User Manual for Setting up of Edge-enabled IoTSim-Stream

1 Preliminary Setup

The following prerequisites are necessary for the application to run the Edge-enabled IoTSim-Stream.

- **Eclipse:** A flexible code editor for developing and debugging web and cloud applications, accessible across multiple platforms.
- **Java:** Used primarily for class implementation.
- **GitHub:** A web-based platform providing version control and collaboration tools for software development projects.
- **JRE (Java Runtime Environment):** Includes the Java Virtual Machine (JVM), responsible for interpreting Java bytecode and running Java applications, along with core libraries and other runtime components required for Java programs.

2 GitHub

The GitHub link where the code is hosted is listed below

https://github.com/GursharnKaurSoni/IoTSim-Stream

3 Directory Structure

To implement the edge layer in the IoTSim-Stream simulator, we introduced the *iotsimstream.edge* package, which contains most of the classes related to edge functionality. Additionally, several classes in other packages, such as *Simulation.Properties*, have been modified. Two new classes, *VMOffersEdgeDatacenter1* and *VMOffersEdgeDatacenter2*, were also added to support the VM flavors of edge devices. Fig. 1 depivts the screenshot of the directory structure for the implemented edge layer.

4 Setup of the repository in Eclipse

Step 1: Open Eclipse IDE $\longrightarrow$ File $\longrightarrow$ Open Projects From File System

