Paper ID: E&TC31

COMPARISON STUDY OF NS-2 AND EXATA SOFTWARES FOR WSN

Dipti D. Khot
M.tech Student ETC Dept.
R. I. T. Sakharale, Maharashtra

Amol Patole
M.tech Student ETC Dept.
R. I. T. Sakharale, Maharashtra

Dr. J.S. Awati
Asst. Prof. ETC Dept.
R. I. T. Sakharale, Maharashtra

Abstract— In recent years wireless sensor network (WSN) is a more popular technique. Wireless technology has many applications in various fields like agricultural, home monitoring, etc. The wireless sensor network faces issues in terms of practical implementation. For WSN simulation many simulators are available like NS-2, NS-3, OMNET++, netsim etc. The main objective of this paper is the comparison study of NS-2 and EXATA simulators for WSN. Both simulators are used in WSN. NS-2 is only a simulator and EXATA is simulator as well as emulator. EXATA is a real-time based, external hardware can be connected.

Keywords- Wireless Sensor Network (WSN) , simulators,NS-2, EXATA, Real-time.

I. INTRODUCTION

In recent years, wsn becomes more popular area for researchers. The network consists of many low-cost sensor nodes and one or more base stations (BS) which gather the sensed data for further processing. nowadays various researchers are coming up with new protocols, new techniques and algorithms in wsn. sometimes these algorithm or new technique or new protocol may not be possible in real time because of time consuming, more complex and costly also. So , to overcome this problem simulators , emulators are the best solution. It gives the better analysis and gives performance of the proposed algorithm or protocol. Various simulators are available for wsn. Ns-2 and exta is mostly used simulators. Ns-2 is a simulator and EXATA is a simulator and emulator as well.

II. OBJECTIVES

- Comparision of ns-2 and EXATA.

III. WSN SIMULATION TOOLS

To develop new technology or new algorithm or protocol various simulation tools are available.

1. NS-2

NS-2 is an event driven packet level network simulator. NS-2 software is used to evaluate the performance of existing network protocols, to evaluate new network protocols before use. Using this software we can create our own protocol. Basically NS is an object- oriented TCL script. It is a set of protocol object and different network. NS-2 is written in two languages: C++ and object oriented tool command language (OTcl). C++ is used because of detailed simulations require for programming language ,easy to configuration, to increase the efficiency of simulation,to provide details of the protocols and their operation,to reduce packet and event processing time. OTcl script set ups the network topology within nodes, initiates an event scheduler and it tell when to start and stop transmitting packets through event scheduler to traffic sources.

OTcl is easy for coding, but it runs slowly. In case of C++, it is slower for coding but runs fastly.

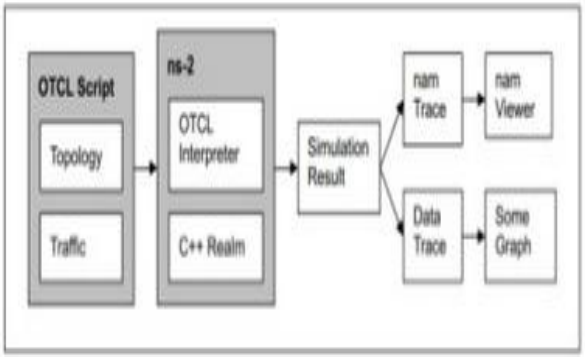

Fig1. NS-2 architecture

Simulation results are stored as trace files. Results we can store by 2 ways:

Paper ID: E&TC31

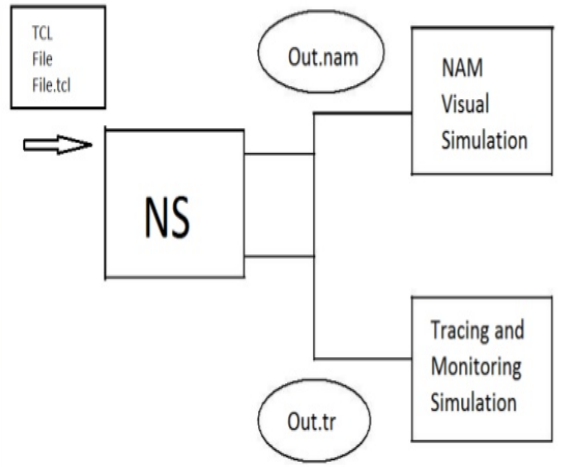

Fig2. NS-2 visualization tools

1. A NAM trace file (file.nam) (Network Animator Tool)
2. A Trace files (file.tr) (XGraph) or (TraceGraph).

Four steps for tcl script program to create NS-2 program:

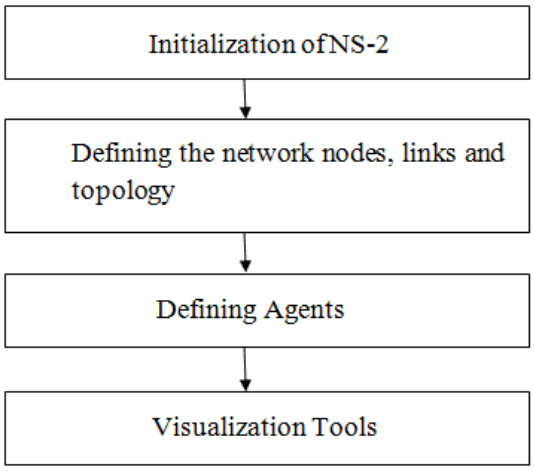

Fig3. Algorithm for TCL script

1.For initialization of ns-2 we have to create new simulator object:
set ns [new Simulator]
2.Defining the network nodes,links and topology:
Node creation: In NS-2, nodes are connected with the help of links. Nodes and links have their own properties with them. Agents are connected or linked with the nodes and for generating the different packets agents are responsible.

Set n0 [$ns node]
Set n1 [$ns node]

Two nodes are created n0 and n1.

- Node links:
  - Simplex link- simplex link is used for creating unidirectional link between two nodes.

$ns simplex-link $n0 $n1 <bandwidth> <delay> <queue_type>

  - Duplex link- duplex link is used for creating bidirectional link between two nodes

$ns duplex-link $n0 $n1 <bandwidth> <delay> <queue_type>

3.Defining Network agents:

Two types of NS-2 agent used for creating or destroying a packet are:
1. Routing agent: Routing agents create and receives routing control packets and commands routing protocols to act accordingly.
2. Transport layer agent: it controls congestion and reliability of a data flow based on a underlying transport layer protocol(e.g UDP or TCP).

- Two types of agents are present in ns-

1.UDP-set udp0 [new agent/udp]:Connectionless transport level protocol or not needed setup for data transfer.
2. TCP – set tcp [new agent/ tcp]: Tcp is a connection oriented reliable protocol.
Objectives of tcp:It avoids congestion at the network.Adapt the transmission rate of packets to the available bandwidth.Create a reliable connection by retransmitting lost packet.

4.Visualization tools:

NAM:

It is a network animator to display output. It gives the visual understanding of the network, also executed directly from the Tcl script.nam have functions like play, stop, pause etc.NAM displays the packets on each link, traffic generated between packets, throughputs.

Paper ID: E&TC31

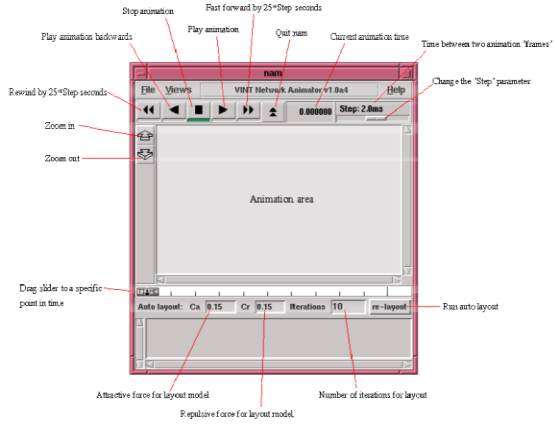

Fig4. NAM output window

2. EXATA

It is simulator as well as emulator software. EXATA can also be connected to systems with real applications. Using this software we cannot create our own protocol, we can use existing protocol. This software is easy to understand. EXATA is a wireless emulator that lets you evaluate on-the-move communication networks faster and with more real-ism than any other emulator.EXATA creates a digital network replica that interfaces with real networks in real time, using real applications.EXATA is a digital representation of networks. EXATA emulation is not a substitute for existing modeling and simulation; it is a whole new category of evaluation/development tool that does what the other products were not designed to do.

EXATA enables you to digitally represent your entire network - devices, software, transmitters, antennas, terrain effects, atmospheric effects, and human interaction effects. You can now represent every variable that will affect the performance of your real network in an EXATA Software Virtual Network.

