Nikhil Khatu
CSC/ECE 570
Professor Thuente

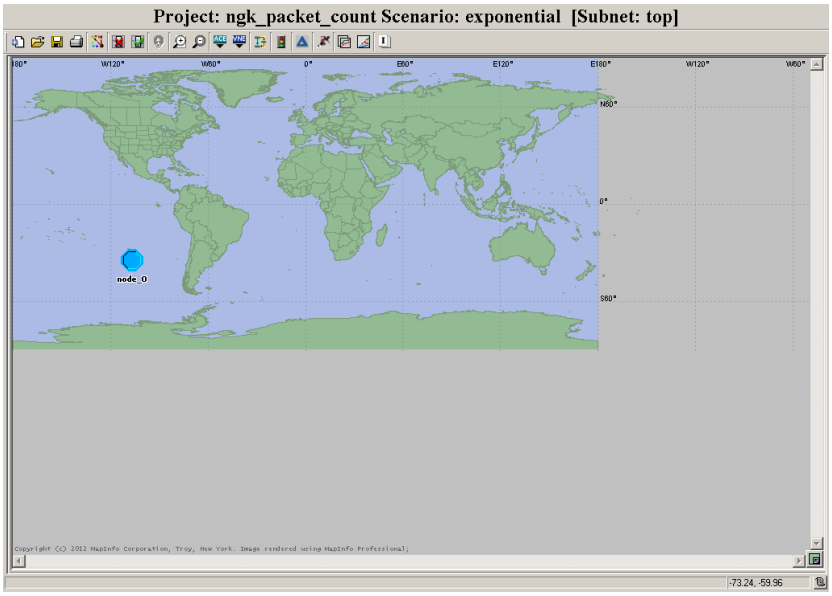
OPNET LAB 3

Based on "Basic Processes" and "Packet Switching I" Tutorials

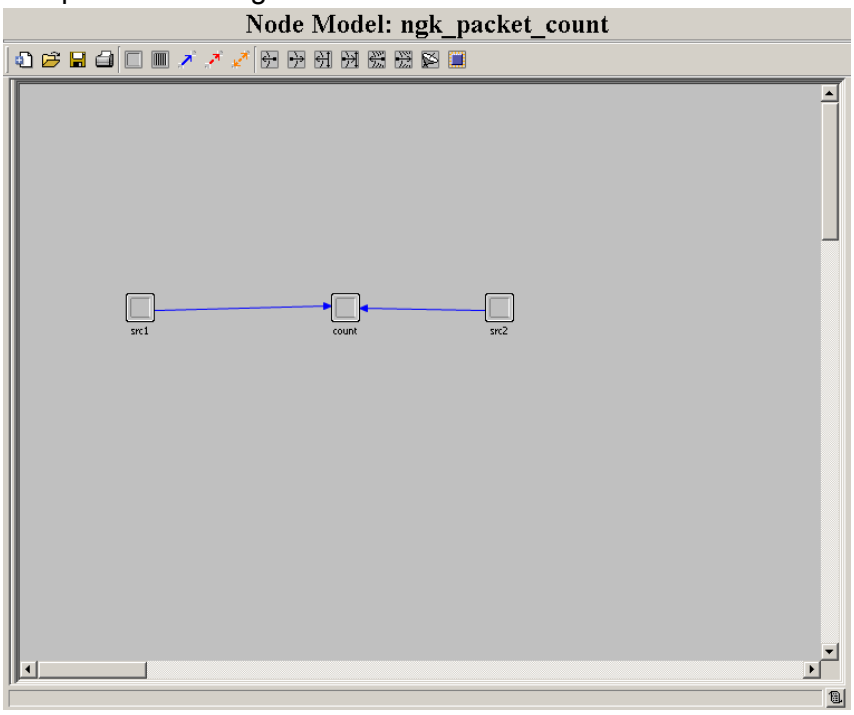
Due Date: Tuesday October 30, 8:00 am

Do the tutorial "Basic Processes"

