SENSORCLOUDSIM: AN APPLICATION SCHEDULING & A SENSOR CLOUD INFRASTRUCTURE SIMULATOR

BY

MERRAD YACINE

A dissertation submitted in fulfilment of the requirement for the degree of Master of Science (Electronic Engineering)

Kulliyyah of Engineering
International Islamic University Malaysia

AUGUST 2019

ABSTRACT

With the enormous growth of IOT sensing devices that is tending to the Internet of everything, the data aggregated by these devices are the biggest data streams generated in the history of the IT. For this reason, aggregating those data to the cloud for leveraging the powerful cloud computing processing and storage is essential. This led to the emergence of the sensor cloud concept which is not only a mere aggregation of the sensors' data to the cloud for further processing, storage and visualization. But also, a virtualization of the sensors, making them accessible to different end users' applications, which are requesting their data. This without the end user application developer being aware of the sensor localization and hardware. So, the revolution brought by the sensor cloud paradigm is that any end user can develop an application that requests any sensor deployed anywhere, by any organization, without knowing the sensor API, localization or hardware, as long as the sensor is registered to the sensor cloud infrastructure. From the above mentioned it is obvious that the scheduling of the applications requesting the sensors is of main concern, and that optimizing the scheduling policy would definitely lead to the optimizing of the quality of the sensing as a service to the end users. However, most of the time, the scheduling policies implanted by the sensor cloud platform provider are hidden from the users and leveraging the platform is not free. This makes experimenting scheduling policies on a real platform either not possible or really cost inefficient. Here is where simulation plays a big role in research. Having a simulation platform where sensor cloud infrastructures agents and components can be modeled, scheduling policies configured, and execution time assessed would be crucial to researcher trying to develop the infrastructure performance and quality of service. In this work we will be discussing the development of such a platform, by extending cloudsim, a famous cloud computing simulation platform, and we will try to come up with an optimum quantum time in a time-sharing scheduling policy by experimenting on the simulator.

خلاصة البحث

مع النمو الهائل لأجهزة الاستشعار IOT التي تميل إلى الإنترنت من كل شيء ، فإن البيانات التي تم تجميعها بواسطة هذه الأجهزة هي أكبر تدفقات البيانات التي تم إنشاؤها في تاريخ تكنولوجيا المعلومات. لهذا السبب ، من الضروري تجميع هذه البيانات على السحابة للاستفادة من معالجة الحوسبة السحابية القوية وتخزينها. وقد أدى ذلك إلى ظهور مفهوم سحابة المستشعر الذي لا يمثل مجرد تجميع لبيانات المستشعرات إلى السحابة لمزيد من المعالجة والتخزين والتصور. ولكن أيضًا ، محاكاة افتراضية للمستشعرات ، مما يجعلها في متناول تطبيقات المستخدمين المختلفين التي تطلب بياناتهم. هذا دون أن يدرك مطور تطبيق المستخدم النهائي توطين المستشعر والأجهزة. لذلك ، فإن الثورة الناتجة عن نموذج سحابة المستشعر هي أن أي مستخدم نهائي يمكنه تطوير تطبيق يطلب أي مستشعر يتم نشره في أي مكان ، من قبل أي مؤسسة ، دون معرفة واجهة برمجة تطبيقات المستشعر أو الترجمة أو الأجهزة ، طالما أن المستشعر مسجل في المستشعر البنية التحتية السحابية. يتضح مما سبق ذكره أن جدولة التطبيقات التي تطلب المستشعرات هي مصدر قلق رئيسي ، وأن تحسين سياسة الجدولة سيؤدي بالتأكيد إلى تحسين جودة الاستشعار عن بعد كخدمة للمستخدمين النهائيين. ومع ذلك ، في معظم الأوقات ، يتم إخفاء سياسات الجدولة التي يزرعها موفر منصة سحابة المستشعر عن المستخدمين والاستفادة من المنصة ليست مجانية. هذا يجعل تجريب سياسات الجدولة على منصة حقيقية إما غير ممكن أو غير فعال بالفعل من حيث التكلفة. هنا حيث تلعب المحاكاة دورًا كبيرًا في البحث. إن امتلاك منصة محاكاة حيث يمكن نمذجة عناصر ومكونات البنية التحتية السحابية لأجهزة الاستشعار ، وتهيئة سياسات الجدولة ، وتقييم وقت التنفيذ سيكون أمرًا ضروريًا للباحث الذي يحاول تطوير أداء البنية التحتية وجودة الخدمة. سنناقش في هذا العمل تطوير مثل هذا النظام الأساسي ، من خلال توسيع cloudsim ، منصة شهيرة لمحاكاة الحوسبة السحابية ، وسنحاول التوصل إلى وقت كوان الأمثل في سياسة جدولة مشاركة الوقت من خلال تجربة المحاكاة.

