CLOUDSIM: A SOFTWARE FRAMEWORK FOR MODELLING CLOUD COMPUTING ENVIRONMENT

Dr.AnanthiSheshasaayee[1], Swetha Margaret T.A.[2]

[1]Research Supervisor, [2]Research Scholar, PG & Research Dept. of Computer Science & Application, Quaid-E- Millath Government College for Women (AN), (India)

ABSTRACT

Advances in computing have opened up many potential possibilities. Previously, the main concern of application developers was the consumption and hosting of applications, keeping in mind the acquisition of resources with a fixed capacity to handle the expected traffic due to the demand for the application, as well as the installation, configuration and maintenance of the whole sustaining stack. With the advent of the Cloud, application deployment and hosting has become highly flexible, easier and less expensive because of the pay-per-use chargeback model offered by cloud service providers through online mediums and portals [1].Cloud computing is a best-fit for applications where users have heterogeneous, dynamic, and competing quality of service (QoS) requirements. Different applications have different performance levels, workloads and dynamic application scaling requirements, but these characteristics, service models and deployment models create a vague situation when we use the cloud to host applications. The cloud creates complex provisioning, deployment, and configuration requirements [1].

Keywords: IaaS, SaaS, PaaS, Pay-Per-Use, Quality of Service (QoS)

I. INTRODUCTION

Cloud computing dispenses infrastructure, platform, and software as services, which is made available on subscription basis, that is services in a pay-as-you-go model to consumers. These services in business are individually referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Clouds aim to power the next generation datacenters by divulging them as a network of virtual services like hardware, database, user-interface and application logic. This allows the users to access and deploy applications from anywhere in the world on-demand at competitive costs depending on users QoS (Quality of Service) requirements. Developers with pioneering ideas for new Internet services are no longer required to make large capital expenditures on hardware and software infrastructures to deploy their services or human expense to operate it. Offer's substantial benefit to IT companies by freeing them from the low level task of setting up basic hardware and software infrastructures and thus enabling more focus on innovation and creation of business values.

As Cloud computing is a rapidly embryonic research area, there is a severe lack of defined standards, tools and methods that can resourcefully tackle the infrastructure and application level complexities. Hence in the near future there would be a number of research efforts both in academia and industry towards essential core algorithms, policies, application benchmarking based execution contexts. Bearing in mind, that none of the current distributed (including Grid and Network) system simulators offer the environment that can be directly used for modelling Cloud computing environments, we present CloudSim: a new, comprehensive, and extensible simulation framework that allows seamless modelling, simulation, and experimentation of emerging Cloud computing infrastructures and application services. By using CloudSim, researchers and industry-based developers can test the performance of a newly developed application service in a controlled and easy to set-up environment. The main advantages of using CloudSim for initial performance testing include [1]:

(i) Time effectiveness: It requires very less effort and time to implement Cloud-based application provisioning test environment.

(ii) Flexibility and applicability: Developers can model and test the performance of their application services in heterogeneous Cloud environments (Amazon EC2, Microsoft Azure) with diminutive programming and deployment effort.

II. CLOUD SIMULATIONS

Cloud service providers charge their users depending upon the space or service provided. In R&D, it is not always possible to have the tangible cloud infrastructure for performing experiments. For any research scholar, academician or scientist, it is not feasible to hire cloud services every time and then execute their algorithms or implementations. For the purpose of research, development and testing, open source libraries are available, which give the sense of cloud services. Nowadays, in the research souk, cloud simulators are widely used by research scholars and practitioners, without the need to pay any amount to a cloud service provider.[3]

Tasks performed by cloud simulators are:

- Modelling and simulation of large scale cloud computing data centers.
- Modelling and simulation of virtualized server hosts with customizable policies for provisioning host resources to VM's.
- Modelling and simulation of energy-aware computational resources.
- Modelling and simulation of data center network topologies and message-passing applications.
- Modelling and simulation of federated clouds.
- Dynamic insertion of simulation elements.
- User-defined policies for allocation of hosts to VMs, and policies for allotting host resources to VMs [2].

