# Chapter 4

**Methodology and Implementation**

*The most important chapter, mainly focused on how we have implemented our work, what tools we have used, how we have used that tool for implantation. In this chapter we have discussed about this methodology, Python and its library functionality we have also discussed how we have used them in our implementation system for producing output and for further use. At last, we have summarized our whole chapter and our work in discussion portion.*

### Introduction

In this section learn some of the platform and simulation tools which can be used for implementing load balancing algorithm in a cloud computing environment. This tool is very useful for measuring the various QoS parameter like response time, waiting time, etc.

### Design Constraints

Design and build our proposed method, we used dataset, libery, pseudocode, simulation tool for analyzing the results of proposed method and compare it with the existing load balancing algorithm.

###### Software Requirement in Proposed Algorithm (minimum)

* PyCharm (IDE)
* Python (version – 3.9.7) (language)
* NumPy (library)
* Pandas (library)
* SciPy
* Matplotlib (library)

###### Hardware Requirement in Proposed Algorithm

* 4 GB RAM
* Intel Pentium Processor or higher

#### 

#### Python

#### Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation.

#### PyCharm

PyCharm is an integrated development environment used in computer programming, specifically for the Python language. I used in the latest version in in PyCharm Ide.

#### NumPy

#### NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. NumPy is a Python package. It stands for ‘Numerical Python’. It is a library consisting of multidimensional array objects and a collection of routines for processing of array

#### Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

#### SciPy

SciPy is a free and open-source Python library used for scientific computing and technical computing. SciPy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

* + 1. **Matplotlib**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

### Dataset

|  |
| --- |
| Task 1 (UJMJQR) Assigned in Server XSRIG. State: Executing |
| Task 2 (ROGIWU) Assigned in Server HNQDQ. State: Executing |
| Task 3 (YTSYIW) Assigned in Server GLJHN. State: Executing |
| Task 4 (UWQBRQ) Assigned in Server IRXNJ. State: Executing |
| Task 5 (GPQHUW) Assigned in Server YKCCF. State: Executing |
| Task 6 (YIYHHO) Assigned in Server KYGWX. State: Executing |

### Methods

We have already created our data. Now we will apply our method to get our desire output.

In the method we used multiple datacenters. A datacenter is a large group of networked computer servers typically used by organizations for the remote storage, processing, or distribution of large amounts of data.

In one datacenter we used multiple virtual servers. A virtual server is a server that shares hardware and software resources with other operating systems (OS). Virtual server read the input files and create multiple tasks. One Task have three properties:

1. X co-ordinate,
2. Y co-ordinate and
3. Time Management (task will create within a time interval)

Here, time management is one kind of multi-processing/ multi-tasking system.

We used multithreading for managing time as well as task scheduling.

Here is the description of all the method materials given below:

1. **Datacenter**

Data center is a physical resource that helps businesses to store, organize, and operate data efficiently. The maintenance cost is high because developers of the organization do maintenance.

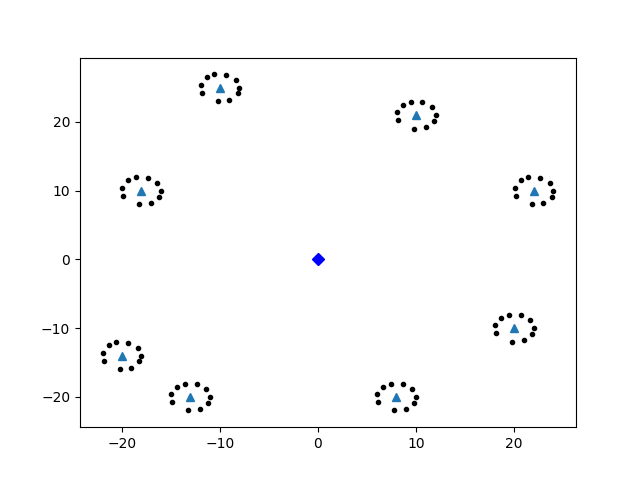


Fig: Load Balancer, datacenter, and virtual Server

1. **Virtual server**

A virtual private server is a virtual machine sold as a service by an Internet hosting service. The virtual dedicated server also has a similar meaning.