APPROVAL PAGE

I certify that I have supervised and read this study and that in my opinion, it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Master of Science (Electronic Engineering).

…………………………………..
Mohamed Hadi Habaebi
Supervisor

…………………………………..
Md. Rafiqul Islam
Co-Supervisor

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Master of Science (Electronic Engineering).

…………………………………..
Internal Examiner

…………………………………..
Internal Examiner

This dissertation was submitted to the Department of Electrical and Computer Engineering and is accepted as a fulfilment of the requirement for the degree of Master of Science (Electronic Engineering).

…………………………………..
Mohamed Hadi Habaebi
Head, Department of Electrical and Computer Engineering

This thesis was submitted to the Kulliyyah of Engineering and is accepted as a fulfilment of the requirement for the degree of Master of Science (Electronic Engineering).

…………………………………..
Ahmad Faris Ismail
Dean, Kulliyyah of Engineering

DECLARATION

I hereby declare that this dissertation is the result of my own investigations, except where otherwise stated. I also declare that it has not been previously or concurrently submitted as a whole for any other degrees at IIUM or other institutions.

Yacine MERRAD

Signature .... Date ....

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

DECLARATION OF COPYRIGHT AND AFFIRMATION OF FAIR USE OF UNPUBLISHED RESEARCH

SENSORCLOUDSIM: AN APPLICATION SCHEDULING & A SENSOR CLOUD INFRASTRUCTURE SIMULATOR

I declare that the copyright holders of this dissertation are jointly owned by the student and IIUM.

No part of this unpublished research may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without prior written permission of the copyright holder except as provided below

1. Any material contained in or derived from this unpublished research may be used by others in their writing with due acknowledgement.

2. IIUM or its library will have the right to make and transmit copies (print or electronic) for institutional and academic purposes.

3. The IIUM library will have the right to make, store in a retrieved system and supply copies of this unpublished research if requested by other universities and research libraries.

By signing this form, I acknowledged that I have read and understand the IIUM Intellectual Property Right and Commercialization policy.

Affirmed by Yacine MERRAD

……..…………………….. …………………………..
Signature Date

ACKNOWLEDGEMENTS

First of all, I praise the Almighty Allah for His guard and guidance on me throughout this research work, then my parents who have guided and supported me throughout my life.

I would like to express my deep gratitude to my supervisor, Associate Prof. Dr. Mohamed Hadi Habaebi for his continuous supervision, support, assistance and useful critiques during this research work, great notions, enthusiastic encouragement and patient and superb guidance. Effective and successful research writing would not be possible without his endless encouragement, support and priceless advices. So therefore, I am deeply grateful to him as he opens my eyes to this new field of research where I find it priceless and passionate approach to pursue.

I would like to express my appreciation and countless thanks to my parents and family members for their support and insightful advices. I would like to extend my appreciation, reverence and thanks to all of my lecturers in Kulliyyah of Engineering. Last and not least, I am willing to thank overall everybody that helps me throughout this journey where it will not be a reality without their continuous support and help

.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

WSN	Wireless Sensor Network
API	Application Programming interface
MIPS	Million Instructions Per Second
BW	Band Width
RAM	Random Access Memory
VM	Virtual Machine
FCFS	First Come First Served
SRJF	Shortest Remaining Job First
RR	Round Robin
GUI	Graphical User Interface
PE	Processing Elements
SDK	Software Development Kit
TQ	Time Quantum
OS	Operating System
QOS	Quality of Service

CHAPTER ONE
INTRODUCTION

1.1 OVERVIEW

Sensor cloud infrastructure is a technology that brings to light the sizable computing resources of the cloud that deals with the processing, storage and analysis of the enormous streams of data. The latter is not only generated by millions of sensing devices deployed in different wireless networks, but it also enables multiple end users requesting the sensing as a service to have access to the sensed data coming from sensors registered to the cloud.

