

Overview of OPNET Modeler

Xavier Pérez Costa

NEC Network Laboratories

Heidelberg, Germany



Outline

- I. Introduction
- II. Usage Prerequisites

III. Modeler Simulator Structure

- I. Hierarchy and Domains
- II. Simulation Output

IV. Simulator Internals

- I. Event-list
- II. Simulation Kernel
- III. Interrupts
- IV. Processes

V. Modeling

- I. Network
- II. Nodes
- III. Links
- IV. Processes

VI. Performing a Simulation



Introduction

• OPNET Technologies, Inc.

- Corporate Overview Founded in 1986 Approximately 300 employees Worldwide coverage via direct offices and channel partners
- Solutions for Enterprises **Solutions** Solutions for Enterprises **Solution**
- Solutions for Service Providers

WDMGuru **VNE**Server **SP**Guru

Solutions for Network R&D Organizations

Modeler WDMGuru Development

- OPNET Modeler
 - Modeling and simulation environment for designing and analyzing communication protocols, network equipment, and end-to-end systems.



• Use of provided models

- Ability to understand C or C++
- Basic understanding of networks

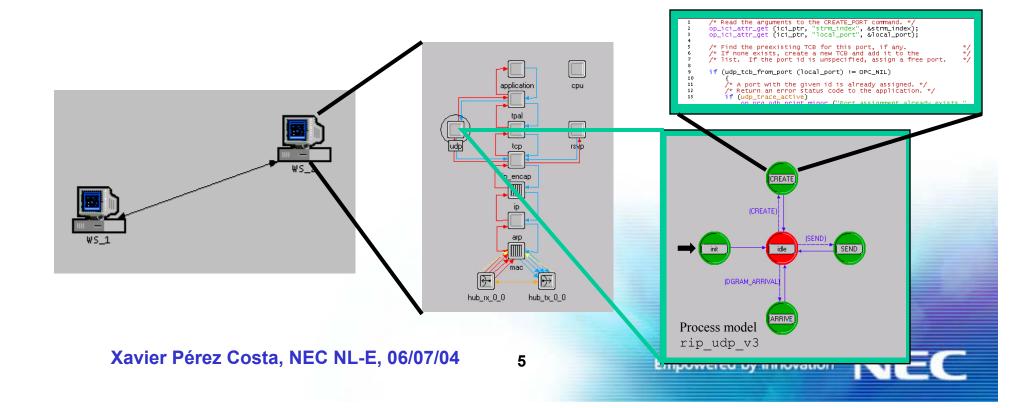
Modification or Development of new models

C or C++ coding skills



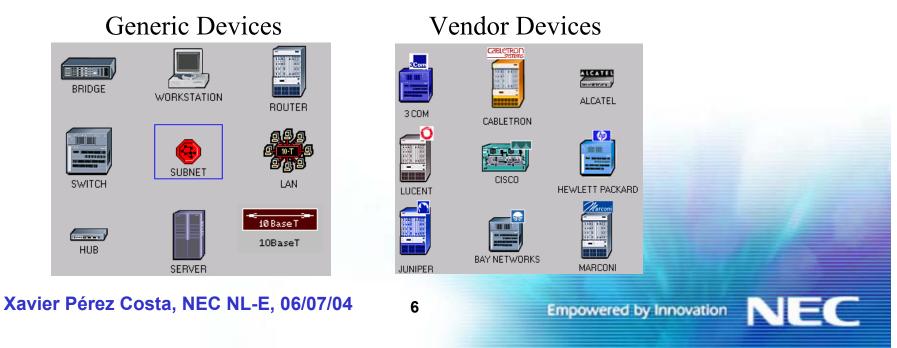
Simulator Structure: The Three-Tiered OPNET Hierarchy

- Three domains: network, node, and process
- Node model specifies object in network domain
- Process model specifies object in node domain



Simulator Structure: Network Domain: Network Objects

- Network models consist of nodes, links, and subnets
- Nodes represent network devices and groups of devices
 - Servers, workstations, routers, etc.
 - LAN nodes, IP clouds, etc.
- Links represent point-to-point and bus links
- Icons assist the user in quickly locating the correct nodes and links
- Vendor models are distinguished by a specific color and logo for each company

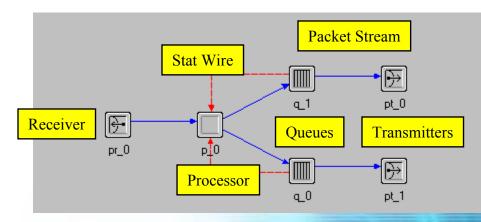


Simulator Structure: **Node Domain**

- Basic building blocks (modules) include processors, queues, and transceivers
 - Processors are fully programmable via their process model
 - Queues also buffer and manage data packets
 - Transceivers are node interfaces