1. **multi-processing**

Multiprocessing is the use of two or more central processing units within a single computer system. The term also refers to the ability of a system to support more than one processor or the ability to allocate tasks between them

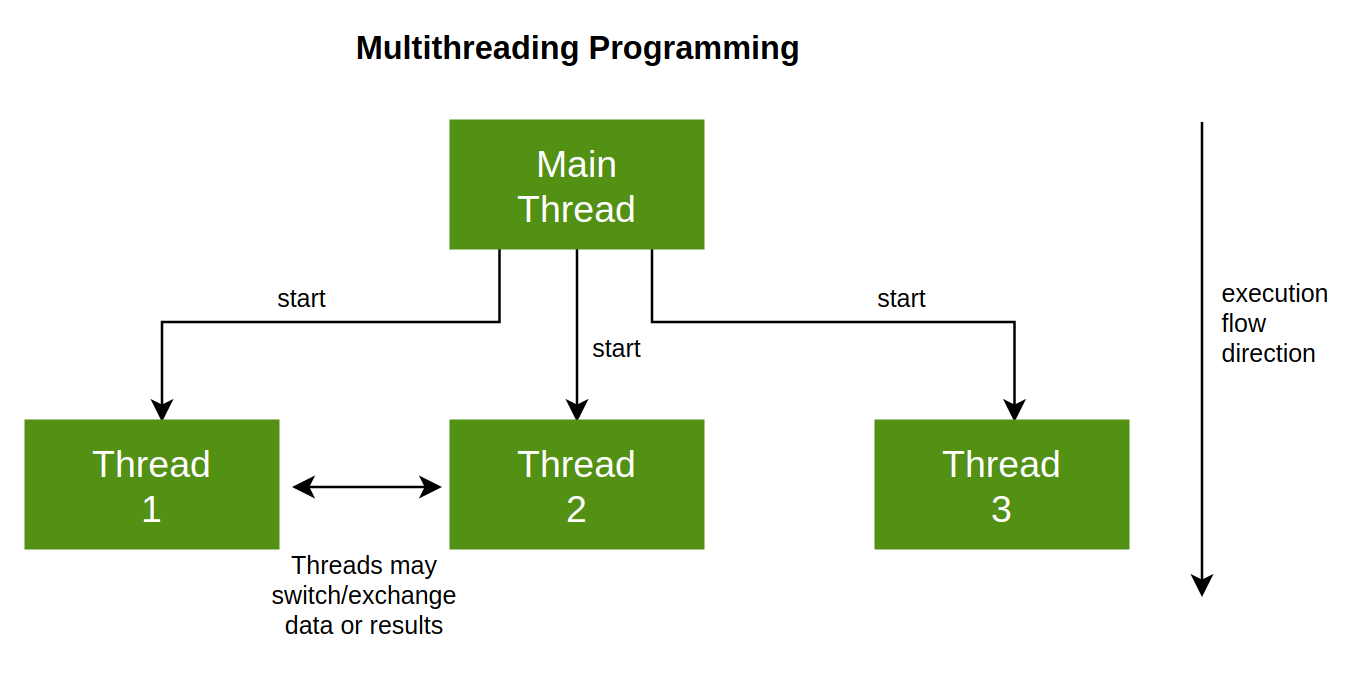
1. **Thread**

A thread is a unit of execution on concurrent programming. Multithreading is a technique which allows a CPU to execute many tasks of one process at the same time. These threads can execute individually while sharing their process resources.

1. **Multi-threading**

**Multithreading in Python** programming is a well-known technique in which multiple threads in a process share their data space with the main thread which makes information sharing and communication within threads easy and efficient. Threads are lighter than processes. Multi threads may execute individually while sharing their process resources. The purpose of multithreading is to run multiple tasks and function cells at the same time.

**How Multithreading Works:**

****

**Figure: Multithreading**

## **4.3 PROCEDURE**

First of all, input files are given to the system. Our system will create the datacenter virtually. The datacenter read the input files and then creates multiple virtual servers. Virtual server creates multiple tasks with the input files. After that, the system will save all the things including datacenter, virtual server as images. After the creation of datacenter and server virtually, the load balancer of Modified Ant Colony (MACO) will run and the server will send files (data) to the load balancer. The load balancer then read all the files and takes the file as input. Our system used multithreading for time management. Here is the description of process of multithreading given:

**Step 1:** At first, we will take input from the dataset and our system will create datacenters and virtual servers automatically based on input.

**Step 2:** Our load balancer will run in one thread. This load balancer will take tasks as input. Using the proposed algorithm, the load balancer will choose the best server. After choosing the best server, our system will assign the tasks to the best server or sent the tasks to the best server.

**Step 3:** Task will start execution in the server if the server will free. If the server will not free, then the task will wait in the queue of that server for further execution. In this way, our load balancer works.

**Step 4:** In this step, our next work will be executed in a new thread. Final thread will always check the server whether the server will remain idle or free (the server has no task). If any server will remain free for a certain time, then the thread will shut down that server. The server will be assignable for any task when the server will free.