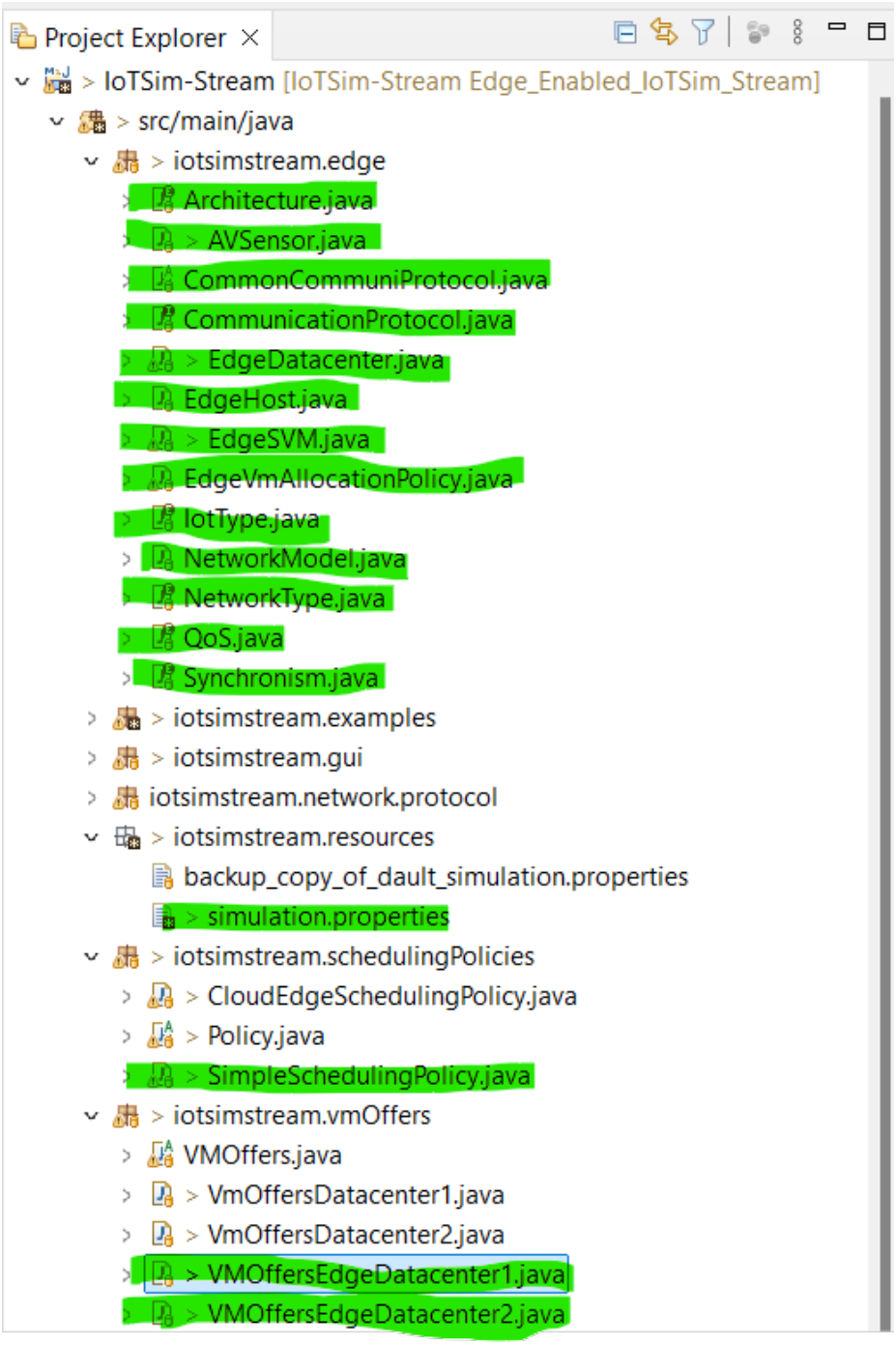
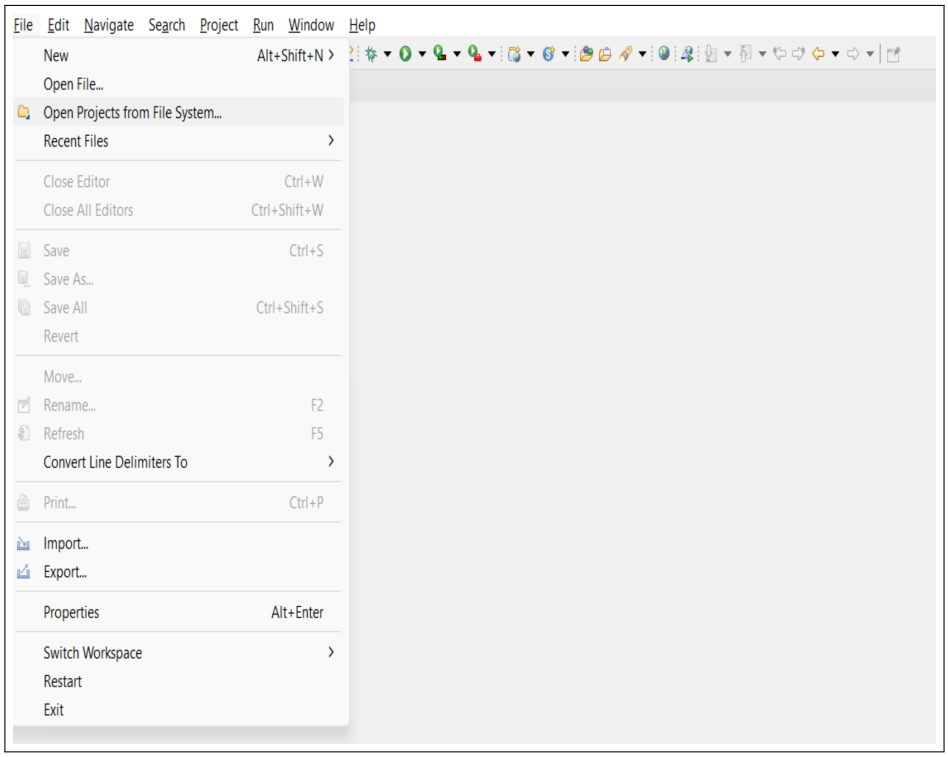
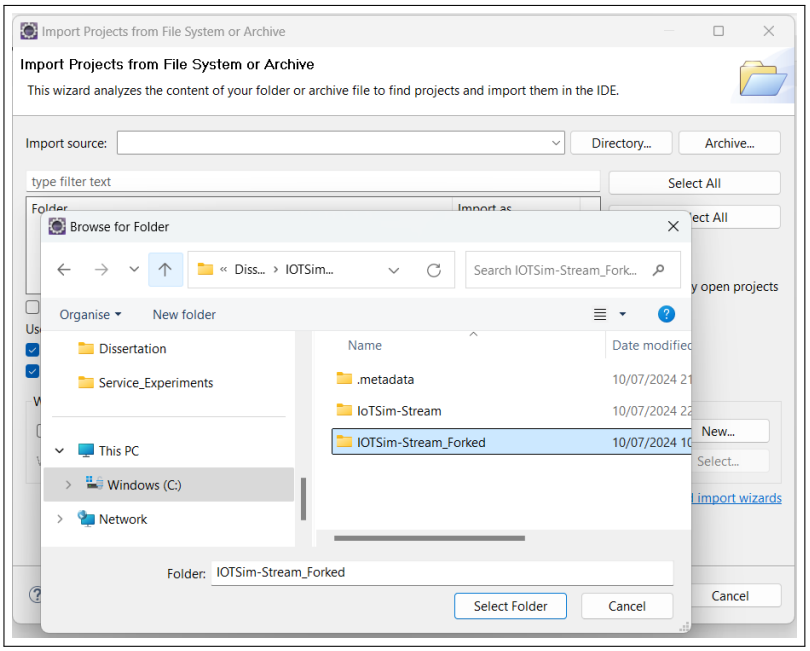