2.1 Why simulation is important for the cloud environment?

Cloud service providers offer elastic, on-demand, measured infrastructure, platforms and software services. In the public cloud, tenants have control over the Operating system, storage and deployed applications. In the public cloud deployment model, the performance of an application deployed in multiple regions (considering the geographical locations) is a matter of concern for organizations. Proof of concepts in the public cloud environment gives a better understanding, but costs a lot in terms of capacity building and resource usage even in the pay-per-use model [1]. CloudSim, which is a toolkit for the modelling and simulation of Cloud computing environments, comes to the rescue. It provides system and behavioral modeling of the Cloud computing components. Simulation of cloud environments and applications to evaluate performance can provide constructive insights to explore such dynamic, massively distributed, and scalable environments.

The principal advantages of simulation are:

- Flexibility of defining configurations
- Ease of use and customization

III. MAGNITUDE OF CLOUDSIM

The CloudSim framework aims to ease and speed-up the process of conducting tentative studies that use Cloud computing as the application provisioning environments. Conducting these kinds of tentative studies using real Cloud infrastructures can be extremely time-consuming due to their sheer scale and complexity. The primary objective of this tool is to provide a generalized and extensible simulation framework that enables seamless modelling, simulation, and experimentation of emerging Cloud computing infrastructures and application services. By using CloudSim, researchers and industry-based developers can focus on explicit system design issues that they want to investigate, without getting concerned about the low level details related to Cloud-based infrastructures and services. [6]

CloudSim is a library for the simulation of cloud scenarios. It provides essential classes for describing data centers, computational resources, virtual machines, applications, users, and policies for the management of various parts of the system such as scheduling and provisioning. Using these components, it is easy to assess new strategies governing the use of clouds, while considering policies, scheduling algorithms, load balancing policies, etc. It can also be used to assess the competence of strategies from various perspectives such as cost, application execution time, etc. It can be used as a building block for a simulated cloud environment and can add new policies for scheduling and load balancing. By using CloudSim, organizations, R&D centers and industry-based developers can test the performance of a newly developed application in a controlled and easy to set-up environment. [7]

3.1 Uses of CloudSim Toolkit

The two important and most acknowledged uses of CloudSim Toolkit are:

- Functionality Leveraged as IS
- New Extensions Introduced

Functionality Leveraged as IS

CloudSim toolkit covers most of the activities taking place within a Data Center in detail. This includes:

- Simulating Data Center hardware definition in terms of physical machines composed of processors, storage devices, memory and internal bandwidth.
- Simulating virtual machine specification, creation and destruction.
- The management of virtual machines, allocation of physical hardware resources for the operation of virtual machines based on different policies (e.g. time-shared and space-shared).
- Simulating the execution of user programs or requests (Cloudlet/Gridlet) on the virtual machines [4].

New Extensions Introduced

In addition to the Data Center operations provided by the CloudSim toolkit, following additional functionality is required for the CloudAnalyst and hence had to be built on top of CloudSim.

- Application users – Autonomous entities are required to act as traffic generators and their behaviour needs to be configurable.
- Internet – The data transmissions across the Internet needs to be realistically modelled with network delays and bandwidth restrictions.
- Simulation defined by time period – CloudSim as a toolkit is designed to process a pre-defined series of events (e.g. submission of n-number of cloudlets.) But for user's purpose we need to convert the simulation to a time-frame limited execution where events are continuously generated by users until a pre-defined time period expires.
- Service Brokers – CloudSim already has the concept of Data Center Brokers which performs a dual role in VM management in multiple data centers and routing traffic to appropriate data centers. But for CloudAnalyst these two main responsibilities were segregated and assigned to two different entities. The Data Center Controller (described below) extends Data Center Broker and is primarily responsible for the VM management within a single data center and load balancing of VM's etc within that single data center. The new entity CloudApp Service Broker were introduced to handle the responsibility of managing the routing of user requests between data centers based on different service brokerage policies[4].

IV. ARCHITECTURE OF CLOUDSIM

The CloudSim layer provides a support for modelling and simulation of cloud environments including dedicated supervision interfaces for memory, storage, bandwidth and VMs. It also provisions hosts to VMs, application execution management and dynamic system state monitoring. A cloud service provider can implement customized strategies at this layer to study the efficiency of different policies in VM provisioning [3].

Cloud Applications

This layer includes applications that are directly available to Cloud customers. These applications may be supplied by the Cloud providers (SaaS providers) and accessed by customers either via a subscription model or a pay-per-use basis. In this layer users deploy their own applications, applications such as Salesforce.com that supply business process models on Clouds (namely, customer relationship management software) and social networks.