Interfaces between modules

- Packet streams
- Statistic wires

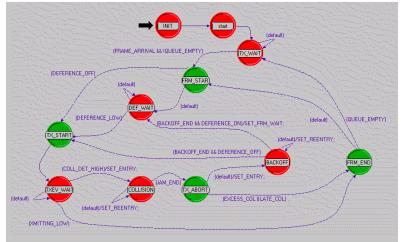




Simulator Structure: **Process Domain**

• OPNET process models consist of:

- State transition diagrams
- Blocks of C code
- OPNET Kernel Procedures (KPs)
- State variables
- Temporary variables



- A process is an instance of a process model
- Processes can dynamically create child processes
- Processes can respond to interrupts



Simulator Structure: **Simulation Output**

Three kinds of output

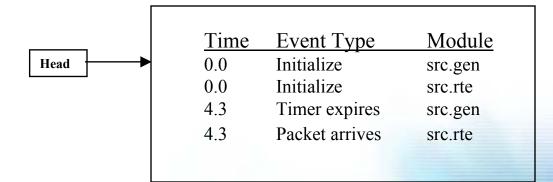
- Vectors
 - List of time-value pairs
- Scalars
 - List of values dependent on parametric input
 - Not plotted vs. time
- Animations
 - Packet flows
 - Node movements

Objects have pre-defined statistics



Simulator Internals: Event List Concepts

- Single global event list
- Shared simulation time clock
- Events scheduled in time order
- Event removed from event list when it completes





Simulator Internals: The Simulation Kernel

- Simulation kernel (SK) manages the event list
- SK delivers each event, in sequence, to the appropriate module
- SK receives requests from processes and inserts new events in the event list

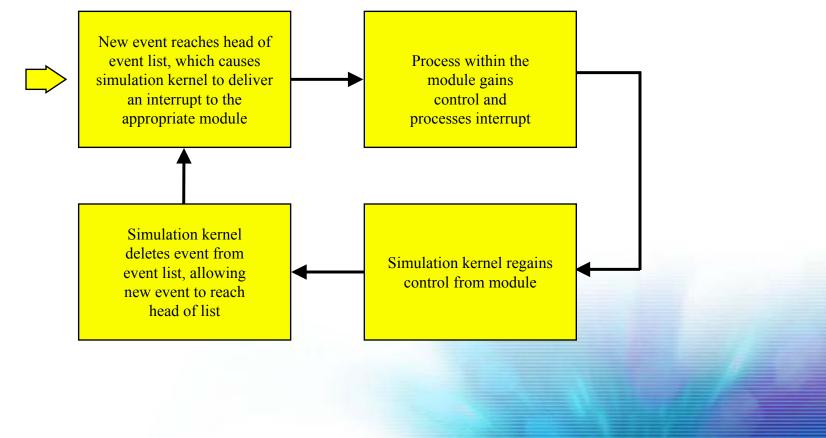


Simulator Internals: Interrupts

- First event in the event list becomes an interrupt
- Delivered by the simulation kernel to the designated module
- Data associated with the event can be obtained by the module
- Processors and queues can have BEGSIM interrupts



Simulator Internals: How Does the Event List Work?



Xavier Pérez Costa, NEC NL-E, 06/07/04

13

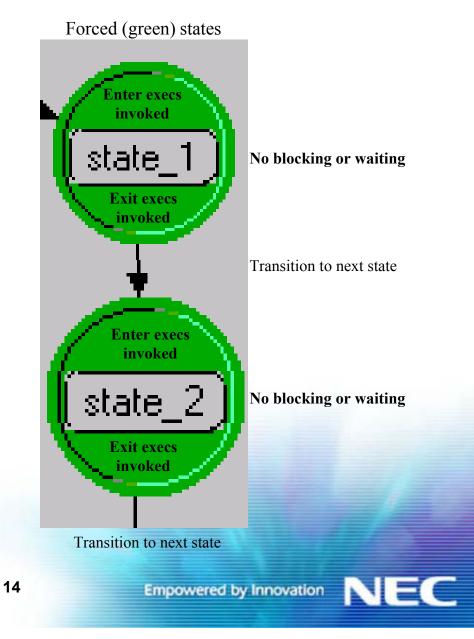
Empowered by Innovation



Simulator Internals: Forced States

- Forced (green) and unforced (red) states differ significantly in execution timing
- In a forced state, the process:
 - Invokes the enter executives
 - Invokes the exit executives
 - Evaluates all condition statements
 - If exactly one condition statement evaluates to true, the transition is traversed to the next state

OPNET convention: code in enter execs only



Simulator Internals: Unforced States

In an unforced state, the process:

