

Simulació de xarxes i sistemes informàtics

- Introducció a la simulació.
- Anàlisi i comparació de paquets d'aplicacions per a la simulació de xarxes : OMNET++ , QUALNET Developer i OPNET IT Guru.
- Estudi de simulació amb OPNET IT Guru : modelat d'una xarxa, simulació i anàlisi de resultats.

Carles Pons
UOC-ETIS 2006

2 – DEDICATORIA I AGRAÏMENTS

Aquest treball està dedicat a la meva família i als meus amics, sobretot a la meva dona Susana a la nostra filla Ona, pel temps que he deixat d'estar amb elles i que espero poder recuperar.

Agraïments a la UOC pel seu sistema d'ensenyament i en especial a l'Àngel per confiar-me aquest treball i per la seva ajuda.

Resum

En l'actualitat, la major part de les empreses i processos productius han crescut en complexitat i han vist augmentats els requeriments de competitivitat, flexibilitat i qualitat en les seves activitats. Constantment s'han d'adaptar als canvis del mercat i innovar.

La falta d'eines analítiques que ajudin i facilitin la presa de decisions, és un dels principals problemes amb els que es troben les empreses que veuen la necessitat de millorar el rendiment dels seus sistemes i processos. Per a poder comprendre la realitat i tota la complexitat que un sistema pot portar, ha estat necessari construir artificialment objectes (models) i experimentar amb ells dinàmicament abans d'interactuar amb el sistema real.

Les eines de simulació ofereixen una plataforma que permet abordar amb èxit un procés de millora contínua de sistemes complexos per als quals, les tècniques analítiques clàssiques basades en l'ús de càlcul diferencial, teoria de probabilitats i mètodes algebraics, no poden ser utilitzades per a formalitzar de manera senzilla la complexitat dels processos.

La simulació per computador és un poderós instrument per a avaluar i entendre el funcionament d'existència i sistemes, particularment cert en casos de xarxes d'ordinadors i sistemes informàtics. Les dades obtingudes durant l'execució del programa de simulació son usades per a calcular les estimacions del funcionament del sistema original.

Les diferents tècniques de simulació per computador han adquirit una importància cada cop mes rellevant en la resolució de diferents tipus de problemes pràctics en camps tals com l'enginyeria, economia, medicina, ciències de la vida, ecologia, etc.

Al llarg d'aquest projecte, s'introduiran, en primer lloc, conceptes generals de sistemes, models i simulació de sistemes. A continuació, s'aprofundirà en la simulació de sistemes informàtics i xarxes, realitzant una comparativa de tres entorns de simulació de xarxes i una simulació pràctica utilitzant un d'ells.

Paraules clau

Events discrets, model, simulació, sistemes i xarxes.

Nom del TFC

Simulació de xarxes i Sistemes Informàtics.

4 – INDEX DE CONTINGUTS

1 - Portada	1
2 - Dedicatòria i agraïments	2
3 - Resum, paraules clau i nom del TFC	3
4 - Índex de continguts	4
5 - Memòria	6
PART I	
5.1 - Introducció a la simulació de sistemes	6
5.2 - Modelat de sistemes orientats a events discrets	11
5.3 - Modelització de l'aleatorietat en sistemes discrets : models estadístics	14
5.4 - Introducció a la teoria de cues	18
5.5 - Simulació de sistemes orientats a events discrets	21
5.6 - Metodologia d'un estudi de simulació	24
5.7 - Conclusions	27
PART II	
5.8 - Modelat i simulació de xarxes	29
5.9 - Software per a la simulació de xarxes de computadors	31
5.10 - OMNET++	33
5.10.1 - Descripció del software, característiques i components	33
5.10.2 - Construcció de models	36
5.10.3 - Simulació : creació d'estadístiques	40
5.10.4 - Execució, visualització i animació	42
5.10.5 - Anàlisi de resultats	45
5.10.6 - Altres recursos	47
5.11 - QUALNET Developer	49
5.11.1 - Descripció del software, característiques i components	49
5.11.2 - Construcció de models	52
5.11.3 - Simulació : creació d'estadístiques	56
5.11.4 - Execució, visualització i animació	57
5.11.5 - Anàlisi de resultats	60
5.11.6 - Altres recursos	61
5.12 - OPNET IT Guru	62
5.12.1 - Descripció del software, característiques i components	62
5.12.2 - Construcció de models	65

5.12.3 - Simulació : creació d'estadístiques	67
5.12.4 - Execució, visualització i animació	68
5.12.5 - Anàlisi de resultats	69
5.12.6 - Altres recursos	71
5.13 - Comparativa	73
5.14 - Conclusions	80
PART III	
5.15 - Estudi de simulació amb OPNET IT Guru	82
5.16 - Objectius	82
5.17 - Model lògico-matemàtic	82
5.18 - Us i verificació del software : construcció del model	83
5.19 - Validació del model	88
5.20 - Disseny de la simulació i execució	88
5.21 - Anàlisi de resultats i conclusions	88
6 - Glossari	92
7 - Bibliografia	95
8 - Annexes	97
8.1 - Llibreria de models per a OMNET++	97
8.2 - Llibreria de models per a QUALNET	99
8.3 - Llibreria de models per a OPNET	101

PART I

5.1 – INTRODUCCIO A LA SIMULACIO DE SISTEMES

Sistemes : components i tipus

Models : modelització, tipus i procés de construcció : matemàtic

Simulació : tècniques, avantatges i inconvenients

Projecte de simulació : cicle de vida i etapes

Simulació d'events discrets : camps d'aplicació

Un sistema és un conjunt d'objectes o elements que interactuen entre sí, i depenen els uns dels altres, amb la finalitat d'aconseguir un objectiu.

Els components bàsics d'un sistema son :

- Entitat : és un objecte d'interès en un sistema.
- Atribut : és la propietat d'una entitat. Una entitat pot tenir diferents atributs.
- Activitat : és el procés que provoca canvis en un sistema.

Es poden classificar, segons on tinguin lloc, en endògena o interna (te lloc dins del sistema), o exògena o externa (te lloc en el medi ambient i afecten al sistema). Un sistema és tancat si no té activitats exògenes i obert si les té.

També es poden classificar segons com es puguin descriure, en determinista (es possible descriure completament el resultat de l'activitat en termes de la seva entrada), o estocàstica (no es possible descriure completament el resultat de l'activitat degut al seu caràcter aleatori).

Estat del sistema és la descripció de les entitats, atributs i activitats en un instant de temps donat. El progrés d'un sistema s'estudia seguint els canvis en l'estat d'aquest. Entenem com a medi ambient del sistema, l'exterior d'aquest. Es important establir els límits entre el medi ambient i el sistema.

Segons el comportament d'un sistema en el domini temporal, es poden classificar en :

- Continus : les variables de l'estat del sistema evolucionen de manera continua al llarg del temps.

- Discrets : les variables de l'estat del sistema evolucionen només en certs instants al llarg del temps.
- Orientats a events discrets : al igual que els discrets, les propietats del sistema canvien únicament en una seqüència d'instants de temps (que obeeix normalment a un patró aleatori), i romanen constants la resta del temps. Per exemple el peatge d'una autopista, on arriben vehicles de manera aleatòria i en el qual es desitja preveure el temps d'espera dels cotxes en les cues en funció del nombre de guixetes obertes.
- Combinats : combinen subsistemes que segueixen filosofies continues o discretes, respectivament.

Un model és la descripció de les característiques d'interès d'un sistema (mecanismes) per tal de preveure, estudiar i analitzar el seu comportament. Es un conjunt de variables juntament amb equacions matemàtiques que les relacionen, i restriccions sobre aquestes variables. No existeix un únic model per a un sistema, ja que depèn de la naturalesa de la informació que reuneixi.

Per a garantir una representació eficient del sistema real, hi ha una sèrie de consideracions bàsiques a tenir presents :

- Un model es desenvolupa sempre a partir d'una sèrie d'aproximacions i hipòtesis, pel que representa parcialment la realitat.
- Un model es crea amb una finalitat específica i ha de ser formulat per a ser útil a aquesta finalitat.
- Un model ha de ser per necessitat un compromís entre la simplicitat i la necessitat de recollir tots els aspectes essencials del sistema en estudi.

La modelització és una etapa present en la major part dels treballs d'investigació. En moltes ocasions, la realitat es bastant complexa com per a ser estudiada directament i es preferible la formulació d'un model que contingui les variables mes rellevants que es donen en el sistema objecte de l'estudi, i les relacions mes importants entre elles.

Els models es poden classificar de diferents maneres :

- Estàtics respecte a dinàmics : Estàtics per a representar el sistema en un cert moment en el temps, i en la seva formulació no es considera l'avanç en el temps. Es molt útil quan el sistema es troba en equilibri (no evoluciona respecte al temps).
Els dinàmics permeten deduir com les variables evolucionen respecte al temps.
- Deterministes respecte a estocàstics : Deterministes si el seu nou estat pot ser completament definit a partir de l'estat anterior i de les seves entrades. Ofereix un únic conjunt de valors de sortida per a un conjunt d'entrades conegudes.

Els estocàstics requereixen d'una o mes variables aleatòries per a formalitzar les dinàmiques d'interès. El model no genera un únic conjunt de sortida.

- Continus respecte a discrets : Continus per a representar l'evolució de les variables d'interès de forma contínua, i discrets per a representar l'evolució de les variables d'interès de forma discreta.

Existeixen molts tipus de models (físics, mentals, simbòlics, etc.), però per a ser usats en entorns de simulació per ordinador, es necessari que formalitzin el coneixement del sistema de manera concisa i clara, sense ambigüitats, i que puguin ser processats per un ordinador.

Aquestes característiques determinen l'ús de models simbòlics matemàtics com a eina per a representar les dinàmiques d'interès de qualsevol sistema en un entorn de simulació. Les entitats del sistema i els seus atributs es representen amb funcions matemàtiques que relacionen les variables.

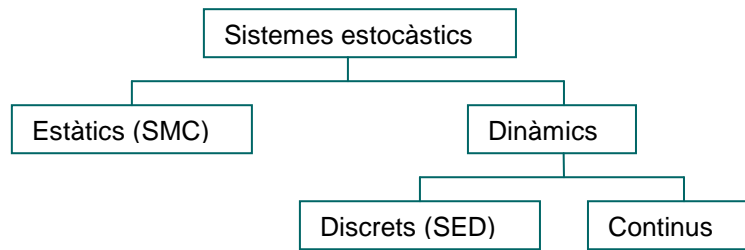


Figura 1 – Classificació dels sistemes estocàstics

Els models de simulació d'events discrets son models dinàmics, estocàstics i discrets en els que les variables d'estat canvien de valor en instants no periòdics del temps. Aquests instants de temps es corresponen amb l'ocurrència d'un event. Un event es defineix com a una acció instantània que pot canviar d'estat el model.

Freqüentment, la resolució dels problemes que es pretenen abordar pot realitzar-se per procediments analítics sobre el model construït (a través d'eines matemàtiques). En altres circumstancies, aquesta resolució analítica no és possible (o complicada o costosa) i es preferible una aproximació de la solució a través de la simulació.

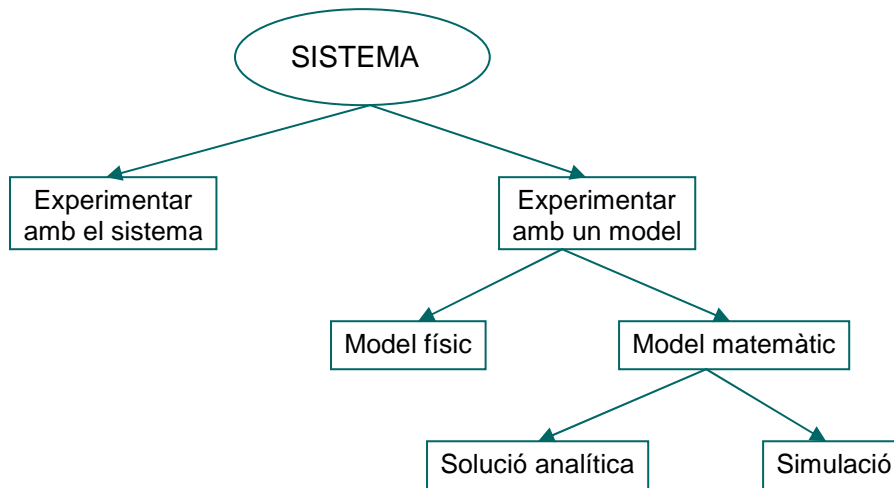


Figura 2 – Esquema de les possibilitats d'experimentació d'un sistema

Es la tècnica que consisteix en realitzar experiments de mostreig sobre un model d'un sistema. La simulació digital, és la tècnica que permet imitar (o simular) en un ordinador el comportament d'un sistema físic o teòric segons determinades condicions particulars d'operació.

La simulació digital d'un sistema orientat a events discrets (Law, Kelton; 2000) es pot fer utilitzant llenguatges de programació de propòsit general (per exemple Pascal, C++ o Java) o amb llenguatges de simulació (com el GPSS, Simscript o Slamsystem) o mitjançant simuladors.

La principal característica d'aquests llenguatges es que ofereixen una llibreria d'instruccions prou ampla i completa per a poder programar qualsevol tipus de model. En general, la codificació del model requereix una inversió de temps i de personal considerable.

El principal objectiu dels entorns de simulació és facilitar l'ús de la simulació en la fase de programació del model, restringint els camps d'aplicació. Mitjançant aquests entorns es redueix considerablement la tasca de programació.

Els avantatges mes destacables que presenten les tècniques de simulació son (Carson; 2003):

La simulació és aplicable en els casos en que no es pot portar a terme una resolució analítica o aquesta és molt complicada o costosa. S'experimenta amb un model matemàtic del sistema, en lloc del propi sistema.

Permet experimentar abans de que existeixi el sistema o quan no es pot experimentar sobre el sistema real al ser aquesta experimentació destructiva. També permet experimentar quan l'experimentació sobre el sistema real és possible però no ètica. Es de molta utilitat en aquells sistemes que evolucionen molt lentament al llarg del temps.

Permet obtenir resultats numèrics sobre el comportament del sistema per a una millor comprensió sobre la dinàmica interna del sistema. També permet comprimir el temps necessari per a comprovar el funcionament del sistema degut a que es treballa amb un rellotge de simulació virtual (és pot manipular).

Les tècniques de simulació poden ser generalment utilitzades com a metodologia segura i econòmica que permet respondre molt satisfactòriament preguntes del tipus “que passaria si es fes aquest canvi en ...”, i contribueix a la reducció del risc inherent a la presa de decisions.

Per altra banda, presenta alguns inconvenients (Banks; 2001), com son el temps i dedicació que pot prendre la construcció d'un bon model, la freqüent ommissió de variables o relacions importants entre elles i la dificultat de conèixer la precisió de la simulació, especialment en lo relatiu al procés de formulació.

Les persones involucrades en el desenvolupament i manteniment de models de simulació, estan d'acord en manifestar clarament que els models simples son preferibles als models complexos, tot i la potencia de càlcul dels ordinadors actuals.

Alguns dels avantatges destacables de treballar amb un model senzill son :

- Son mes fàcils d'implementar, validar i analitzar.
- Es menys perjudicial refusar un model simple que un de complexa en el que s'ha invertit una gran quantitat de temps.
- Es mes fàcil de canviar si les condicions o hipòtesi d'operació sobre el sistema canvien.
- Permet oferir resultats correctament vàlids abans de les dates d'entrega.

Un projecte de simulació és dinàmic per naturalesa. Els resultats s'obtenen a mesura que es va desenvolupant, posant de manifest nous problemes i/o limitacions, provocant molts cop la reconsideració de la orientació presa al principi.

Les etapes d'un projecte de simulació son (Banks; 1999) :

- Formulació del problema : es defineix el problema que es pretén estudiar i els objectius.
- Disseny del model conceptual : s'especifica el model a partir de les característiques dels elements del sistema que es vol estudiar i les seves interaccions.
- Construcció del model : a partir del model conceptual i les dades.
- Verificació i validació : es comprova que el model es comporta com era d'esperar i que existeix una correspondència correcte entre el sistema real i el model.
- Anàlisi : dels resultats de la simulació amb la finalitat de detectar problemes i recomanar millores i/o solucions.
- Documentació : proporcionar documentació sobre el treball realitzat.
- Implementació : posar en pràctica les decisions efectuades.

Les aplicacions de la simulació compren una gran quantitat d'àmbits i sectors: l'industrial ([Banks; 1998](#)), el militar, l'informàtic i de telecomunicacions, el de serveis, econòmic-financer, el social, etc.

Adreçant-se als articles publicats en la Winter Simulation Conference de diferents anys, [WSC](#), es pot obtenir una perspectiva més exhaustiva dels diferents camps d'aplicació de la simulació.

Exemples més concrets son :

- Fabricació de processos : s'utilitza tant en el disseny com per a l'ajuda de presa de decisions operacionals.
- Logística : contribueix de manera significativa a la millora dels processos, inclosos la gestió d'inventaris i la cadena de subministraments.
- Transport : : ferroviari, aeri o terrestre. Per exemple, la interconnexió de diferents mitjans de transport o la simulació visual de models de trànsit.
- Sanitat : s'utilitza per a la millora de, per exemple, un departament hospitalari, o per a la coordinació metge d'una regió.
- Negocis : s'utilitza per a la simulació dels processos administratius i de negoci d'una empresa.
- Construcció : s'utilitza, per exemple, per a la planificació de la construcció d'una obra civil o la gestió de recursos.
- Emergències : s'utilitza per a la gestió de la emergència, per exemple el pla d'evacuació d'edificis.
- Telecomunicacions : s'utilitza per a avaluar i comparar dissenys alternatius.
- Serveis en general : públics, gestió de restaurants, banca, asseguradores, ...

5.2 – MODELAT DE SISTEMES ORIENTATS A EVENTS DISCRETS

Formalització de models conceptuals : especificació formal

Models de flux

Xarxes de cues

Xarxes de Petri

Xarxes Petricolorejades

En els apartats anteriors s'ha justificat la necessitat d'obtenir un model conceptual dels processos d'interès del sistema previ a la recollida de dades i a la construcció del model. Es necessari un procés d'abstracció que permeti formalitzar tot el coneixement que es té del comportament del sistema en un model fàcil d'interpretar.

Per a formalitzar un model conceptual es necessari respectar algunes propietats ([Overstreet, CM; 1982](#)):

- El formalisme ha de ser independent dels constructors i eines que ens ofereixen els entorns de simulació.
- El model formalitzat ha de poder ser analitzat per a determinar relacions entre components i avaluar alternatives que permetin la simplificació del model.
- El formalisme ha de permetre una fàcil transformació a les representacions suportades pels entorns de simulació.
- Alguns aspectes del model poden deixar-se sense especificar.

El model conceptual es pot formalitzar de diferents maneres, a través de : nivells de flux, xarxes de cues, xarxes de petri i xarxes de petricolorejades.

Els models de flux, representen gràficament el conjunt d'activitats que formen una tasca. El seu avantatge principal es la seva simplicitat ([Harrington, HJ; 1991](#)). En cada activitat, s'ha d'especificar ([Ortner, Stary; 1998](#)):

- Precondicions i postcondicions s'han de complir a l'inici d'una activitat i al finalitzar.
- Qui té control sobre l'activitat.
- Quines son les altres activitats requerides per a completar-la.
- Entrada/sortida de l'activitat : les dades i el flux d'informació necessaris per a la realització d'una tasca.

El principal problema dels diagrames de flux es que no contempen de forma explícita el conjunt de recursos necessaris per al correcte funcionament del procés.

En les xarxes de cues, es representen els diferents servidors on es duren a terme les activitats que es desenvolupen en el sistema mitjançant un conjunt de cues.

Els elements bàsics que formen un procés de cues son ([Kendall; 1953](#)) :

- Entitats temporals que competeixen entre sí per a capturar els recursos.
- Els recursos que proporcionen un servei a les entitats temporals.
- L'espai d'espera de les entitats temporals quan el recurs al que desitgen accedir està ocupat. Aquest espai s'anomena cua.

Permeten formalitzar de manera molt natural clients o peticions, recursos i processos presents en gran part dels sistemes.

Segons el nombre de cues i el nombre de servidors, es poden fer els diagrames dels tipus d'estructures típiques :

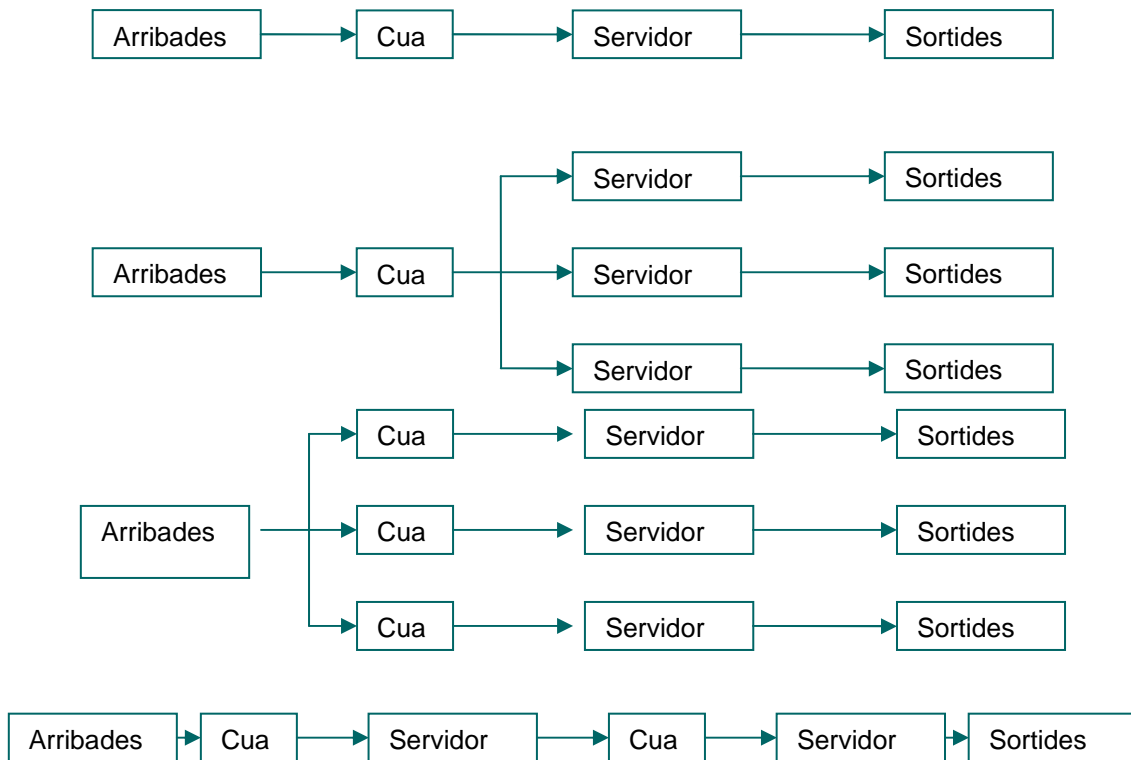


Figura 3 - Una cua / un servidor
 una cua / múltiples servidors
 varies cues / múltiples servidors
 una cua / servidors seqüencials

Les xarxes de Petri (Petri CA; 1966) permeten representar de forma natural un model. Els events estan associats a transicions i les activitats a llocs. Per altre banda, permet determinar de manera immediata tots aquells events que poden sorgir quan el sistema es troba en un estat determinat i tots els events que poden sorgir per aquest nou event.

Es defineix com un graf dirigit bipartit, amb un lloc inicial anomenat marcació inicial. Els dos components principals de la xarxa de Petri son el llocs i les transicions. Gràficament, els llocs es dibuixen com a cercles i les transicions com a barres o rectangles. Les arestes del graf s'anomenen arcs i tenen un pes específic (enter positiu) i van d'estat a transició i a l'inrevés.

Les xarxes petricolorejades, pertanyen a la família de les RdP. La seva diferencia ve marcada per les consideracions en RdPC de colors i funcions lineals associades als seus arcs. Tenen la capacitat d'associar a cada objecte (marca) un tipus de dades (conjunt de valors) denominat color de la marca.

5.3 – MODELS ESTADÍSTICS DE SIMULACIO

Conceptes bàsics

Desenvolupament del model

Adquisició i anàlisi de dades

Ajust de la funció de distribució

Validació de l'ajust d'una funció de distribució

Generació de mostres de distribucions aleatòries : generació de nombres pseudo-aleatoris uniformes en (0,1)

Els models estadístics de simulació, s'utilitzen per a representar aquelles activitats amb efectes que no poden ser descrits completament en termes de les entrades i de l'estat actual del sistema (processos estocàstics), es a dir, que necessiten de variables aleatòries per a formalitzar l'evolució del sistema que es representa.

Conceptes bàsics per a la construcció i anàlisi d'un model d'un procés aleatori :

- Experiment : procés amb resultats no coneguts amb seguretat. Els conjunt de possibles resultats d'un experiment s'anomena espai de mostreig S . Un resultat possible s'anomena punt de mostreig en l'espai de mostreig.
- Variable aleatòria : X ho és si pot tenir assignat qualsevol valor (no previsible) d'un rang finit (variable aleatòria discreta) o infinit (variable aleatòria continua) de possibles valors. Es una funció que assigna un número a cada possible resultat. Encara que la seqüència exacta de valors que seran assignats a una variable aleatòria, no pot ser prevista, si que es possible conèixer els rang de valors en els que pot variar, així com la probabilitat de tenir assignat un cert valor.
- Distribució de probabilitat : permet relacionar un conjunt de valors o mesures amb la seva freqüència relativa d'aparició.
- Funció de densitat de probabilitat $f(x)$: descriu la probabilitat que una variable aleatòria X assumeixi un cert valor C .
- Funció de distribució acumulativa $F(x)$: descriu la probabilitat de que una variable aleatòria X prengui un valor mes petit o igual que un cert valor x_i .

