to the broker for further submitting it to the respective DataCenters for its placement and execution management during the simulation run.

How to run my first cloudsim simulation scenario

- Create one/more Cloudlet and submit the cloudlet list to the broker for further task scheduling on the active virtual machines for its processing during the simulation run.
- Starts the simulation, this will initiate all the entities and components created above and put them into execution for supporting various simulation operations.
- Stop the simulation, concludes simulation and flush all the entities & components before the exit of a simulation run.
- Print results when the simulation is over, where you will be able to display which cloudlet executed on which virtual machine along with how much time it spent in execution, its start time as well as its finish time.