In traditional WSN, the user of the network is the same owner agent, considering the application API used for the development of applications accessing the sensing nodes is specified by the sensor's manufacturer. Besides, this obliges the end user to be aware of sensor API, its hardware and location in the network. Sensor cloud infrastructure adds a sensor virtualization layer developed on the cloud, which hides hardware and location details from the end user. Consequently, enables him to access the sensing services of various WSN, which are deployed in different geographical areas by different organizations, only by paying some fee, as if the sensor node was part of his device hardware.

Accordingly, what is brought by this technology is enabling multiple end users to access cloud registered sensor devices. For this reason, the scheduling policy in allocating the sensed data to various applications requesting is of a big concern. Furthermore, improving the efficiency of the application scheduling policy is directly linked to the improvement of the quality of the sensing service delivered to the end user application. However, what studies and researches show is very little inquiry to the enhancement of the application scheduling policy in the sensor cloud infrastructure. This is mainly because cloud providers do not permit the configuration of the cloud resources scheduling policies and keep them invisible to the user.

Even if cloud providers do permit the scheduling configuration, leveraging cloud resources are not free, making it a very cost wise inefficient platform for experimenting new algorithms. Hence, this is where simulation has a big role to play. The latter permits the trial of free and unlimited configurations and gives results, which can be analyzed to assess the performance. So, simulation platforms are great scientific tools allowing time and cost-efficient empirical experimenting. In this work, a toolkit will be designed for simulating and modeling sensor cloud infrastructure that permits the scheduling configuration and gives time aware result.

1.2 STATEMENT OF THE PROBLEM

Simulator would suit perfectly. By all means, a simulator allows the repeated testing no cost, evaluation of the services under different scenarios, the detection of issues affecting the performance. A lot of cloud furnisher such as Amazon Web Services and Microsoft Azure are offering the Sensing as a Service to different end users, and are embracing the Sensor cloud paradigm. For example, Microsoft azure has designed Azure IOT hub service, which enables organizations or privates to register their sensors and made them available for different end user applications.

Despite all that, there is no simulator that permits the modeling of the sensor cloud infrastructure, no simulator that clearly models the sensors, the cloud infrastructure and the end user applications, with the logical components that match them all together has been designed. Researchers are finding it difficult to assess applications where such infrastructures are involved. As a support to that claim, (Sudip

Misra ; Anuj Singh ; Subarna Chatterjee ; Mohammad S. Obaidat, 2014) work is cited. The work is entitled: A Sensor-Cloud-Based Architecture for the Integration of Military Tri-Services Operations and Decision Making. As the title denounces, the work is about proposing a solution to real world application using sensor cloud infrastructure.

In this work, the authors assessed the concept they developed using simulation. Clearly, they used Cloudsim simulator, which did not allow very realistic modeling. The author modeled the sensors using the VM class, which is normally used for modelling virtual machines in a host residing in a cloud data center and in cloud computing. For application, they used the cloudlet class, which is indeed the class appropriate for modelling the applications. That is to say, the author used a class within the software to model the sensors.That class should be used to model another component from the sensor cloud infrastructure, which are the VMs. These clearly demonstrate the need to customize CloudSim to allow researches and even non-researchers to model in a more realistic and proper manner, sensor cloud infrastructure and asses the performances of their architecture. Here are the problems and issues that motivated us to conduct our work, summarized in the following points:

1. The absence of a simulating platform that can model each of the sensor cloud infrastructure components (the sensor nodes, the cloud computing components including the datacenters, the host, VMs, and the end applications) as well as simulate the performance of the architecture.
2. The lack of studies conducted on assessing the performance of different scheduling policies under different scenarios in sensor cloud infrastructure. For those purpose this research will be conducted in the aim of achieving the objectives that will be state in the very next session of this chapter.

1.3 RESEARCH OBJECTIVES

The main objectives of this research are as follows:

1. Extending Cloudsim software, for developing a framework where sensor cloud infrastructure can be modeled, simulated and assessed.
2. Assessing and comparing the performance of different scheduling of the application requests on the sensors and deducing the most suitable one.
3. Deducing the effect and impact on the requests scheduling performance of the request load in term of the requests frequency of arrivals and the requests length.