En el desenvolupament d'un model, les variables d'interès d'un sistema que no es poden preveure amb exactitud, és modelaran com a processos estocàstics, definint un rang de valors que pot assumir la variable aleatòria associada al procés estocàstic, així com la probabilitat de prendre cada un dels valors possibles.

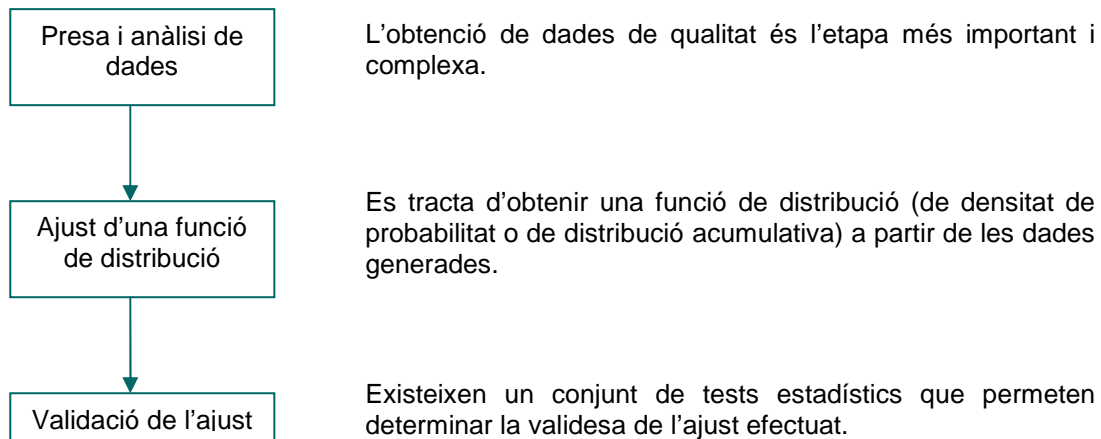


Figura 4 – Desenvolupament d'un model estadístic (Prat; 1997)

S'ha de determinar i buscar la informació necessària i les fonts que les puguin proporcionar. Si la informació obtinguda és insuficient, s'hauran d'efectuar hipòtesis raonables per a prosseguir l'estudi.

Un dels problemes més usuals en aquesta etapa, és que les dades no es representen en el format necessari per a definir el model.

En el modelat d'un component aleatori d'un sistema, tan sols han de registrar-se les dades referents al procés estocàstic, sense considerar les causes ni l'activitat aleatòria ni les seves conseqüències.

En general, se suposa que els valors aleatoris de la mostra son independents i estan idènticament distribuïts (IID). Això suposa que tots els valors de la mostra segueixen una distribució de probabilitat.

Dues de les tècniques heurístiques mes utilitzades per a avaluar la independència dels valors d'una mostra son :

- Gràfiques d'autocorrelació : la independència es determina calculant els valors d'autocorrelació per a diferents valors de retard (correlació separades per k valors).
- Diagrama de dispersió : és un gràfic. Es recomanen mes de 50 parells de valors per a treure conclusions (Prat: 1997).

La distribució de la probabilitat, permet relacionar un conjunt de valors o mesures amb la seva freqüència relativa d'aparició.

Distribucions de probabilitat empíriques (Guash; 2002) :

- Si els valors d'una mostra son continus, la distribució de la probabilitat es pot descriure a través d'un histograma. Per a obtenir-lo, la mostra de valors s'agrupa en intervals disjunts adjacents.
- Si els valors d'una mostra son discrets, la distribució de la probabilitat es pot descriure a través d'un diagrama de barres. Per a cada possible valor de la mostra x_i , l'alçada de la barra corresponent p_i serà la proporció de valors de la mostra que son iguals a x_i .

Distribucions de probabilitat teòriques (funcions més utilitzades) (Guash; 2002) :

- Continues : funció de distribució exponencial, funció de distribució gamma, funció de distribució Lognormal, funció de distribució Normal, funció de distribució Uniforme, funció de distribució Weibull.
- Discretes : funció de distribució de Bernoulli, funció de distribució Uniforme discreta, funció de distribució Binomial, funció de distribució de Poisson, funció de distribució geomètrica.

Quan el procés estocàstic es continu i no es troba limitat a valors discrets, la variable aleatòria pot prendre un nombre infinit de valors possibles en el rang d'interès. En aquest casos, per a descriure la probabilitat de que la variable aleatòria prengui un valor determinat, s'utilitza la funció de densitat de probabilitat.

A partir de la distribució de probabilitat es pot calcular la distribució acumulativa afegint per a cada interval la probabilitat de que la variable aleatòria prengui el valor en algun interval inferior o igual al que s'està avaluant.

En l'actualitat, aquestes tècniques estan automatitzades a través de nombroses eines estadístiques que seleccionen un conjunt de distribucions, las parametrizen i les prioritzen per ordre de qualitat d'ajust.

Les tècniques bàsiques son dues :

- Tècniques basades en test d'hipòtesis, una regla que permet acceptar o refusar una hipòtesi (per exemple, test de Chi-Cuadrat, test de Kolmogorov-Smirnov).
- Tècniques heurístiques, que utilitzen representacions gràfiques.

La generació de mostres aleatòries que segueixen una determinada distribució de probabilitat, es realitza en dues passes (Guash; 2002) :

- Generar una variable aleatòria u_i que segueixi una distribució uniforme entre 0 i 1 ($U[0,1]$). Per a obtenir aquesta seqüència de nombres aleatoris uniformement distribuïts s'utilitzen generadors de variables aleatòries (GNA). Actualment s'utilitzen mètodes recursius pels GNA. Exemples son els generadors congruencials lineals (GCL) i els generadors congruencials multiplicadors (GCM) que encara son més eficients que els anteriors.

- Transformar el valor u_i en un altre que segueixi la distribució escollida mitjançant un generador de variables aleatòries (GVA). L'algorisme més utilitzat es el mètode de la transformada inversa.

Per a la utilització de nombres aleatoris en la simulació, s'ha de tenir en compte :

- La independència estadística : que la mostra generada sigui independent i idènticament distribuïda (IID).
- La densitat de valors generats : propietat que té relació amb la dimensió del període i del cicle de la sèrie generada.
- La eficiència : propietat que té relació amb el temps d'inici i d'execució.
- La reproductibilitat : poder reproduir de forma fàcil i sistemàtica qualsevol sèrie de nombres aleatoris, a través de mètodes de generació que ho permetin.

Per a avaluar la qualitat de la generació de nombres aleatoris, existeixen gran quantitat de test ([Cao, 2002](#)), per exemple :

- Test de freqüències o de Chi-quadrat : comproven l'aleatorietat global. Per a funcions discretes o discretitzables.
- Test sèrie : detecta zones sense recobrir.
- Test de gap : detecta distribucions no esperades de pautes.
- Test de yule-walker : compara la suma de cinc dígit amb la suma esperada. Detecta pautes.
- Test de distàncies : comprova l'aleatorietat global entre dígit, treballant amb la distància entre punts en el pla.
- Test de K-S : comprova l'aleatorietat global i local. Per a funcions contínues.
- Test run : s'aplica només a la prova d'independència i no a la de l'aleatorietat. Es dels qualificats com a fort.

Per a la generació de variables aleatòries existeixen diferents mètodes, depenent la seva elecció de criteris d'eficiència, qualitat o rang acceptat de paràmetres ([Guash; 2002](#)) :

- Transformada inversa : aprofita les possibilitats d'inversió de la funció de distribució. Exemple d'aplicació: distribució exponencial, geomètrica, Weibull.
- Composició : combinació convexa d'altres funcions de distribució més fàcils de mostrejar. Per exemple Box-Muller, Marsaglia.
- Convolució : representació de la variable resultat com a suma d'altres variables aleatòries. Per exemple Bernoulli.
- Acceptació/Refús : s'utilitza quan només es coneix la funció de densitat de probabilitat i no es pot invertir la funció de distribució acumulada.
- Caracterització : aprofita les propietats estadístiques o matemàtiques de la distribució. Per exemple, la distribució lognormal, Poisson pel mètode multiplicatiu.
- Comparació : compara entre series $U(0,1)$. Per exemple, per a la distribució exponencial.

5.4 – INTRODUCCIO A LA TEORIA DE CUES

Introducció : nocions bàsiques

El procés de Poisson i la distribució exponencial

Variables associades a un sistema de cues.

Característiques dels models de cua

Principi de la teoria de cues

Tipus de cues

Xarxes de cues

La teoria de cues va sorgir a començaments del segle passat per a estudiar els problemes de tràfic que es presentaven en les comunicacions telefòniques. Entre 1903 i 1905 Erlang és el primer en tractar aquest problema de forma científica, i estableix la unitat de tràfic telefònic. Posteriorment, l'àmbit d'aplicació de la teoria de cues ha estat enorme ([Banks; 2001](#)): esperes per a ser atesos en establiments comercials, esperes per a processar programes informàtics, esperes per poder creuar un carrer amb un vehicle en una ciutat, esperes per establir una comunicació o recepció d'informació d'un servidor web, a través d'Internet, i moltes d'altres.

En general, un sistema de cues consisteix en un o diversos servidors que donen servei a un o mes usuaris que accedeixen al sistema. El procés d'arribades el regula una font generadora d'usuaris i les arribades seran de forma aleatòria. Aquesta font generadora d'usuaris pot ser finita o infinita.

Per a modelar correctament un sistema d'una cua, necessitem identificar els següents elements ([Cao; 2002](#)) :

- Font d'entrada : conjunt d'usuaris que poden arribar a sol·licitar el servei.
- Client : tot usuari de la font d'entrada que sol·liciti servei.
- Capacitat de la cua : el nombre màxim de clients que poden estar fent cua.
- Disciplina de la cua : quan a la cua hi ha més d'un usuari, quan el servidor queda lliure ha de determinar quin dels usuaris en espera serà el que rebí el servei. Els models més importants de disciplina de cues son :
 - FIFO (first-in-first-out) : se li dona servei al primer que ha arribat. La cua està ordenada per ordre d'arribada.
 - LIFO (last-in-last-out) : se li dona servei a l'últim que ha arribat. La cua està ordenada en sentit invers a l'arribada dels usuaris.
 - RSS o SIRO (service-in-random-order) : selecciona els clients de forma aleatòria.
 - RR (round robin) : se li dona un espai de temps de servei a cada usuari. Només es dona en algunes circumstàncies, sobretot en l'àmbit de la informàtica.

- Mecanisme de servei : es el procediment pel qual se li dona servei als clients que ho sol·liciten. S'ha de conèixer el nombre de servidors del mecanisme o la distribució de probabilitat si fos aleatori, i la distribució de probabilitat del temps que porta a cada servidor donar un servei.
- La cua : conjunt de clients en espera (que ja han sol·licitat el servei).
- Sistema de la cua : és el conjunt format per la cua i el mecanisme de servei, juntament amb la disciplina de la cua.

Els usuaris que surten del sistema, poden fer-ho a l'exterior o poden integrar-se en un altre sistema similar, llavors, es parla de cues entrelaçades o xarxes de cues.

En la majoria dels sistemes de cues, el procés d'arribades segueix un distribució de Poisson (Cao; 2002). Es demostra que si es dona aquesta circumstancia, la duració dels intervals entre arribades té una distribució exponencial o una combinació continua d'exponencials, es a dir, una distribució gamma, que rep el nom de distribució erlangiana, o distribució K.

En un model de cues, el temps no és ni continu ni ho fa en les passes discretes equidistants, però procedeix event a event. S'organitzen en una llista tots els possibles events que puguin predeterminar en un moment donat. Els events passat son esborrats de la llista, i els events poden generar d'altres i posar-los a la llista. Quan es fa referència als models de cua, hi almenys tres tipus diferents d'objectes : servidors, clients i cues. Tècnicament hi ha un d'addicional que guarda la pista dels events i els organitza. En els models de cua, el temps entre les arribades així com els temps emprat per a servir a un client és aleatori, seguint una certa distribució aleatòria. Els models d'events discrets són dinàmics : els estats de servidors, les cues i els clients depenen dels estats anteriors.

Principi de la teoria de cues : una font genera nous clients que arriben al sistema, aquest fan cua davant d'un servidor on esperen fins a rebre el servei del servidor. Un cop servits, abandonen el sistema.

Les cues i els servidors son objectes estàtics o recursos, mentre que els clients son objectes dinàmics. Segons les combinacions de diferents alternatives realitzades per a un model en concret, alguns tipus de models tenen solució analítica. Paràmetres interessants son :

- Nombre d'objectes en el sistema.
- Nombre d'objectes en cua.
- Nombre d'objectes que estan sent servits.
- Temps d'espera.
- Temps d'espera en cua.
- Temps d'espera en el servidor.

En els casos simples, aquests valors es poden calcular analíticament, però en els casos complexos han de determinar-se a través de la simulació. Per a ser analíticament factibles, els models de cua han de complir les condicions següents : que la taxa d'arribada sigui més baixa que la taxa de servei, i que l'arribada i els processos d'extracció obeeixin les distribucions particulars.

Els models més habituals per a un sistema de cua, en la que se suposa que la distribució del temps entre arribades i la del temps de servei són exponencials, són :

- **M/M/1** : en el que la distribució del temps entre dos arribades consecutives de clients al sistema és una $\exp(\lambda)$, independentment del nombre de clients que hi hagi dins del mateix, la distribució del temps de servei és $\exp(\mu)$ i només hi ha un servidor. Per defecte no hi ha restriccions al nombre de clients en cua, la font és infinita, la disciplina de la cua és FIFO.
- **M/M/S** : és la generalització del model anterior en el cas que hi hagi s servidors i amb les mateixes característiques per defecte.
- **M/M/1/K** : es tracta del model M/M/1 però amb limitació K per a la capacitat de la cua. La font és infinita i la disciplina de la cua FIFO.
- **M/M/1/s/K** : és una extensió del model anterior en el cas que existeixi un nombre sencer qualsevol, s , de servidors. Les taxes d'arribada són gairebé iguals al model M/M/1, mentre que les de servei són iguals a M/M/s.
- **M/M/1/∞/H** : la font és finita d' H clients potencials. Hi ha un únic servidor, la disciplina de cua és FIFO i la cua no té límit.
- **M/M/s/∞/H (sense recanvis)** : generalitza l'anterior a un nombre sencer, s , qualsevol de servidors. La font és finita H , cua il·limitada amb disciplina FIFO.
- **M/M/s/∞/H (amb recanvis)** : és una modificació de l'anterior que respon a la idea d'ocupar l'espai buit que deixa un client en la font potencial, a l'entrar en el sistema, amb un nou client (recanvi), que en un principi no s'hi trobava.
- **M/M/∞** : es tracta d'un model amb infinits servidors, distribució del temps entre dos arribades consecutives de clients al sistema és una \exp i la distribució del temps de servei és $\exp(\mu)$, sense limitació de cua, font infinita i disciplina FIFO.

Una xarxa de cues és una xarxa en la que cada node està constituït pel sistema com una cua (amb un o més servidors). Es tracta d'un graf orientat en el que es poden produir transicions de clients que surten servits d'un node (cua) cap a un altre node.

Aquestes xarxes serveixen per a modelitzar aquells sistemes en que els usuaris necessiten ser atesos per diversos servidors diferents. Un exemple clar són les xarxes d'ordinadors.

Les xarxes de cues poden ser de diferents tipus :

- Obertes : aquelles en les que es poden donar arribades de clients des de fora de la xarxa i sortides de clients a fora de la xarxa.

Una xarxa de Jackson oberta, és una xarxa de cues oberta que verifica tres propietats :

- Cada node té un mecanisme de servei consistent en s_i servidors amb temps de servei de idèntica distribució, exponencial de paràmetre μ_i .
- Els clients que arriben al node i des de fora del sistema ho fan segons un procés de Poisson d'intensitat λ_i .
- El flux de clients segueix l'esquema de transicions instantànies aleatòries.

La manera de poder trobar les diferents quantitats d'interès en una xarxa de Jackson oberta ve donada pel Teorema de Jackson.

- Tancades : aquelles en les que no es poden donar arribades de clients des de fora de la xarxa i sortides de clients a fora de la xarxa.

Una xarxa de Jackson tancada, és una xarxa de cues tancada amb K nodes o subsistemes, en la que cada node té s_i servidors en el seu mecanisme de servei, sent tots els del node i amb temps de servei de distribució, exponencial de paràmetre μ_i . Com en qualsevol xarxa tancada, al no haver entrades ni sortides de clients, és indispensable especificar el nombre de clients dins de la xarxa, N , que romandrà constant per sempre.

- Cues en sèrie : es tracta d'una col·lecció de K cues que es succeeixen les unes a les altres de tal manera que només es possible l'entrada de clients des de fora del sistema a la primera d'elles, produint-se la sortida després del servei de l'últim node.
- Xarxa de cues cíclica : es tenen K subsistemes de cues tipus $M/M/s_i$ (com l'anterior) però connectades de forma circular. Es tracta, per tant, d'una xarxa de Jackson tancada en la que cada client que surt d'un node, es dirigeix al següent amb una probabilitat 1.
- Cues en sèrie amb bloqueig : es tracta de cues en sèrie però amb capacitat limitada per a algunes o totes les cues.

5.5 – SIMULACIÓ DE SISTEMES ORIENTATS A EVENTS DISCRETS

Elements d'un model de simulació

Elements del simulador

Estratègies de simulació

Entorns de simulació d'events discrets

Per a poder construir, simular i analitzar un model, en la major part dels projectes de simulació, l'usuari final tan sols necessita coneixements sobre aspectes d'interfície del simulador. Però també és important el coneixement els mecanismes interns dels simuladors a fi de facilitar el modelat de situacions complexes i la detecció, comprensió i correcció d'errors.

Els elements d'un model de simulació (Guash; 2002) son :

- Entitats permanents o recursos : son elements estàtics ja que el seu nombre no canvia al llarg de la simulació. S'utilitzen per a descriure els mitjans gracies als quals es poden executar les activitats. Defineixen també qui o que executa l'activitat. Es parametriza per característiques com capacitat, velocitat o temps de cicle.
- Entitats temporals : son elements que es creen i es destrueixen al llarg de la simulació. S'utilitzen per a descriure els objectes que es processen en el sistema, com poden ser clients o documents. També son els objectes que arriben, es processen i surten del sistema.
- Atributs : permeten caracteritzar a les entitats. Cada atribut correspon a una propietat (per exemple el tamany), així com tota informació que hi hagi en el sistema referent a les entitats temporals (per exemple la posició). Son imprescindibles per al control de flux d'entitats del sistema.
- Activitats : son les accions que tenen lloc en el sistema. Generalment involucren a mes d'una entitat, pel que el seu inici està lligat a la presencia de totes les entitats involucrades. Un aspecte essencial és la durada de l'activitat, que ha de ser conegut per a poder determinar l'instant de finalització d'aquesta.
- Events : son accions instantànies (no consumeixen temps) i que poden canviar el valor d'una variable d'estat del sistema modelat. Normalment, en cada event s'inicien o finalitzen mes d'una activitat. Dues maneres de classificar els events :
 - Condicionats i no condicionats : per a que s'activin els condicionats es necessari que s'acompleixin una o mes condicions, mentre que els no condicionats estan planificats per a la seva execució i no depenen de condicions.
 - Interns o externs : els interns estan causats per condicions en el model i els externs no.
- Cues : estructura determinada a partir d'una col·lecció d'entitats, ordenada de forma lògica. Les entitats que estan en una cua pateixen un retard de durada indeterminada.

La major part dels simuladors tenen implementat els següents mòduls ([Law, Kelton; 2000](#)):

- Generador de nombres pseudo-aleatoris uniformes en $U(0,1)$: generadors GCL o GCM.
- Generador de variables aleatòries : generadors GVA.
- Relotge de la simulació : variable que emmagatzema el temps de simulació.
- Llista d'events futurs (LEF) : llista que emmagatzema els instants de temps quan han d'activar-se els events no condicionats.
- Mòduls per a la generació d'estadístiques : conjunt de variables i rutines per a obtenir les mesures de comportament d'interès.
- Mòdul d'inicialització : executa el conjunt de tasques necessàries per a inicialitzar el model i el simulador.
- Rutina d'actualització del rellotge del simulador : subprograma que determina l'instant de temps en el que s'ha d'activar l'event següent, així com el que ha de sortir disparat.

- Rutina de tractament de l'event : existeix una per a cada tipus d'event, que actualitza els valors d'alguna de les variables d'estat i pot generar noves entrades a la llista d'events futurs.
- Codi executiu : controla l'execució de la simulació.

Estat del sistema : col·lecció de variables necessàries per a descriure el sistema en qualsevol instant de temps.

Una de les característiques mes importants i determinants de les eines de simulació, és l'estratègia per a la selecció de l'event següent.

La majoria dels simuladors implementen diverses estratègies de simulació amb l'objectiu d'aprofitar el potencial de cada una. Tres estratègies prou importants son :

- Programació d'events (Event Scheduling) : la principal característica es que tan sols es poden codificar les rutines de tractament d'events no condicionats, pel que el codi associat als events condicionat s'ha d'incloure en el codi d'events no condicionats.
Després d'inicialitzar el model, el codi executiu explora tots els futurs events no condicionats i mou el rellotge de la simulació fins l'event no condicionat més proper en el temps. A continuació, crida al codi associat a l'event per a efectuar els canvis en el model provocats per l'event.
El mecanisme que permet que tots els events s'executin en ordre cronològic es basa en la LEF. El contingut i la longitud de la LEF canvien contínuament al llarg de la simulació.
- Exploració d'activitats (Activity Scanning) :
Els models son desenvolupats a partir del conjunt d'activitats que s'han d'executar i l'estratègia radica en la identificació de la natura de les entitats, dels conjunts d'entitats (temporals o permanents) i del conjunt de condicions que permetin que succeeixi l'activitat. L'exploració d'activitats utilitza un increment fix del temps i en cada instant de temps explora totes les activitats per a decidir si alguna d'elles pot començar.
- Interacció de processos (Process Interaction) :
Es centra en els processos que pateix l'entitat a mesura que flueix pel sistema i en el seu cicle de vida. D'aquesta manera es diferencien clarament les entitats temporals de les permanents (o recursos).
Els models es descriuen com a diagrames de blocs on les entitats temporals flueixen a mesura que avancen en el seu cicle de vida. Una entitat temporal es pot trobar en diferents estat al llarg del seu cicle : activa, demorada, parada.

Algunes de les eines de simulació disponibles en el mercat :

Arena	Rockwell Software	www.rockwellsoftware.com
Automod	Autosimulations, Inc	www.autosim.com
GPSS/H	Wolverine Soft Corp	www.wolverinesoftware.com
Quest	Delmia Corp	www.delmia.com
MicroSaint	Micro Analysis Design, Inc	www.madboulder.com
ProModel	Promodel Corp	www.promodel.com
Simul8	Simul8 Corp	www.simul8.com
FlexSim	Flexim Soft Prod, Inc	www.flexsim.com
Witness	Lanner Group	www.lanner.com

5.6 – METODOLOGIA D'UN ESTUDI DE SIMULACIÓ

Passes a seguir per a una correcta simulació

Fiabilitat dels resultats obtinguts. Interval de confiança

Temps de simulació i nombre d'events

Validació, verificació i credibilitat

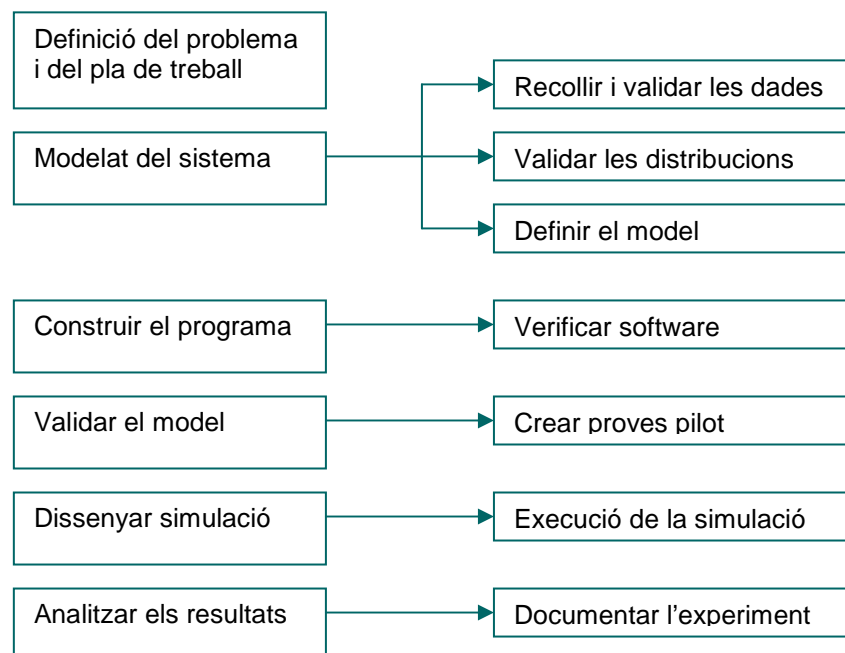


Figura 5 – Fases d'una simulació

Una correcta simulació es aquella que ens informa del que realment ens interessa sobre el model i ens dona una certa fiabilitat del resultats. Les passes a seguir son (Carson; 2003) (Banks; 1999) :

- Enunciar explícitament els objectius que es pretenen, els interrogants que es plantegen, les hipòtesis que es volen demostrar i les diferents possibilitats a considerar.
- Crear un model lògic-matemàtic. És fonamental modelar els fenòmens aleatoris del sistema i les seves característiques.
- Dissenyar o utilitzar un programa d'ordinador per a simular el model i verificar que el programa es comporta com es desitja.
- Validar el model a través de proves pilot amb diferents inputs i comparar els resultats estadísticament (Sargent; 1999). El model ha de reflectir convenientment el model real.
- Dissenyar la simulació i executar-la utilitzant el model per a respondre a les preguntes que inicialment se'ns plantejaven.
- Reunir, processar i analitzar les dades generades com a solucions del model i en termes de validesa i fiabilitat estadística. Documentar l'experiment.

