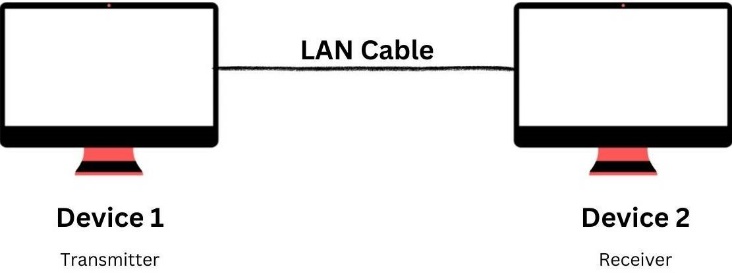
**Practical No 4**

**1.Aim: -** WAP to simulate Point to Point(simple) topology.

**2.Objective: -**

* Understanding of Point-to-Point topology
* Implementation of Point-to-Point topology using NS3

**3.Theory: -**

Point to Point (P2P) topology is a network topology that connects two nodes or devices (hub, router, switch, etc.) directly using a private, dedicated line (LAN cable). In this topology, one of the connected nodes serves as the transmitter, while the other acts as the receiver.

**Advantages of Point-to-Point topology:**

* Point to Point topology provides a dedicated communication channel between two nodes, ensuring faster data transfer rates.
* Since only two devices are connected, identifying and troubleshooting issues is simpler than any other network topology, as each node or connection can be tested individually.
* P2P topology offers increased security as data is transmitted directly between two nodes, reducing the chances of data interception or theft.

**Disadvantages of Point-to-Point topology:**

* With P2P networks, adding new devices means establishing a separate link between each new device and the existing network, which can be time-consuming and expensive.
* When a link fails, or a device goes offline, it can cause problems as there are no alternative routes through which data can be transmitted.
* P2P topology is unsuitable for large areas due to the need for a direct connection.

**Steps of Simulation:**

1. Set of libraries
2. Define namespace
3. Take logs
4. Number of Computers.
5. Choose the way of Communication
6. Install the way of communication on nodes
7. Ask to Follow the rules.
8. Assign IP address to communicate
9. Assign the type of address
10. Create a type of Server
11. Install the node as server
12. Create a client
13. Install a client

**4.Program: -**

#include "ns3/core-module.h"

#include "ns3/network-module.h"

#include "ns3/internet-module.h"

#include "ns3/point-to-point-module.h"

#include "ns3/applications-module.h"

#include "ns3/netanim-module.h"

using namespace ns3;

NS\_LOG\_COMPONENT\_DEFINE ("FirstScriptExample");

int

main (int argc, char \*argv[])

{

CommandLine cmd (\_\_FILE\_\_);

cmd.Parse (argc, argv);

Time::SetResolution (Time::NS);

LogComponentEnable ("UdpEchoClientApplication", LOG\_LEVEL\_INFO);

LogComponentEnable ("UdpEchoServerApplication", LOG\_LEVEL\_INFO);

NodeContainer nodes;

nodes.Create (2);

PointToPointHelper pointToPoint;

pointToPoint.SetDeviceAttribute ("DataRate", StringValue ("5Mbps"));

pointToPoint.SetChannelAttribute ("Delay", StringValue ("2ms"));

NetDeviceContainer devices;

devices = pointToPoint.Install (nodes);

InternetStackHelper stack;

stack.Install (nodes);

Ipv4AddressHelper address;

address.SetBase ("10.1.1.0", "255.255.255.0");

Ipv4InterfaceContainer interfaces = address.Assign (devices);

UdpEchoServerHelper echoServer (9);

ApplicationContainer serverApps = echoServer.Install (nodes.Get (1));

serverApps.Start (Seconds (1.0));

serverApps.Stop (Seconds (10.0));

UdpEchoClientHelper echoClient (interfaces.GetAddress (1), 15);

echoClient.SetAttribute ("MaxPackets", UintegerValue (1));

echoClient.SetAttribute ("Interval", TimeValue (Seconds (1.0)));

echoClient.SetAttribute ("PacketSize", UintegerValue (1024));