1.4 RESEARCH METHODOLOGY

First a flow chart will be given illustrating on how the research would be conducted systematically. The research methodology, then, goes in the aim of achieving the objectives stated previously. As for the optimum quantum time, it will be obtained after multiple simulations, trying different value of the quantum time. For this aim, wi a scenario will be simulated where three different sensors will be aggregating their sensed data to the cloud in a periodic manner, and nine end user applications would request the data for further processing.

To enumerate, let assume the sensors to be named s1, s2, and applications: app1, app2, app3, app4, app5, app6, app7. Combinations would be tried where for example: app1, app2, app3 will be all requesting s1 data. app4, app5, app6 will be requesting s2. app7, app8, app9 will request s3. Applications are modeled in the simulator by a class where it is needed to precise the length of the workload in terms of MIPS, and the VM on which it should run. Therefore, the applications will be having different workloads.

Furthermore, different combinations of applications requesting the sensors would be tried, and for each combination, an optimum time quantum is come up with for scheduling the applications in a round robin fashion. This will be achieved after simulating the scenarios using different time quantum. First, a simulation experimen is conducted using an initial value for the tq, then the tq will be incremented with a certain step value until the performance in terms of throughput (number of applications terminated in a unit of time) and average application execution time will start deteriorating. After that, tq would be decreased by a smaller step value until the performance start decreasing. The optimal value of tq will be the last value before the performance in terms of applications execution time start decreasing.

For this purpose, the tool that will be used is the customized version of CloudSim simulator. Chiefly, we will be using CloudSim as base and extending it by adding some classes, to make a realistic framework for simulating sensor cloud infrastructure under different applications scheduling policies. round robin with different time quantum will be used. Nonetheless, before discussing the added extensions, first CloudSim toolkit, its software architecture, how does its model cloud computing should be discussed.