S'ha de tenir present que una sola simulació proporciona un únic valor de entre molts possibles resultats, que pot ser diferent en cada simulació com a conseqüència del caràcter aleatori d'algunes de les seves variables. Per això s'utilitzen les tècniques d'inferència estadística per a establir el nombre de simulacions necessari per a proporcionar un nivell desitjat de confiança.

Aquestes tècniques, s'hauran d'usar també a l'hora de reunir dades que serveixin com a entrades. Expressem la confiança en el resultat com la probabilitat de que el valor del sistema real estigui comprès en un interval (interval de confiança) que tingui per centre el valor estimat. La determinació d'aquests intervals de confiança en les simulacions resulta prou complexa degut a que les sortides no solen ser independents i les condicions inicials poden influir considerablement en els resultats.

Si la variable de sortida s'obté com la mitja d'un conjunt de sortides obtingudes per una determinada mostra, serà més fiable quan més gran sigui en tamany mostral. D'aquesta manera, en les simulacions, té un gran rellevància paràmetres com el temps de simulació i el nombre d'events.

Aquest dos paràmetres estan íntimament lligats amb la fiabilitat amb la que podem acceptar els resultats de la simulació.

El temps de simulació permet especificar quant temps suposem que està en funcionament el sistema, per a observar el seu comportament al llarg d'aquest període, el que implicarà si estem calculant valors mitjos, per exemple, calcular la mitja del valor que pren aquest paràmetre al llarg de tot el període de simulació.

De manera intuïtiva es pot apreciar com la fiabilitat d'un resultat que faci referència a la mitja d'una certa variable, augmentarà amb el temps de simulació si la resta d'entrades es manté constant. D'igual manera el nombre d'events seran el nombre d'arribades o sortides que es

produeixin durant la simulació i d'igual manera que per al temps de simulació, s'estableix que a major nombre d'events, en iguals condicions, la fiabilitat es veu incrementada.

Per tant, es de vital importància que el nombre d'events que es succeeixin en una simulació sigui suficient per a garantir que els resultats siguin estadísticament fiables.

La dificultat mes important en un estudi de simulació, és construir un model lògic-matemàtic que representi amb exactitud el sistema real. Segons (Banks; 2001), hi ha tres passes bàsiques per a una construcció correcte del model :

- Recopilar dades i informació del sistema real.
- Dissenyar el model conceptual.
- Implementar el model conceptual en l'ordinador.

Hi ha tres conceptes importants a tenir present :

- Validació : procés per a comprovar que el model conceptual representa correctament al sistema real.
- Verificació : procés per a comprovar que la implementació en l'ordinador del model conceptual sigui correcte.
- Credibilitat : capacitat del model per a ser acceptat pels responsables del sistema.

5.7 – CONCLUSIONS

La simulació com a metodologia de treball és una activitat molt antiga i inherent al procés d'aprenentatge del ser humà.

Per a poder comprendre la realitat i tota la complexitat que un sistema pot comportar, es necessari construir artificialment models que representin aquesta realitat per a poder experimentar amb ells dinàmicament, abans d'interactuar amb el sistema real.

La simulació es proposa analitzar el comportament dels sistemes discrets que no es poden analitzar amb eines tradicionals de tipus matemàtic amb la finalitat de prendre decisions. Es centra en l'estudi de sistemes complexos que no es poden, o que no és econòmicament rentable, simular en la realitat, i es pot aplicar durant totes les fases de disseny d'un sistema discret, des de el seu disseny inicial fins al disseny detallat incloent l'explotació i el manteniment del sistema.

Presenta alguns inconvenients com son el temps i dedicació que pot prendre la construcció d'un bon model, la freqüent omisió de variables o relacions importants entre elles i la dificultat de conèixer la precisió de la simulació, especialment en lo relatiu al procés de formulació.

Per a analitzar, estudiar i millorar el comportament d'un sistema mitjançant les tècniques de simulació digital, és necessari desenvolupar primer un model conceptual que descrigui les dinàmiques d'interès i després implementar-lo en un simulador per a poder analitzar els resultats.

Aquestes tècniques tenen cada cop més rellevància en la resolució de diferents tipus de problemes pràctics. L'ús d'aquestes tècniques és un camp interdisciplinari molt ampli, tant per la varietat de sistemes que poden ser considerats com per la diversitat de contextos que es poden descriure.

Per a representar aquelles activitats els efectes de les quals no poden ser descrites completament en termes de les entrades i de l'estat actual del sistema, son de vital importància els models de simulació estocàstics (models estadístics). Aquests models permeten, mitjançant l'ús de variables aleatòries, formalitzar l'evolució del sistema.

Son molts els sistemes o camps d'aplicació en els que es desconeixen les conseqüències de l'esdeveniment d'un fet en concret, així com la influència sobre el rendiment global del procés en una variació en la seqüència d'events que es produiran com a conseqüència d'un cert event situat en algun instant anterior en el temps.

Així doncs, els camps d'aplicació són innumerables i creixents : fabricació de processos, logística, transport, sanitat, emergències, telecomunicacions, serveis en general (públics, gestió de restaurants, banca, asseguradores, etc.), negocis, construcció, etc.

Tot i que la simulació digital és relativament recent, el creixement quasi exponencial de la tecnologia, que ofereix cada cop processadors amb una capacitat i velocitat de càlcul més elevada, augura que la simulació i les tècniques de simulació digital seran d'una importància cada cop més rellevant, tant en benefici de la humanitat (previsió en cas de fenòmens naturals, malalties, canvi climàtic, desforestació, etc.) com en contra (el que alguns anomenen "defensa").

PART II

5.8 – MODELAT I SIMULACIO DE XARXES

Les xarxes de comunicacions estan incrementant contínuament la seva complexitat integrant serveis de telefonia i xarxes de dades. Com a conseqüència del ràpid creixement, el temps reduït del mercat i la fusió de tecnologies, la predicció del funcionament d'una xarxa i l'eliminació de les fallades de protocols esdevenen tasques dificultoses.

La possibilitat d'extrapolar el funcionament d'una xarxa en un experiment a escala té interès per a avaluar i comparar dissenys alternatius, amb l'objectiu d'assolir un funcionament òptim que s'ajusti a les necessitats de l'organització. Aquest anàlisi es pot dur a terme a través de models analítics, models de simulació o la mesura directe del seu comportament. A través dels models de simulació podem avaluar un sistema que encara no existeix.

Durant la fase de disseny d'una xarxa s'han d'ajustar determinades propietats i paràmetres d'aquesta (topologia, protocols, cablejat, etc.), prevenint d'aquesta manera la formació de colls d'ampolla, el col·lapse i la infrautilització.

La part més feixuga del modelat d'un sistema, és la recollida i validació de dades per a definir el model conceptual. A continuació, es relacionen les passes que entenc necessàries per a caracteritzar una xarxa, de manera que es pugui disposar de tota la informació important per a que el model representi amb la major exactitud al sistema real. En el cas de que no existeixi el sistema real, algunes passes es poden ometre, i en d'altres, la informació s'hauria "d'inventar" d'acord amb les necessitats d'interès de l'estudi de simulació.

Les passes a seguir per a una correcta caracterització d'una xarxa son :

- 1) Caracteritzar les aplicacions : nom, tipus, nombre d'usuaris i nombre de hosts o servidors.
- 2) Caracteritzar els protocols de xarxa : nom, tipus, nombre d'usuaris i nombre de hosts o servidors.
- 3) Documentar la xarxa : topologia, esquema d'adreces.
- 4) Identificar els colls d'ampolla potencials determinant el tràfic local i el no local.
Per segment de xarxa, determinar : origen i destí local, origen local i destí no local, origen no local i destí local, origen i destí no local.
- 5) Identificar interessos de l'organització i les entrades en el disseny de la xarxa : entenent l'estructura de l'organització, identificant dades crítiques o operacions i tenint en compte les polítiques financeres que poden afectar al disseny.
- 6) Documentar la disponibilitat de la xarxa : cost/hora per departament i de l'organització per interrupció de la xarxa.

- 7) Caracteritzar el funcionament de la xarxa : relacions entre hosts.
- 8) Caracteritzar la fiabilitat de la xarxa : MB totals, nombre de frames total, nombre d'errors de CRC (Cyclic redundancy check), nombre d'errors de la capa MAC (col·lisions, errors del software del Token Ring, operacions anell FDDI), nombre total de frames broadcast/multicast.
- 9) Caracteritzar l'ús de la xarxa : taxes relatives i absolutes de l'ús en la xarxa de IP, IPX, netBIOS, SNA, etc.
- 10) Caracteritzar l'estat dels routers principals : nom router, ús cpu, entrades i sortides de cua per hora, paquets perduts per hora, paquets ignorats per hora.
- 11) Caracteritzar l'administració del sistema de la xarxa documentant el tipus de plataforma i les eines d'administració de la xarxa.
- 12) Resumir la salut de la xarxa : saturacions en segments de la xarxa, % ús de la xarxa, temps de resposta, col·lisions de paquets, temps d'espera en cua, nombre de pèrdues de buffer, nombre de paquets ignorats, etc.

En el cas de disposar d'un sistema real, la recollida de les dades es pot realitzar a través de monitors, benchmarks i eines d'administració del sistema.

Per a dur a terme la simulació, com ja s'ha indicat abans en la part I, la simulació digital d'un sistema orientat a events discrets ([Law, Kelton; 2000](#)) es pot fer :

- Utilitzant llenguatges de programació de propòsit general (per exemple Pascal, C++ o Java): es disposa de major flexibilitat a l'hora de construir el model però té un elevat cost de temps de desenvolupament del projecte.
- Amb llenguatges de simulació (com el GPSS, Simscript o Slamsystem) : suposa menys temps en el desenvolupament al disposar de llibreries de classes, objectes i funcions, però es perd flexibilitat en la construcció del model.
- Mitjançant entorns de simulació : acostuma a ser la que presenta una menor flexibilitat tot i que estan dissenyats per a analitzar una tipologia concreta de sistema. Faciliten enormement el desenvolupament del projecte i estalvien molt de temps respecte als anteriors.

En el cas d'utilitzar un software de simulació, es convenient tenir present una sèrie de condicions per a escollir el software més adient per al projecte ([Banks; 1998](#)) :

- El tipus de sistema que és capaç de modelar : si es d'àmbit general o està específicament dissenyat per a simular xarxes, producció industrial, etc.
- La seva capacitat per a importar i exportar fitxers en formats estàndards.
- La seva compatibilitat amb codi extern generat en C/C++, Java o VB, i amb editors de texts, paquets estadístics, etc.

- L'existència de llibreries de classes o mòduls ja predefinits en el programa per a agilitzar el procés de modelat.
- Les seves possibilitats pel que fa referència al modelat de processos aleatoris.
- La velocitat d'execució del model resultant.
- La portabilitat a altres entorns informàtics.
- La flexibilitat a l'hora d'adaptar detallar el model.
- Els anàlisis estadístics i informes dels resultats de la simulació.
- La facilitat d'us i aprenentatge.
- El nivell i la qualitat de la documentació tècnica disponible.
- El nivell i la qualitat de l'assistència tècnica.
- El cost (diners).
- Les possibilitats gràfiques i d'animació de la simulació.

5.9 – SOFTWARE DE SIMULACIÓ DE XARXES DE COMUNICACIONS

A continuació es relacionen alguns dels paquets de software per a la simulació de xarxes de comunicacions disponibles en el mercat :

OMNET++ www.omnetpp.org/

Technical University of Budapest, Department of Telecommunications (BME-HIT)

ns2 www.isi.edu/nsnam/ns/

DARPA, SAMAN, CONSER i amb moltes col·laboracions.

GLOMOSIM <http://pcl.cs.ucla.edu/projects/glomosim/>

UCLA Parallel Computing Laboratory amb PARSEC.

SWANS <http://jist.ece.cornell.edu/>

Cornell Research Foundation, Inc.

QUALNET www.scalable-networks.com/products/qualnet.php

Scalable Network Technologies, Inc.

CNET www.csse.uwa.edu.au/cnet/

School of Computer Science & Software Engineering, The University of Western Australia.

Dr. Chris McDonald.

OPNET www.opnet.com/

OPNET Technologies, Inc.

TRAFFIC V2.0 www.erlang-software.com/

Erlang Software, Inc.

NCTUNS 2.0 <http://nsl.csie.nctu.edu.tw/nctuns.html>

Network and System Laboratory (NSL), Department of Computer Science and Information Engineering, National Chiao Tung University (NCTU), Taiwan,

GTNETS www.ece.gatech.edu/research/labs/MANIACS/GTNetS/

Georgia Institute of Technology, Atlanta

PERFORMANCE PROPHET <http://dps.uibk.ac.at/projects/prophet/>

Fundat per la [Austrian Science Fund \(FWF\)](#) com a part de l'[Aurora Project](#)

D'aquests paquets d'aplicacions, s'analitzaran, compararan i valoraran :



Instal·lant-los sobre una plataforma Windows XP Prof sp2 amb un Pentium IV a 3,2 MHz.

S'utilitzarà com a compilador de C++ el que ve integrat en Visual Studio .NET 2003. Per a la interfície gràfica de l'entorn virtual de QUALNET Developer s'utilitzarà el paquet de software Sun Java 2 SDK, Standard Edition 1.4.2.

5.10 – OMNET++

Versions/ Edicions	<p>OMNEST v3.2 (Comercial)</p> <p>OMNEST Demo v3.0 (Limitada i caduca en 30 dies)</p> <p>OMNET++ v3.2 (Es de llicència pública acadèmica)</p> <p>Aquesta edició no està limitada en el nombre de models de simulació disponibles per a diferents protocols de xarxa i tecnologies respecte a la versió comercial.</p>
Requeriments del sistema	<p>Processador : Intel Pentium II, III, 4 o compatible (350 Mhz o mes)</p> <p>RAM : 256 MB.</p> <p>Espai de disc : 200 MB.</p> <p>Pantalla: 600 x 800 o major resolució, 256 colors o mes.</p> <p>Plataformes : Windows 2000 i XP.</p> <p>Per a crear nous models es pot utilitzar Microsoft visual C++ 6, 7 (http://www.microsoft.com/). També es poden utilitzar Cygwin (www.cygwin.com) o MinGW (www.mingw.org/) per a construir aplicacions tcl/tk es necessari els binaris BLT i part de XFree86, hi ha documentació disponible al respecte en la distribució.</p>
Venedor / Desenvolupador	<p>Omnet++ ha estat desenvolupat per András Varga i la Technical University of Budapest, Department of Telecommunications (BME-HIT). Actualment es manté i es desenvolupen nous models en la Comunitat Omnet++ en www.omnetpp.org, des d'on es pot obtenir el paquet d'aplicacions i les diferents llibreries de models desenvolupades.</p> <p>L'edició comercial es pot a través d'Omnest Global Inc.</p> <p>Website: www.omnest.com.</p>

5.10.1 – Descripció del software

OMNET++ (Objective Modular Network Tesbed in C++) és un simulador d'events discrets orientat a objectes. La seva principal area d'aplicació és la simulació de xarxes de comunicació, disseny i validació de protocols de comunicació i disseny d'arquitectures. Degut a la seva arquitectura genèrica i flexible s'usa en d'altres àrees com la simulació de sistemes IT, xarxes de cues, arquitectures de hardware i en processos de negoci.

S'utilitza per a dissenyar dispositius de xarxa i tecnologies experimentals, així com avaluar dissenys alternatius per a determinar costos i localitzar colls d'ampolla. Permet experimentar amb xarxes de cues simulant, per exemple, call-centers.

Utilitza el llenguatge C++ i disposa d'una API C++ orientada a l'objecte.

Característiques i components

Omnet++ es compon de :

- Llibreria de kernel de simulació : integrable en productes de tercers com a llibreria de classe C++. El kernel de simulació suporta temps real.
- Compilador per al llenguatge de descripció de topologies NED.
- Editor gràfic de xarxes per als arxius NED (GNED).
- GUI per a l'execució de les simulacions sobre els executables de simulació (Tkenv).
- Un interfície de línia de comandes per a l'execució de la simulació (Cmdenv)
- Eina per a l'anàlisi de vectors a través d'estadístiques (Plove).
- Eina per a l'anàlisi de escalars a través d'estadístiques (Scalars).
- Eines per a la documentació de models (opp_neddoc). Els models s'autodocumenten, les eines de documentació generen documentació de qualitat amb diagrames, taules, referències creuades. S'integra amb Doxygen.
- Utilitats (eines de generació de llavors de nombres aleatoris, eina de creació makefile, etc.)
- Documentació, exemples de simulació, material de contribució, etc. El codi font està ben documentat per a augmentar la comprensió, ajudar a depurar i permetre modificacions.

Existeix un gran esforç de desenvolupament, tant en l'entorn de simulació com en el de les llibreries (IPv6, TCP, Mobility, ...). Amb aquest entorn es poden executar simulacions compilades que necessiten d'una eina gràfica, mentre que per a desenvolupar-ne de noves, es necessari implementar els mòduls i compilar-los.

Proveeix una arquitectura de components per als models. Els components (mòduls) estan programats en C++ , després muntats en components més grans i models que usen un llenguatge d'alt nivell (NED).

Implementa la orientació a processos i la orientació a events, i força una jerarquització dels models que facilita la programació. Un model consisteix en mòduls jeràrquicament anuats, que es comuniquen mitjançant el pas de missatges.

Presenta dues interfícies d'execució : gràfica i de comandes, el que permet una depuració més senzilla. El mode gràfic permet una simulació pas a pas (de missatges), o de forma continua (amb diferent granularitat). La interfície visual és una eina didàctica i de debugging fonamental (en front de l'animació offline, que no aporta res).

Si no es disposa de l'executable, per a executar qualsevol simulació es necessari compilar primer el codi font. La única dificultat es obtenir aquest codi font (a través de llibreries disponibles o per desenvolupament) i executar un compilador C++.

Ofereix suport estàndard. Les entrades i sortides poden ser en text pla i/o XML, facilitant el procés amb productes de tercers. La integració de base de dades es possible.

Per al rastreig, depuració i animació de les simulacions, disposa d'una GUI on es pot clicar sobre els objectes i variables en codi C++, anivellant els canvis al vol.

Dona suport a base de dades. La topologia de la xarxa i la configuració del model pot ser importada d'una base de dades i també exportada a aquesta.

Referències de l'API d'OMNET++ (generades per a la llibreria de simulació per Doxygen 1.3.9.1)

Per a poder analitzar millor el paquet d'aplicacions, cal destacar algunes categories i classes corresponents dins de l'API :

Classes troncales de simulació :

- cObject - classe base per a la majoria de classes d'OMNET++.
- cModule, cCompoundModule i cSimpleModule – representen m+mòduls en la simulació. S'implementen els nous models a través de fer subclasses amb cSimpleModule i sobreescrivint la seva activity() o la funció del handleMessage().
- cMessage – representa events i missatges enviats entre mòduls.
- cGate – representa les portes dels mòduls.
- cPar - representa els paràmetres del mòdul i dels missatges.
- cSimulation – emmagatzema tots els mòduls de la xarxa i l'estructura de les dades per a futurs events.

Classes de contenidors :

- cQueue: representa la cua prioritària per a objectes derivats de cObject.
- cArray: un array dinàmic per a emmagatzemar objectes derivats de cObject. Generació de nombres aleatoris : Hi ha diverses funcions que generen variacions aleatòries des de diferents distribucions. Entre aquestes destaquen :

Distribucions contínues		Distribucions discretes
uniform	chi_square	Intuniform
exponential	student	bernoulli
normal	cauchy	binomial
truncnormal	trianglognorma	geometric
gamma	weibull	negbinomial
beta	pareto	poisson
erlang		

Generadors de nombres aleatoris i algoritmes generadors. Per defecte utilitza cMersenneTwister.

Classes d'utilitats : faciliten la creació de models. Entre elles cal destacar :

- cXMLElement – fa possible la configuració XML per a mòduls simples
- cTopology – suports enrutament en telecomunicacions.
- cStringTokenizer – per a obtenir cadenes de caràcters
- cFSM – per a crear l'estat final de les màquines.
- cWatch – fa visible les variables en Tkenv.

Interfície d'usuari : la classes cEnvir proporciona la interfície per a l'entorn de simulació.

Extensions per a la simulació paral·lela

- cCommBuffer - Buffer de la capa de comunicacions per a la simulació paral·lela.
- cParsimCommunications – Capa de comunicacions bàsiques per a la simulació paral·lela.

Llibreria de models de simulació d'events discrets

Les llibreries disponibles es troben, per lo general, molt documentades. Degut al desenvolupament modular de les mateixes, es possible utilitzar únicament aquells blocs d'interès.

Disposa de llibreries de models per a :

- Xarxes TCP/IP (TCP, UDP, PPP, IPv4 i 6, etc.)
- Protocols LAN/MAN (Ethernet, FDDI)
- Protocols per a ad-hoc frameworks i wireless.
- Altres models de simulació

En l'[annex 1](#) es relacionen tots els models disponibles.

5.10.2 – Construcció de models

Les passes en la construcció d'un model de simulació :

1. Definir la estructura de la xarxa : llenguatge NED (.ned).
2. Definir els missatges (.msg).
3. Implementar el mòduls simples i altres codis C++ (.cpp)
4. Crear un fitxer executable (.exe)

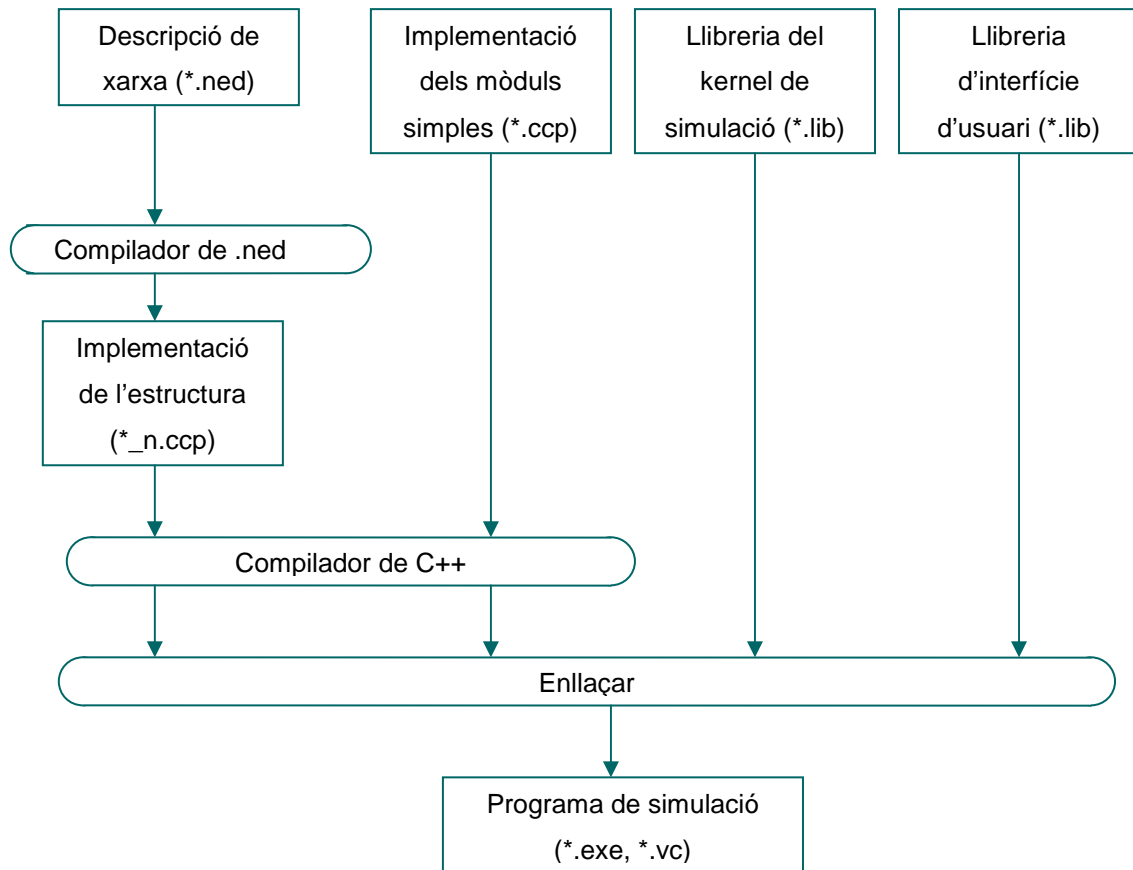


Figura 6 – Diagrama de construcció d'un model de simulació amb OMNET++

Definir l'estructura de la xarxa (topologia). Llenguatge NED

El software incorpora un editor gràfic GNED, (tot i que podem utilitzar d'altres editors de text.) i utilitza el llenguatge .ned. GNED permet Importar directives des d'un altre arxiu de descripció de xarxa i definir els canals a través de tipus de connexions amb diferents característiques. També permet definir mòduls simples, mòduls compostos i xarxes :

Mòduls simples :

- Paràmetres: numèric, const, bool, string, xml.
- Portes: in o out, poden ser arrays.

