Experiment No. 5

Aim: Study assignment on Edge and Fog Simulators

Introduction :

In recent years, computing and information storage have been moved to the cloud, using paradigms that allow managing the data that users generate daily. In addition, the reduction in the cost of smart devices has led to new challenges and the need to create new infrastructures capable of managing the data they generate.

The proliferation of these devices and new types of services and applications led to the emergence of the Internet of Things (IoT) paradigm and the consequent challenges to being able to read, store or even process the immense amount of information that all these devices use or generate continuously. Initially, there were no paradigms that could process all this information,exceeding the capacities and bandwidth available on the network. For these reasons, paradigms such as Cloud Computing ,Edge Computing , Fog Computing or Mobile Edge Computing emerged. Firstly, Cloud Computing was used as a measure to solve these problems, taking computing to the cloud and thus allowing the distribution of information among the different servers that make up the infrastructure to reduce congestion and speed up data processing.When Cloud Computing was implemented worldwide as a paradigm for information storage and processing, the market advanced rapidly allowing the population to use smart devices (smartphones, smart TVs, smartwatches) , which substantially increased the need for the use of network resources and servers processing information. This led to the emergence of new computing paradigms such as Edge Computing and Fog Computing , which brought information computing to the edge of the network by bringing computing and storage capacity closer to users, reducing latency, and increasing information availability.

The Edge Computing paradigm is a rapidly growing field of research that covers a wide spectrum of features and technologies that bring information processing and storage to the edge of the network. This paradigm includes other aspects such as Mobile Cloud Computing or even Fog Computing.Finally, the Fog Computing paradigm is a layered model whose main objective is to preserve the benefits of the cloud for the deployment of applications or services capable of processing or storing end-user information in distributed nodes. This allows, together with Edge nodes, to reduce the response time of the tasks or processes that are generated during their use.However, designing these infrastructures may not be practical and, in addition,generating testbeds for this type of scenario can involve very high costs as there are many devices and connections involved. For these reasons, prior studies must be carried out to analyze the feasibility of the infrastructure and the desired scenarios. This is why the solution chosen worldwide is based on the use of simulators that make it possible to define an environment by introducing the desired scenarios, applications, or devices and

obtain results that make it possible to decide whether or not it is feasible to implement(such as bandwidth or CPU usage). Currently, there are many simulators in the field capable of defining Cloud or Edge environments with their consequent characteristics (bandwidth, devices, processes, links, CPU, etc.), in addition to the definition of the topology. However, there is no simulator capable of covering all possible aspects of each experiment such as, for example, the mobility of sensors or mobile devices. Therefore, this work presents the development of a simulator called ENIGMA (gEneric Iot edGe siMulAtor), based on the SimGrid simulation tool, capable of simulating Edge and Fog Computing environments that facilitate the user the study and analysis of scenarios or Cloud environments,allowing the specification of different characteristics of the components (CPU, RAM,bandwidth, power consumption, etc.), the analysis of different types of applications and even the study of environments that include the mobility of sensors or mobile devices.

This will allow users to easily create environments, allowing them to define the components and connections available to them. In addition, building these infrastructures without prior knowledge is usually not easy, and generating test benches for certain scenarios involves high costs, so using simulators capable of performing these tasks speeds up and lowers the costs of implementing such infrastructures.It is also vitally important to take into account aspects such as the scope of the simulators, since nowadays, if Smart Cities are taken into account, the number of continuously connected devices in an environment can become very large, and the larger the territory, the more the simulators must support. Therefore, scalability is another important aspect in our work, so that many of the existing computers can obtain results in most of the created environments, an aspect that few simulators have, since they focus on smaller and more specific cases. In addition, there are not a large number of simulators capable of providing mobility in sensors or devices, so it becomes complicated to determine a good configuration to study the infrastructure.That is why, in addition, we present an extension to SimGrid, which is a set of tools for the simulation of large-scale distributed computational systems, where a set of capabilities for component mobility is added. These mobility capabilities provide SimGrid with several features that facilitate the execution of simulations for multiple

infrastructures and applications. In addition, this extension to the simulator has been designed as an API of functions that are not restricted to a specific subset of infrastructures, which adds versatility to its use.