ApplicationContainer clientApps = echoClient.Install (nodes.Get (0));

clientApps.Start (Seconds (2.0));

clientApps.Stop (Seconds (10.0));

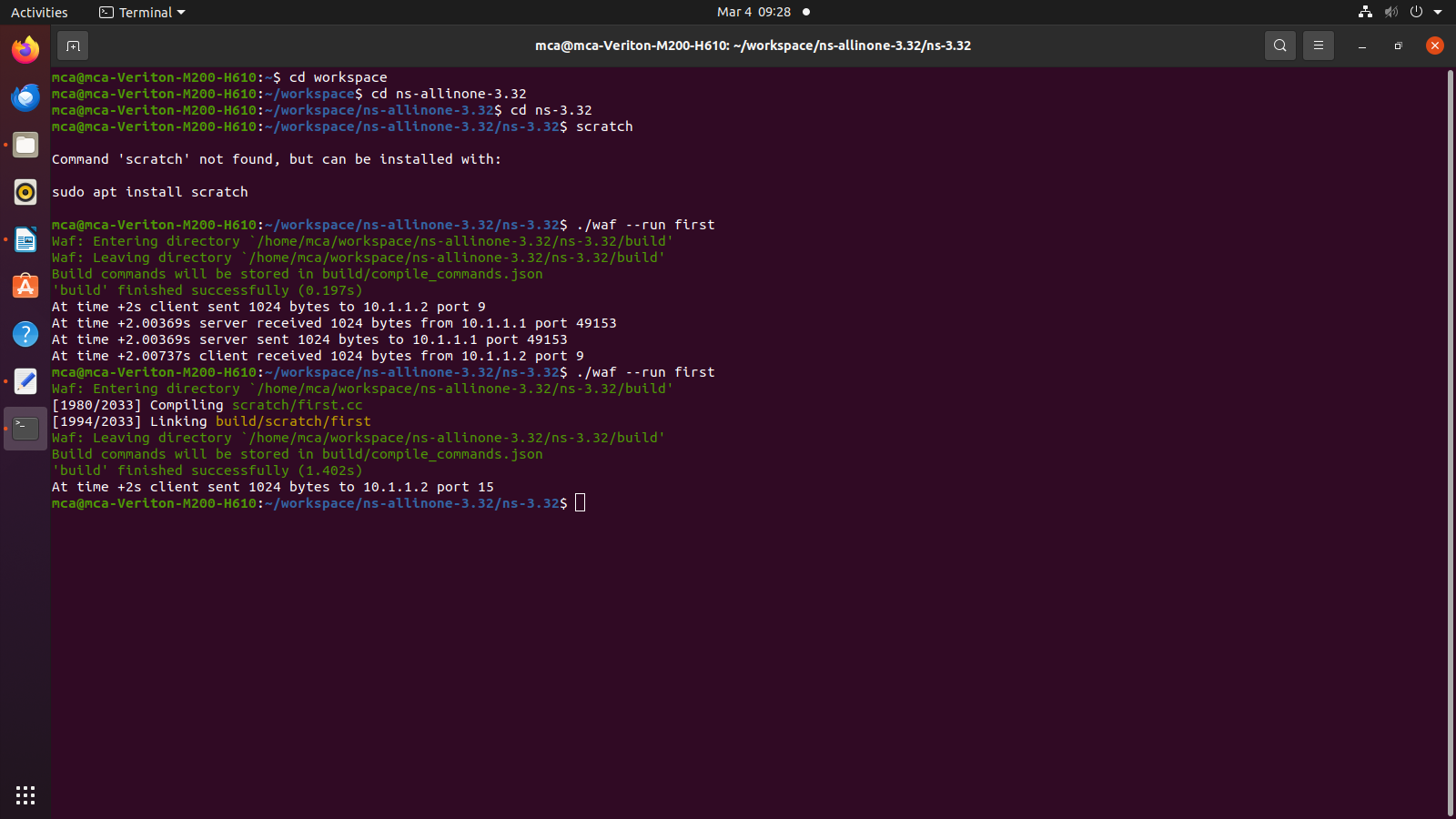