Mòduls compostos :

- Permeten compondre mòduls complexos a partir d'altres (submòduls).
- S'usen per a interconnectar mòduls.
- Es poden realitzar múltiples nivells de composició.

Xarxes :

- Instanciació de la xarxa.

- Poden contenir assignació de paràmetres

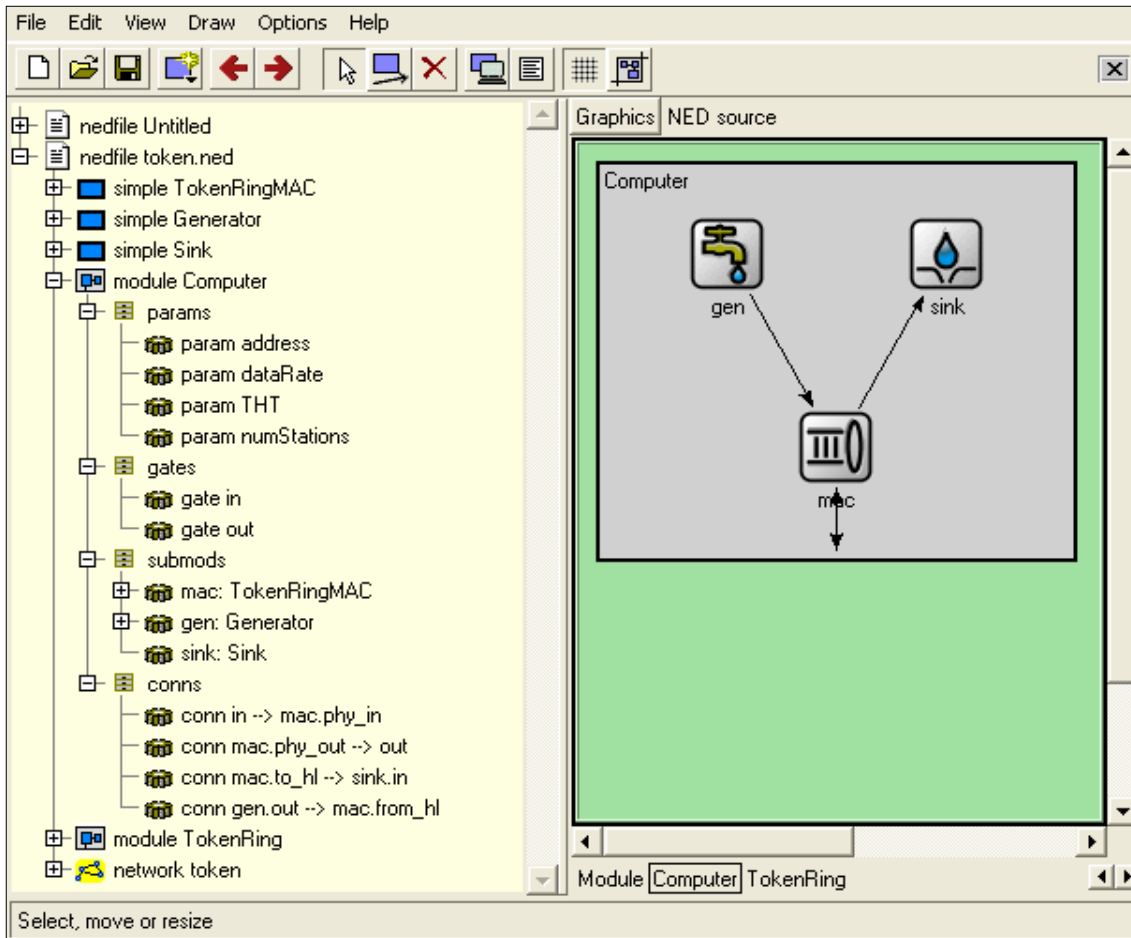


Figura 7 – Exemple de vista gràfica de l'estructura d'una estació en TokenRing amb l'editor GNED.

A la finestra de l'esquerra hi ha l'estructura jeràrquica dels mòduls utilitzats. Es pot observar en aquest exemple, que el mòdul computadora consta de submòduls prèviament creats i definits.

A la finestra de la dreta hi ha la vista gràfica (que pot ser vista de codi) d'una estació que consta de :

- (gen) mòdul per a generar paquets per a MAC amb els següents paràmetres : nombre d'estacions, adreça, missatge, volum del missatge, temps entre missatges i portes de sortida
- (sink) mòdul de destrucció de paquets i creació d'estadístiques amb portes d'entrada.
- (mac) mòdul que, en aquest exemple, implementa el protocol TokenRing amb els següents paràmetres : taxa de dades (Token Ring en bps), temps de possessió del token, adreça mac de l'estació, volum màxim de la cua en paquets.
- Connexions parametritzades en la direcció i portes utilitzades.

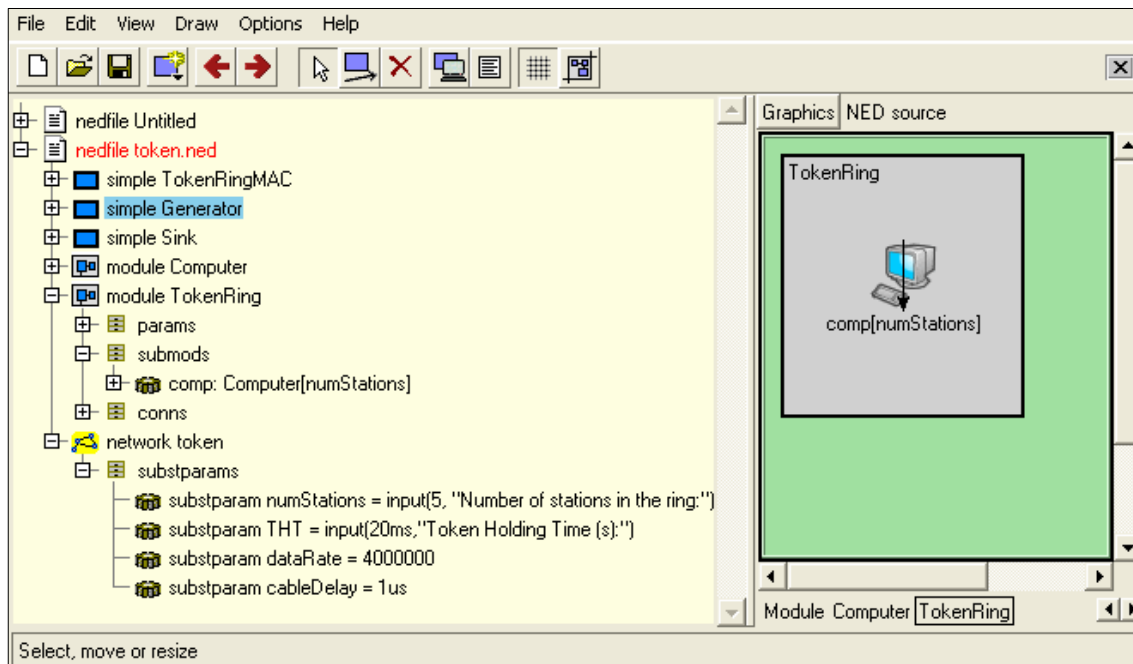


Figura 8 – Exemple de vista gràfica de l'estructura d'una xarxa d'estacions en TokenRing amb l'editor GNED.

En aquesta figura, es pot observar com la vista ara és la d'una xarxa TokenRing, quan en l'anterior (que correspon al mateix projecte), només teníem una computadora. El paràmetre numStations permet formar la xarxa amb el nombre d'estacions que es desitgi.

Amb GNED, també es poden utilitzar expressions : constants, paràmetres de referència, operadors, documents XML, funcions (i crear-ne), valors aleatoris.

Per a compilar els arxius .ned es disposa de l'eina "nedtools".

Definir els missatges

Omnet++ aporta definicions de missatges amb una sintaxis concreta i senzilla que es generen automàticament en codi C++.

Implementar el mòduls simples

Per a programar els mòduls simples, es necessari definir el comportament d'aquests :

- Iniciació : generalment, en el codi, en "initialize()" es crea un objecte cMessage, i s'envia a l'exterior a través de la porta de sortida (gate out).
- Tractament de missatges : el kernel de simulació deixa el missatge en aquest mòdul (com a argument) per a ser tractat en funció de la definició de missatges anterior.

- Activitat
- Finalització

Cada mòdul simple es programa com una classe C++.

- Hereta de cSimpleModule

Crear un fitxer executable (.exe)

Es poden generar dos tipus d'executables:

- Interfície d'usuari gràfic: Útil per a depurar i comprendre. Tkenv.
- Interfície de consola: Mes eficaç per a realitzar les simulacions per lots. Cmdenv.

Per a la compilació, Omnet++ permet generar automàticament els Makefile que ens ajudaran a compilar i executar i enllaçar el programa per a crear un executable. Amb el compilador de C++ que incorpora Visual Studio .NET, amb la instrucció `opp_nmakemake` es genera el `Makefile.vc`

Creat el Makefile, es procedeix a la compilació per a crear un fitxer `.obj` que s'enllaçarà a les llibreries necessàries (de simulació de kernel i de interfície d'usuari) per a generar l'executable `.exe`. Amb el compilador de C++ que incorpora Visual Studio .NET, amb la `nmake -f Makefile.vc` es genera l'executable.

5.10.3 – Simulació : creació d'estadístiques

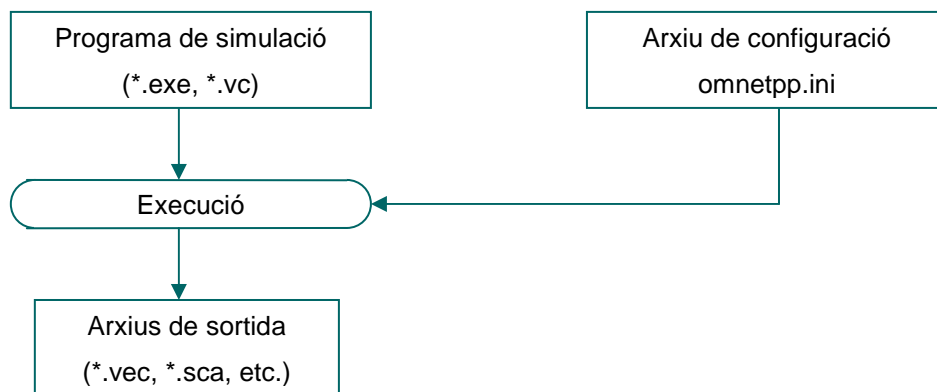


Figura 9 – Diagrama d'execució d'un model de simulació

La interacció entre la interfície d'usuari i el kernel de simulació defineixen la interfície, de manera que canviant l'arxiu del model, la simulació pot córrer sota diferents interfícies.

Configurar l'arxiu .ini

Mitjançant la creació de l'arxiu omnetpp.ini es configura la simulació. Les seccions més importants de l'arxiu son :

- [General] : conté les opcions generals per a aplicar a l'execució de la simulació. Dintre d'aquesta secció podem trobar diferents subseccions :
 - Ini_warnings : si està habilitat, es llisten els noms dels arxius ini d'entrada per a cada valor utilitzat.
 - Preload-ned-files = llista d'arxius .ned a carregar.
 - network = nom de la xarxa
 - Output-vector-file = omnetpp.vec
 - Output-scalar-file = omnetpp.sca
 - Pause-in-sendmsg = no
- [Run 1], [Run 2], ... : conté les opcions de la preexecució.
- [Cmdenv] : opcions específiques de la interfície de línia de comandes per a l'execució de la simulació.
- [Tkenv] : opcions específiques d'interfície gràfica.
- [Parameters] : valors per a paràmetres del mòdul que no en tenien.
- [OutputVectors] : opcions per a guardar les dades dels vectors de sortida.
- [OutputScalars] : opcions per a guardar les dades dels scalars de sortida.

Permet l'exportació/importació d'aquest arxiu.

Configurar el recull de dades vectorials i escalars

Per a recollir estadístiques sobre el rendiment, es poden crear arxius amb les dades vectorials i escalars de la sortida per al seu posterior anàlisi. La seva configuració es troba en les seccions corresponents de l'arxiu omnetpp.ini. També es poden configurar des de Tkenv com es veurà més endavant.

Configurar la generació de nombres aleatoris

El tipus de generador (Mersenne Twister, minimal standard, Akaroa o d'altres) es defineix en l'arxiu omnetpp.ini. Permet a selecció automàtica i manual de la llavor, que es defineixen a [Run1..2..3..].

Configurar Tkenv i Cmdenv

Tkenv és una eina molt completa que disposa d'un entorn gràfica mitjançant el qual es pot realitzar :

- animació dels missatges.
- gràfiques d'estadístiques (histogrames, etc.).
- vectors de sortida durant la simulació.
- finestres diferents per a la sortida de cada mòdul.
- finestra de missatges.
- execució event a event, normal i ràpida.
- Inspector de finestres.
- Pausa i reinici de la simulació

La seva configuració es troba en les seccions corresponents de l'arxiu omnetpp.ini, i pot ser executat en dos modes : normal (permet depuració) i exprés (per a simulacions llargues).

5.10.4 – Execució, visualització i animació

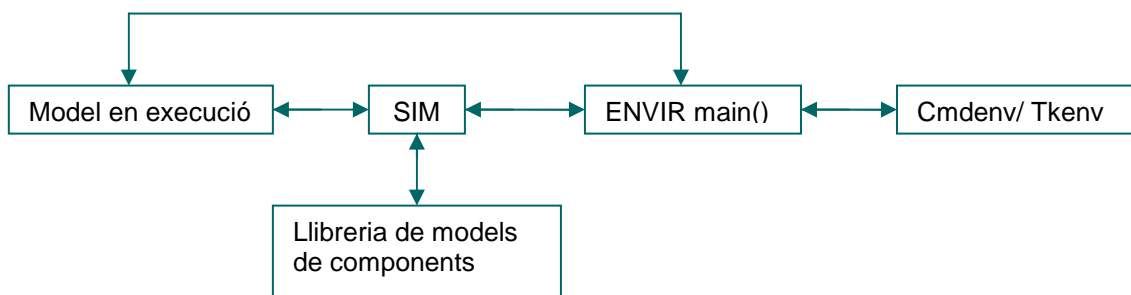


Figura 10 – Arquitectura d'un programa de simulació

SIM és el kernel de simulació i la llibreria de classes. Existeix com una llibreria per a enllaçar el programa de simulació.

ENVIR és una altre llibreria que conté tot el codi que és comú per a totes les interfícies de l'usuari. Proveen serveis com el fitxer ini.

Cmdenv i Tkenv son implementacions específiques de les interfícies de l'usuari.

La llibreria de models de components és un mòdul simple de definicions i la seva implementació en C++, tipus de composició dels mòduls, canals, xarxa, tipus de missatges i tot allò que pertanyi als mòduls. S'ha d'enllaçar al programa de simulació.

A l'executar el programa de simulació, si està configurat per a Tkenv, obtindrem les següents finestres des d'on es pot configurar la simulació i diverses opcions que no hagin estat declarades en l'arxiu omnetpp.ini.

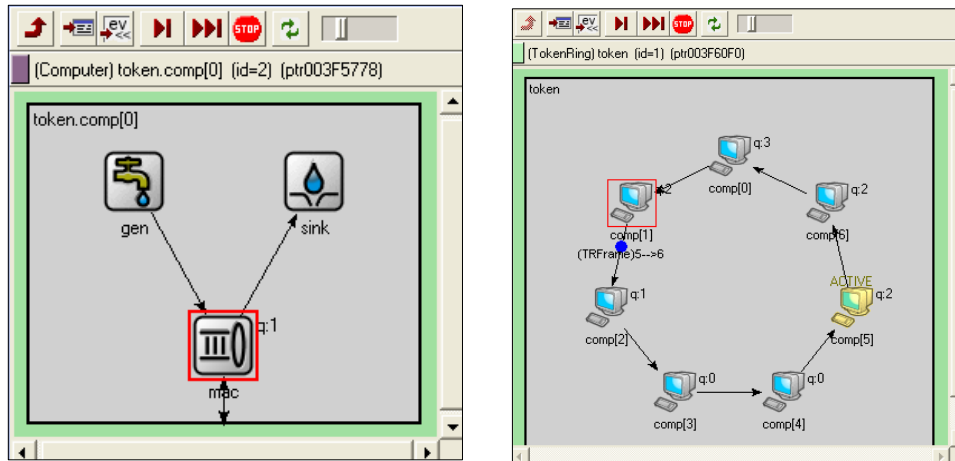


Figura 11 – Finestres de visió animada de la simulació, la de l'esquerra representa a la computadora activa i la de la dreta la xarxa i el pas del token.

Els diferents objectes es poden clicar durant la simulació per a veure les propietats i configuracions (per exemple : submòduls que inclou, paràmetres i portes).

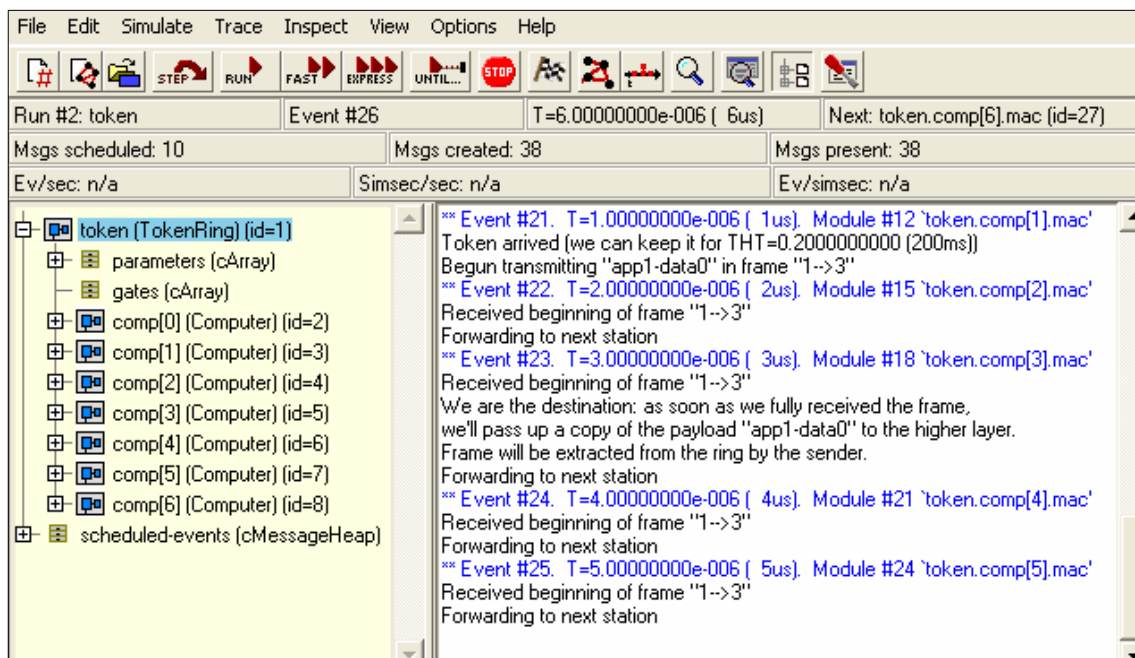


Figura 12 – Finestra de control de la simulació.

En la figura anterior es mostra la finestra de control de la simulació, on a l'esquerra es pot observar l'estructura dels mòduls (6 computadors en xarxa TokenRing), i a la dreta informació dels events que es succeeixen durant la simulació.

Es molt útil l'inspector d'objectes (dins de Inspect en el menú), que ofereix la possibilitat de parametritzar la simulació sobre un objecte especificat (per exemple la variable numPacketsToSend, implementada en el mètode d'inicialització en els mòduls simples).

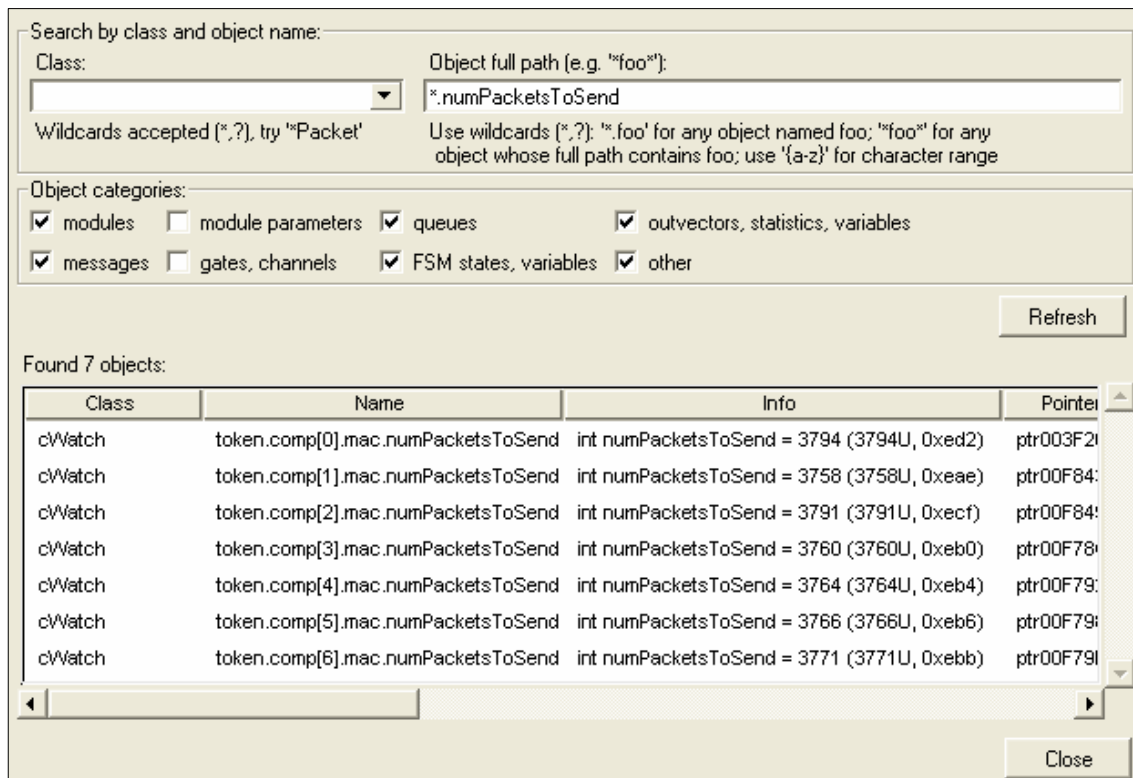


Figura 13 – Finestra de l'Inspector d'objectes

D'aquesta manera es poden obtenir estadístiques i vectors de sortida en temps de simulació sobre l'objecte especificat..

En la següent figura es poden veure dues finestres de diagrames, la de l'esquerra representa temps que triga en fer una volta el token, i la de la és un histograma dels temps de vida dels missatges que arriben.

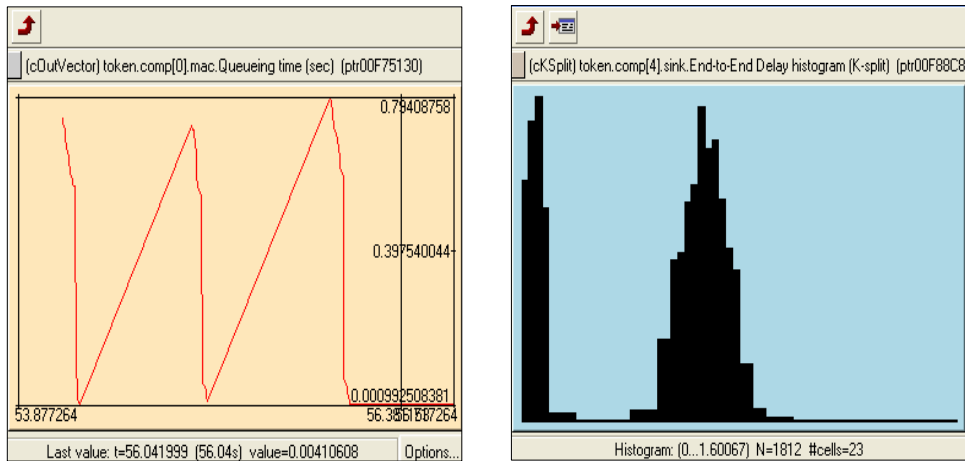


Figura 14 – Finestres de diagrames.

5.10.5 – Anàlisi dels resultats

Estadístiques vectorials

Si s'ha configurat el recollir vectors durant la simulació, OMNET++ crea un arxiu .vec. Mitjançant l'eina PLOVE, es poden visualitzar aquests arxius per a analitzar el funcionament i rendiment de l'objecte d'interès.