As mentioned, this article aims to present an alternative to existing simulators with ENIGMA, capable of simulating Edge and Fog Computing environments with mobility capabilities of IoT devices. This simulator will offer users the possibility to generate large-scale infrastructures, where hundreds or thousands of components are involved in the simulation. For this, a series of previous steps have been necessary, such

as validating the project with another existing project in the field and evaluating the results shown in a small simulation environment. As an important aspect of this work, and as the title indicates, this simulator is

intended to be scalable in order to simulate large Edge and Fog Computing environments. In this context, we want to emphasize that by scalability we are looking for a simulator capable of using applications with hundreds or thousands of components without compromising the use of computational resources. In this way, a series of experiments will be carried out to evaluate the scope of several of the most widely used simulators in recent years, studying the limitations they have in larger environments with more elements involved in the simulation. Thus, it will be possible to analyze the differences in this area with the developed simulator.

A Toolkit for Modeling and Simulation of Resource Management Techniques in Internet of Things, Edge and Fog Computing Environments with the following new features:

- Mobility-support and Migration Management
- Supporting real mobility datasets
- Implementing different random mobility models
- Microservice Orchestration
- Dynamic Distributed Clustering

Any Combinations of Above-mentioned Features

Full Compatibility with the Latest Version of the CloudSim (i.e., CloudSim 5) and Previous iFogSim

Version and Tutorials How to run iFogSim2 ?

1. Eclipse IDE:
2. Create a Java project
3. Inside the project directory, initialize an empty Git repository with the following command: git init
4. Add the Git repository of iFogSim2 as the origin remote:
5. git remote add origin https://github.com/Cloudslab/iFogSim
6. Pull the contents of the repository to your machine:
7. git pull origin main
8. Include the JARs to your project

Run the example files (e.g. TranslationServiceFog_Clustering.java,

CrowdSensing_Microservices_RandomMobility_Clustering.java) to get started

IntelliJ IDEA:

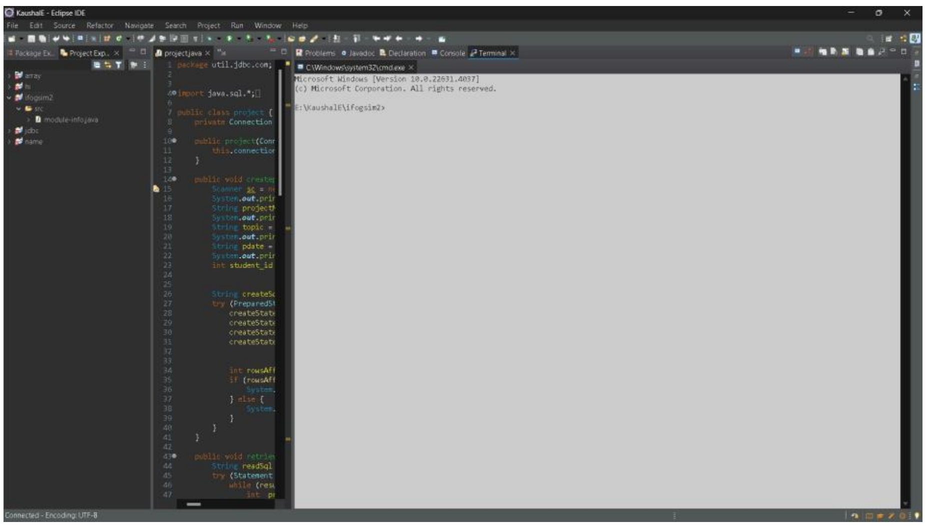
Clone the iFogSim2 Git repository to desired folder:git clone https://github.com/Cloudslab/iFogSimSelect "project from existing resources" from the
"File" drop-down menu