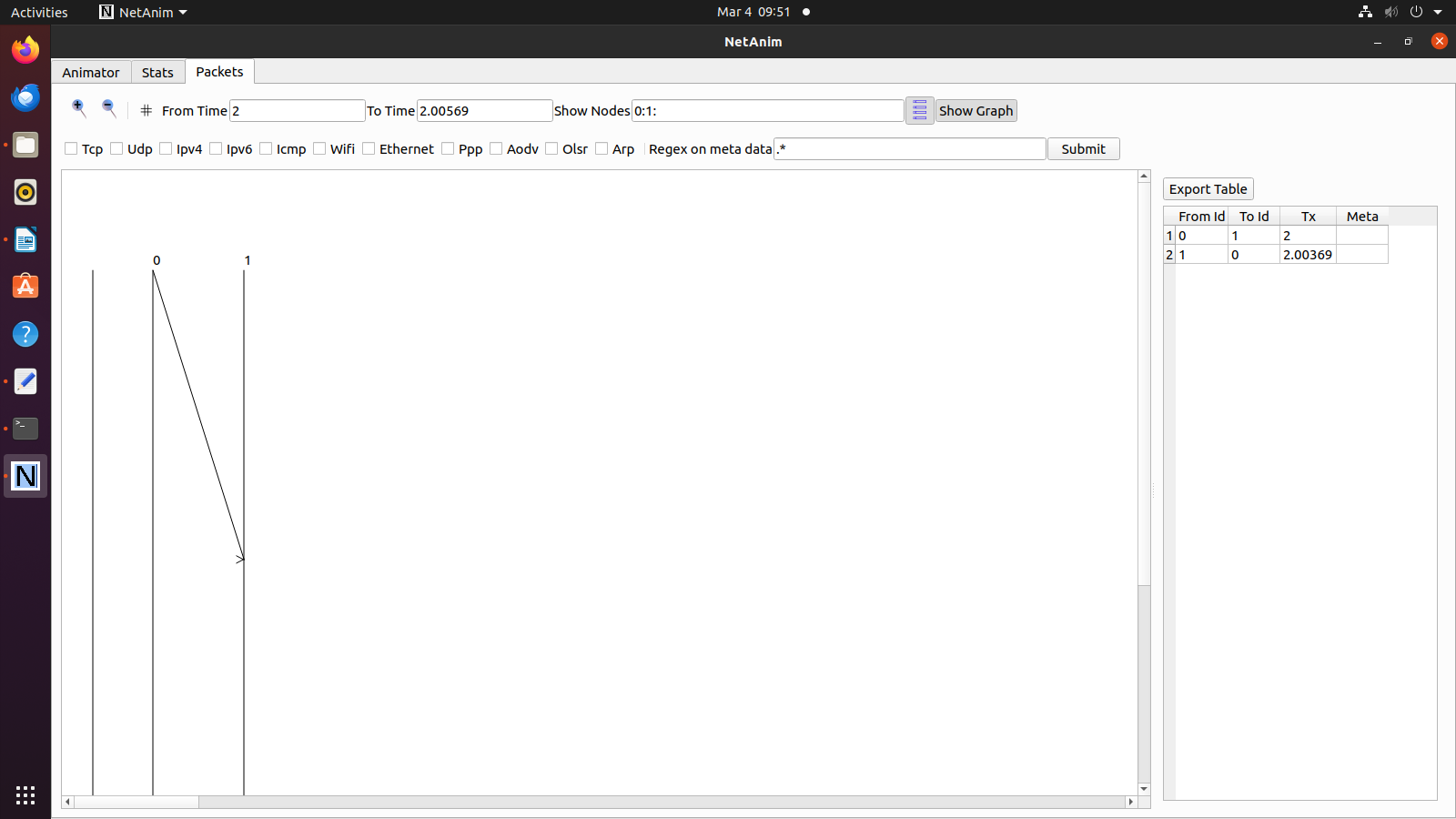
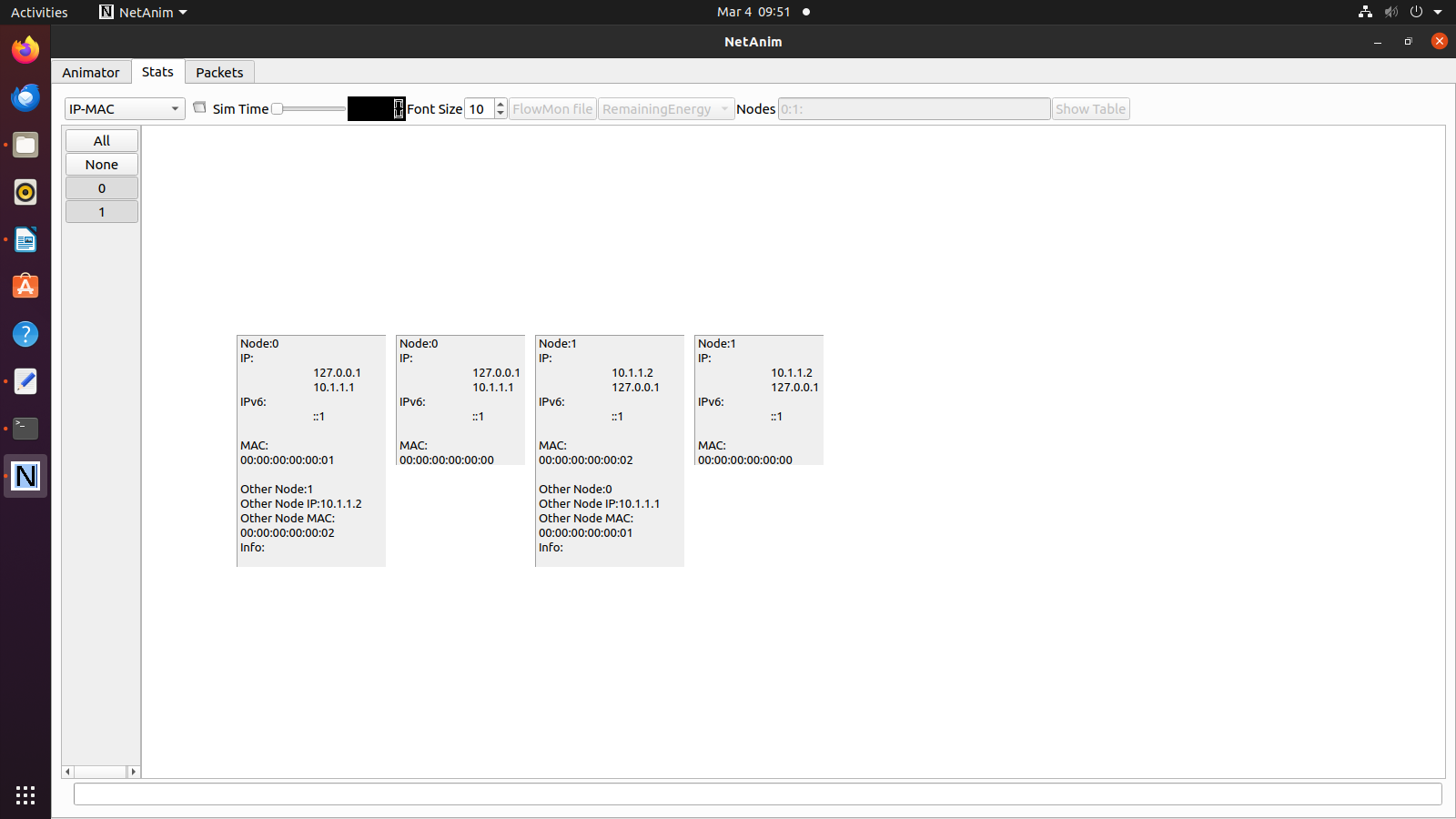
AnimationInterface anim("animT.xml");