- Invokes the enter executives
- Places a marker at the middle of the state
- Releases control to the simulation kernel and becomes idle
- Resumes at the marker and processes the exit execs when next invoked

state xit execs process when invocatio occurs Transition to next state nter execs invoked state 2 Next invocation

Unforced (red) states

Blocking, waiting for invocation

Start of invocation

End of invocation

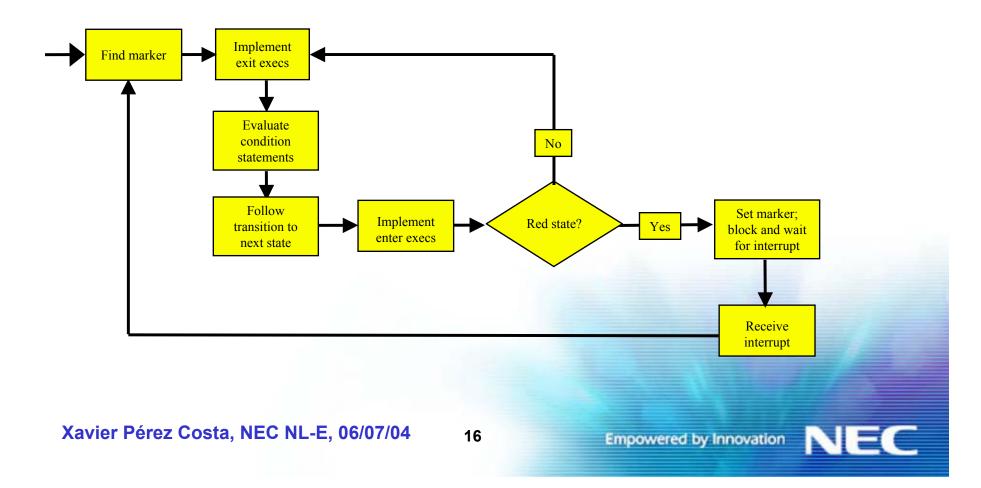
Blocking, wait for next invocation

Xavier Pérez Costa, NEC NL-E, 06/07/04



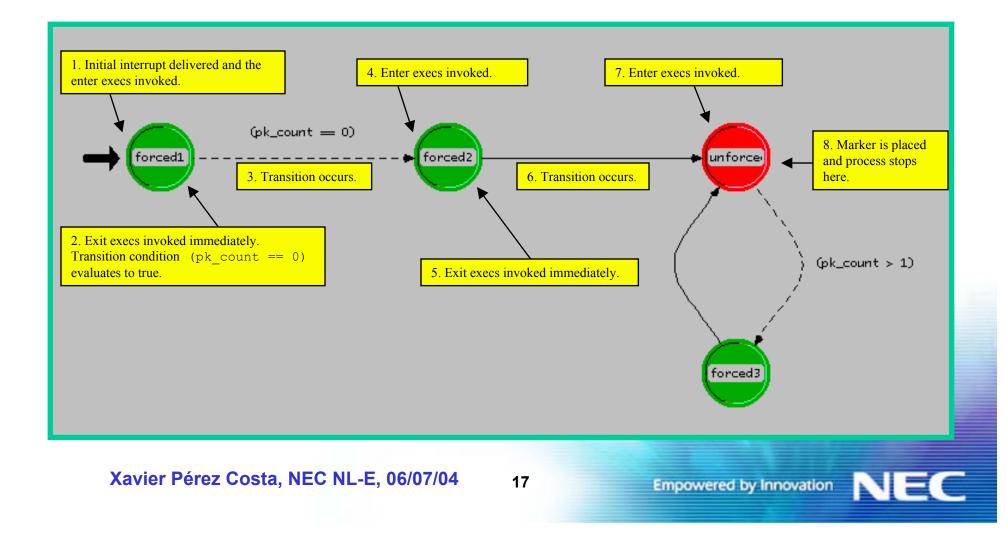
Simulator Internals: How a Process Handles an Interrupt

- Flow diagram showing how a process handles an interrupt:
 - (except the initial interrupt)



Simulator Internals: Process Model Example

Model with three forced states and one unforced state



Modeling : Creating Network Models

There are two ways to create new network models

- Manual creation
 - Drag and drop
 - Rapid configuration
- Import from network management tool



Modeling : Rapid configuration

- Rapid configuration allows you to quickly create networks of any size
- Available topology configurations:

Bus; Ring; Star; Tree; Unconnected Net; Mesh (Full or Randomized)

 You control the number of nodes, the node and link models used, how nodes will be arranged, and node locations within the workspace

	Rapid Configuration:	Star	×
	Models:		
Rapid Configuration	Center node model:	hernet32_hub	
Configuration: Star	Periphery node model: et	hernet_wkstn 💌 Number: 20	
	Link model: 10)BaseT 🗾	
Seed <u>D</u> K <u>C</u> ancel	- Placement:		
	Center:		
	X: 50	Radius: 26.9516	
	Y: 50.0967		
	Select Models	<u></u> K	Cancel
Xavier Pérez Costa, NEC NL-E, 06/07/04	19	Empowered by Innovation	NEC