Figura 15 – Interfície de l'eina PLOVE.

Es carrega una sortida de vectors a la finestra esquerra i es passen els que es vulguin a la finestra dreta per a iniciar la planificació de vectors. Es pot ajustar l'estil del dibuix i duplicar vectors.

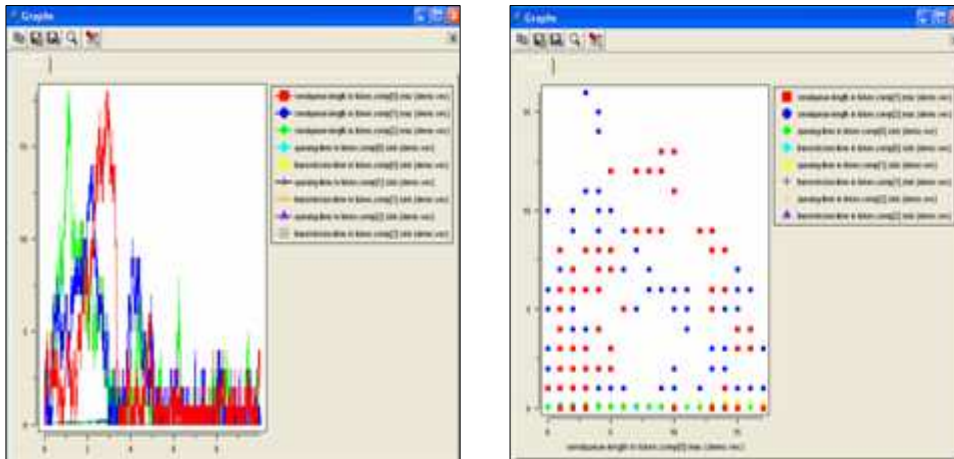


Figura 16 – Exemple de visualització i anàlisi de les dades vectorials mitjançant diferents diagrames amb PLOVE

El tipus de línia per a cada vector pot ser configurat com les opcions de dibuix més freqüents : límit d'eixos, escalabilitat, títols, etiquetes, etc.

Permet exportar els gràfics i el format vector, filtrar els resultats (promig, truncament de valors extrems, valorar densitats per histogrames) i aplicar diferents filtres al mateix vector.

Estadístiques escalars

Per a poder comparar models sota diversos paràmetres diferents, si s'ha configurat, OMNET++ guarda les dades escalars en arxius amb l'extensió .sca. A través de l'eina SCALARS es poden dibuixar diagrames de barres, arguments x i y, exportar les dades i els fitxers per a posteriors anàlisis en fulls de càlcul o d'altres programes.

Mostra les dades en una taula amb columnes : nom de l'arxiu, nombre controlat, nom del mòdul on ha estat registrat i el valor. Permet filtrar les línies i també carregar arxius remots escalars i analitzar-los junts.

Directory	File	Run#	Module	Name	Value
C:\Programa:	demo.sca	4	largeNet.switchA.mac[1]	frames rcvd	34.0
C:\Programa:	demo.sca	4	largeNet.switchA.mac[1]	simulated time	2.354
C:\Programa:	demo.sca	4	largeNet.switchA.mac[0]	frames rcvd	131.0
C:\Programa:	demo.sca	4	largeNet.switchA.mac[0]	frames sent	122.0
C:\Programa:	demo.sca	4	largeNet.switchA.mac[0]	simulated time	2.354
C:\Programa:	demo.sca	4	largeNet.switchA.relayUnit	dropped frames	64.0
C:\Programa:	demo.sca	4	largeNet.switchA.relayUnit	processed frames	270.0

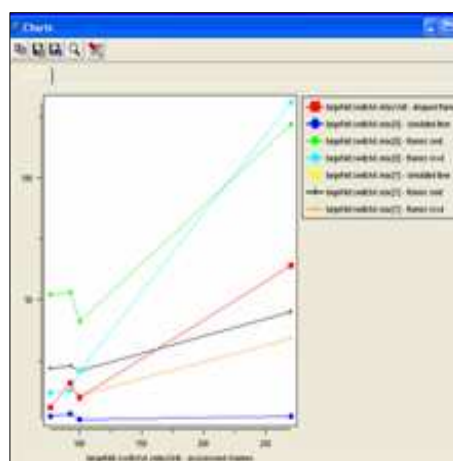
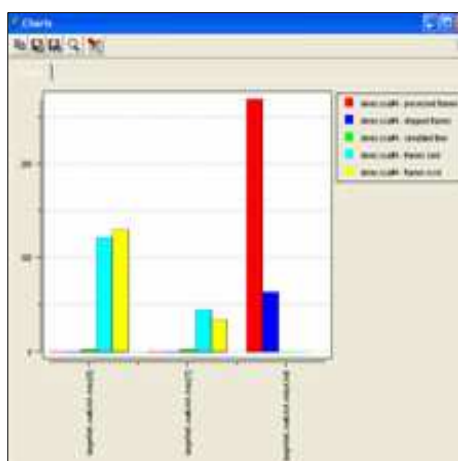


Figura 17 – Exemple de visualització i anàlisi de les dades escalars mitjançant diferents diagrames amb SCALARS.

5.10.6 – Altres recursos

- eines i utilitats per a la generació de documentació : amb l'eina opp_netdoc es pot generar documentació en HTML d'arxius .ned i .msg. Amb opp_netdocgenerated, la documentació cataloga mòduls simples i compostos, i presenta els seus detalls (descripció, portes, paràmetres, codi assignat, etc.) . També inclou diagrames de xarxa clicables exportats de l'editor gràfic GNED i diagrames d'us de mòdul i d'herència per a missatges. Pot treballar conjuntament amb Doxygen.
- Altres eines i utilitats per a l'anàlisi i visualització a mes de Plove i Scalars. Amb aquest objectiu, es pot guardar la sortida en diversos formats (postdata, imatge, etc.), copiar i

enganxar dades en fulls de càlcul (OpenOffice, Excel, etc.) i exportar a programes numèrics (Octave, MathLab, etc.). Exemples d'aquestes utilitats son :

- Grace : és un editor GPL per a Unix i Windows que permet exportar gràfiques a PS, PDF, MIF, SVG, PNM, JPEG y formats de PNG.

- ROOT : és un marc d'anàlisi de dades mitjançant objectes destinat a la planificació i els gràfics. Està basat en CINT(intèrpret de C/C++) pel tractament de codis C/C ++ . La pàgina de Curt Brune en Stanford (<http://www.slac.stanford.edu/curt/omnet++/>) mostra diversos exemples.

- GnuPlot : per a traçar gràfics 3D. Disponible per a Unix i Windows. Es poden copiar i enganxar gràfics.

- Documentació : manuals, faqs, notícies, anuncis, tècniques, projectes d'universitats, treballs, publicacions, fòrums.
- Comunitats d'usuaris : <http://www.omnetpp.org/> , <http://community.omnest.com/pmwiki/>

5.11 – QUALNET Developer

Versions/ Edicions	<p>QUALNET Developer v3.9 (Comercial) QUALNET Developer Demo v3.9</p> <p>La versió demo ha estat facilitada Scalable Networks Technologies Inc., prèvia sol·licitud. Es només per a empreses i estudiants universitaris. Si SNT accepta la sol·licitud, envia la llicència per a 2 setmanes i l'accés a la descàrrega a través del email. Aquesta demo està limitada en el nombre de models de simulació disponibles per a diferents protocols de xarxa i tecnologies respecte a la versió comercial.</p>
Requeriments del sistema	<p>Processador : x86 i amd64 compatible (a 3.2 GHz o +).</p> <p>RAM : 64 MB per a simulacions sense GUI, 256 MB amb GUI, 128 MB i més per a simulacions amb GUI de grans xarxes.</p> <p>Espai de disc : per a la versió demo 230 MB, per a la comercial depèn dels mòduls adquirits, però mínim 500MB.</p> <p>Pantalla: 600 x 800 o major resolució, 256 colors o mes.</p> <p>Plataformes : Windows NT, 2000 i XP.</p> <p>Per a la interfície gràfica de l'entorn virtual, és necessari el paquet de software Sun Java 2 SDK, Standard Edition 1.4.2. (http://java.sun.com/j2se/1.4.2/download.htm.)</p> <p>Per a compilar models nous, es necessari Visual C++ 6.0 amb Visual Studio 6.0 Service Pack 5 (http://msdn.microsoft.com/vstudio/sp/vs6sp5/default.asp) o superior.</p>
Venedor / Desenvolupador	<p>Scalable Network Technologies, Inc. 6701 Center Drive West, Suite 520 - Los Angeles, CA 90045 Website: http://www.scalable-networks.com o http://www.qualnet.com</p>

5.11.1 – Descripció del software

QUALNET Developer és un paquet d'aplicacions que proporciona un entorn virtual per al modelat i simulació de xarxes amb i sense fils. La seva principal area d'aplicació és la simulació de xarxes de comunicació, disseny i validació de protocols de comunicació i disseny d'arquitectures.

Es el derivat comercial de GLOMOSIM (<http://pcl.cs.ucla.edu/projects/glomosim/>) i utilitza la simulació en paral·lel proporcionada per el llenguatge de simulació d'events discrets PARSEC (<http://pcl.cs.ucla.edu/projects/parsec/>), que permet simulacions per a radio i xarxes sobre arquitectures seqüencials paral·leles.

L'actiu més important d'aquest software és la velocitat de la simulació a temps real de grans xarxes de comunicacions amb tràfic dens. Ofereix una predicció bastant exacte del funcionament d'una xarxa per a un gran ventall d'exigències d'aplicació i usos.

Els principals avantatges que ofereix son :

- Prototipatge ràpid dels protocols
- Comparativa de l'avaluació del funcionament de protocols alternatius en cada capa
- Mesures integrades en cada capa.
- Disseny d'apilament de capes modular.
- API estàndard per a la composició de protocols a través de les diferents capes.
- Suport a través de l'adaptabilitat per a les execucions paral·leles.
- Eines GUI per a la simulació de models/sistemes.

Característiques i components

QualNet Developer aporta un entorn per al disseny de protocols, creació i animació d'experiments i anàlisi de resultats d'aquests experiments. Tot el que es pot fer, es realitza a través del les següents eines :

- Qualnet simulator : motor de simulació altament escalable. Aporta una gran fidelitat en els models de xarxes de mils de nodes.
- Scenario Designer : eina gràfica per al disseny d'experiments de simulació. Permet definir la distribució geogràfica, les connexions físiques i els paràmetres funcionals dels nodes de la xarxa.
- Animator : eina gràfica per a la configuració i visualització de l'animació de simulació. Animació dels escenaris creats on es pot veure el flux de tràfic i crear gràfics dinàmics del funcionament durant la simulació.
- Analyzer : eina gràfica per a la l'anàlisi dels resultats de la simulació. Es poden visualitzar i analitzar les diferents estadístiques obtingudes i comparar-les entre elles.
- Protocol Designer : eina gràfica per al disseny de protocols. Té dues funcions bàsiques : el disseny de nous protocols de xarxes i simplificar els mecanismes per a incorporar protocols al simulador. Permet definir els processos i events d'un model de protocol.
- Packet Tracer : eina gràfica per al seguiment dels paquets durant la simulació.

Proporciona una interfície gràfica basada en Java i una implementació de protocols basada en C++.

Disposa d'un disseny modular. El model de capes té una arquitectura similar a les xarxes físiques, un API ben definit per a la interacció de protocols i una unitat d'interacció (missatge, paquet o temporitzador). Disposa de la possibilitat d'emulació de la xarxa suportant migració directa de codi entre el model i xarxes operacionals.

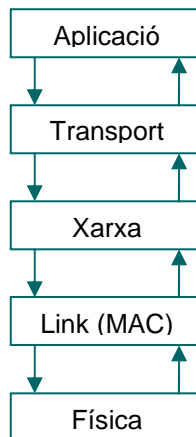


Figura 18 – Esquema de l'estructura de capes de protocol del model

La capa d'aplicació és la responsable de la generació del tràfic i del nivell d'enrutament de l'aplicació. La capa de transport proveeix els serveis de transmissió de dades a la capa d'aplicació. La capa de xarxa és la responsable de la distribució de paquets. La capa MAC proveeix dels enllaços per a la distribució de paquets. La capa física és la responsable de rebre i enviar els bits pels canals corresponents.

Es un simulador basat en events discrets que ofereix suport per a diferents xarxes a través de les següents llibreries de models: MANET, QoS, amb fils, sense fils, cel·lulars i satèl·lit.

Estructura dels directori

Directorio	Descripció
Addons	Paquets opcionals
Application	Protocols de la capa aplicacions i generadors de tràfic
Bin	Arxius executables
Data	Arxius amb les dades per al modelat
Docs	Documentació
Gui	GUI

Importer	Codi per a l'importador de Qualnet
Include	Arxius comuns
Lib	Arxius d'execució per al GUI
License-dir	Llicència
Mac	Protocols capa MAC
Main	Dissenys bàsics i makefiles
Mobility	Models mòbils
Network	Protocols d'enrutament de la capa de xarxa
Phy	Capa física i de propagació
Tcplib	Aplicacions basades en TCP
Transport	Protocols capa de transport
Util	Codi per a cues i planificacions
Verification	Arxius d'exemple i sortides
Win32	Arxius del projecte per a Visual Studio

Llibreria de models de simulació d'events discrets

Disposa de 4 llibreries, estàndard, IPv6, QoS i MANET, que comprenen models per a aplicacions QoS, transport, multicast, cues, programació de tasques, enrutament, MAC, mobilitat, dispositius, etc.

En l'[annex 2](#) es relacionen tots els models disponibles.

5.11.2 – Construcció de models

Hi ha dues maners d'usar QUALNET, a través d'una interfície gràfica o de línies de comandes (utilitzant els arxius .config de text).

Per a la construcció de models a través de la línia de comandes : cal crear l'arxiu .config utilitzant de el model adjunt en la distribució default.config (en /bin), afegir els nodes, aplicacions i enllaços en l'arxiu .config. Per a dur a terme la simulació, cal executar l'arxiu creat .config. Hi ha diferents arxius d'entrada i de sortida :

Entrada

- .config : paràmetres generals de la simulació.
- .nodes : nodes en la xarxa a simular.
- .app : configuracions de les aplicacions.

- .fault : fallades probables en un node.
- .routes : entrades d'enrutament.

Sortida

- .stat : sumari de les dades estadístiques.
- .trace : rastres dels paquets (capturats amb l'eina Tracer).

```

#####
# Name of experiment. Results are written to EXPERIMENT-NAME.stat.
EXPERIMENT-NAME default
# Simulation time. Units:
#
# 100NS - 100 nanoseconds      100   - 100 seconds (default case)
# 100US - 100 microseconds     100M  - 100 minutes
# 100MS - 100 milliseconds     100H  - 100 hours
# 100S  - 100 seconds          100D  - 100 days
SIMULATION-TIME 15M
# Value used to seed the random number generator. The random number
# generator is used in several models, such as the UNIFORM and RANDOM
# node-placement models.
SEED 1

```

Figura 19 – Mostra la part de la configuració del nom, temps i llavor en un arxiu .config

```

Current Sim Time[ls] = 729.057255955 Real Time[ls] = 0 Completed 81%
Current Sim Time[ls] = 738.007763088 Real Time[ls] = 0 Completed 82%
Current Sim Time[ls] = 747.033341271 Real Time[ls] = 0 Completed 83%
Current Sim Time[ls] = 756.019458084 Real Time[ls] = 0 Completed 84%
Current Sim Time[ls] = 765.005634126 Real Time[ls] = 0 Completed 85%
Current Sim Time[ls] = 774.010665898 Real Time[ls] = 0 Completed 86%
Current Sim Time[ls] = 783.025416876 Real Time[ls] = 0 Completed 87%
Current Sim Time[ls] = 792.011854624 Real Time[ls] = 0 Completed 88%
Current Sim Time[ls] = 801.034911295 Real Time[ls] = 0 Completed 89%
Current Sim Time[ls] = 810.006477214 Real Time[ls] = 0 Completed 90%
Current Sim Time[ls] = 819.011379199 Real Time[ls] = 0 Completed 91%
Current Sim Time[ls] = 828.003698406 Real Time[ls] = 0 Completed 92%
Current Sim Time[ls] = 837.022473956 Real Time[ls] = 0 Completed 93%
Current Sim Time[ls] = 846.041444693 Real Time[ls] = 0 Completed 94%
Current Sim Time[ls] = 855.009326237 Real Time[ls] = 0 Completed 95%
Current Sim Time[ls] = 864.030836357 Real Time[ls] = 0 Completed 96%
Current Sim Time[ls] = 873.017081682 Real Time[ls] = 0 Completed 97%
Current Sim Time[ls] = 882.040571011 Real Time[ls] = 0 Completed 98%
Current Sim Time[ls] = 891.066135913 Real Time[ls] = 0 Completed 99%
Current Sim Time[ls] = 900.000000000 Real Time[ls] = 0 Completed 100%
Executed 380368 events in 0.7650 sec

```

Figura 20 – Mostra l'execució de l'arxiu .config, es a dir la simulació.

```

1, , [0], Physical, 802.11,Signals transmitted = 419
1, , [0], Physical, 802.11,Signals received and forwarded to MAC = 998
1, , [0], Physical, 802.11,Signals locked on by PHY = 1114
1, , [0], Physical, 802.11,Signals received but with errors = 116
1, , [0], Physical, 802.11,Energy consumption (in mWthr) = 225.025
1, , [0], MAC, 802.11MAC,Packets from network = 94
1, , [0], MAC, 802.11MAC,UNICAST packets sent to channel = 71
1, , [0], MAC, 802.11MAC,BROADCAST packets sent to channel = 23
1, , [0], MAC, 802.11MAC,UNICAST packets received clearly = 111
1, , [0], MAC, 802.11MAC,BROADCAST packets received clearly = 136

```

Figura 21 – Mostra el contingut de l'arxiu .stat amb estadístiques de les dades recollides durant la simulació.

La construcció de models mitjançant la interfície gràfica es realitza a través del dissenyador d'escenaris. Les passes per a la construcció del model són les següents :

- Seleccionar l'escenari.
- Es seleccionen els dispositius per als diferents nodes.
- Es seleccionen i configuren les aplicacions per a connectar cada parell de nodes.
- Es creen els enllaços.
- Es prepara l'escenari per a la simulació.

La simulació es du a terme des de l'animador, escollint la velocitat i el detall d'animació, i seleccionant les estadístiques entre les disponibles.

Qualnet Developer integra en una sola interfície gràfica totes les eines necessàries per a la construcció, simulació i anàlisi de resultats dels models.

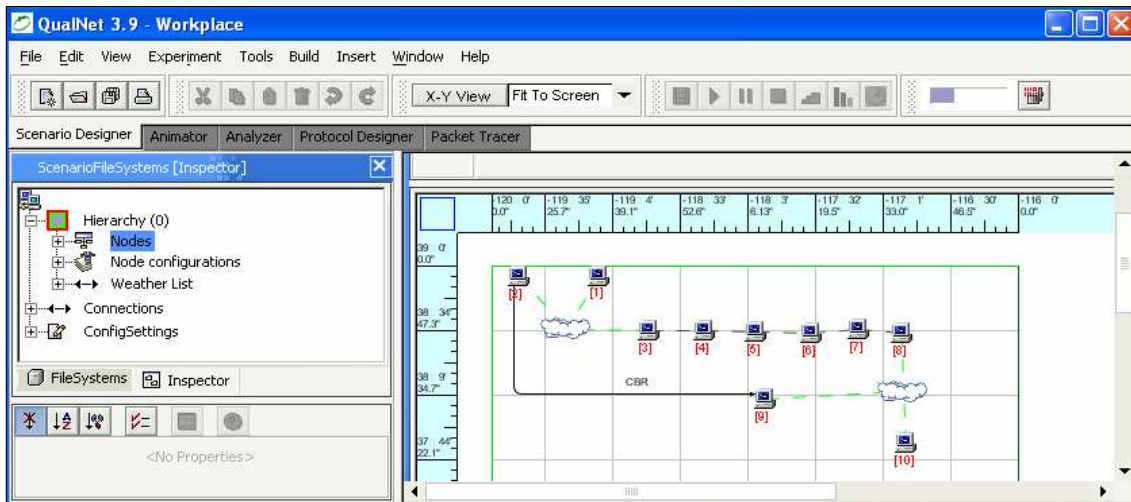


Figura 22 – Entorn gràfic general (i interfície del dissenyador d'escenaris en concret) on es poden observar les 5 eines mencionades i les diferents finestres del dissenyador d'escenaris.

Tots els components (objectes) són configurables i parametrizables des de l'inspector de components.

La construcció del model es du a terme des de el dissenyador d'escenaris. Com s'observa en la figura anterior, el dissenyador d'escenaris es divideix en tres zones : a la dreta la zona de gràfics, a l'esquerra a dalt els arxius del sistema i l'inspector de components, a l'esquerra a baix l'explorador de propietats.

Per a crear un escenari es disposa de diferents taules d'objectes : dispositius, aplicacions, enllaços, components de xarxes i d'altres components.



Figura 23 – Exemple de la taula d'aplicacions

Un model de xarxa creat es compon de :

- Una jerarquia de nodes : cada node es compon de hosts i de subxarxes. Cada host disposa d'una configuració d'interfície (capa física/radio, protocol MAC, protocol de xarxa, protocol d'enrutament), aplicacions pròpies, mobilitat, etc. Tots els nodes són configurables: mobilitat, capa física/radio, protocol MAC, ARP, protocol de xarxa, capa de transport, propietats dels dispositius, configuracions d'enrutament, etc.

- Connexions : origen, destí, prioritats, etc.
- Opcions de configuracions generals : posicionament dels nodes, opcions de les connexions, opcions de les capes, opcions de tràfic, etc.

Tot model es compon d'un o mes models d'entorn, d'aplicacions (generació de tràfic), de dispositius, d'especificacions (p.e. mobilitat)

Destaca la possibilitat de posicionament automàtic de nodes. Especificant un nombre de nodes, els distribueix uniformement, aleatòriament utilitzant una llavor aleatòria o en distàncies exactes.

5.11.3 – Simulació : creació d'estadístiques

Diferents aspectes a tenir en compte en la simulació d'un model :

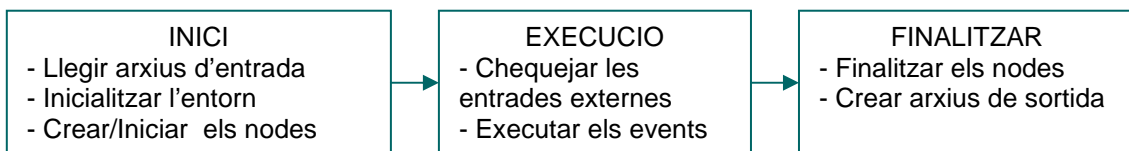


Figura 24 – Esquema del cicle de vida d'una simulació

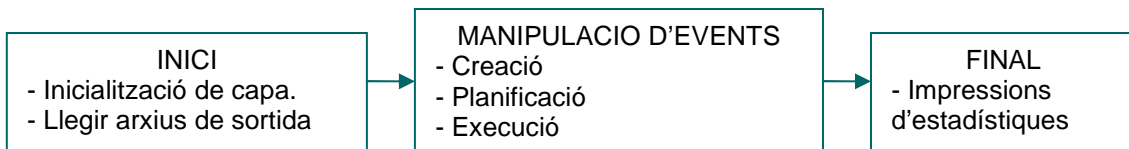


Figura 25 – Esquema del cicle de vida d'un node

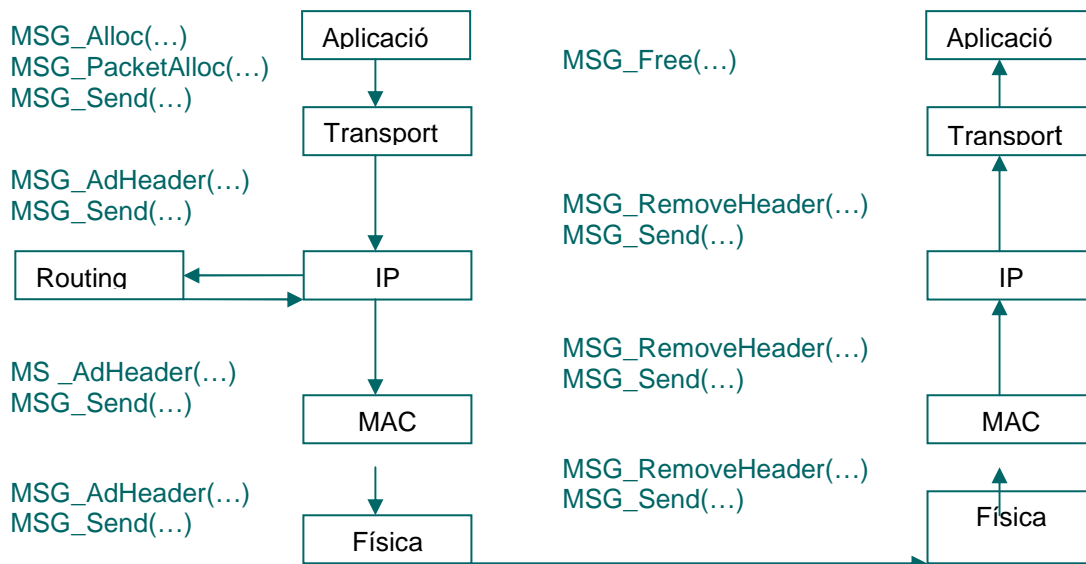


Figura 26 – Esquema del cicle de vida d'un paquet

Abans de dur a terme la simulació s'han d'haver configurat en les opcions generals de configuració del model les estadístiques a recollir.