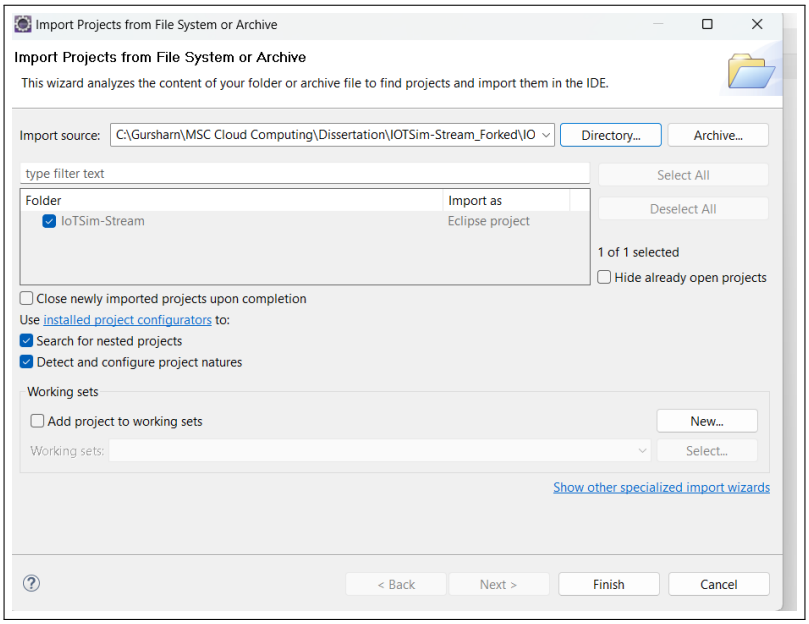
Fig. 1: Directory structure for the Edge-enabled IoTSim-Stream