Modeling : Device Creator

- Automatically create a particular device with a specific configuration
- Launched from Topology menu





Modeling : **Object Attributes**

- All objects have attributes that control aspects of their behavior
- Attributes may vary from one model to the next one
- Attribute values may vary between objects of the same model type
- Right-click on an object and select Edit Attributes to view or change its attributes
- Can be promoted
 - Allows you to set the value at a higher layer
- Allows you to specify a range of values at runtime Xavier Pérez Costa, NEC NL-E, 06/07/04

	Attribute	Value		
0	_ name	http_server		
0		ethernet_server		
	Applications			
	± CPU			
_	E IP			
	IP Host Parameters	()		
2	IP Processing Information	Default		
0		None		
_	± SIP			
0	- Server Address	Auto Assigned		
	E Servers			
~				
0		Default		
	Apply changes to selected objects	_ □ A <u>d</u> vance		
	<u>Find Next</u>	<u> </u>		

- Online Documentation from the Help menu
- Model help accessible by right-clicking object icons in the object palette or by right-clicking objects in the Project workspace and selecting View Node Description
- Tool Tips by holding the mouse cursor over any object to get a brief description of that object
- Attribute help accessible by clicking on the question mark next to the attribute

		I IP			
	0	🛨 IP Host Parame	ters	()	
	0		nformation	Default	
	0	± IP QoS Parame	ters	None	
		∃ <mark>SIP</mark>			
	0	🗄 SIP Proxy Serve	er Parameters	()	
	2	- Server Address		Auto Assigned	
Xavier Pérez Costa, NEC NL-	E, 0	6/07/04	22	Empowered by Innovation	C

Modeling : Deriving New Models

- Derive a new model based on any existing model
- Can alter the attributes of the newly-derived model

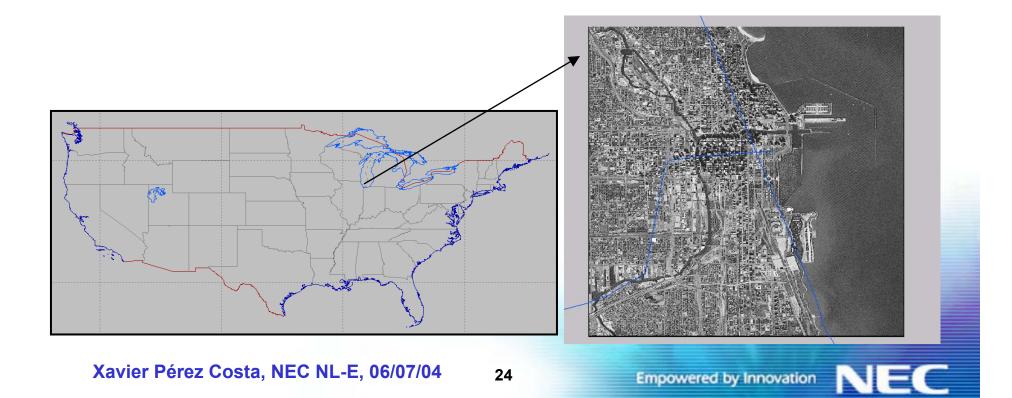
* Node Model Description: ethernet_server				
Comments:		Parent Model:		
General Node Functions:		ethernet_server_int		
The ethernet_server model represents a server node with server applications running over TCP/IP and UDP/IP. This no		View Parent		
- Keywords:	Supported Node Types:	·		
	lode Type Default Icon ked server			
Attribute Name	Initial Value	View Properties		
Application: ACE Tier Configuration	Unspecified	1		
Application: Supported Services	None			
CPU Background Utilization	None			
CPU Resource Parameters	Single Processor			
IP Host Parameters	[]			
IP Processing Information	Default			
IP QoS Parameters	None			
<u>E</u> dit <u>D</u> erive New D <u>o</u> cumentation.	. <u>V</u> iew Self-Description	<u>C</u> lose		

Empowered by Innovation NEC

23

Modeling : Map Backgrounds

- Maps can be loaded as backgrounds for network models
- Map backgrounds provide a physical context for model specification
- Distance between nodes can be a factor affecting simulation results