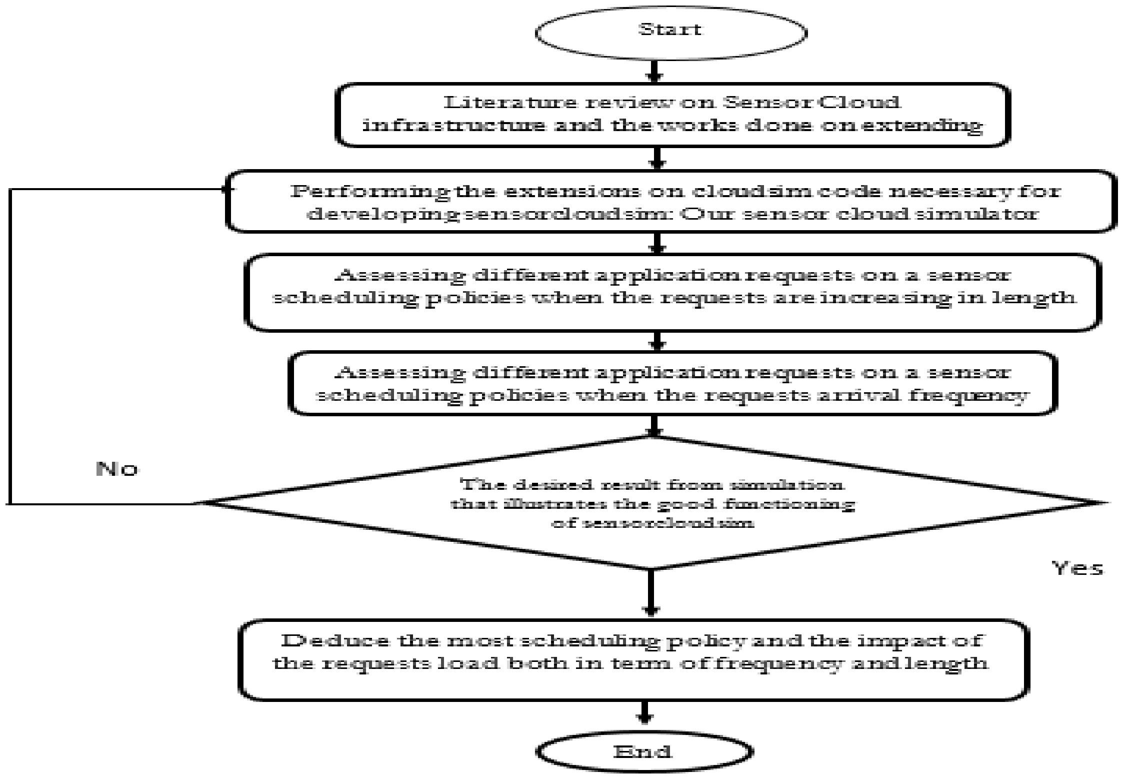

Figure 1.1 Research Methodology Flowchart

1.5 RESEARCH SCOPE

The scope of this work will not focus on any of the sensor cloud issues related to the nodes, whether by themselves or to the sensors network such as bandwidth consumption, ineffective power consumption, and latency. Rather, we will be aiming to improve the quality of the sensing as a service, delivered to different end user applications requesting data from sensors registered to the cloud. The sensing QoS improvement will be brought by our work, with an optimal application scheduling policy.

As for the simulation platform, the aim is to develop a CloudSim based customized platform, which enables the modeling of the sensor cloud infrastructure agents and grants the user to input his scheduling polices configuration. The metrics displayed by the framework in the simulation output will be the application execution start time, its finish waiting and turnaround time.

1.6 DISSERTATION OUTLINE

The dissertation is organized as such: Chapter One presents a brief introduction to the challenges facing regular wireless sensor networks and the solution brought by linking the WSN to the cloud and the emergence of the sensor cloud concept, it then demonstrates the problem statement, research objectives, and scope. Chapter Two consists of an overview of cloudsim simulator, its entities, allocation policies and data flow will be described, then a literature review on the sensor cloudim paradigm and infrastructure and on the works that has been done to extend cloudsim for developing customized simulator platforms will be given. Chapter Three presents the design and work flow of sensorcloudsim and describes in details all the extensions added to cloudsim and describes as well the simulation scenarios that will be performed in order to conduct the first study on requests scheduling and the impact of the frequency of the request's arrival and their length on the scheduling performance. Results are discussed and analyzed in Chapter Four. Chapter Five concludes and provides recommendations for future work.

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

First in this chapter, an overview on the sensor cloud paradigm will be given. the concept will be explained as well as the technology and architecture of this infrastructure. Then in the next section an overview on CloudSim simulation platform will be given where the main simulation components and data flow will be explained. Following that the next section of the chapter will discuss the work related to the research. This section of the chapter will be divided into two sub ones. The first part of the related work is on analyzing some of the work done on the sensor cloud infrastructure then, personal feedback on the related work would be given. In the second subsection some of the works that inspired us on how to extend cloudsim, famous toolkit to a more suitable framework, where all of the sensor cloud infrastructure agents and workflow can be modeled will be discussed.

2.2 SENSOR CLOUD INFRASTRUCTURE OVERVIEW

Nowadays, enormous progress is made in microelectronics and Micro-Electro-Mechanical Systems (MEMS). Small sensor nodes with processing capabilities are able to work on cell batteries, some of them are Internet Protocol (IP) including a TCP/IP interface module, while some others are more resource constrained and are only supporting wireless communication protocols such as Bluetooth, ZigBee, or Wi-Fi for forwarding their sensed data. Additionally, such devices are being deployed in the form of networks in various constrained and remote environment with various topologies (mountains, farms, underwater, embedded in wearable clothes), to serve different including healthcare monitoring (Peiris, V,2013), (O'Donovan,2009), environmental/earth sensing and industrial monitoring (Saleem K,2009). This led by consequence to the emergence of what is now known as Wireless Sensor Networks (WSN).

WSN can be defined as a network of devices, denoted as nodes, which can sense the environment and communicate the information gathered from the monitored field (e.g. an area or volume) through wireless links. Despite the benefits brought by the WSN, it is still facing many challenges and the concept showed many limitations. One of WSN limitations is the fact that it is deployed to serve one application, in other terms the sensor network owner is the network user. To develop an application that makes the use of the sensors as resources, the developer must use the restrained Application Program Interface (API), which is provided by the sensor node manufacturer and must be aware of the location as well as the information related to the sensor node in order to communicate with it. However, this excludes unaware individuals of the hardware and location of the sensor node and all the information regarding the network to make use of the WSN.