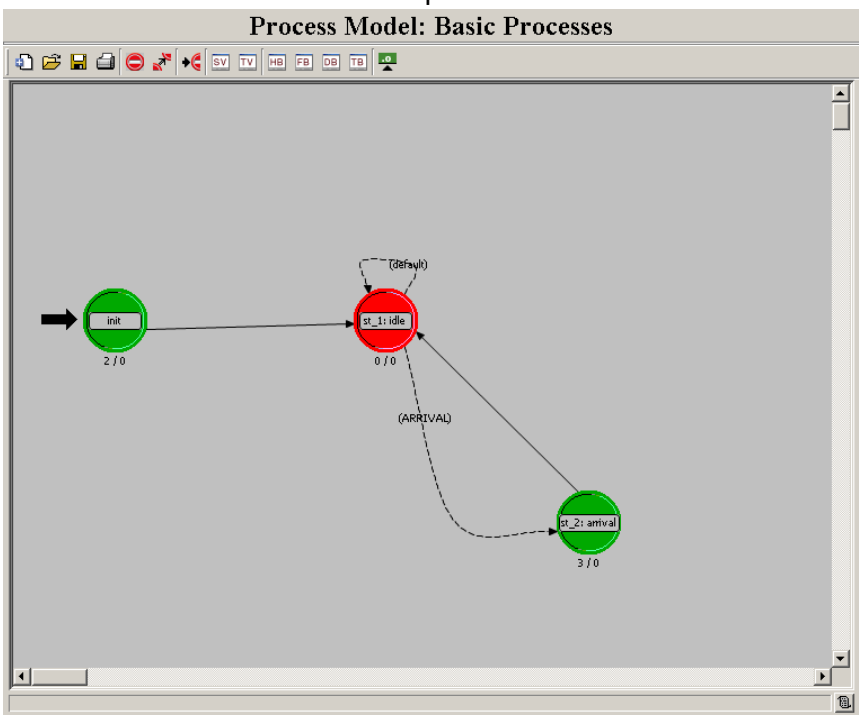
Question 1: For the one fixed processor module that generates packets use constant interarrival time of 2.0. For the promoted value, do two cases:
In this tutorial we will setup a simple counter. Here is the network model with the created node model.



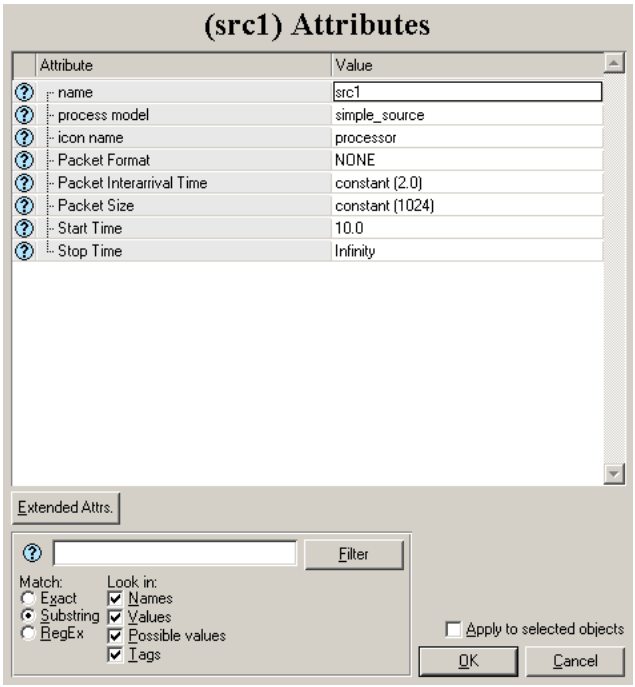
The packet counting node model:



The Process model of the count processor:

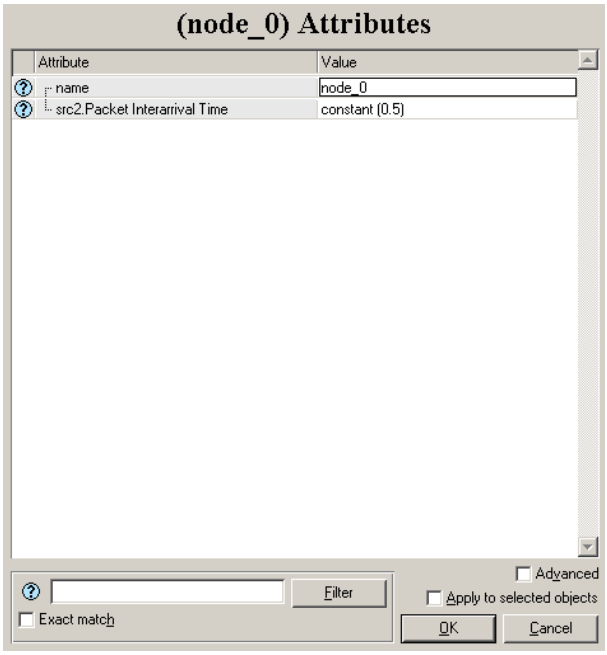


The constant interarrival time is set to 2.0 for the fixed processor module.