Modeling : Node Objects

Modules are the basic building blocks of node models

- Processors
- Queues
- Transceivers
 - Transmitters
 - Receivers
- Antennas



Xavier Pérez Costa, NEC NL-E, 06/07/04

modelsFully programmable

Processors

Queues

Offer all the functionality of processors

General-purpose building blocks of node

Modeling : Processors and Queues

 Can also buffer and manage a collection of data packets q

Processor

D

Queue

26

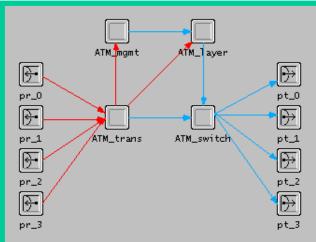


Modeling : Transceivers

- Transceivers are the interfaces between objects inside a node and communication links outside of it
 - Transmitters are the outbound interfaces
 - Receivers are the inbound interface

Some transceiver attributes:

- Number of channels
- Data rate
- Supported packet formats



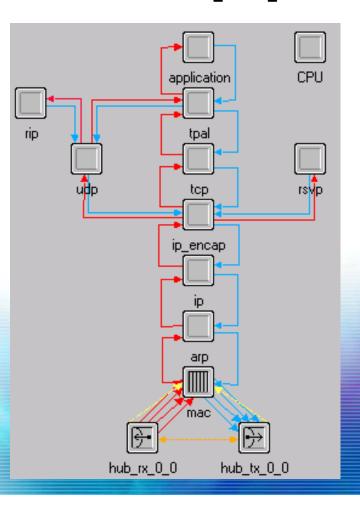
Point-to-poin	t transceivers	Bus trans	sceivers	Radio trans	ceivers	
Ð	Ð	X	Ħ	• <u>**</u>	X	
Transmitter	Receiver	Transmitter	Receiver	Transmitter	Receiver	
Xavier P	érez Costa,	NEC NL-E, 0	6/07/04	27	Empowered	by Innovation



Modeling : Example of a Node Model

Node models can support:

- Layering of protocol functions
- Dynamic inter-module monitoring
- Arbitrary node architectures



Node Model : ethernet_wkstn_adv

Modeling : Specifying Node Interfaces

Node Model: in	itials_recei	ver_n	d			
e Edit Interfac	es Objects	Wind	ows I	Help		
	el Attributes Interfaces		Þ		H]
Node	Statistics					_
Self (Description					
1 Vada Takaufasasi inikiste vas	siuce od					
Node Interfaces: initials_rece Comments:	liver_na					
Keywords:		Node Types: —				
Reywolds.		Node Types.	Supported	Default Icon		
		ixed	ves	server		
		nobile	no	301701		
Add	<u>D</u> elete	atellite	no			
Attributes:						
Attribute Name	Status	Initial Value			A Re	name/Merge
TIM source set		none			Ed	lit Properties.
altitude	set	0.0			in rependes.	
altitude modeling	set	relative to subnet-platform enabled				
condition financial cost	set set	enabled 0.00				
minimized icon	set	circle/#70809	90			
abasa	sot				-	
Documentation				[<u>0</u> K	<u>C</u> ance

Specify various characteristics of the node

- Rename attributes
- Set attribute values
- Hide attributes
- Specify node type
- Add comments

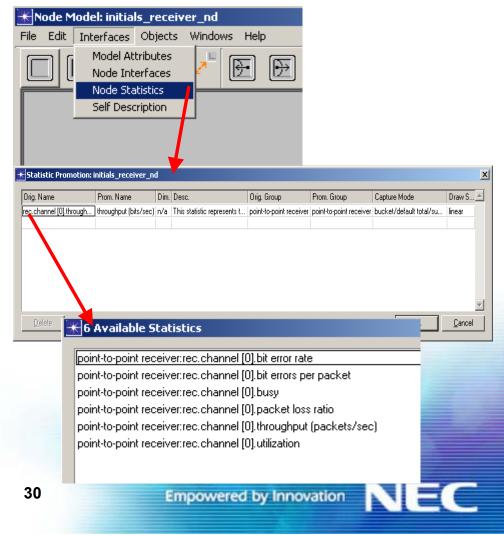
Xavier Pérez Costa, NEC NL-E, 06/07/04

29



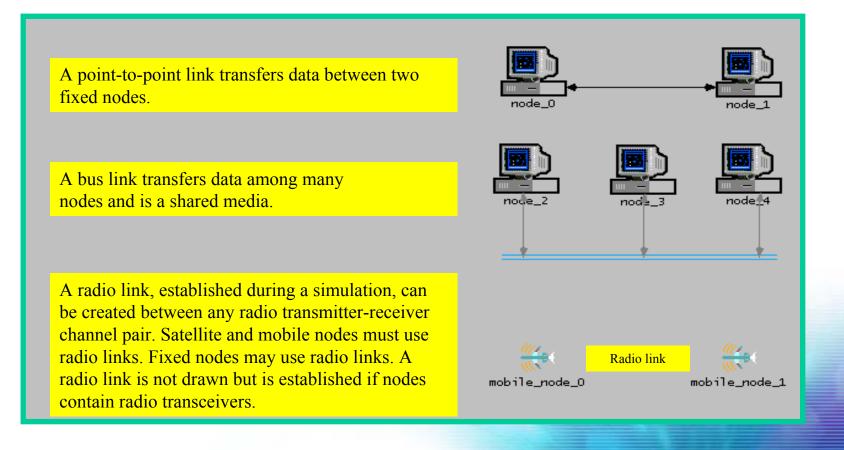
Modeling : Specifying Available Node Statistics

- Specify statistics available from project editor
- Selecting a statistic from the "Available Statistics" table adds the statistic to the "Statistic Promotion" table



Modeling : Link Types

 Link objects model physical layer effects between nodes, such as delays, noise, etc.