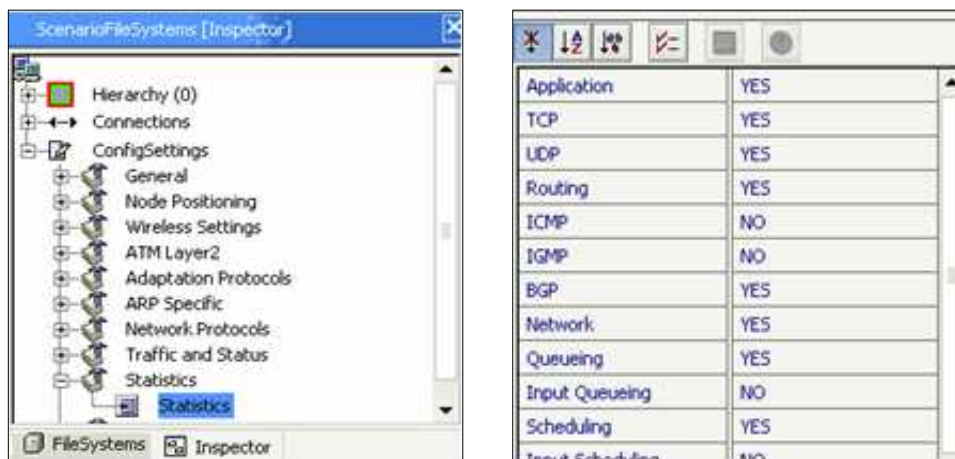


Figura 27 – Exemple de selecció d'estadístiques des de l'inspector d'arxius del sistema del dissenyador d'escenaris

5.11.4 – Execució, visualització i animació

Des de l'animador s'executa i s'anima el model creat en el dissenyador d'escenaris. La finestra es divideix en tres zones :

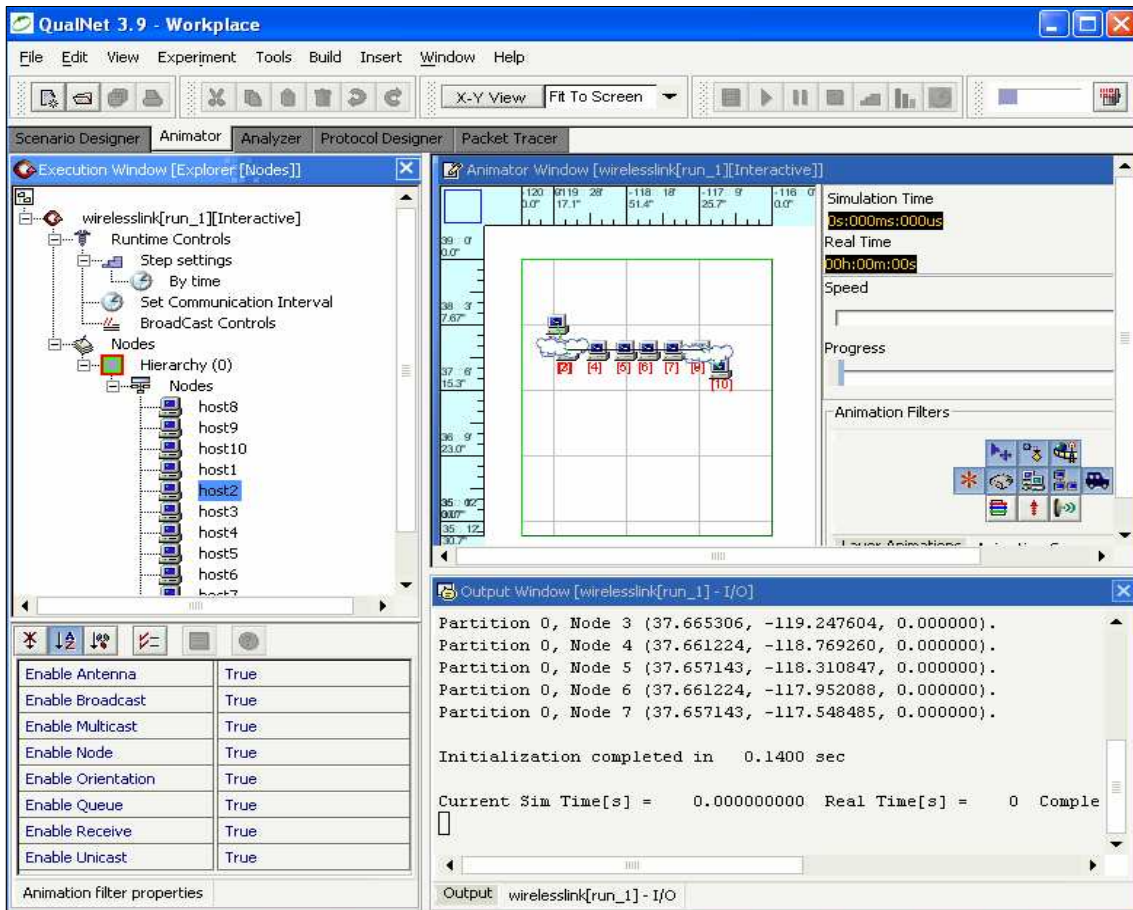


Figura 28 – Interfície gràfica de l'animador de la simulació.

- Animació : disposa de control de temps (període de temps dins de la simulació en el que s'està, temps real des de que s'està executant la simulació). Permet accelerar/diminuir la velocitat de la simulació. Disposa diversos efectes d'animació i propagació per a les antenes.

Permet l'aplicació de filtres per a l'animació des de dos perspectives diferents. Disposa de filtres per habilitar/deshabilitar capes de l'animació (mac, xarxa, transport, aplicació, etc.) i per a comandes de l'animació (node, enllaç, broadcast, col·lisions, cues, etc.).

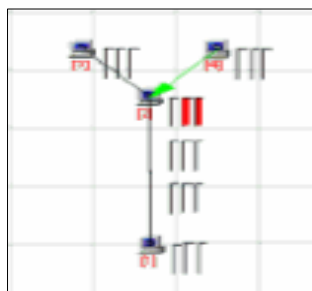


Figura 29 – Exemple d'animació amb el filtre de cues habilitades.

- Execució : es poden observar els paràmetres definits a tots els objectes a sota. Permet configurar diferents aspectes de la simulació com son el temps de simulació, els intervals de comunicacions , el nombre de cicles o el factor d'escala.
- Missatges sobre les dades de sortida.

L'animació és configurable en diferents paràmetres com per exemple el temps de simulació i els intervals de comunicació.

Disposa d'execució en segon pla on especificar el nombre d'experiments a realitzar i els valors de les variables i l'assignació de llavors.

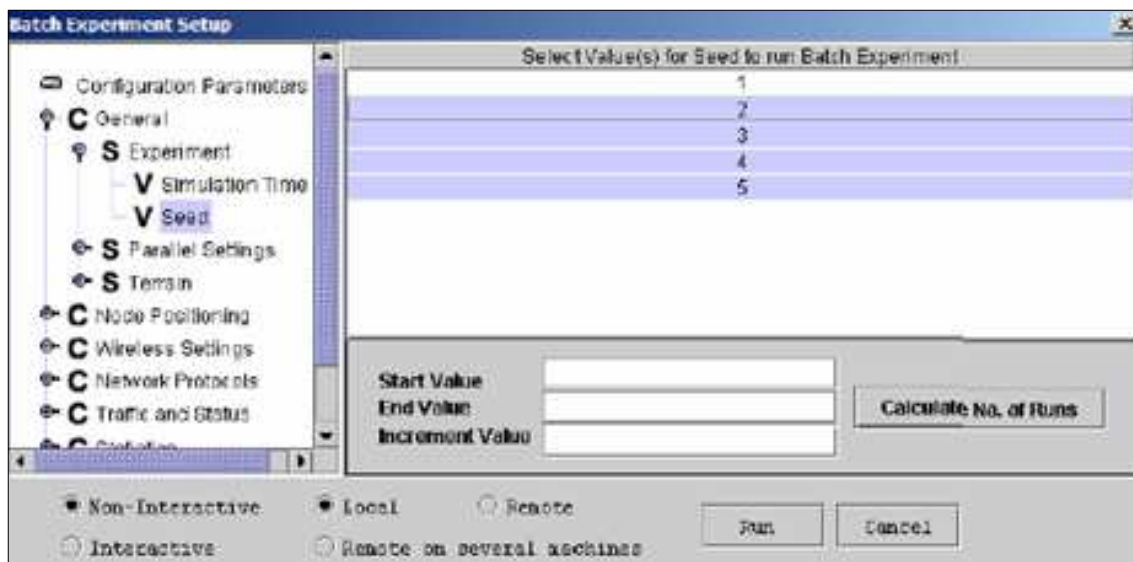


Figura 30 – Exemple d'especificació de llavor per a la recol·lecció d'estadístiques dinàmiques en l'execució en segon pla.

Permet la recol·lecció dinàmica d'estadístiques.

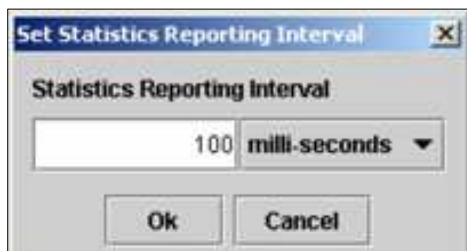


Figura 31 – Exemple d'especificació d'interval de recol·lecció d'estadístiques cada 100 ms

5.11.5 – Anàlisi de resultats

L'anàlisi de resultats es realitza a través de l'eina Analyzer. Un cop executada la simulació (tants cops com s'hagi especificat) els resultats es poden analitzar a partir de diferents gràfics (p.e. diagrames de barres i histogrames) que reflexen estadístiques de les dades de sortida classificades per capes/protocols

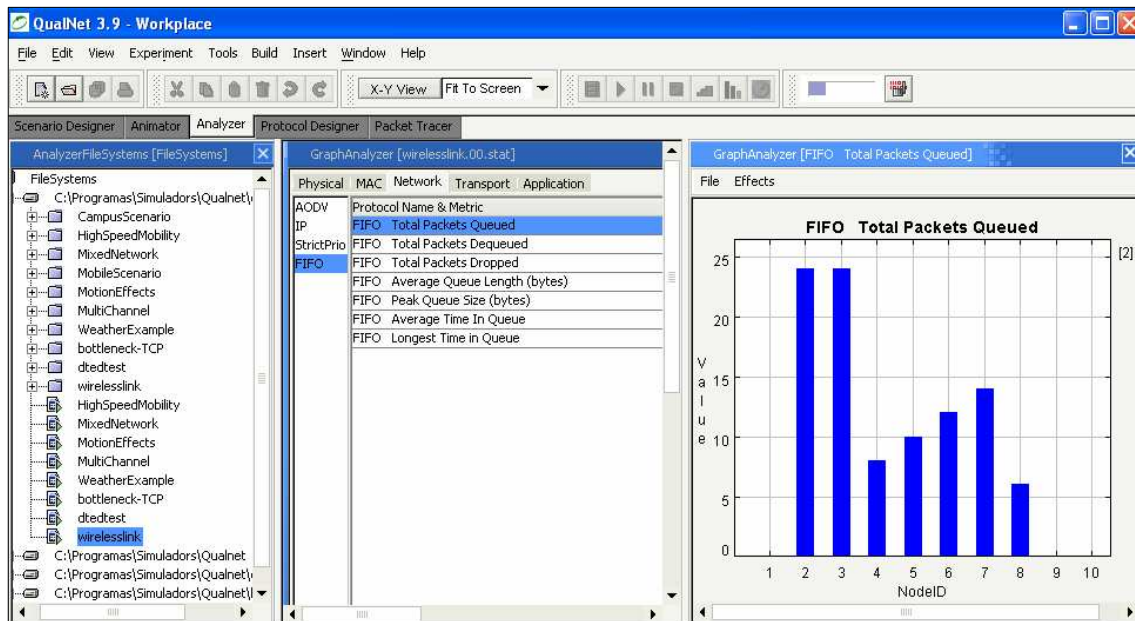


Figura 32 – Interfície gràfica de l'analitzador de les estadístiques de sortida on es mostra un diagrama de barres dels paquets en cua FIFO de cada node de la xarxa.

Les característiques més destacables de l'analitzador son :

- Generació ràpida de les estadístiques.
- Genera bastants tipus de figures per a una correcta comparació de resultats.
- Genera arxius de dades ASCII d'aplicacions neutrals per a importar a altres programes com poden ser Gnuplot o Excel.

L'analitzador suporta la comparació d'un únic experiment per node, la comparació d'experiments múltiples i els histogrames.

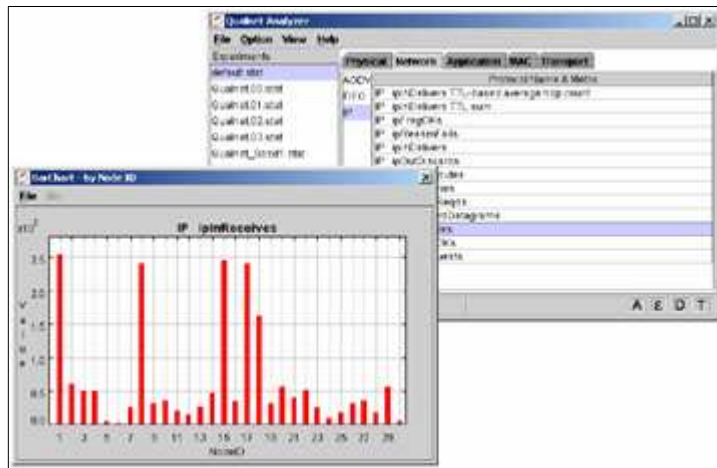


Figura 33 – Exemple diagrama de barres sobre un únic experiment

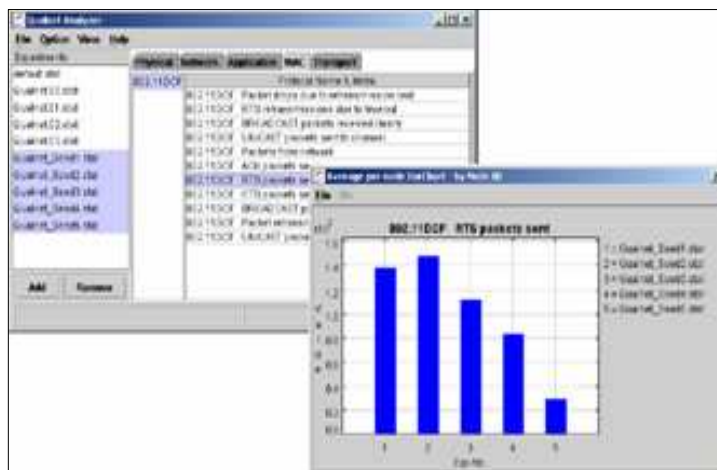


Figura 34 – Exemple diagrama de barres sobre múltiples experiments

5.11.6 – Altres recursos

- Actualitzacions del software.
- Documentació : manuals, faqs, notícies, anuncis, tècniques, projectes d'universitats, treballs, publicacions, fòrums.
- Comunitat d'usuaris:
http://www.qualnet.com/training_and_support/support/forums/index.php

5.12 – OPNET IT Guru

Versions/ Edicions	OPNET IT Guru (Comercial) OPNET IT Guru Academic Edition v1995 (Limitada i es necessari sol·licitar llicència d'ús a OPNET que no caduca)
Requeriments del sistema	Processador : Intel Pentium III, 4 o compatible (500 Mhz o mes). RAM : 256 MB. Espai de disc : per a l'edició acadèmica 200 MB, per a la comercial depèn dels mòduls adquirits, però mínim 500MB. Pantalla: 1024 x 768 o major resolució, 256 colors o mes Plataforma : Windows NT, Windows 2000 o Windows XP (amb sp1).
Venedor / Desenvolupador	OPNET Technologies, Inc. 7255 Woodmont Avenue - Bethesda, MD 20814 Website: http://www.opnet.com

5.12.1 – Descripció del software

OPNET IT Guru proporciona un entorn virtual de xarxa que modela el comportament d'una xarxa per complet, incloent-hi els routers, commutadors, protocols, servidors i aplicacions en xarxa.

Aquest entorn de treball es de gran utilitat per als responsables d'informàtica i I+D, dissenyadors de xarxes, administradors, operadors i personal de manteniment de xarxa, ja que permet :

Diagnosticar el funcionament i els problemes de configuració :

- Visualitzar la xarxa, fluxos de tràfic i transaccions d'aplicacions.
- Detectar errors de configuració d'ajust de xarxa.
- Detectar la causa i l'origen d'un funcionament pobre d'un servidor, client, xarxa, etc.

Validar canvis en la xarxa abans d'implementar-los i preveure el comportament de la xarxa davant de futurs escenaris, com ara creixement de transit, fallades de xarxa, etc.

- Millores de capacitat.
- Millores dels protocols.
- Validar polítiques de QoS.

Definir un pla per al creixement :

- Establir pressupostos justificats quantitativament.
- Millores en el pla de creixement de tràfic o noves instal·lacions.
- Optimització en l'ús de noves tecnologies.

Característiques i components

Disposa d'un entorn gràfic de qualitat que incorpora diferents eines accessibles des de l'editor de projectes, que és l'editor gràfic des d'on es du a terme tot el cicle de modelització i simulació, a través de les fases de disseny, construcció i simulació.

Aquestes eines permeten especificar els models amb molt de detall, identificar els elements, executar la simulació i analitzar les dades de sortida.

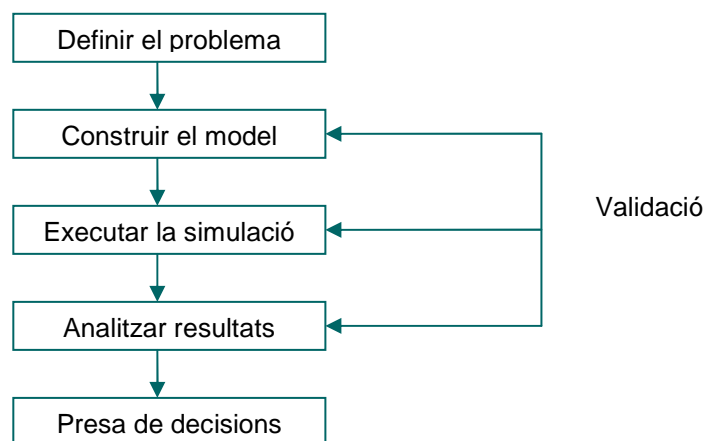


Figura 35 – Fases en la construcció d'un model i simulació.

Utilitza una estructura jeràrquica per a la construcció dels models. Cada nivell de jerarquia descriu diferents aspectes del model a simular. Els models utilitzats en una capa poden ser usats per un altre model en una capa més alta.

Disposa d'un motor de simulació amb la possibilitat de modificar l'ús de la memòria durant la simulació. La simulació es pot dur a terme en un espai de temps relativament curt.

Disposa d'una detallada llibreria de models que dona suport als protocols i tecnologies existents.

Permet la generació automàtica de la simulació al compilar els models en codi executable. Permet la depuració de l'execució de la simulació generant dades de sortida.

Llibreria de models de simulació d'events discrets

La major part de les característiques i eines estan incloses en la llibreria de models estàndards de la distribució. Determinades característiques i mòduls s'han d'adquirir apart i tenen un cost econòmic.

En l'[annex 3](#) es relacionen tots els models disponibles.

Entre les característiques i mòduls que proporcionen eines per a treure el màxim profit del producte i que no figuren en la llibreria estàndard, destaquen :

- Els mòduls ACE :
 - ACE (Application Characterization Environment) : aplicació per a l'anàlisi de xarxes. Permet a les empreses identificar d'arrel problemes existents en les prestacions de les aplicacions en xarxa, i resoldre aquests problemes de manera eficient i amb baix cost. A més permet quantificar l'impacte dels canvis realitzats. Pot interactuar amb d'altres analitzadors de protocol.
 - AAC (ACE Advanced Console module) : habilita la captura de paquets per al seu posterior anàlisi amb ACE.
 - ACE decode : motor de descodificació protocol/aplicació per a ACE.
- NetDoctor : motor d'anàlisi de la configuració de la xarxa per a encaminaments, seguretat, polítiques i canvis d'administració.
- Server modeling : per a la planificació de la capacitat dels servidors i modelat amb simulació d'events discrets. Permet a la plataforma el modelat independent de sistemes de servidor i components. Està especialitzat en models per entendre l'impacte de canvis significatius del seu ambient de tractament, canvis de càrregues de treballs, consolidació del servidor i de distribucions de les aplicacions.
- DOCSIS : simulació d'events discrets protocol DOCSIS, per a planificació i I+D.

5.12.2 – Construcció de models

Com ja s'ha mencionat, utilitza una estructura jeràrquica :

- Xarxa : especifica la topologia física de la xarxa de comunicacions, definint la posició i la interconnexió entre cada entitat. Permet especificar un model de xarxa complex, composta per diferents subxarxes intercomunicades. Cada subxarxa es compon de nodes i enllaços.
- Nodes : defineix els nodes i els fluxos de dades. La creació de dispositius i la comunicació d'aquests amb el nivell de xarxa, son expressats com a mòduls interconnectats. Aquests mòduls s'agrupen en dos grups : els que porten predefinits característiques i paràmetres configurables (p.e. generadors de paquets, receptors de radio, transmissors punt a punt, etc.) i els que son altament programables (p.e. processos, cues, etc.).

Cada node es descriu per un diagrama de fluxos de blocs de dades. Cada bloc de dades té la seva funcionalitat definida per un model de procés. Els mòduls estan interconnectats.

- Procés : descriu el flux lògic, el comportament del procés i els mòduls de cues. La comunicació entre processos es suportada per interrupcions. Un procés pot crear nous processos per a realitzar subtasques.

L'editor de Projectes es el principal escenari per a crear una simulació de xarxa de comunicacions. A partir d'aquest editor es pot construir un model de xarxa (utilitzant models de la llibreria estàndard), seleccionar les estadístiques a recollir, executar la simulació i veure els resultats.

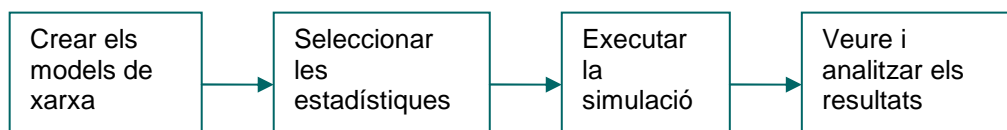


Figura 36 – Esquema d'us de l'editor de projectes

Des de l'editor de projectes :

Es crea un projecte : relacions d'escenaris (cada escenari pot contenir diferents aspectes del model).

Es defineix l'escenari : parametrizant el tipus de xarxa (mundial, empresa, campus, oficina, lògica, a escollir dels mapes), l'extensió de la xarxa (m., km, o milles) i les tecnologies a utilitzar (3Com, atm, ethernet, fddi, token_ring, etc.) associant-les amb la paleta d'objectes.

Es defineix el model de xarxa. Per a crear una topologia de xarxa, OPNET incorpora topologies predefinides per a facilitar la creació del model (bus, anell, estrella, arbre i sense connexions) configurables en diversos paràmetres.

Disposa d'una paleta d'objectes per a cada tecnologia i permet arrastrar els objectes a l'escenari (drag and drop).

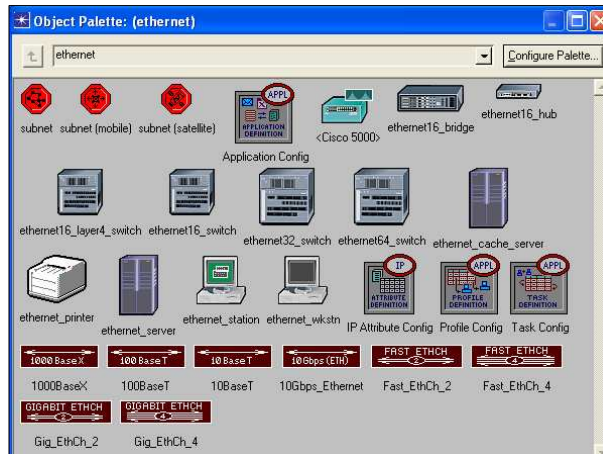


Figura 37 – Exemple de paleta d'objectes disponibles per a ethernet

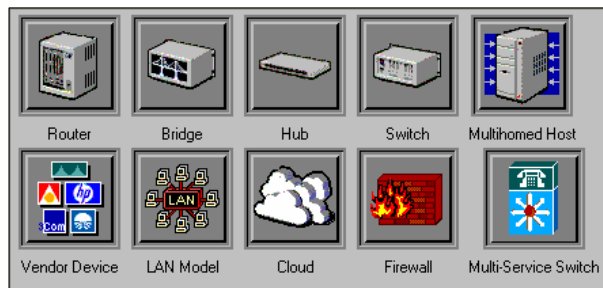


Figura 38 – Exemple de paleta d'objectes disponibles

Permet la construcció parcial del model utilitzant un model ja creat i la importació de models, de nodes (determinats productes de venedors), d'arxius de text / XML i de configuracions d'encaminadors.