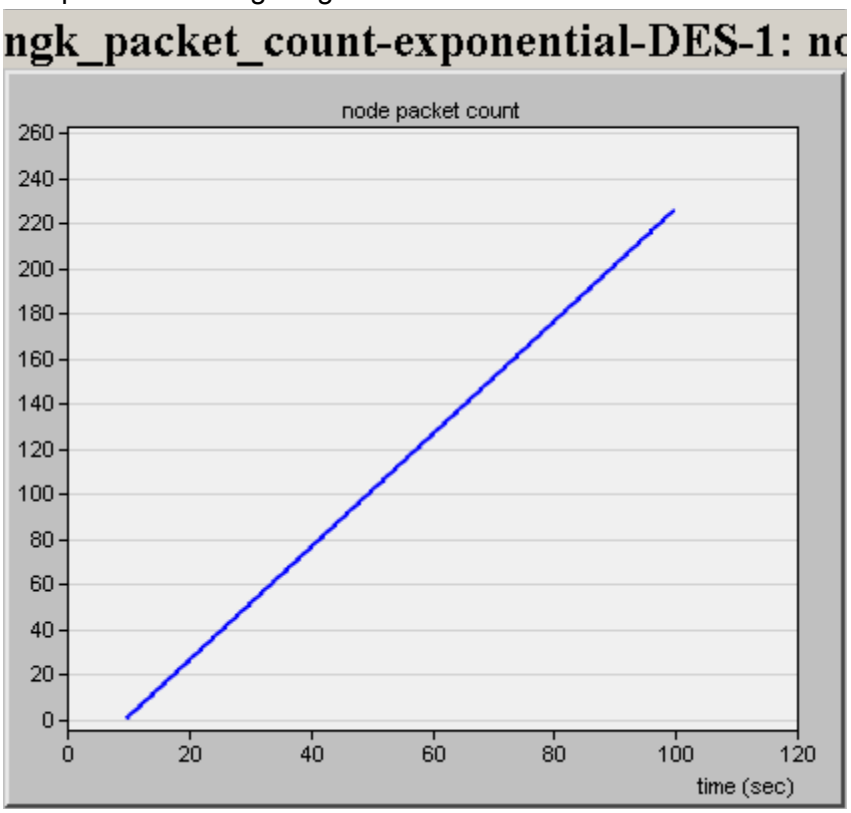


1a) Constant interarrival time of 0.5
Provide the figure on number of packets received as in the tutorial.

Promoted value now has a 0.5 constant interarrival rate:

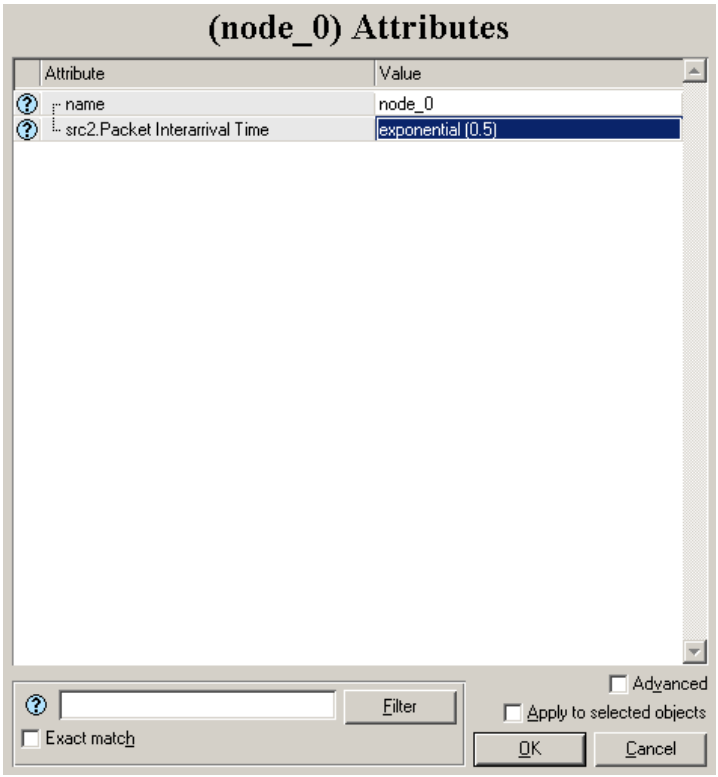


The packet count figure generated from the simulation:

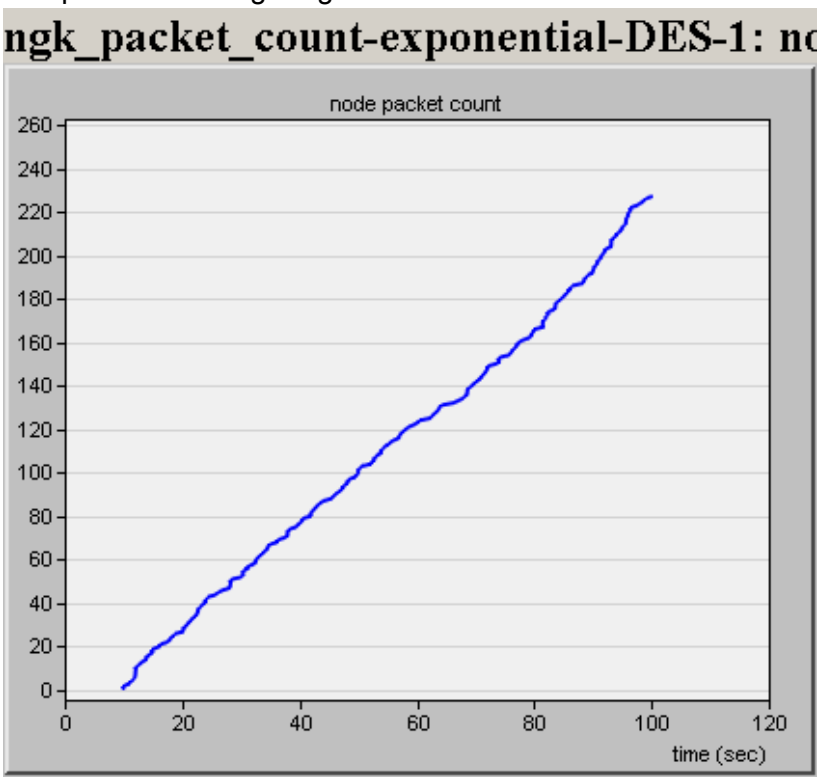


1b) Exponential interarrival time with mean 0.5
Provide the figure on number of packets received as in the tutorial.For the constant – exponential case, provide the number of packets received figure for 30 – 60 seconds.

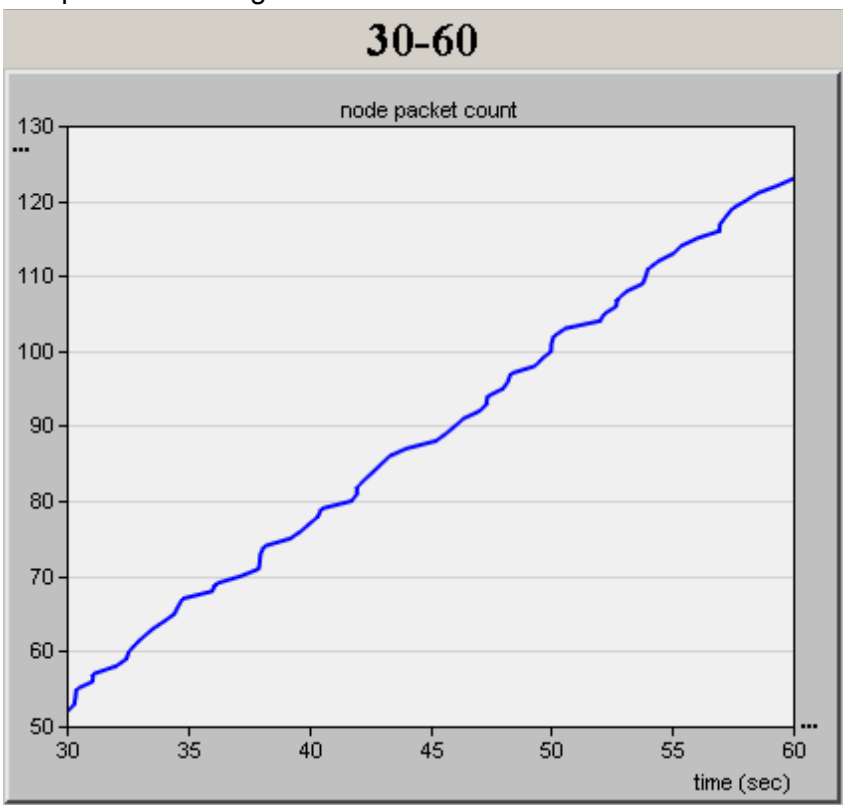
The promoted value now has an interarrival time of 0.5.



The packet count figure generated from the simulation run:

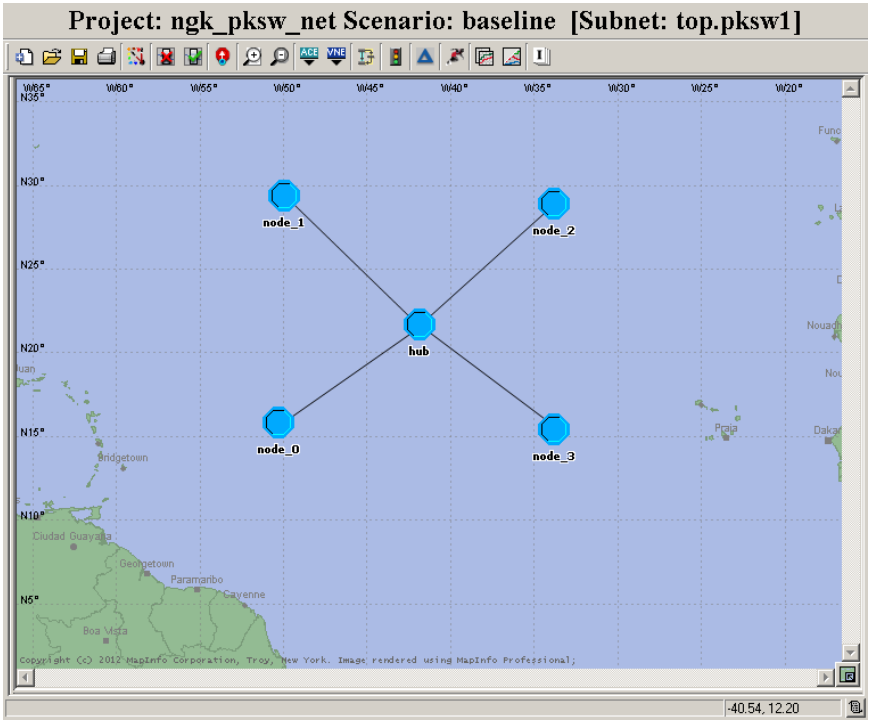


The packet count figure from 30 seconds to 60 seconds:

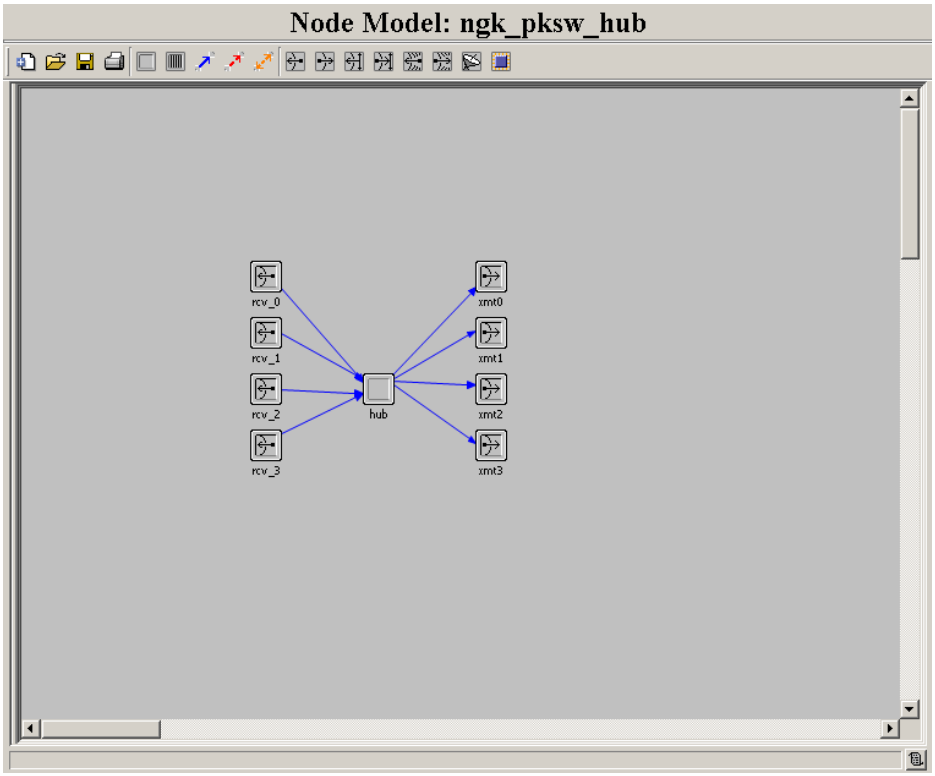


Do the tutorial “Packet Switching I”