Xavier Pérez Costa, NEC NL-E, 06/07/04

31

Empowered by Innovation



Modeling : Link Editor

- Create or modify links
- Choose link types
- Modify attributes

		nitials_pt		00							
e <u>E</u> dit	Link	<u>W</u> indows	Help								
Comme	ents: —										
										_	<u></u>
Keywor	rds: —						Ī	 Supported Li 	nk Types: ——		
						*		Link Type	Supported		Palette Icon
								ptsimp	yes		simp_pt_lk
								ptdup	yes		dup_pt_lk
								bus	no		
						Ψ.		bus tap	no		
		Add		Del	ete						
Attribut	es:						J [.				
Attribute	e Name	9			Status		1	nitial Value			Define
arrowhe	eads				set		ŀ	iead and tail			Rename/Merge
ber					set			1.0			Thendinez merge
channe					set		1				Edit Properties
closure					set			lbu_closure			
coll mod	iel				set			lbu_coll		-	
color					set		TF	GB000		_	
Docume	entation	n Loca	al Statistics	1							Help

Xavier Pérez Costa, NEC NL-E, 06/07/04



Modeling : Verify Links

Verify links before running a simulation



- Ensures that point-to-point and bus link connections are valid
 - Enough transmitters and receivers to support all of the incoming and outgoing links
 - Data rates of the connected transmitter and receiver match the data rate of the link
 - Transceivers support the attached link technology



Modeling: Process Models

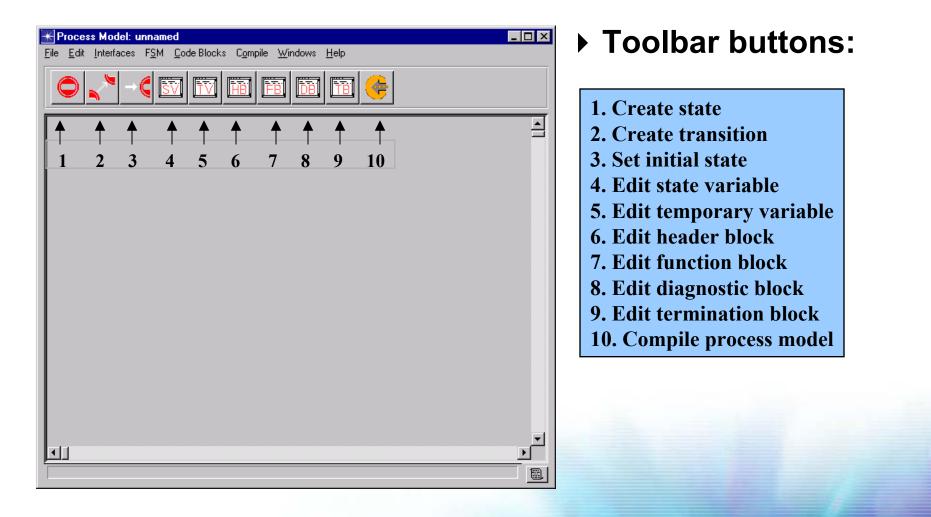
Process models represent algorithms

- Communications protocols and algorithms
- Shared-resource managers
- Queuing disciplines
- Specialized traffic generators
- Statistic-collection mechanisms
- Control processes