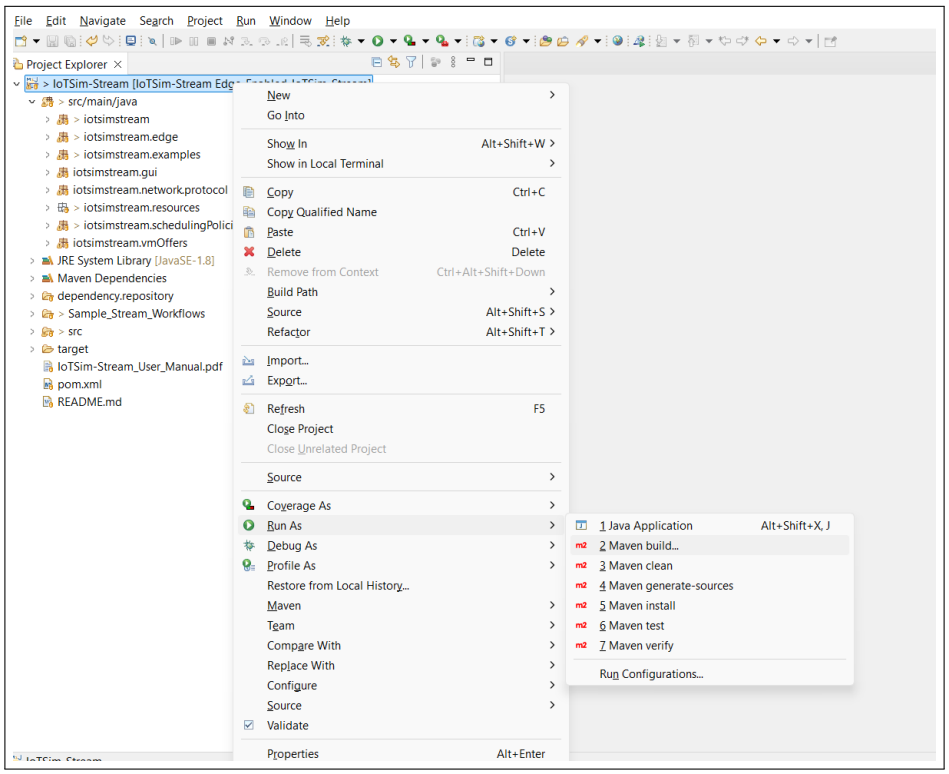
Step 2: Click on Directory and then select the folder corresponding to IoTSim-Stream_Forked which is the folder name that has been given by in which the repository has been cloned (*git clone* which has been mentioned in *section 1.* After that click on Select Folder.



Now click on the Finish



Step 3: Right-click on the Edge-enabled IoTSim-Stream project, find the "Run As" option, and then click on "Maven build."



Step 4: To process the graph application, we must first provide values for the keys defined in the Simulation.properties files for both the cloud and edge layers according to user requirements. Users can customize these values as needed as shown in Fig. 3. To know the details about each key refer to Table 1.In the scheduling.policy key user can provide either of the following policies.

- **SimpleSchedulingPolicy**
- **CloudEdgeSchedulingPolicy**

Step 5: In the following step, we will execute the linear graph, with its graphical and XML-based representations provided below.

Step 6: Right-click on the Edge-enabled IoTSim-Stream project, find the "Run As" option, and then click on "Maven install."

Once the mvn install command is executed, it triggers a full build of the project, including compiling the source code and running tests. The "Build Success" message shown in the screenshot below indicates that the mvn install command completed successfully.

```
simulation.time = 300
scheduling.policy = iotsimstream.schedulingPolicies.CloudEdgeSchedulingPolicy
dag.file = C:\\App1.xml
cloud.datacenter = 2
engine.network.bandwidth = 250
engine.network.latency = 0.03
datacenter.hosts#0 = 1000
vm.delay#0 = 20
vm.offers#0 = iotsimstream.vmOffers.VmOffersDatacenter1
host.cores#0 = 64
host.memory#0 = 144000
host.storage#0 = 14000000
core.mips#0 = 1000
internal.bandwidth#0 = 770
internal.latency#0 = 0.00077
external.bandwidth#0 = 170
external.latency#0 = 0.028

datacenter.hosts#1 = 1000
vm.delay#1 = 20
vm.offers#1 = iotsimstream.vmOffers.VmOffersDatacenter2
host.cores#1 = 64
host.memory#1 = 176000
host.storage#1 = 15000000
core.mips#1 = 2000
internal.bandwidth#1 = 780
internal.latency#1 = 0.00075
external.bandwidth#1 = 180
external.latency#1 = 0.026

# Edge Properties
edge.datacenter = 2
edge.vm.offers#0 = iotsimstream.vmOffers.VMOffersEdgeDatacenter1
edge.vm.delay#0=20
edge.hosts#0= 1000
edge.host.cores#0 = 20
edge.host.cores.mips#0=1000
edge.host.storage#0= 1000
edge.host.type#0= RASPBERRY_PI
edge.host.memory#0=100000
edge.external.bandwidth#0 = 180
```