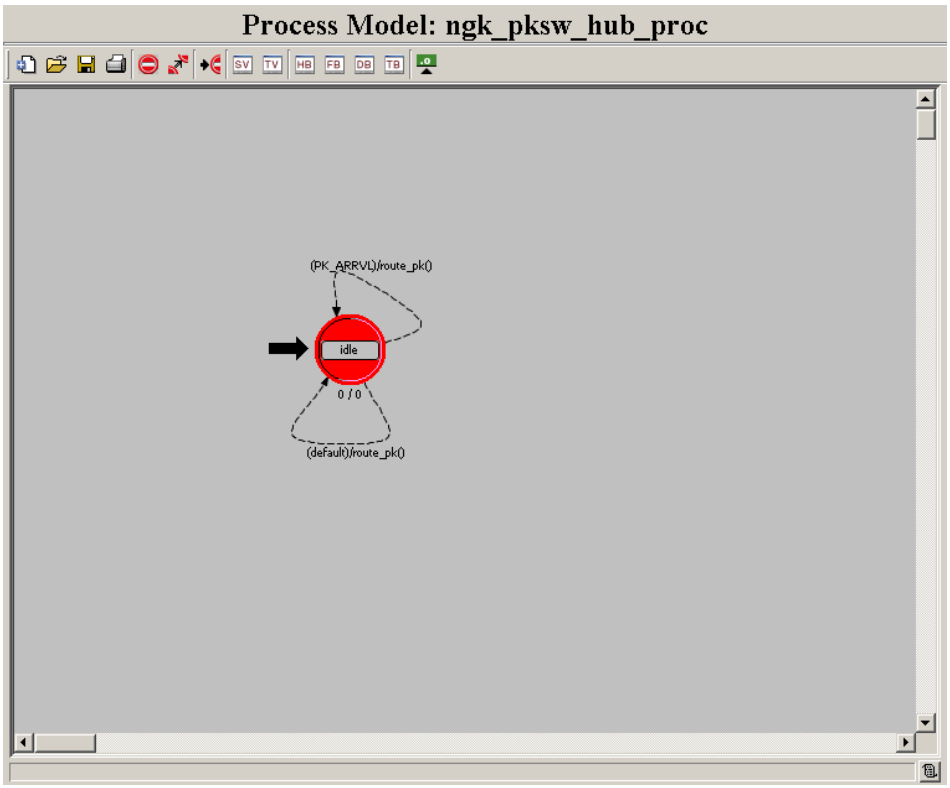
Network model for the Packet Switched scenario:



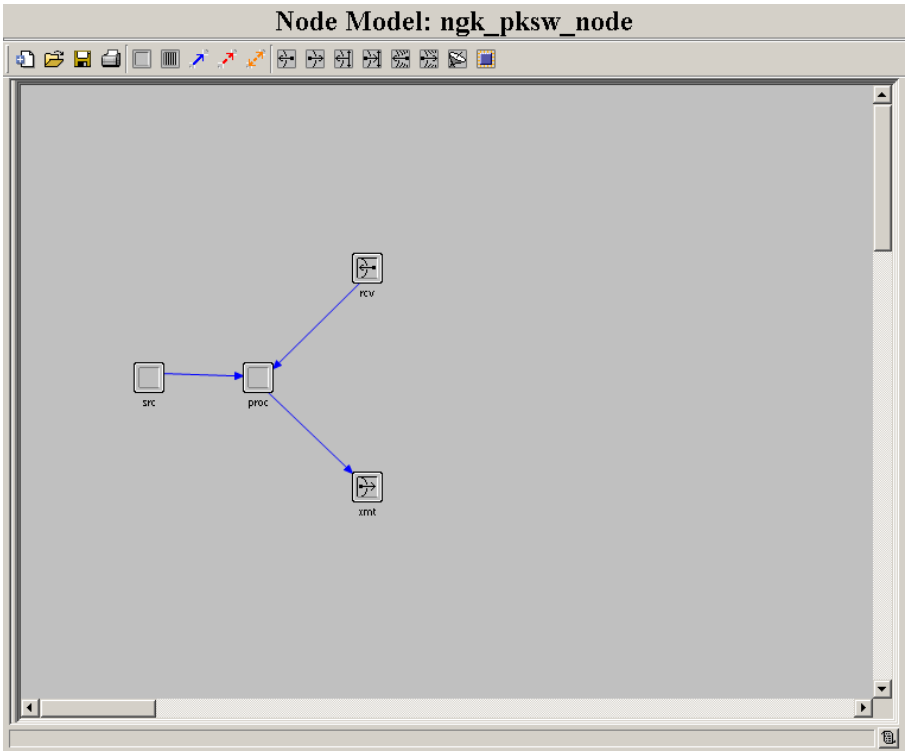
The hub node model:



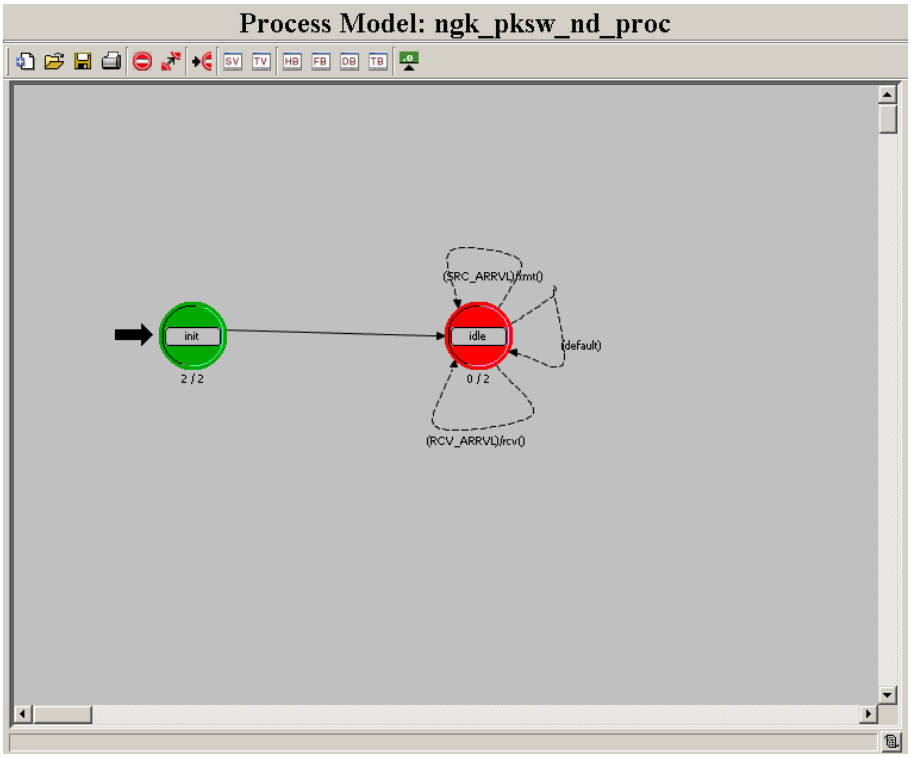
The hub process model:



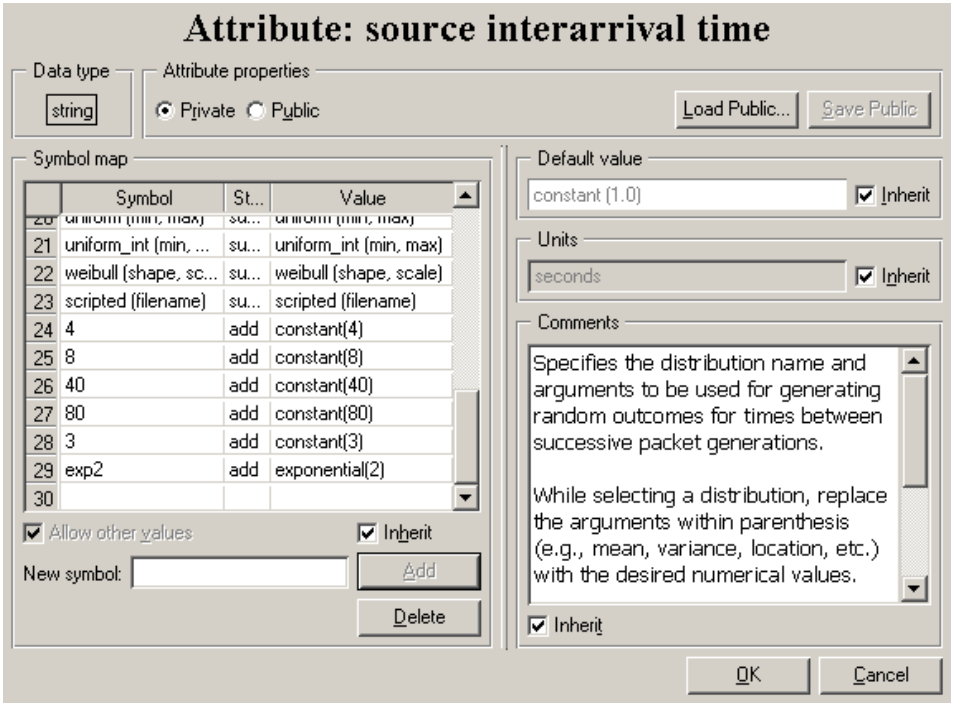
The node model used for Node_0, Node_1, Node_2, and Node_3:



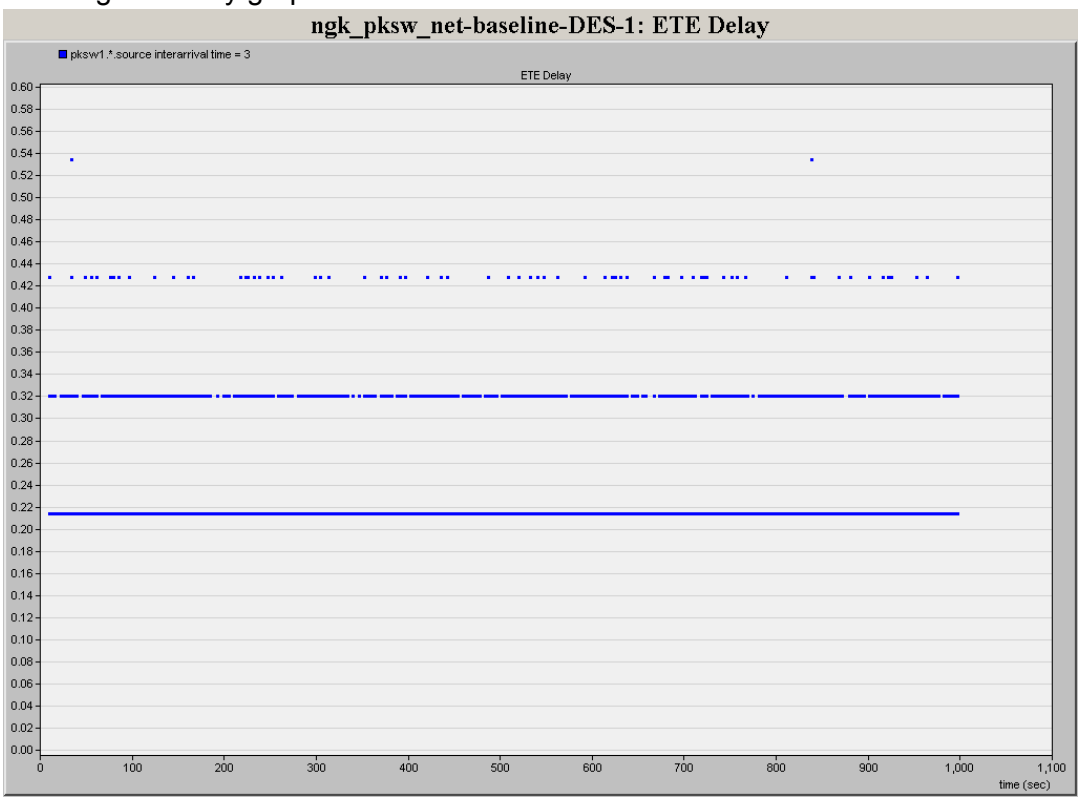
The process mode used in the ‘proc’ processor module:



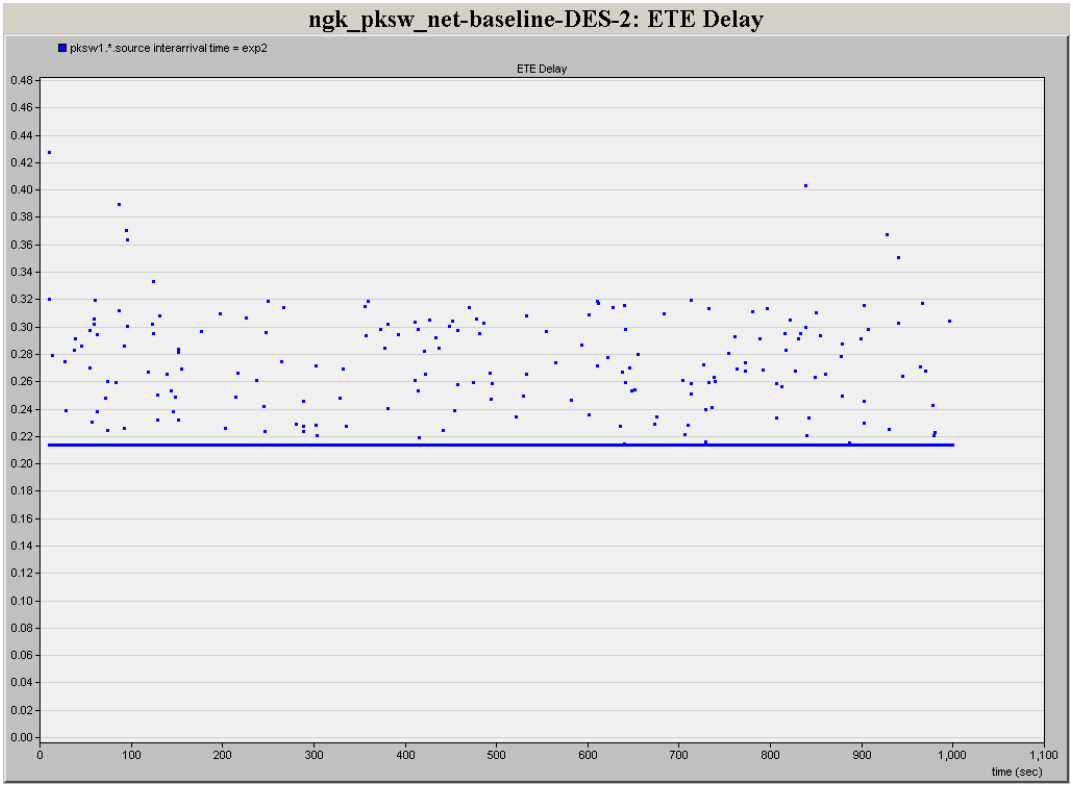
Before the simulation run two more symbols (3 and exp2) are added to the interrival time attribute list:



2a) Do the tutorial and run when it generates packets using constant interarrival time of 3.0. Give a good delay graph:



2b) Run when it generates packets using exponential interarrival time of 2.0.
Give a good delay graph:



Give the time average utilization for both (one graph).