Process editor provides the features for creating process models



Modeling : Process Editor



Xavier Pérez Costa, NEC NL-E, 06/07/04

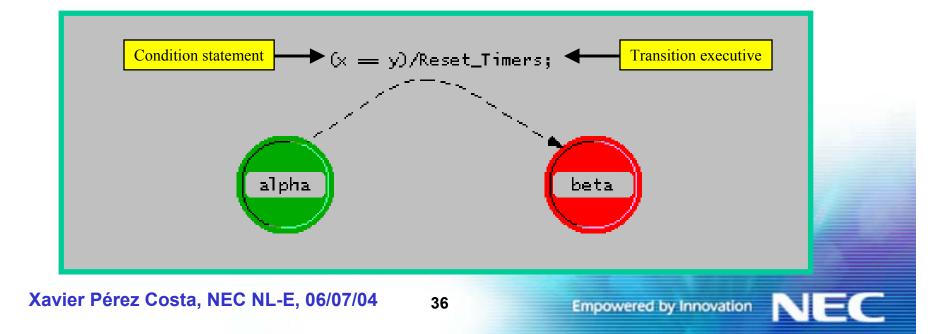
35



Modeling: State Transitions

Transitions connect states

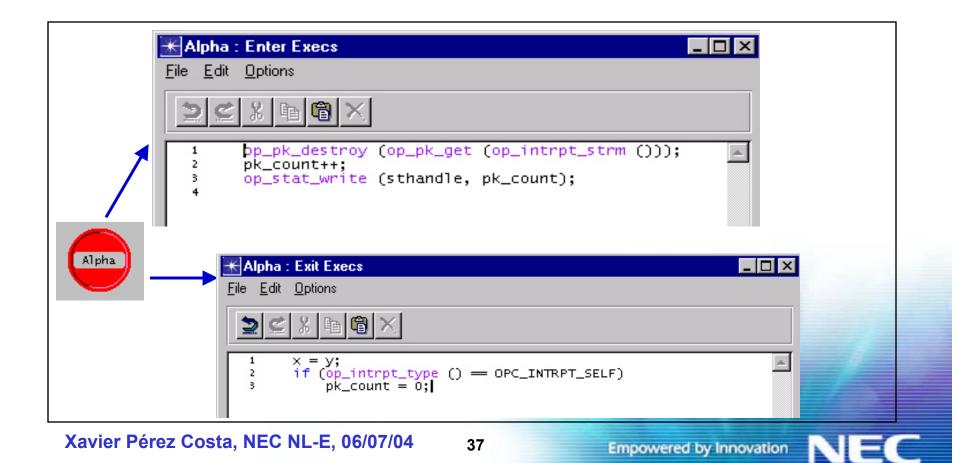
- Conditional
- Unconditional
- Transition executive
- If the condition statement (x == y) is true, the transition executive (Reset_Timers) is invoked



Modeling: State Executive Blocks

Each state has two executive blocks

- Enter executives are invoked upon entering a state
- Exit executives are invoked before exiting a state



Modeling: Kernel Procedures (KP)

- Pre-written functions for difficult, tedious, or common operations
- KPs free users from addressing memory management, data structure, handling event processing, etc.
- KPs focus on communication modeling
- All KPs begin with prefix op_*



Modeling: Common Kernel Procedures

Packet Package: op_pk_create () op_pk_create_fmt () op_pk_copy () op_pk_get () op_pk_total_size_get ()	Subq Package: op_subq_pk_insert () op_subq_pk_remove () ID, Topo and Internal	Interrupt Package: op_intrpt_schedule_self() op_intrpt_type() op_intrpt_strm() op_intrpt_code()	Distribution Package : op_dist_load () op_dist_outcome ()
op_pk_nfd_set () op_pk_nfd_get () op_pk_send () op_pk_send_delayed () op_pk_destroy ()	Model Access Packages: op_id_self () op_topo_parent () op_topo_child () op_ima_obj_attr_get ()	Simulation and Event Packages: op_ev_cancel () op_sim_time ()	

Naming convention for kernel procedures op_<family name describing object >_<action>



Performing a Simulation: Configuring Simulations

- Scenarios automatically provide a default duration and random number seed for simulations
- Users can set simulation attributes by choosing Configure/Run Discrete Event Simulation from the DES menu, or by clicking on the "running man" icon:



Configure/Run DES: bank_net-switched Image: Simulation Set Info Number of runs in set: 1 Pause between each run in set: Common Reports Global Attributes Object Attributes Traffic Growth		Configure Simulation Dialog Box			
E Duration: 30 30 Seed: 128	Duration: 30 minute(s)		Name	Description	
Values per statistic: 100 Update interval: 500000	events		Duration	Duration of simulation, in simulated time	
✓ Enable DES log Maximum log entries: 200			Values Per Statistic	Number of values to be collected for each statistic	
Simulation Kernet: Based on 'kernet_type' prefere	ence		Seed	Random number generation seed	
	Run C	ancel Apply Help			

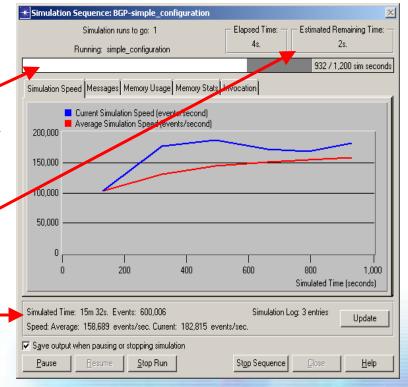
Xavier Pérez Costa, NEC NL-E, 06/07/04

Empowered by Innovation



Performing a Simulation: Running a Simulation

- The Simulation Sequence window shows the progress of simulation
- Elapsed time bar displays the progress of the simulation
 - Appears after 1,000,000 events by default
- Elapsed/Remaining Time: Real time elapsed and remaining time
- Simulation Time: Simulation
 time elapsed and number of events processed