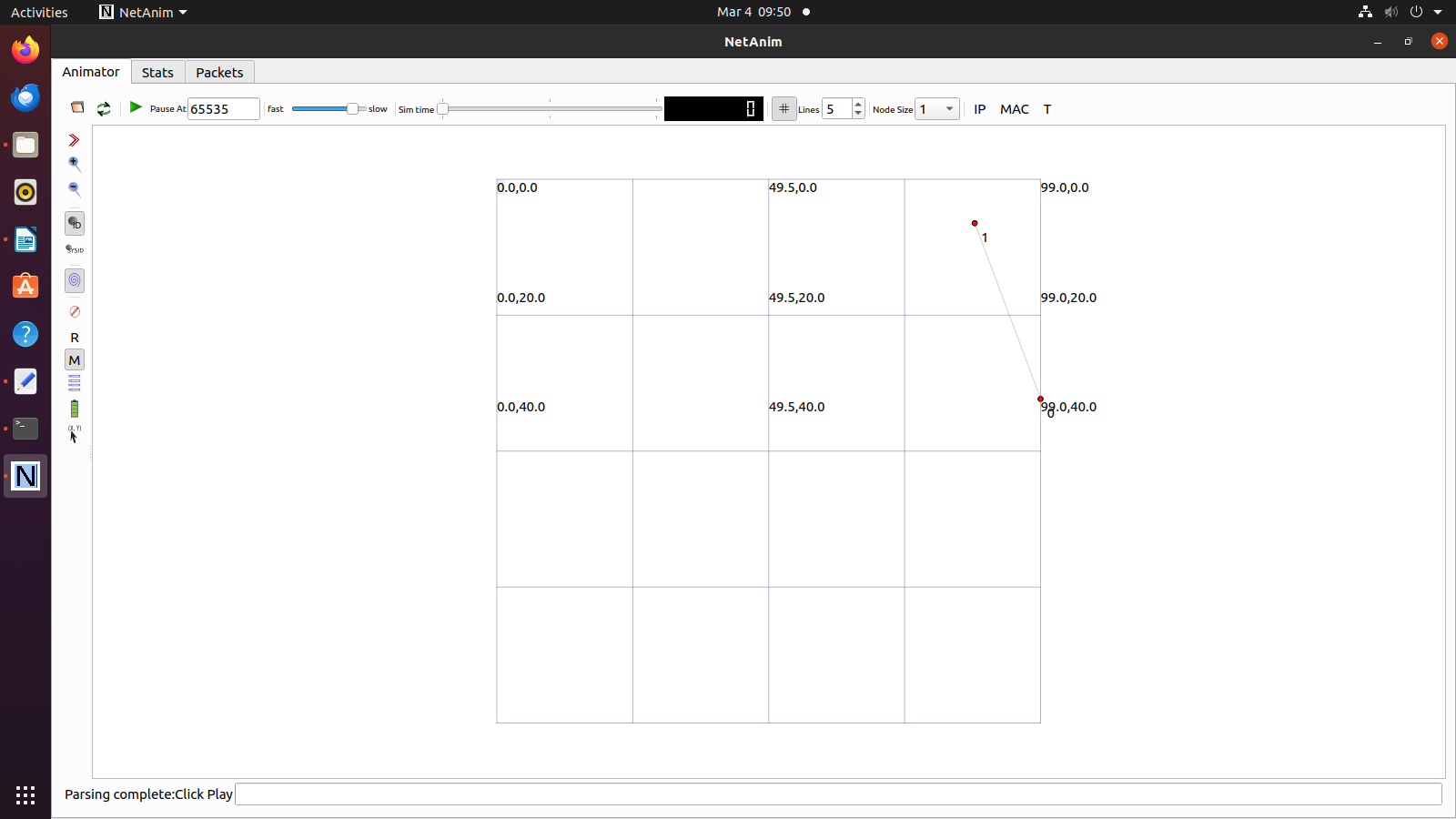
Simulator::Run ();

Simulator::Destroy ();

return 0;

}

**5. Output: -**



**6. Conclusion: -**

Successfully implemented Point-to-Point topology using NS3.