Fig. 3: Simulation property file

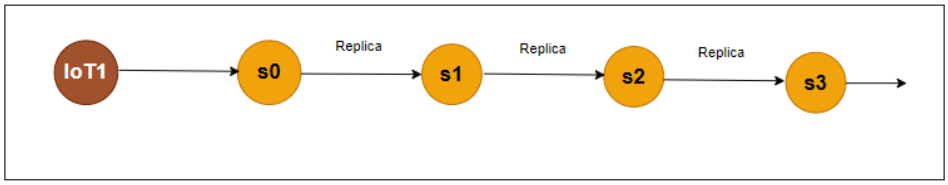
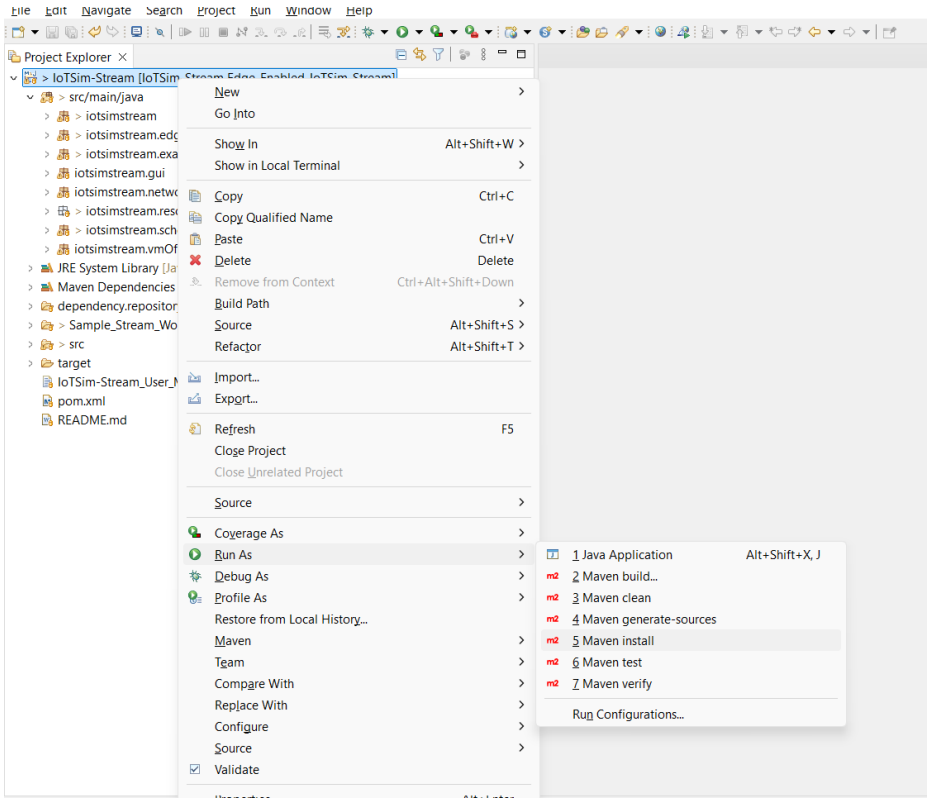
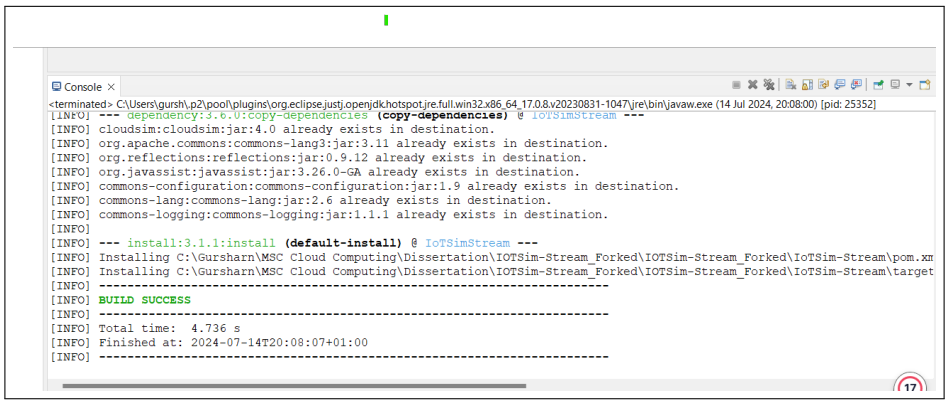