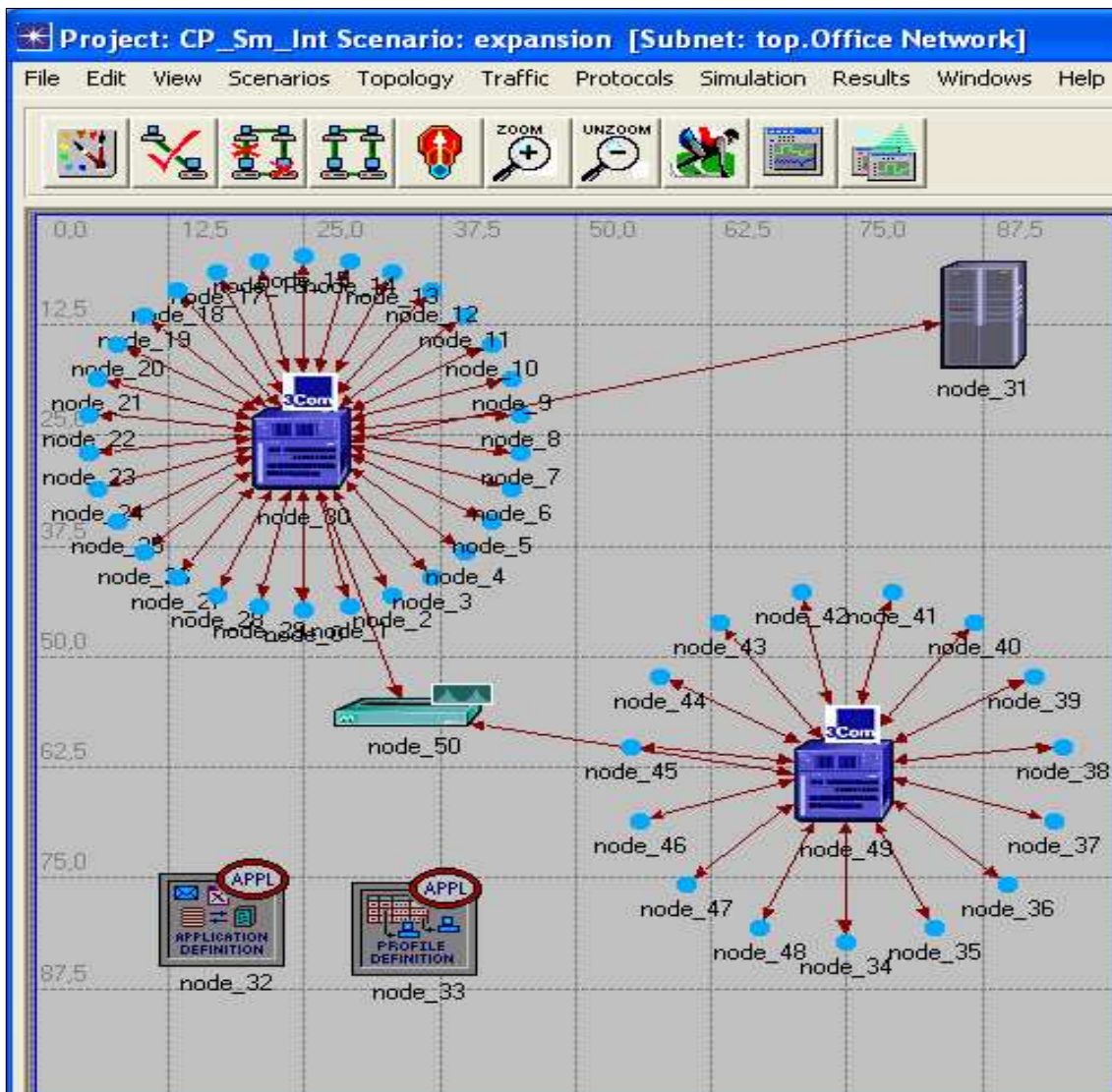


Figura 39 – Exemple d'escenari amb un model de xarxa amb dues subxarxes en estrella.

5.12.3 – Simulació : creació d'estadístiques

Permet la recollida d'estadístiques individuals de cada node o globals durant la simulació, pel que es poden seleccionar diferents anàlisis estadístics als diferents dispositius, aplicacions, enllaços i protocols i les diferents propietats dels mateixos.

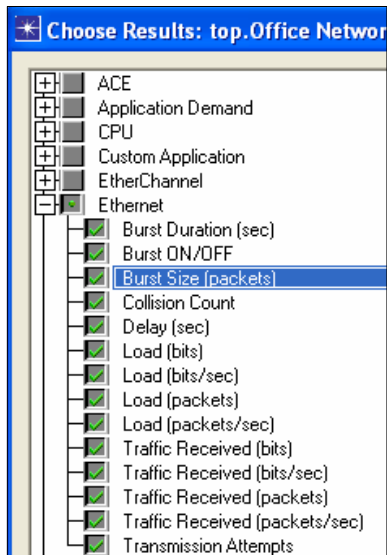


Figura 40 – Opcions d'estadístiques individuals (per a un node)

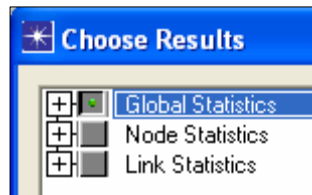
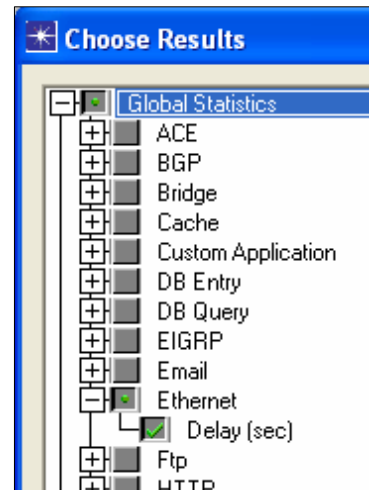


Figura 41 – Opcions d'estadístiques globals (p.e. selecció del retard de la xarxa sencera)



Per a executar les simulacions, es disposa de diferents opcions, incloent l'execució interna i externa, configurables i parametrizables.

5.12.4 – Execució, visualització i animació

La simulació s'obté a partir del simulador que incorpora OPNET en la seva versió i s'executa en el sistema del host. En base als resultats observats durant l'execució de la simulació, si cal, es poden canvis en l'especificació dels models i executar simulacions addicionals.

Permet parametrizar la simulació : temps de durada, valors, interval i habilitar o deshabilitar atributs.

Mentre s'executa la simulació, en una finestra es mostra el progrés.

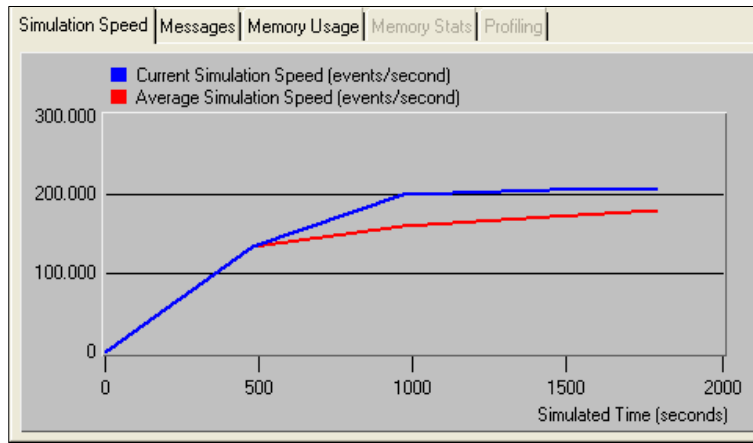


Figura 42 – Exemple de gràfic de dades sobre la simulació

Quan s'ha executat, en la mateixa finestra es mostra el resultat (en Message)

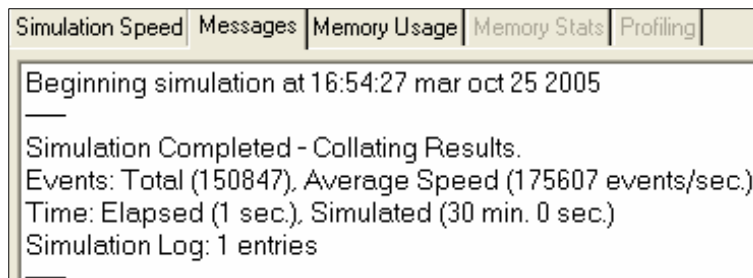


Figura 43 – Exemple de Message d'una execució d'una simulació

OPNET permet exportar les dades recollides.

5.12.5 – Anàlisi de resultats

Un cop realitzada la simulació del model, es poden visualitzar els resultats de manera gràfica amb el mateix editor de projectes.

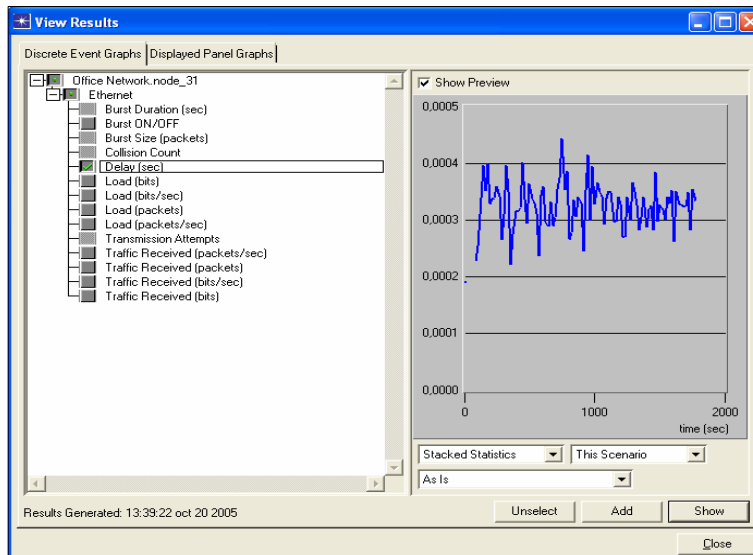


Figura 44 – Exemple gràfic del retard de la càrrega.

Comparació de resultats

Destaca la possibilitat de poder expandir la xarxa i comparar els resultats de la primera simulació amb la segona expandida, de manera que es poden fer les comparacions necessàries fins a determinar la millor estructura i/o identificar problemes.

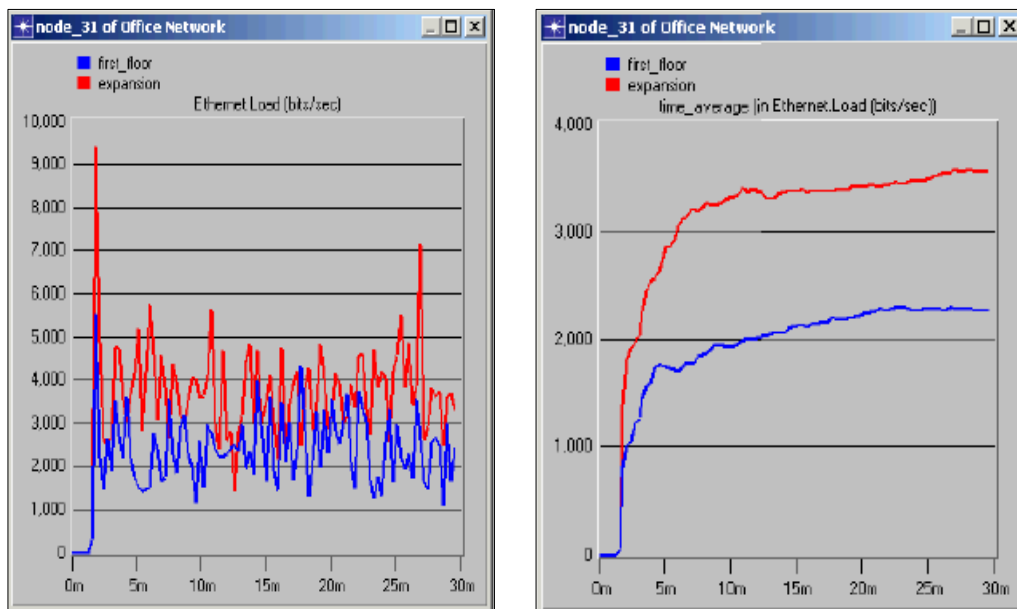


Figura 45 – Gràfiques comparatives de la xarxa i de la xarxa expandida

5.12.6 – Altres recursos

- Mòdul DOCSIS per a la generació de documentació.
- Mòdul per a l'anàlisi i visualització : captura dels rastres de les transaccions, visualització de les transaccions, diagnòstic de problemes de configuració, validació de solucions.



Figura 46 – Exemple de l'aplicació de rastreig de les transaccions amb l'agent de captura de paquets. En la darrera figura, a dalt mostra les sol·licituds i respostes entre client i servidor, al centre, mostra els paquets a nivell de xarxa, a baix, especifica les transaccions responsables de coll d'ampolla.

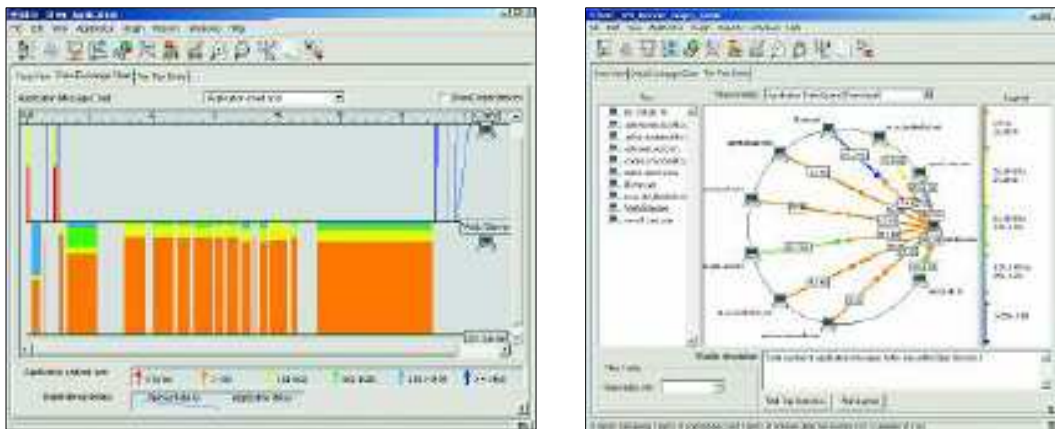


Figura 47 – Exemple de visualització de fluxos transaccions a nivell d'aplicació i de paquets de xarxa. Comprensió de les interaccions i dependències entre clients, la xarxa, servidors d'aplicacions i servidors de base de dades.

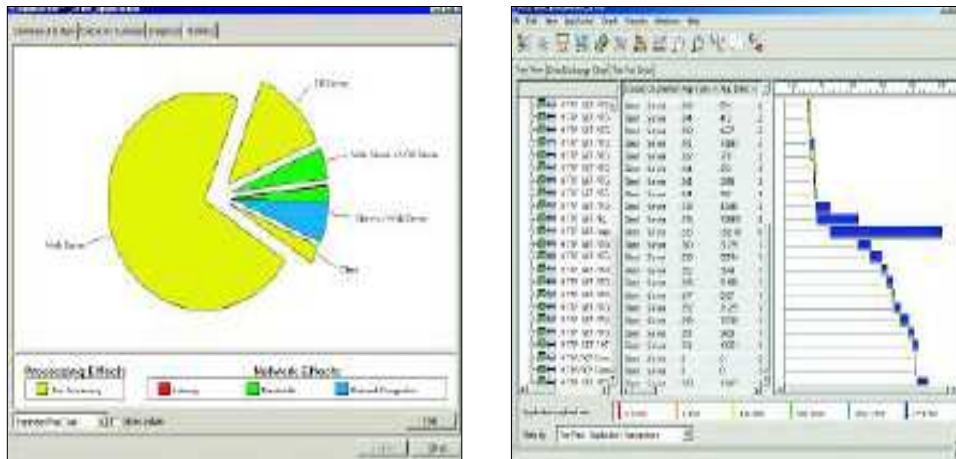


Figura 48 – Exemple de diagnòstic de problemes de funcionament o posta a punt mitjançant l'aplicació AppDoctor.



Figura 49 – Exemple de validació de solucions sobre ample de banda, configuració de protocols, velocitat dels servidors, etc.

- Mòdul NETDOCTOR : motor d'anàlisi de la configuració de la xarxa per a encaminaments, seguretat, polítiques i canvis d'administració.
- Comunitats d'usuaris : <http://www.opnet.com/services/university/>

5.13 – COMPARATIVA

Camps d'aplicació	
QUALNET Developer OMNET++	<p>La principal area d'aplicació d'aquests dos paquets d'aplicacions son la simulació de xarxes de comunicació, disseny i validació de protocols de comunicació i disseny d'arquitectures. També s'usen en d'altres àrees com la simulació de sistemes IT, xarxes de cues, arquitectures de hardware i en processos de negoci.</p> <p>Permeten dissenyar dispositius de xarxa i tecnologies experimentals, avaluar dissenys alternatius per a determinar costos i localitzar colls d'ampolla, experimentar amb xarxes de cues, realitzar simulacions per radio i xarxes sobre arquitectures seqüencials paral·leles.</p>
OPNET IT Guru	<p>Esta orientat a empreses que disposen d'aplicacions en xarxa i necessiten d'una infraestructura IT per a garantir un funcionament satisfactori del seu negoci. Permet realitzar diagnòstics de funcionament i de problemes de configuració, i validar canvis en la xarxa abans d'implementar-los i preveure el comportament de la xarxa davant de futurs escenaris, com ara creixement de transit, fallades de xarxa, etc. Els beneficis més destacables son la reducció de costos, la millora de l'eficiència operacional, l'increment de la productivitat del negoci i la reducció de riscos.</p>

Distribucions gratuïtes	
OMNET++ v3.2	Es de llicència pública acadèmica
QUALNET Developer Demo	Es de llicència privada acadèmica facilitada per SNT prèvia sol·licitud. Vàlida per a 14 dies
OPNET IT Guru AE v1995	Es de llicència privada acadèmica facilitada per OPNET prèvia sol·licitud. No prescriu.

Limitacions distribucions gratuïtes	
OMNET++ v3.2	No està limitada ni en les característiques de la distribució ni en les llibreries de models.
QUALNET Developer Demo	No està limitada ni en les característiques de la distribució ni en les llibreries de models, però sí en el temps d'ús.
OPNET IT Guru AE v1995	No està limitada en les característiques de la distribució però sí en les llibreries de models i en els mòduls (aporten característiques i recursos diferents, com p.e. ACE).

Requeriments	
OMNET++	Per a crear nous models es pot utilitzar Microsoft visual C++ 6, 7 També es poden utilitzar Cygwin o MinGW. Per a construir aplicacions tcl/tk es necessari els binaris BLT i part de XFree86.
QUALNET Developer	Per a la interfície gràfica de l'entorn virtual, és necessari el paquet de software Sun Java 2 SDK, Standard Edition 1.4.2. Per a compilar models nous, es necessari Visual C++ 6.0 amb Visual Studio 6.0 Service Pack 5 o superior.
OPNET IT Guru	Les eines necessàries estan integrades en la distribució.

Característiques i funcionalitats principals	
OMNET++	Disseny d'escenari, animació de la simulació, anàlisi de resultats, seguiment de paquets, disseny de protocols i generació de documentació.
QUALNET Developer	Disseny d'escenari, animació de la simulació, anàlisi de resultats, seguiment de paquets, disseny de protocols.
OPNET IT Guru	Disseny d'escenari, simulació i anàlisi de resultats. Les característiques d'animació, de seguiment de paquets, de disseny de protocols i de generació de documentació formen part de mòduls integrables (amb un cost econòmic).

Llibreries de models	
OMNET++	Disposa de models de la majoria de tècniques i protocols per a xarxes amb i sense fils, incloent models de mobilitat. La majoria s'han de compilar primer. L'edició comercial OMNEST les incorpora totes ja compilades.
QUALNET Developer	Disposa de models de pràcticament totes les tècniques i protocols per a xarxes amb i sense fils, mobilitat, cel·lulars i satèl·lit.
OPNET IT Guru	L'edició comercial disposa d'una llibreria estàndard que integra la majoria de tècniques i protocols. Les llibreries dels models de mobilitat, cel·lulars i satèl·lit s'han d'adquirir apart (amb un cost econòmic). L'edició acadèmica, que és bastant limitada, disposa de models d'algunes de les tècniques i protocols per a xarxes amb i sense fils.

Entorn : interfícies i eines	
OMNET++	<p>Interfície per a la construcció del model.</p> <p>Interfície gràfica per a l'execució.</p> <p>Interfície línia de comandes per a l'execució.</p> <p>Interfície per a l'anàlisi d'estadístiques vectorials.</p> <p>Interfície per a l'anàlisi d'estadístiques escalars</p> <p>Eina per a la generació de documentació (amb diagrames, taules, etc.) que pot treballar conjuntament amb Doxygen.</p> <p>Suport estàndard per a les entrades i sortides (text pla i/o XML).</p>
QUALNET Developer	<p>Interfície gràfica única que inclou diferents eines gràfiques per a la construcció del model (scenario designer), execució de la simulació (animator), anàlisi de resultats (analyzer), disseny de protocols (protocol designer) i rastreig de paquets (paquet tracer).</p> <p>Interfície línia de comandes per a la construcció i execució.</p>
OPNET IT Guru	<p>Interfície gràfica única per a la construcció del model, execució de la simulació i anàlisi de resultats (editor de projectes).</p> <p>La resta d'aplicacions/mòduls, com poden ser ACE, NetDoctor o DocSis, disposen de la seva pròpia interfície gràfica.</p>

Dificultat en la construcció de models	
OMNET++	Es necessiten coneixements del llenguatge de programació C++ per a implementar els mòduls simples i construir l'executable. Es triga bastant fins a haver creat l'executable.
QUALNET Developer	No es necessiten coneixements de cap llenguatge de programació, tot i que per a construir el model mitjançant la línia de comandes disposa d'un codi propi.
OPNET IT Guru	No es necessiten coneixements de cap llenguatge de programació. La distribució inclou topologies, dispositius i tecnologies predefinides.

Interfície gràfica per a la construcció de models i prestacions	
OMNET++	Disposa de vista gràfica i de codi per a la construcció del model. Disposa de vista de l'estructura jeràrquica dels mòduls que es van creant, el que permet un correcte seguiment del modelat i una fàcil parametrització de tots els mòduls per a una implementació posterior dels simples en C++. No es poden tenir diferents escenaris oberts. Només permet copiar codi. No permet la visió completa de la xarxa fins a la simulació. Permet importació/exportació de topologies de xarxa i configuracions del model.
QUALNET Developer	L'estructura jeràrquica dels nodes en vista en arbre permet una fàcil parametrització de tots els objectes. Permet una visió de la xarxa sencera. Disposa de taules de dispositius, tecnologies i protocols que es poden afegir a l'escenari mitjançant "drag&drop". Permet copiar i arrossegar dispositius i grups sencers de dispositius a d'altres escenaris oberts. Tots els nodes i enllaços es poden clicar per a veure les seves propietats i configurar-los.
OPNET IT Guru	Disposa de taules de dispositius, tecnologies i protocols que es poden afegir a l'escenari mitjançant "drag&drop". Permet copiar i arrossegar dispositius i grups sencers de dispositius a d'altres escenaris oberts. Permet una visió de la xarxa sencera. Tots els nodes i enllaços es poden clicar per a veure les seves propietats i configurar-los.