In a traditional WSN, the sensed data are forwarded to a server, so it can be later accessed by the user application (which is - as mentioned earlier - the same network owner). Another drawback of this architecture is the rigidity of the server hardware, that is why, if there is a big scaling in the sensor network, chances are that the server hardware will support it. In the present work, classical issues regarding WSN such as security, power consumption, routing, processing capabilities, Operating System (OS) will not be addressed, rather the focus will be on two major drawbacks of the traditional WSN, which are:

1) The single user centricity of WSN based applications, as referred in (Yuriyama,

M,2010).

2) The limited scalability of the network in traditional WSN.

Notably, sensor cloud infrastructure is a paradigm that solves the two above-mentioned limitations of the traditional WSN. It is nowadays almost replacing the traditional architecture, which is being more and more outdated. In detail, sensor-cloud architecture has been conceived as a potential solution for multi organization WSN deployment, and data access (M. M. Hassan, 2009), (A. Alamri et al., 2013). It is the combination of two well-established technologies, which are the WSN technology, and the cloud computing technology. According to Micro Strains, a sensor cloud is formally defined as a unique sensor data storage, visualization and remoting management platform that leverages powerful cloud computing technologies to provide excellent data scalability, rapid visualization, and user programmable analysis (A. Alamri et al., 2013).

To clarify, knowing the power that cloud computing offers, we can clearly deduce that the scalability of the network is no more a problem. Indeed, what the cloud sensor architecture proposes is to implement a middleware on the cloud between the physical sensors and the end user, which consists of a virtualization layer of the sensors. On that note, this virtual layer aims to hide the sensor information from the end user, and to share the sensed data between multiple and user applications running on different platforms, and different OSs. This concept of sensor virtualization is what makes sensor cloud architecture so revolutionary. Moreover, the sensor owner can register the sensors to the cloud platform and make it accessible (by mean of an access fee), to different end users, using different applications, running on different hardware, implementing different Oss.

Those users can, thus, access sensed data from different sensors deployed on different geographical locations, owned by different organization, without being aware of sensor location and hardware details, enabling an unprecedented connection and marketing of sensors networks, and sensed data, accessible by literally everyone. Cloud computing has enabled to provide the different services to end users, such as Infrastructure as a Service (IaaS) (Bhardwaj, 2010), Platform as a Service (PaaS) (Boniface, 2010), Software as a Service (SaaS) (Buxmann, P, 2008). Sensor Cloud is now introducing Sensor as a Service (SeaaS) (Alam, S, 2010). This, of course, allows end users to use different sensors, from different networks, as if they were running on their own hardware.

In the following section, we will present a critical analysis on some of the related work that has been done on the topic of the sensor cloud. It will identically provide a summary of previous data, with a brief stating of the pros and cons of the research done on the topic. Finally, some of the gaps that make opportunities for future research will be further discussed.

2.3 CLOUDSIM SIMULATION PLATFORM OVERVIEW

CloudSim is open source software, developed in clouds lab at the University of Melbourne, Australia. The software has been developed as a toolkit for modeling and simulating cloud-computing components, which are data centers host servers and VMs, in addition to application services. Namely, it is suitable to simulate the scheduling of different applications and services for cloud infrastructures. The software is a set of layered Java packages that anyone can download and run on eclipse Java SDK. It is a very flexible software.

2.3.1 CloudSim Entities, Allocation Policies and Data Flow

Whenever a user simulates a scenario using CloudSim, the framework creates an entity called CIS (Cloud Infrastructure Service). CIS is a registry where the framework registers all the cloud resources that the user has created. Basically, the cloud computing components a user can create are the datacenters and the Hosts on each data centers, as well as the virtual machines sharing the host resources. The user has to specify the computing resources allocated to each host in terms of:

- Number of processing units (number of CPU cores).
- RAM
- Bandwidth
- VMs that are created on each host sharing its computing resources
- Tasks or applications are modeled by the cloudlet class in the framework, characterized by the length of task in terms of MIPS

The framework then, creates a broker responsible for submitting the tasks to the datacenters after getting information about the datacenters created including their hosts, computing resources and VMs from the CIS entity. There are three allocations and scheduling policies under which this framework model is working. The policies are:

- VM allocation policy: used by data center for allocating VMs to the hosts
- VM scheduler policy: used by hosts to schedule VMs
- Cloudlet scheduler policy: used by VMs to schedule cloudlets on them

All the policies are either timeshared or space shared. Figure 5 (Calheiros, 2011) illustrates the layered architecture of Cloudsim.