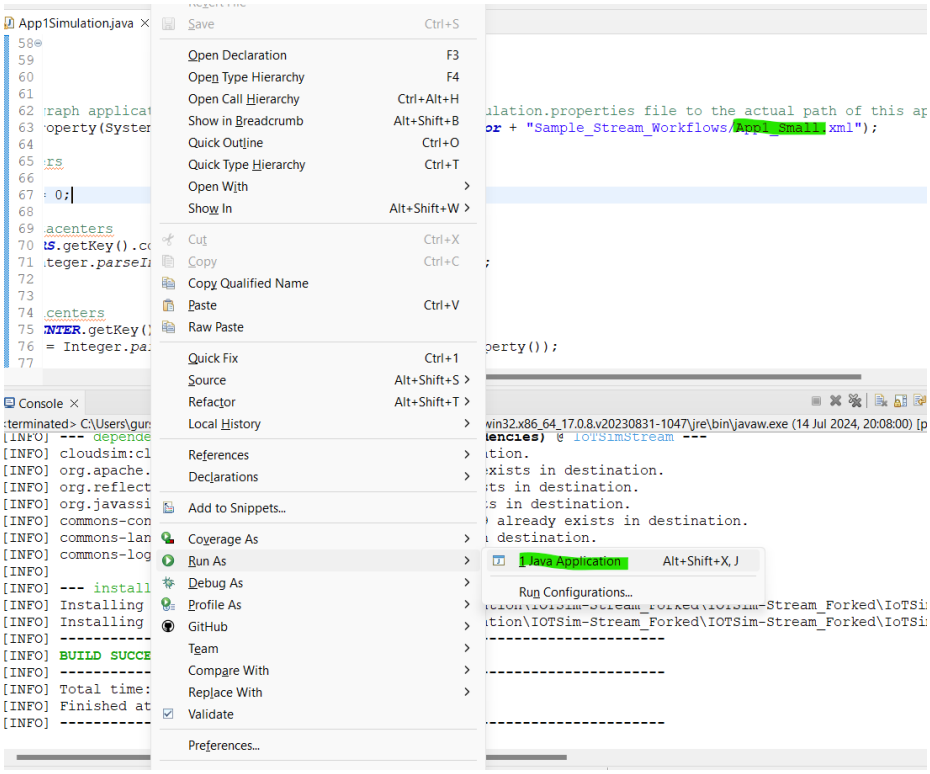
Table 1: Parameters for Edge environment in simulation properties