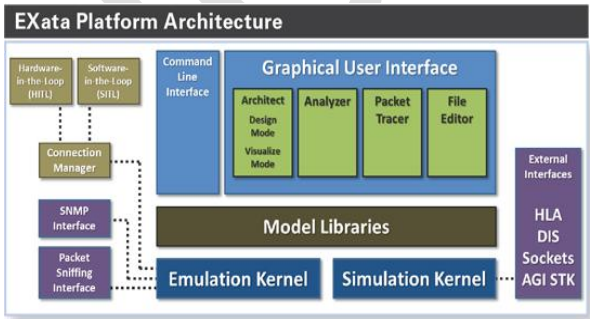

Fig5. EXATA architecture

IV. RESULTS

- Ns-2: for simulation here 10 nodes are considered.where node 3 is a server where all data will collect.

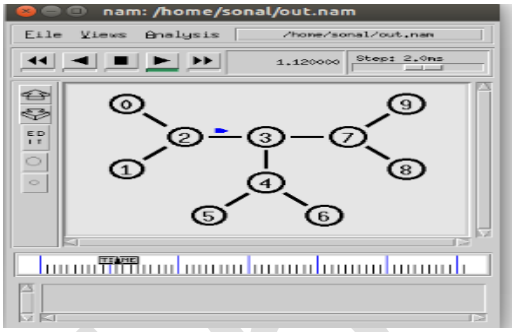

Fig6. Scenario of NS-2 simulation

- EXATA: scenario consist of nodes , links and wireless network. Scenario shown in following figure.

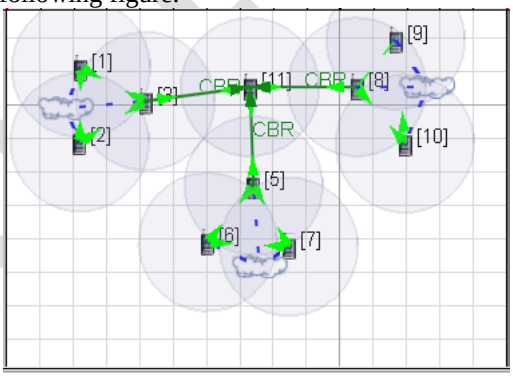

Fig 7. Scenario of EXATA

Comparison between NS-2 and EXATA

Table No.1

Sr. No.	NS-2	EXATA
1	Open source	Licenced
2	Operating on Ubuntu	Operating on windows and linux
3	Only simulator	Simulator and emulator
4	Only for testing	Real time interfacing
5	Complex	More user friendly
6	Easy for algorithm development	Complex for algorithm development
7	on-the-move communication networks slower than EXATA tool.	on-the-move communication networks faster and with more realism than NS-2.
8	2D visualization	3D visualization

9	Complicated to link number of nodes	Easy to link number of nodes
10	Research tool	Research as well as industry tool
11	Weather enverionments can not tested	Weather enverionments can be tested
12	Cyber effects and impacts can nottested	Cyber effects and impacts can be tested

CONCLUSION

The NS-2 and EXATA both are the network simulators . NS-2 provides only a simulator while EXATA has simulator as well as emulator. EXATA is a real-time application based simulator where external hardware can be join. NS-2 is a open source tool where EXATA is a liecened. Therefore NS-2 simulator can be only used in research and teaching field while EXATA can be used for developing industrial applications. Table 1 gives comparison between these simulators. Considering various parameters taken into consideration for performance comparison of various WSN simulators, NS-2 and EXATA Simulator can be considered best simulators to carry out all sort of research in area of Wireless Sensor Networks.

REFERENCES

[1]Anand Nayyar, Rajeshwar Singh, " A Comprehensive Review of Simulation Tools for Wireless Sensor Networks (WSNs) ", Journal of Wireless Networking and Communications, pp 19-47 ,2015.

[2] Sundani, H., Li, H., Devabhaktuni, V. K., Alam, M., & Bhattacharya, P. (2011). Wireless sensor network simulators a survey and comparisons. International Journal of Computer Networks (IJCN), 2(5), 249-265.

[3] Imran, M., Said, A. M., & Hasbullah, H. (2010, June). A survey of simulators, emulators and testbeds for wireless sensor networks. In Information Technology (ITSim), 2010 International Symposium in (Vol. 2, pp. 897-902). IEEE.

[4] Varga, A., & Hornig, R. (2008, March). An overview of the OMNeT++ simulation environment. In Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops (p. 60). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[5] Discrete Event Simulation: URLhttp://en.wikipedia.org/wiki/Discrete_event_simulation#Network_simulators". Accessed on 17 August, 2014.