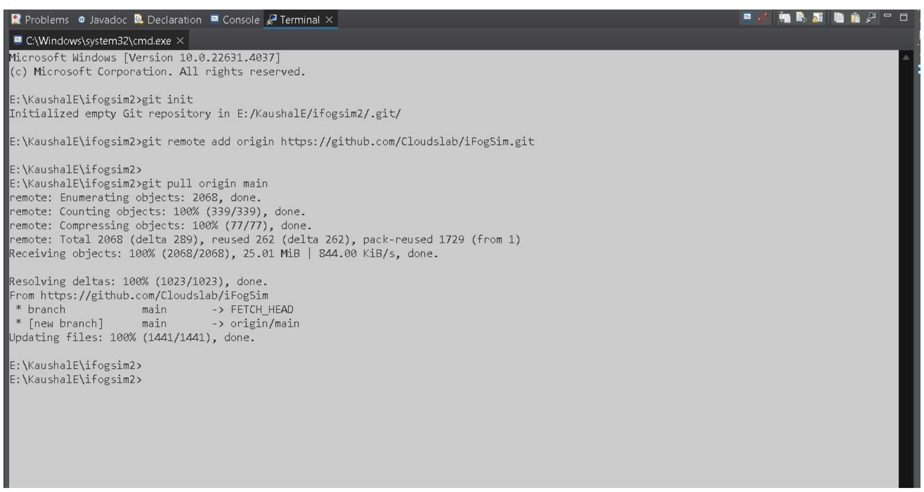
Verify the Java version

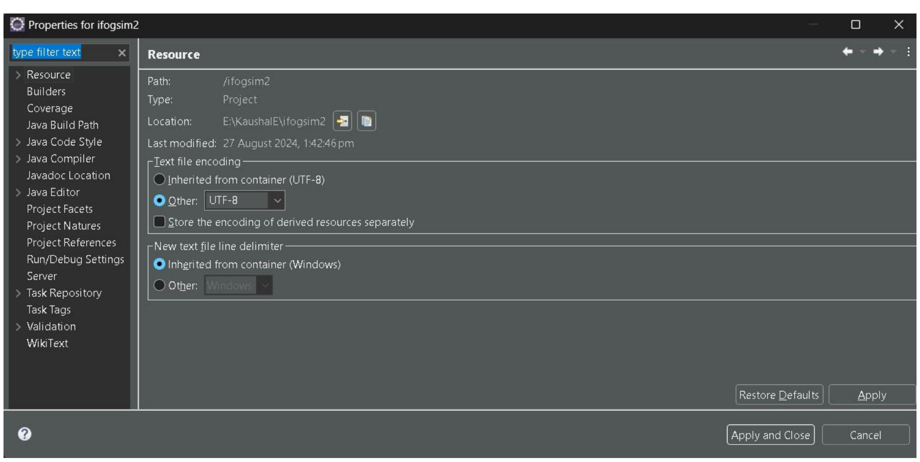
Verify the external libraries in the "JARs" Folder are added to the project

Run the example files (e.g.TranslationServiceFog_Clustering.java,CrowdSensing_Microservices_RandomMobility_Clustering.java) to get started

Output:

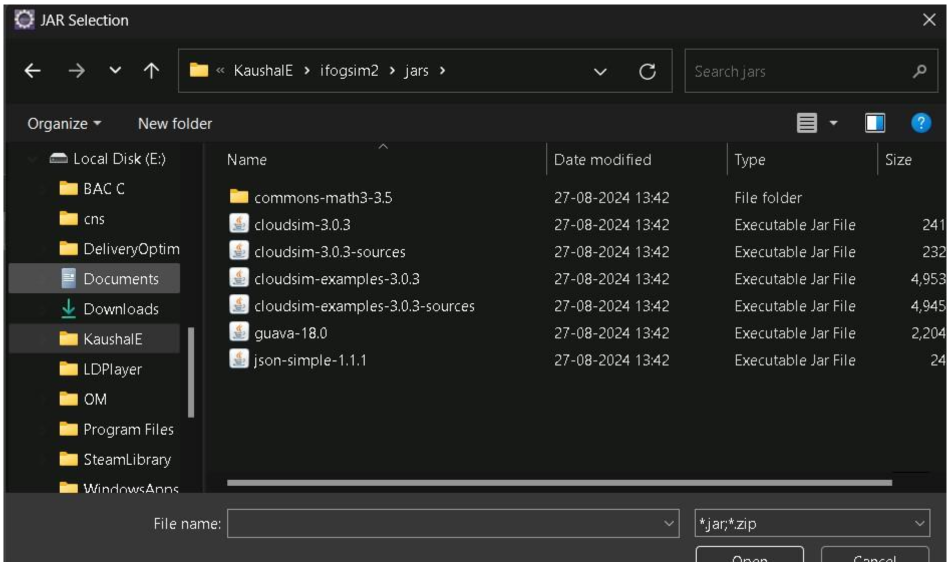

Problems
Javadoc
Declaration
Console
Terminal
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
E:\KaushalE\ifogsim2>