Parameter	**Description**	**Value**
simulation .time	User-defined requested time to run the simulation	*seconds*
scheduling.policy	Policy for the provision and scheduling VM	*CloudEdgeSchedulingPolicy*
edge.datacenter	Number of edge data centers required to run the simulator	*Integer*
edge.host.type#index	Type of edge device	*Raspberry PI, Udoo Board*
edge.vm.offers#index	Path of java class for vm offerings by edge datacenter	*packagename.classname*
edge.vm.delay#index	Average delay of VM boot time	*Integer*
edge.hosts#index	Number of host in the edge datacenter	*Integer*
edge.host.cores#index	Number of cores(PEs) available for each host	*Integer*
edge.host.cores.mips#index	MIPS of each core	*Integer*
edge.host.storage=index	Amount of storage available in each host	*Integer*
edge.host.memory#index	Amount of memory available in each host	*Integer (unit MB)*
edge.internal.bandwidth#index	Internal bandwidth available for each VM in each edge datacenter	*Integer (unit MB)*
edge.internal.latency#index	Network delay between VMs within edge datacenter	*Integer (unit MB/s)*
edge.external.bandwidth#index	External bandwidth available by edge datacenter for transferring stream to another edge datacenter	*Integer (unit MB/s)*
edge.external.latency#index	Network delay between two edge datacenter	*Integer (unit MB/s)*


File Edit Navigate Search Project Run Window Help
Project Explorer
IoTSim-Stream
src/main/java
iotsimstream
iotsimstream.gui
JRE System Library
Maven Dependencies
dependency.repositor
Sample_Stream_Wo
src
target
pom.xml
README.md
New
Go Into
Show In Alt+Shift+W
Show in Local Terminal
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Project
Source
Coverage As
Run As
Debug As
Profile As
Restore from Local History...
Maven
Team
Compare With
Replace With
Configure
Source
Validate
1 Java Application Alt+Shift+X, J
2 Maven build...
3 Maven clean
4 Maven generate-sources
5 Maven install
6 Maven test
7 Maven verify
Run Configurations...

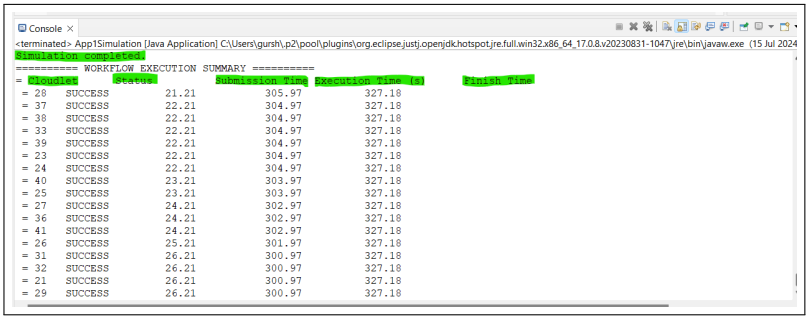

Console
<terminated> C:\Users\gursh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (14 Jul 2024, 20:08:00) [pid: 25352]
[INFO] --- dependency:3.6.0:copy-dependencies (copy-dependencies) @ IoTSimStream ---
[INFO] cloudsim:cloudsim:jar:4.0 already exists in destination.
[INFO] org.apache.commons:commons-lang3:jar:3.11 already exists in destination.
[INFO] org.reflections:reflections:jar:0.9.12 already exists in destination.
[INFO] org.javassist:javassist:jar:3.26.0-GA already exists in destination.
[INFO] commons-configuration:commons-configuration:jar:1.9 already exists in destination.
[INFO] commons-lang:commons-lang:jar:2.6 already exists in destination.
[INFO] commons-logging:commons-logging:jar:1.1.1 already exists in destination.
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ IoTSimStream ---
[INFO] Installing C:\Gursharn\MSC Cloud Computing\Dissertation\IOTSim-Stream_Forked\IOTSim-Stream_Forked\IoTSim-Stream\pom.xm
[INFO] Installing C:\Gursharn\MSC Cloud Computing\Dissertation\IOTSim-Stream_Forked\IOTSim-Stream_Forked\IoTSim-Stream\target
[INFO] --
[INFO] BUILD SUCCESS
[INFO] --
[INFO] Total time: 4.736 s
[INFO] Finished at: 2024-07-14T20:08:07+01:00
[INFO] --

