Writing a use case for simulation: Writing graph and event file for use case

Pham Nguyen Quang

May 16, 2017

1 Introduction

1.1 Context

Network simulation is part of the internship project from Bui Quang Minh, Pham Nguyen Quang, Nguyen Chi Thanh and Vo Huynh Quang Kiet at Frankfrut university of applied sciences.This project is about building a generic network optimizing system using evolutionary algorithm (EA). The network simulation take graph and event file as inupt for the simulation. There are specific format and attributes from the graph and events file which can be read. This document will describe in detail how to write a graph and an event file (together these are called use case) for the network simulation. This document will be changed in the future when new features are added.

1.2 Requirements

- Basic knowledge on networking.
- Basic knowledge on xml, graphml and json file format.
- Optional: Basic programming skill.

2 Writing a use case

2.1 General requirement

- A use case will consist of one graph file and at least one event file.
- Graph file is written in graphml format and follow the general graphml format standard.
- Event file can be written in either json or xml format.
- An event file must be written using the same attribute list that are exists in the graph file (where the attributes applies) e.g: When the node id in the graph file are n1,n2,n3 and n5, the node id in the events that are presented in the event file must be any 2 nodes with the exact naming from n1, n2, n3 or n5.

2.2 Writing a graph file

A graph file is written in graphml format and follow the graphml format standard, an example of a graphml file that is used with this simulation is found in the use case section. Attribute for graph edges and nodes must obey the exact properties of the following tables

- Table 1: Edge Attribute list

Name	Attribute name	Attribute type	Attribute data	example	mandatory
Source Node	source	String	Node ID of the source Node	n1	yes
Source Data Rate	DataRateSrc	String	Data Rate of the source	1Mbps	no
Source MTU	MTUSrc	Double	MTU of the source	1500	no
Source Interframe Gap	InterframeGapSrc	String	Interframe Gap of the source	96ns	no
Destination Node	target	String	Node ID of the destination Node	n5	yes
Destination Data Rate	DataRateDst	String	Data Rate of the destination	500kbps	no
Destination MTU	MTUDst	Double	MTU of the destination	1500	no
Destination Interframe Gap	InterframeGapDst	String	Interframe Gap of the destination	38ns	no
Delay	Delay	String	Delay of the network	10ms	no

- Note:
  - The item listed on the table above will be changed as the project progress.
  - There are no special requirement for the node, this may change in the future as the project progresses.

2.3 Writing event files

Event file should be written in either json or xml file, the attributes for the event must follow the exact naming and properties intended as stated in the following table.

- Table 2: Event Attribute list

Name	Attribute Nane Tag	Attribute data	Example	Manatory
Source Node	srcNode	Node ID of the source Node	n1	yes
Destination Node	dstNode	Node ID of the destination Node	n2	yes
Protocol	protocol	Transmission protocol	tcp, ssh	no
Data Transfer	datatransfer	The amount of data transfer in bytes	12345	yes
Start time	start	Time (in seconds) to start the event	54321	yes

- Note:
  - The item listed on the table above will be changed as the project progresses.
  - Transfer protocol can be "tcp", "udp", or "udptest" (Automated Simulations document); more protocol (ssh, http...) will be added as the time they are implemented.

2.3.1 Writing Json event file

Json event file, apart from following standard json file format, must follow exactly the following naming method and properties listed below.

- The root element (the very first " ") must named events.
- The attributes must have the name tag as described in the table above.

2.3.2 Writing xml event file

Json event file, apart from following standard json file format, must follow exactly the following naming method and properties listed below.

- The root element (the very first " ") must named as events.
- The subsequence child element must be named as event
- The XML attributes inside the event element must have the name tag according to the table above.

3 Conclusion

This document briefly described the necessary attributes and format for graph and event file. The sample of the graph and event file can be found on the repository under use case. This document will be changed in the future as the project progresses.