User-Level Middleware

This layer includes the software frameworks such as Web 2.0 Interfaces (Ajax, IBM Workplace) that helps developers in creating rich, cost-effecting user-interfaces for browser-based applications. The layer also provides the programming environments and composition tools that ease the creation, deployment, and execution of applications in Clouds [4].

Core Middleware

This layer implements the platform level services that provide runtime environment enabling Cloud computing capabilities to application services built using User-Level Middleware. Core services at this layer include

Dynamic SLA Management, Accounting, Billing, Execution monitoring and management, and Pricing. The well-known examples of services operating at this layer are Amazon EC2, Google App Engine, and Aneka [4].

System Level

The computing power in Cloud environments is supplied by a collection of data centers, which are typically installed with hundreds to thousands of hosts. At the System Level layer there exist massive physical resources (storage servers and application servers) that power the data centers. These servers are evidently managed by the higher level virtualization services and toolkits that allow sharing of their capacity among virtual instances of servers. These VM's are isolated from each other, which aid in achieving fault tolerant behaviour and isolated security context. [1]

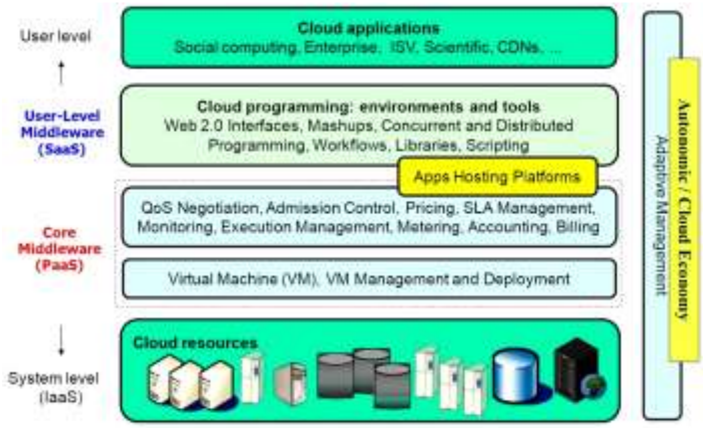

Fig 1: CloudSim Architecture

4.1 Implementing the Cloud with CloudSim

The CloudSim layer provides support for modelling and simulation of cloud environments including dedicated management interfaces for memory, storage, bandwidth and VMs. It also provisions hosts to VMs, application execution management and dynamic system state monitoring. A cloud service provider can implement customised strategies at this layer to study the efficiency of different policies in VM provisioning.The user code layer exposes basic entities such as the number of machines, their specifications, etc., as well as applications, VMs, number of users, application types and scheduling policies. The main components of the CloudSim framework are [9]:

- **Regions:** It models geographical regions in which cloud service providers distributes resources to their customers. In cloud analysis, there are six regions that correspond to six continents in the world.
- **Data centers:** It models the infrastructure services provided by various cloud service providers. It encapsulates a set of computing hosts or servers that are either heterogeneous or homogeneous in nature, based on their hardware configurations [9].
- **Data center characteristics:** It models information regarding data center resource configurations.
- **Hosts:** It models physical resources (compute or storage).
- **The user base:** It models a group of users considered as a single unit in the simulation, and its main responsibility is to generate traffic for the simulation.
- **Cloudlet:** It specifies the set of user requests. It contains the application ID, name of the user base that is the originator to which the responses have to be routed back, as well as the size of the request execution

commands, and input and output files. It models the cloud-based application services. CloudSim categorizes the complexity of an application in terms of its computational requirements. Each application service has a pre-assigned instruction length and data transfer overhead that it needs to carry out during its life cycle.

- **Service broker:** The service broker decides which data center should be selected to provide the services to the requests from the user base.
- **VMM allocation policy:** It models provisioning policies on how to allocate VMs to hosts.
- **VM scheduler:** It models the time or space shared, scheduling a policy to allocate processor cores to VMs [9].