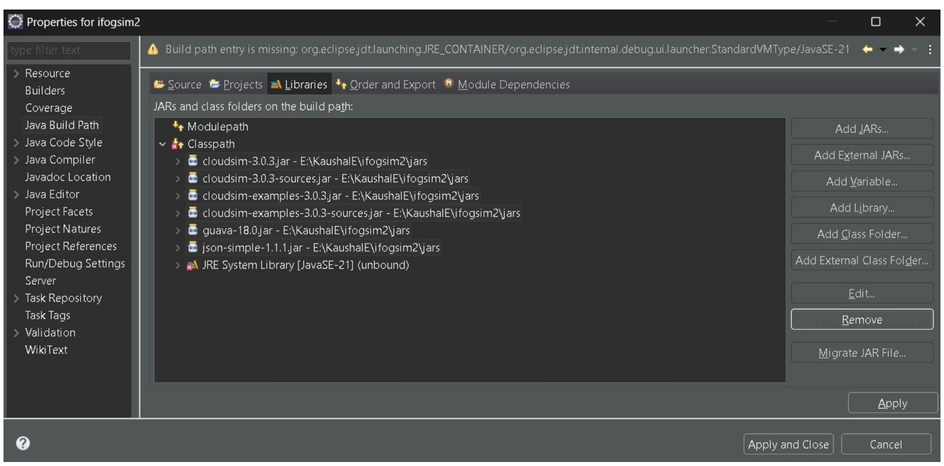

Problems
Javadoc
Declaration
Console
Terminal
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
E:\KaushalE\ifogsim2>git init
Initialized empty Git repository in E:/KaushalE/ifogsim2/.git/
E:\KaushalE\ifogsim2>git remote add origin https://github.com/Cloudslab/iFogSim.git
E:\KaushalE\ifogsim2>
E:\KaushalE\ifogsim2>git pull origin main
remote: Enumerating objects: 2068, done.
remote: Counting objects: 100% (339/339), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 2068 (delta 289), reused 262 (delta 262), pack-reused 1729 (from 1)
Receiving objects: 100% (2068/2068), 25.01 MiB | 844.00 KiB/s, done.
Resolving deltas: 100% (1023/1023), done.
From https://github.com/Cloudslab/iFogSim
* branch main -> FETCH_HEAD
* [new branch] main -> origin/main
Updating files: 100% (1441/1441), done.
E:\KaushalE\ifogsim2>
E:\KaushalE\ifogsim2>

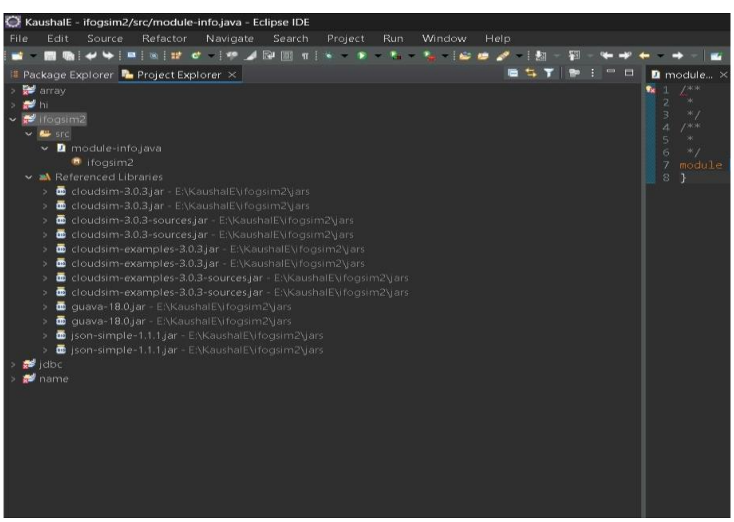

Properties for ifogsim2
type filter text
Resource
Builders
Coverage
Java Build Path
Java Code Style
Java Compiler
Javadoc Location
Java Editor
Project Facets
Project Natures
Project References
Run/Debug Settings
Server
Task Repository
Task Tags
Validation
WikiText
Resource
Path: /ifogsim2
Type: Project
Location: E:\KaushalE\ifogsim2
Last modified: 27 August 2024, 1:42:46 pm
Text file encoding
Inherited from container (UTF-8)
Other: UTF-8
Store the encoding of derived resources separately
New text file line delimiter
Inherited from container (Windows)
Other: Windows
Restore Defaults
Apply
Apply and Close
Cancel