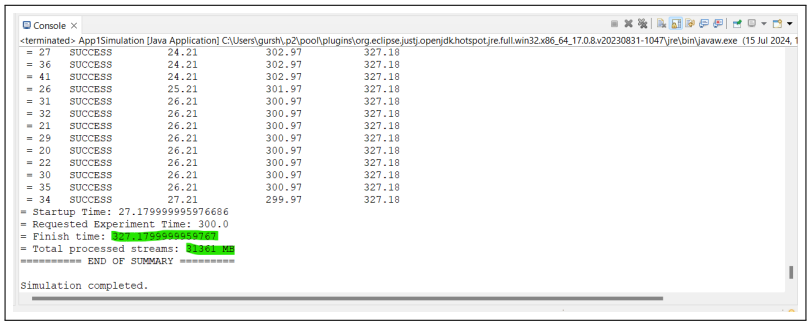
Step 7: To process the graph application, navigate to the *iotsimstream.examples* package. For example, in the App1Simulation.java file, modify the DAG file name (XML file) located in the Sample_Stream_Workflows directory within the runSimulation method to specify the file you want to process. In the example below, we have processed the App1_Small.xml file. After executing *App1Simulation.java*



as a Java file, the simulator will begin running, and you can view the output of the execution in the console section of Eclipse.

Similarly, by modifying the DAG file name (XML file), in the *App1Simulation.java* users can execute the graph application on both the cloud and/or edge layers. This customization involves adjusting policies and parameters related to the cloud and edge layers in the *simulation.properties* file.


Console
<terminated> App1Simulation [Java Application] C:\Users\gursh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (15 Jul 2024
Simulation completed.
========== WORKFLOW EXECUTION SUMMARY ==========
= Cloudlet Status Submission Time Execution Time (s) Finish Time
= 28 SUCCESS 21.21 305.97 327.18
= 37 SUCCESS 22.21 304.97 327.18
= 38 SUCCESS 22.21 304.97 327.18
= 33 SUCCESS 22.21 304.97 327.18
= 39 SUCCESS 22.21 304.97 327.18
= 23 SUCCESS 22.21 304.97 327.18
= 24 SUCCESS 22.21 304.97 327.18
= 40 SUCCESS 23.21 303.97 327.18
= 25 SUCCESS 23.21 303.97 327.18
= 27 SUCCESS 24.21 302.97 327.18
= 36 SUCCESS 24.21 302.97 327.18
= 41 SUCCESS 24.21 302.97 327.18
= 26 SUCCESS 25.21 301.97 327.18
= 31 SUCCESS 26.21 300.97 327.18
= 32 SUCCESS 26.21 300.97 327.18
= 21 SUCCESS 26.21 300.97 327.18
= 29 SUCCESS 26.21 300.97 327.18


Console
<terminated> App1Simulation [Java Application] C:\Users\gursh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (15 Jul 2024, 1
= 27 SUCCESS 24.21 302.97 327.18
= 36 SUCCESS 24.21 302.97 327.18
= 41 SUCCESS 24.21 302.97 327.18
= 26 SUCCESS 25.21 301.97 327.18
= 31 SUCCESS 26.21 300.97 327.18
= 32 SUCCESS 26.21 300.97 327.18
= 21 SUCCESS 26.21 300.97 327.18
= 29 SUCCESS 26.21 300.97 327.18
= 20 SUCCESS 26.21 300.97 327.18
= 22 SUCCESS 26.21 300.97 327.18
= 30 SUCCESS 26.21 300.97 327.18
= 35 SUCCESS 26.21 300.97 327.18
= 34 SUCCESS 27.21 299.97 327.18
= Startup Time: 27.179999995976686
= Requested Experiment Time: 300.0
= Finish time: 327.17999999557674
= Total processed streams: 31361 MB
========== END OF SUMMARY ==========
Simulation completed.