V. POLICIES AND ALGORITHMS

CloudSim models scheduling of CPU resources at two levels: Host and VM. At Host level, the host shares fractions of each processor element (PE) to each VM running on it. Because resources are shared among VMs, this scheduler is called VmScheduler. The scheduler a host uses is a parameter of the Host constructor. In the VM level, each virtual machine divides the resources received from the host among Cloudlets running on it. Because in this level resources are shared among Cloudlets, this scheduler is called CloudletScheduler. The scheduler a VM uses is a parameter of its constructor. In both levels, there are two default policies available: the first policy, xSpaceShared (x stands for VmScheduler or Cloudlet Scheduler), required PEs by Cloudlets/VMs are exclusively allocated. It means that if there are more running elements (VMs or Cloudlets) than available PEs, the last elements to arrive wait on a queue until enough resources are free. In the second policy, xTime-Shared, fraction of available PEs are shared among running elements, and all the elements run simultaneously [8].

Policies for VM scheduling and Cloudlet scheduling can be used in any combination. For example, you can use VmScheduler Time-Shared and Cloudlet Scheduler Space-Shared, or you can use VmScheduler Time-Shared and Cloudlet Scheduler Time-Shared. It is possible even having a host running VMs with different Cloudlet scheduling policies, or a data center with hosts with different VM Scheduling policies. The VmScheduler models the behaviour of scheduling at virtual machine level like VMMs such as Xen and VMware. Therefore, if you want to model behaviour of this kind of software regarding distribution of resources among VMs running in the same host, this is the place where your new policy should be implemented [8].

Similarly, Cloudlet Scheduler models the behaviour of scheduling at operating system level: given a number of applications currently running in the system, how available CPU resources should be divided among them? If you want to model this behaviour, Cloudlet Scheduler is the class to be extended. There are several places in CloudSim where we can implement the algorithm depending on what the algorithm is intended to do. Here are several examples of classes that may need be modified or extend:

- Data center Broker -- modifying the way VM provisioning requests are submitted to data center and the way cloudlets are submitted and assigned to VMs.
- VmAllocaton Policy -- you need to extend this abstract class to implement your own algorithms for deciding which host a new VM should be placed on. You can also implement dynamic VM reallocation

algorithms by implementing the optimize Allocation method, which is called at every time frame and passed with the full set of current VMs in the data center.

- VmScheduler -- implementing algorithms for resource allocation to VMs within a single host.
- Cloudlet Scheduler -- implementing algorithms for scheduling cloudlets within a single VM.
- PowerVm Allocation Policy Migration Abstract -- a template class for implementing power-aware dynamic VM consolidation algorithms that use VM live migration to dynamically reallocate VMs at every time frame. The main method to be overridden is optimize Allocation [10].

VI. CONCLUSION

The recent efforts to design and develop Cloud technologies focus on defining novel methods, policies and mechanisms for efficiently managing Cloud infrastructures. To test these newly developed methods and policies, researchers need tools that allow them to evaluate the hypothesis prior to real deployment in an environment where one can reproduce tests. Simulation-based approaches in evaluating Cloud computing systems and application behaviours offer significant benefits, as they allow Cloud developers: (i) to test performance of their provisioning and service delivery policies in a repeatable and controllable environment free of cost; and (ii) to tune the performance bottlenecks on commercial Clouds. In order to offer a built-in support CloudSim simulates the currently available Clouds into the environment. Modelling and simulation of such environments that consist of providers encompassing multiple services and routing boundaries present unique challenges [1].

REFERENCES

[1] Buyya, Rajkumar, Rajiv Ranjan, and Rodrigo N. Calheiros. "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities." High Performance Computing & Simulation, 2009.HPCS'09.International Conference on.IEEE, 2009.

[2] Calheiros, Rodrigo N., et al. "Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services." arXiv preprint arXiv:0903.2525 (2009).

[3] Calheiros, Rodrigo N., et al. "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms." Software: Practice and Experience 41.1 (2011): 23-50.

[4] Wickremasinghe, Bhathiya. "CloudAnalyst: A CloudSim-based tool for modelling and analysis of large scale cloud computing environments." MEDC project report 22.6 (2009): 433-659.

[5] Shi, Yuxiang, Xiaohong Jiang, and Kejiang Ye. "An energy-efficient scheme for cloud resource provisioning based on cloudsim." Cluster Computing (CLUSTER), 2011 IEEE International Conference on.IEEE, 2011.

[6] http://www.opensourceforu.com/2014/03/cloudsim-framework-modelling-simulating-cloud-environment

[7] http://www.cloudbus.org/cloudsim/

[8] http://cloudsim-setup.blogspot.in/

[9] http://cloud-simulation-frameworks.wikispaces.asu.edu/

[10] http://cloud-simulation-frameworks.wikispaces.asu.edu/