Performing a Simulation: Viewing Results

42

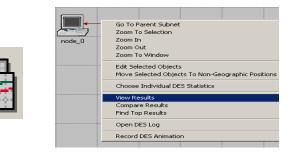
Results can be displayed by:

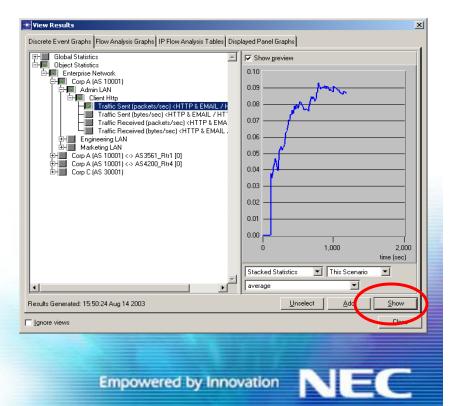
- Selecting the View Results button on the tool bar
- Selecting View Statistics from the DES menu
- Right-clicking the project workspace and selecting from the pop-up menu

View Results dialog box allows the user to select the results to display.

- *Note*: Only the statistics you chose for collection will be available

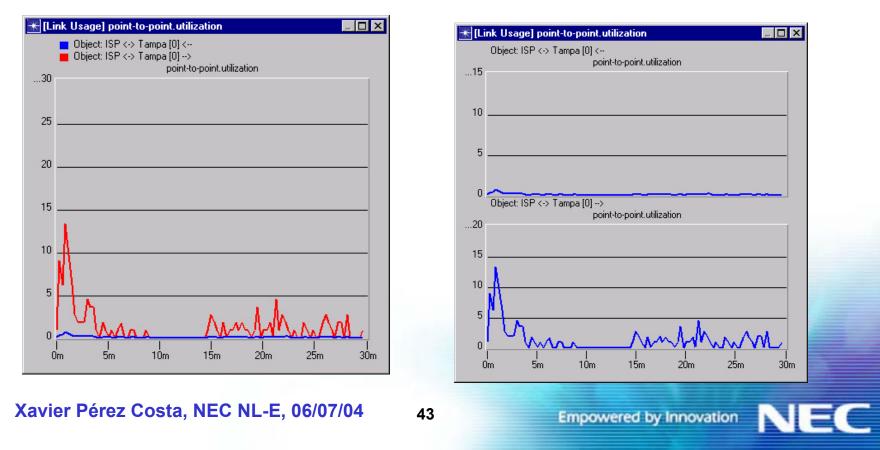
The Show button in the View Results dialog box displays a graph of the selected statistics





Performing a Simulation: Viewing Results (cont.)

- Multiple graph panels can be displayed at the same time
- Each panel can contain one or more traces in an Overlaid or Stacked layout



Performing a Simulation: Statistic Attributes

- Right-clicking on a statistic while in the Choose Results dialog box presents a menu of statistic attributes
- Statistic attributes include:
 - Change Collection Modes
 - Statistic Description
 - Change Draw Style

Choose Results Global Statistics ACE Task Response Time Change Collection Mode Traffic Received (byte Statistic Description Traffic Received (pacl Change Draw Style Traffic Sent (bytes/sed Traffic Sent (packets/sec) ATM. Cache Custom Application DB Entry DB Query DSR Email Fto HTTP Pv6 Mobile IP PNNI

Empowered by Innovation

NEC

Performing a Simulation: Statistic Collection Modes

- Normal mode: Every data point is collected from a statistic
- Sample mode: The data is collected according to a user-defined time interval or sample count
- Bucket mode: All the data points in a bucket are collected and processed according to a userdefined parameter

45

- Max
- Min
- Sum
- Count
- Sample mean
- Time average

🛞 Task Resp	onse Time (se	c).captu 🗙	1
Capture mode:	bucket	-	
○ Every:		seconds	
○ Every:		values	
● <u>T</u> otal of:	_	values	
Bucket mode:	sample mean	-	
✓ <u>R</u> eset			
Advanced			
[<u>0</u> K	<u>C</u> ancel	NEC

The material to prepare this presentation has been extracted from:

- [1] Overview of OPNET's Product Portfolio, OPNETWORK2003, Session 1000.
- [2] Overview of Modeler and ODK, OPNETWORK2003, Session 1006.
- [3] Introduction to using Modeler, OPNETWORK2003, Session 1201.
- [4] Habits of highly-effective OPNET users, OPNETWORK2003, Session 1213.
- [5] www.opnet.com