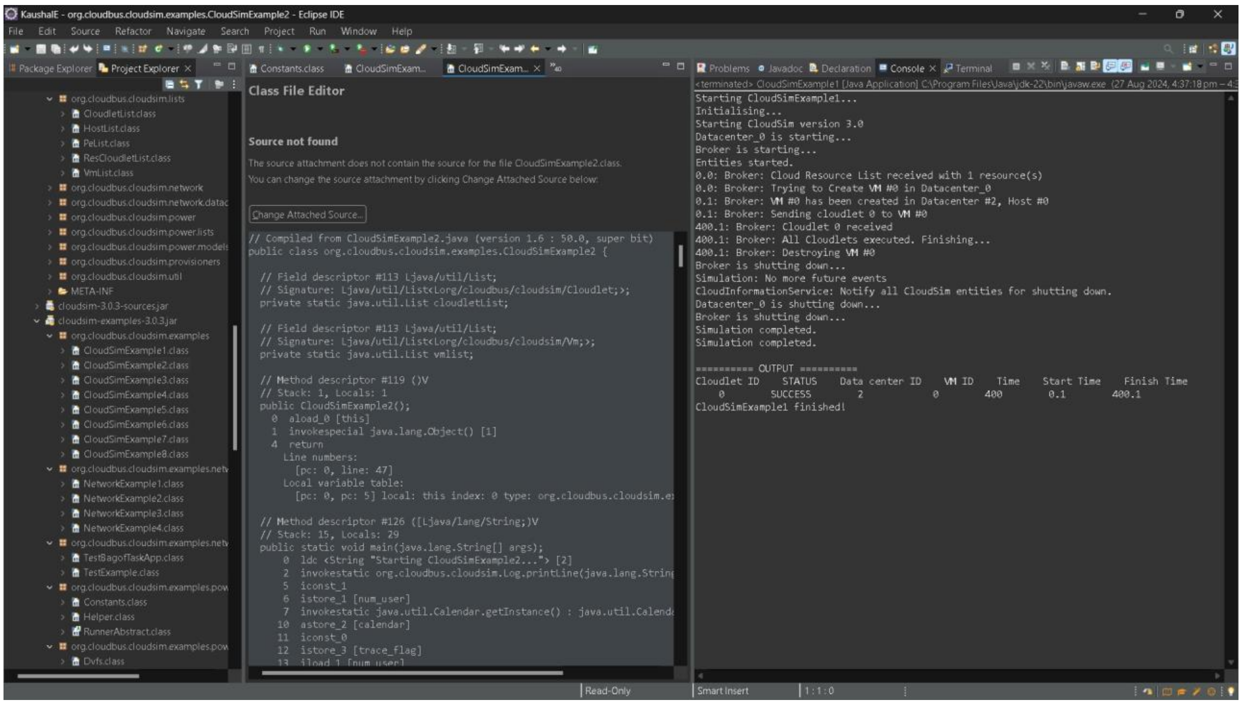
Jawahar Education Society's
A.C. Patil College of Engineering, Kharghar
Department of CSE(IoT-CS BC)

Subject : Edge and Fog Computing









Conclusion:

This Experiment focuses on the study and analysis of Edge and Fog Simulators, essential tools for evaluating the feasibility of complex IoT infrastructures before their practical implementation. With the rapid growth of IoT devices and the need for efficient data processing and storage, paradigms like Edge and Fog Computing have emerged to bring computation closer to the data source, reducing latency and improving performance. However, creating and testing these infrastructures can be costly and complex. Therefore, simulators like ENIGMA, based on SimGrid, offer a scalable and versatile solution for simulating large-scale environments, including features such as mobility support, microservice orchestration, and dynamic clustering. These tools allow researchers and developers to experiment with various configurations and analyze the impact on resources, making them invaluable for optimizing IoT deployments.