Configuració de la simulació	
OMNET++	<p>Configuració de l'arxiu .ini (arxius a carregar, sortides, etc.)</p> <p>Configuració del recull de dades per a la creació d'estadístiques vectorials i escalars.</p> <p>Permet la generació de nombres aleatoris</p> <p>Configuració per a execució en interfície gràfica o per línia de comandes.</p>
QUALNET Developer	<p>Disposa d'un apartat propi per a parametritzar les opcions generals de configuració del model.</p> <p>Disposa d'un apartat propi per a determinar i parametritzar les estadístiques a recollir.</p> <p>Permet la generació de nombres aleatoris</p>
OPNET IT Guru	<p>Permet la recol·lecció de diferents estadístiques individuals de cada node o globals.</p> <p>Per a executar les simulacions, es disposa de diferents opcions, incloent l'execució en segon plà, configurables i parametritzables.</p>

Execució visualització i animació	
OMNET++	<p>Interfície gràfica TKenv : permet depuració en temps d'execució, simulació pas a pas i continua, clicar sobre els objectes i variables per a analitzar canvis al vol, simulacions paral·leles, veure els vectors de sortida durant l'execució, animació dels missatges, finestres diferents per a la sortida de cada mòdul, finestres de missatges, controls pausa, inici, fi .</p> <p>Interfície línia de comandes : millor per a realitzar simulacions per lots.</p> <p>Dos modes : normal (permet depuració) i exprés (per a simulacions llargues)</p>

<p>QUALNET Developer</p>	<p>Disposa de control de temps de la simulació i de temps real des de que s'està executant la simulació. Permet accelerar/disminuir la velocitat de la simulació. Disposa diversos efectes d'animació i propagació per a les antenes.</p> <p>Permet l'aplicació de filtres per a l'animació des de dos perspectives diferents. Disposa de filtres per habilitar/deshabilitar capes i comandes de l'animació. Permet veure els missatges de sortida.</p> <p>Disposa d'execució en segon pla on especificar el nombre d'experiments a realitzar i els valors de les variables i l'assignació de llavors.</p> <p>Permet la recol·lecció dinàmica d'estadístiques.</p>
<p>OPNET IT Guru</p>	<p>Permet parametritzar la simulació : temps de durada, valors, interval i habilitar o deshabilitar atributs.</p> <p>Mentre s'executa la simulació, en una finestra es mostra el progrés.</p>

<p>Anàlisi de resultats</p>	
<p>OMNET++</p>	<p>Dues interfícies gràfiques : PLOVE anàlisis d'estadístiques vectorials + filtre de resultats (promig, densitat, etc.) i SCALARS anàlisis d'estadístiques escalars + filtre.</p> <p>Disposa de diferents diagrames (barres, histogrames, etc.)</p> <p>Permet importació/exportació en diversos formats.</p> <p>Altres eines : Grace, Root, Gnuplot.</p>
<p>QUALNET Developer</p>	<p>Estadístiques de sortida classificades per capes i protocols.</p> <p>Disposa de diferents diagrames (barres, histogrames, etc.)</p> <p>Genera arxius de dades ASCII d'aplicacions neutrals per a importar a altres programes com poden ser Gnuplot o Excel.</p> <p>Suporta la comparació d'un únic experiment per node i la comparació d'experiments múltiples.</p>
<p>OPNET IT Guru</p>	<p>Disposa de diferents diagrames (barres, histogrames, etc.)</p> <p>Permet importació/exportació en diversos formats.</p> <p>Destaca la possibilitat de poder expandir la xarxa i comparar els resultats de la primera simulació amb la segona expandida, de manera que es poden fer les comparacions necessàries fins a determinar la millor estructura i/o identificar problemes.</p>

Altres recursos	
OMNET++	<p>Eines i utilitats per a la generació de documentació.</p> <p>Eines i utilitats per a l'anàlisi i visualització a mes de Plove i Scalars.</p> <p>Documentació, exemples de simulació, material de contribució, codi documentat.</p> <p>Comunitat d'usuaris.</p>
QUALNET Developer	<p>Actualitzacions del software.</p> <p>Documentació : manuals, faqs, notícies, anuncis, tècniques, projectes d'universitats, treballs, publicacions, fòrums.</p> <p>Comunitat d'usuaris:</p>
OPNET IT Guru	<p>Actualitzacions del software.</p> <p>Documentació : manuals, faqs, notícies, anuncis, tècniques, projectes d'universitats, treballs, publicacions, fòrums.</p> <p>Comunitat d'usuaris:</p>

5.14 – CONCLUSIONS

En aquest apartat, s'ha ofert un primer contacte amb els simuladors de xarxes de computadors. Això ha inclòs una comprensió de la importància dels simuladors, així com conèixer les característiques fonamentals dels mateixos.

Els diferents simuladors ofereixen els mòduls necessaris per a programar de manera àgil un model. Es poden destacar : la generació de nombres aleatoris que responen a certes funcions de distribucions particulars, gestió automatitzada del temps de simulació, gestió automatitzada de les rutines de tractament dels events que determinen el comportament del sistema, algorismes d'anàlisi dels resultats i generació d'informes.

Totes aquestes característiques repercuteixen en una reducció considerable del temps de programació del model.

La complexitat dels simuladors no ha permès fer un anàlisi i comparació exhaustiva i en profunditat, empitjorada per la dificultat en obtenir edicions gratuïtes de simuladors amb les màximes prestacions.

Segons els criteris que proposa la norma [ISO 9126](#), a continuació s'exposen les conclusions finals de l'anàlisi i comparativa realitzada, en aquest apartat, dels tres softwares de simulació.

Funcionalitat

Construcció del model

Tots tres disposen d'una descomposició jeràrquica del model que permet una millor estructuració.

Opnet IT Guru no disposa de mecanismes de programació equivalents a llenguatges d'alt nivell.

Omnet++ és el que ofereix una construcció del model més difícil ja que exigeix coneixements de programació en C a l'hora de implementar els mòduls simples.

Simulació i animació.

Opnet IT Guru no disposa de la possibilitat d'animació tot i que la pot emmagatzemar en un fitxer per a una posterior visió.

Omnet++ i Qualnet Developer permeten la importació de gràfics.

Entrades i sortides.

Tots tres softwares disposen de distribucions de probabilitat per a modelar el comportament dels diferents elements. Disposen de la possibilitat de definir i analitzar estadístiques de les dades recollides. Permeten efectuar simulacions múltiples amb diferents seqüències de nombres aleatoris. Permeten la lectura i escriptura a arxius.

Facilitat

Opnet IT Guru i Qualnet Developer tenen una gran capacitat a l'hora de definir gràficament els elements del model i les seves relacions. També tenen una gran capacitat d'interacció entre el mouse i l'entorn.

Tots tres softwares disposen d'una documentació de qualitat, de manual interactiu, generació automàtica d'estadístiques, opcions de depuració i seguiment, exemples de referència i mecanismes de detecció d'errors.

Eficiència

La velocitat d'execució és millor en Opnet IT Guru i Qualnet Developer que en Omnet++. Omnet++ i Qualnet Developer permeten executar el model amb animació, a marxa ràpida i en procés per lots, mentre que Opnet IT Guru no disposa d'animació.

Fiabilitat

Opnet IT Guru i Qualnet Developer, al tenir un cost econòmic considerable, ofereixen en front a Omnet++ :

- Una certificació de qualitat del software.
- Una tecnologia més robusta en el desenvolupament de l'entorn.
- Mecanismes de correcció d'errors.
- Millores proporcionades a l'usuari final.

Portabilitat

Opnet IT Guru inclou totes les eines necessàries integrades en la distribució, mentre que Omnet ++ i Qualnet Developer necessiten d'entorns i compiladors addicionals.

PART III

5.15 – ESTUDI DE SIMULACIO AMB OPNET IT GURU

Com ja s'ha comentat en la Part I de la memòria, les passes a seguir son [\(Carson; 2003\)](#) [\(Banks; 1999\)](#) :

1. Enunciar explícitament els objectius que es pretenen, els interrogants que es plantegen, les hipòtesis que es volen demostrar i les diferents possibilitats a considerar.
2. Crear un model lògic-matemàtic. És fonamental modelar els fenòmens aleatoris del sistema i les seves característiques.
3. Dissenyar o utilitzar un programa d'ordinador per a simular el model i verificar que el programa es comporta com es desitja.
4. Validar el model a través de proves pilot amb diferents inputs i comparar els resultats estadísticament [\(Sargent; 1999\)](#). El model ha de reflectir convenientment el model real.
5. Dissenyar la simulació i executar-la utilitzant el model per a respondre a les preguntes que inicialment se'ns plantejaven.
6. Reunir, processar i analitzar les dades generades com a solucions del model i en termes de validesa i fiabilitat estadística. Documentar l'experiment.

5.16 – Objectius

La xarxa del banc MANSENLAIRE es connecta a Internet a través d'un firewall. Els usuaris utilitzen diverses aplicacions online com son: email, web, autorització de targetes de crèdit i impressió d'arxius. A més a més, alguns usuaris estan descarregant arxius de música i vídeo piratejats. De moment, el tràfic il·lícit no està bloquejat.

L'aplicació més crítica del banc és l'autorització de targetes de crèdit, és necessari que la petició obtingui resposta en menys de 2 segons.

L'objectiu és avaluar la política d'implementació del firewall per tal de bloquejar el tràfic il·lícit i analitzar el seu efecte en el funcionament de les aplicacions i en l'ús dels enllaços.

5.17 – Model lògic-matemàtic

Mitjançant diferents monitors hardware i software, així com diferents benchmarks, obtenim les dades necessàries per a poder modelar el sistema de la manera més convenient per als nostres objectius.

En aquest estudi de simulació, el model lògic-matemàtic se suposa ja creat, pel que es podran observar les dades d'aquest model en els paràmetres assignats als diferents dispositius, protocols i aplicacions.

Es simularà la xarxa en una hora del dia de bastant tràfic per a avaluar el funcionament de l'aplicació crítica.

5.18 – Utilització i verificació del software : construcció del model

Les característiques i components de la xarxa del banc son les següents :

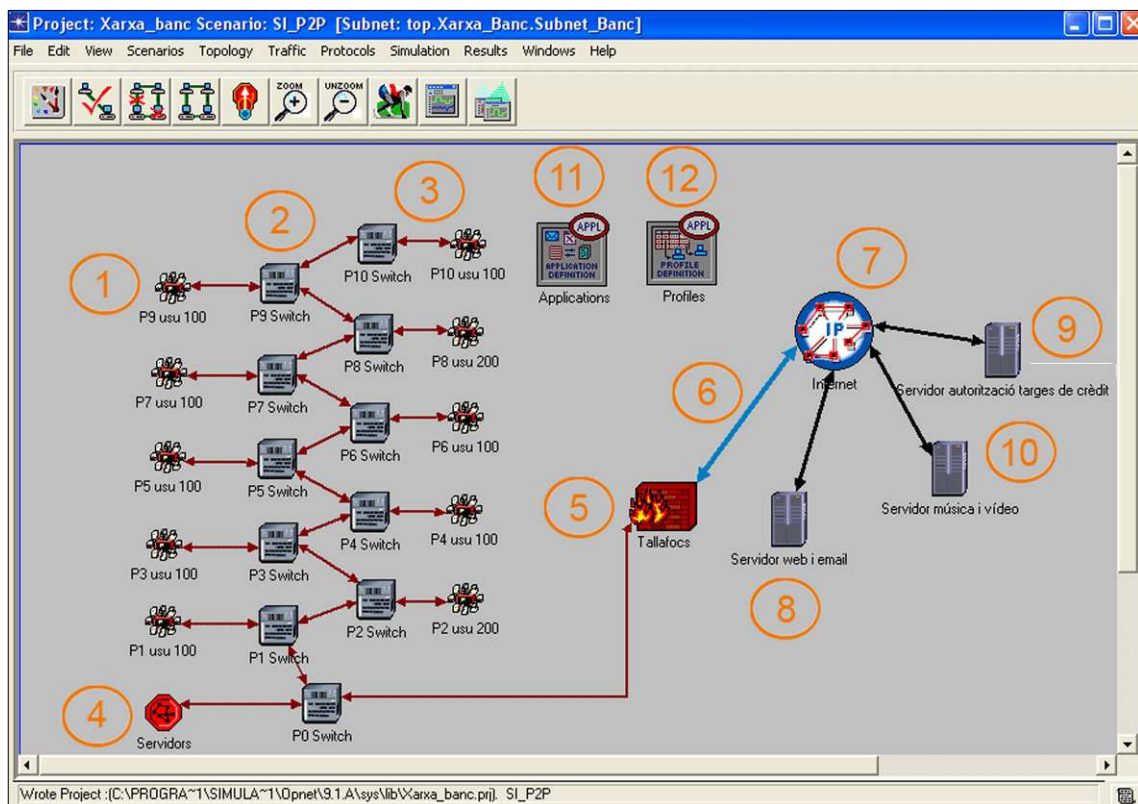


Figura 50 – Estructura de la xarxa del banc

1. Lans



L'edifici del banc consta de 11 plantes. Per a cada planta de la 1 al la 10 hi ha una LAN de determinat nombre d'usuaris. L'objecte 10BaseT_LAN representa un LAN Ethernet en tecnologia switch. Pot contenir qualsevol nombre d'estacions i un servidor. El tràfic client pot ser dirigit tant al servidor intern com al servidor extern.

Suporta les aplicacions FTP, Email, Database, Custom, Rlogin, Vídeo, X windows, HTTP i algunes d'altres. Aquestes aplicacions corren sobre TCP o UDP. Per a cada aplicació es pot especificar el tràfic per grups de clients, permetent caracteritzar la LAN sencera.

Altres atributs son : switching speed (default = 500,000pkts/sec) i LAN Server Name (default = AutoAssigned)

2. Switchs



Cada planta de l'edifici disposa d'un switch. El model de node ethernet16_switch representa un switch que suporta interfícies ethernet. Implementa l'algorisme Spanning Tree per a assegurar un lliure retorn a la topologia de xarxa. Es comunica amb els altres mitjançant l'envio de BPDU (Bridget Protocol Data Units). Els paquets son rebuts i processats pel switch basant-se en la configuració del spanning tree.

Suporta els protocols : Spanning Tree Bridge Protocol (IEEE 802.1D), Ethernet (IEEE 802.3).

Utilitza les interconnexions (Port Interface Description) : 16 Ethernet connections a la taxa de dades especificada (10, 100, 1000 Mbps).

Restriccions: només pot connectar LANs del mateix tipus.

3. Enllaços



Els enllaços son 10BaseT duplex que representen la connexió ethernet. Connecten qualsevol combinació de estacions, hub, bridge, switch i LANs (excepte hub a hub). El format dels paquets és ethernet. La taxa de dades és de 10Mbps.



Enllaços que connecten dos nodes sobre IP. El format del paquet és ip3_dgram amb una taxa de dades T1 (1.544 Mbps).

4. Subnet : servidors email i impressió d'arxius

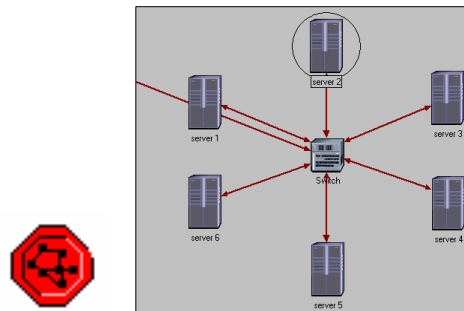


Figura 51 – Subnet i estructura de la mateixa. 6 servidors (d'impressió arxius i email).

La subxarxa consta de 6 servidors i 1 switch. El switch té les mateixes característiques que els altres, i els servidors corresponen al model de node `eth4_slip4_multihomed_server` node que representa a un servidor amb suport per a aplicacions amb client-server que corren sobre TCP/IP i UDP/IP. Requereix una quantitat fixa de temps d'enrutament de cada paquet determinat per la taxa d'IP forwarding del node. Els paquets s'enruten basats en first-come-first-serve i poden trobar cues als ports, depenent de la taxa de transmissió de l'interface de sortida.

Suporta els protocols: RIP,UDP,IP,TCP,OSPF.

Diferents atributs son l'adreça de transport, la taxa d'IP forwarding, la funció d'IP gateway, el mode de procés RIP, la informació de la connexió TCP, el tamany màxim del segment TCP i la capacitat del buffer de recepció TCP.

5. Tallafocs



El firewall és un model de node `ethernet2_slip8_firewall` que representa un gateway basat en IP amb característiques de firewall i suport de servidor. Suporta 2 ethernet i 8 interfícies de línees en sèrie amb taxes de dades seleccionables.

Els protocols : Routing Information Protocol (RIP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP) o Gateway Routing Protocol (IGRP) poden ser usats automàticament. Els paquets s'enruten basats en first-come-first-serve i poden trobar cues als ports, depenent de la taxa de transmissió de l'interfície de sortida.

Suporta els protocols: TCP, RIP, UDP, IP, Ethernet, Fast Ethernet, Gigabit Ethernet, OSPF, BGP, IGRP. Suporta 1 connexió ethernet i 1 connexió IP de línia en sèrie, les dues amb taxa de dades definible.

Admet diferents atributs com son l'adreça de transport, la taxa d'IP forwarding, la funció d'IP gateway, el temps d'inici RIP, el mode de procés RIP, informació de servidor proxi.

6. Enllaç WAN



L'enllaç ppp_adv point-to-point connecta 2 nodes amb interfícies en sèrie (p.e. enrutadors amb ports PPP) amb taxa de dades definible. El format dels paquets son ip_dgram_v4 i ipx_pkt.

7. Internet



El model de node ip32_cloud node model representa mes de 32 interfícies de línees en sèrie amb taxa de dades seleccionable. Els paquets que arriben son enrutats a una apropiada interfície de sortida basant-se en la seva adreça IP de destí. Requereix d'un temps fixat per a enrutar cada paquet, determinat per l'atribut latència de cada node. Els paquets s'enruten basats en first-come-first-serve i poden trobar cues als ports, depenent de la taxa de transmissió de l'interfície de sortida.

Suporta els protocols: RIP, UDP, IP, OSPF, BGP, IGRP, TCP. Les interconnexions: 32 connexions IP de línies en sèrie amb taxa de dades seleccionable.

Admet diferents atributs com son la latència dels paquets i el rati del rebuig de paquets.

8 i 9. Servidor web i d'autorització de targetes de crèdit



El model de servidor ppp_server model representa un node servidor amb aplicacions de servidor corrent sobre TCP/IP i UDP/IP. La velocitat operacional ve determinada per la taxa de dades de l'enllaç de connexió.

Suporta els protocols : RIP, UDP, IP, TCP, OSPF. Admet diferents atributs com son la taula de configuració de les aplicacions del servidor, adreça de transport, la taxa d'IP forwarding, la

funció d'IP gateway, el mode de procés RIP, la informació de la connexió TCP, el tamany màxim del segment TCP i la capacitat del buffer de recepció TCP.

10. Servidor música i vídeo



El model de node ppp_wkstn node representa una estació de treball amb aplicacions client-servidor corrent sobre TCP/IP i UDP/IP. Requereix una quantitat fixa de temps d'enrutament de cada paquet determinat per la taxa d'IP forwarding del node. Els paquets s'enruten basats en first-come-first-serve i poden trobar cues als ports, depenent de la taxa de transmissió de l'interfície de sortida.

Suporta els protocols : RIP, UDP, IP, TCP, OSPF.

Admet diferents atributs com son la taula de configuració de les aplicacions del servidor, adreça de transport, la taxa d'IP forwarding, la funció d'IP gateway, el mode de procés RIP, la informació de la connexió TCP, el tamany màxim del segment TCP i la capacitat del buffer de recepció TCP.

11. Aplicacions



El node "Application Config" s'utilitza per a especificar les aplicacions que s'usaran en el model. En aquest cas son : transferència de música i vídeo, impressió d'arxius, autorització de targetes de crèdit, email i navegació web.

12. Profiles



El node "Profile Config" s'utilitza per a crear els perfils d'usuari que es poden especificar en diferents nodes en la xarxa per a generar el tràfic de la capa d'aplicació.

5.19 – Validació del model

Realitzats diferents experiments amb diferents entrades i comparant els resultats estadísticament, obtenim el model que millor reflecteix el model real en els aspectes d'interès.

5.20 – Disseny de la simulació i execució

L'objectiu és determinar el temps de resposta (per a comprovar que sigui inferior a 2s) a les peticions d'autorització de targetes de crèdit. Per tant, es seleccionen les estadístiques :

- Globals : "DB Query" , temps de resposta (sec).
- Enllaços : "Point-to-point" , us (←)

Es configura la simulació amb 1 hora de durada, llavor 128, 100 valors per estadística, intervals de 100000 events.

5.21 – Anàlisi de resultats i conclusions

Superposant en el mateix gràfic el temps de resposta i el promig d'aquests, es pot observar que el temps de resposta és superior a 2 segons.

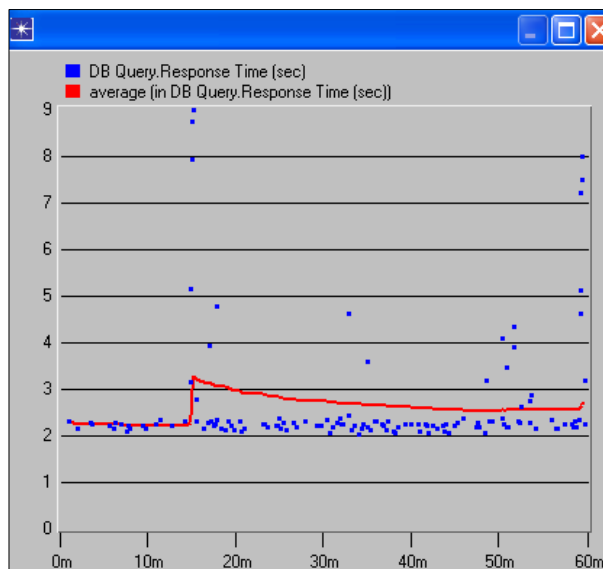


Figura 52 – Gràfic temps resposta.

En el gràfic d'ús de l'enllaç WAN es pot observar que és molt alt, el que contribueix a ampliar el temps de resposta.

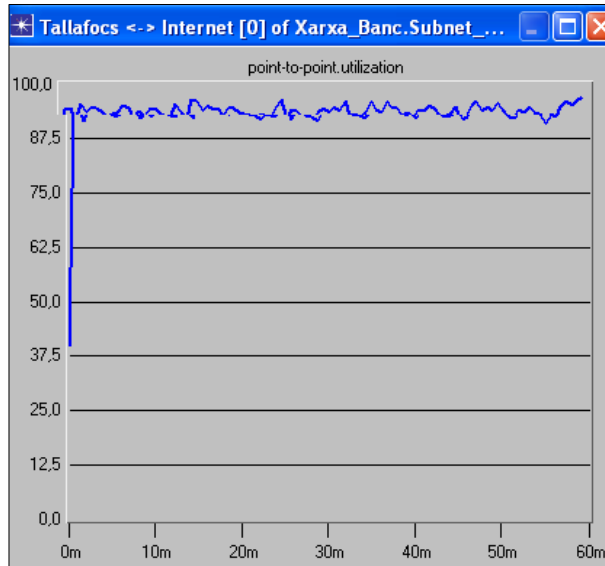


Figura 53 – Gràfic ús enllaç WAN.

Si en les mateixes condicions de treball, s'implementa el firewall bloquejant les aplicacions P2P, s'obtenen els següents resultats :

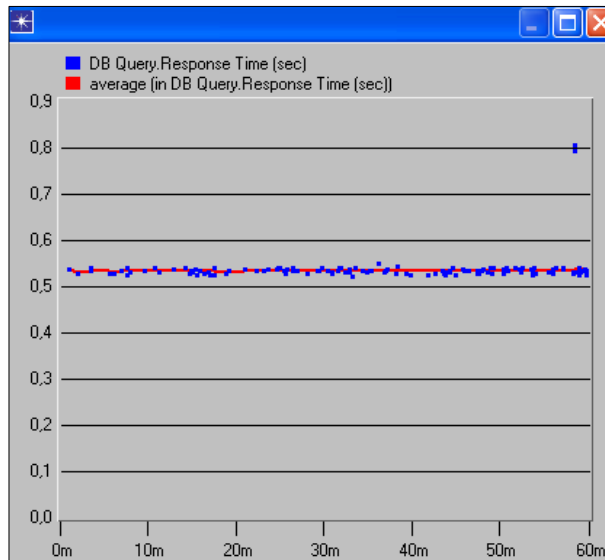


Figura 54 – Gràfic temps resposta.

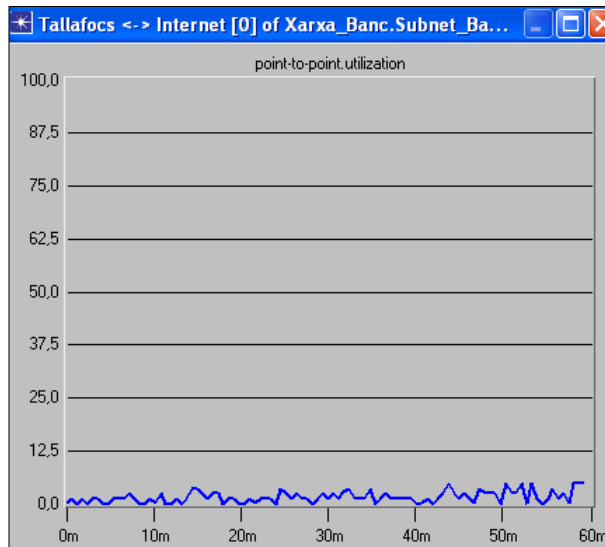


Figura 55 – Gràfic ús enllaç WAN.

Si es comparen els resultats :

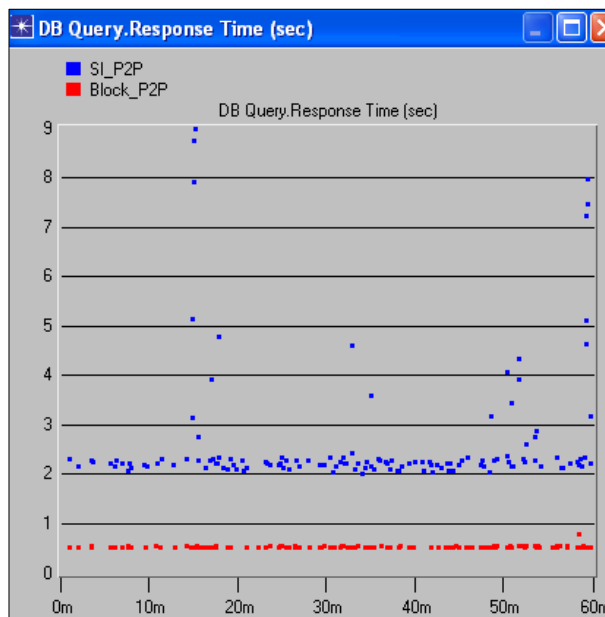


Figura 56 – Gràfic comparativa temps de resposta amb i sense P2P.

Com es suposava, el resulta d'implementar el tallafocs ha tingut una incidència important en el funcionament de l'aplicació d'autoritzacions de targetes de crèdit, de manera que pot atendre les peticions en menys d'un segon.

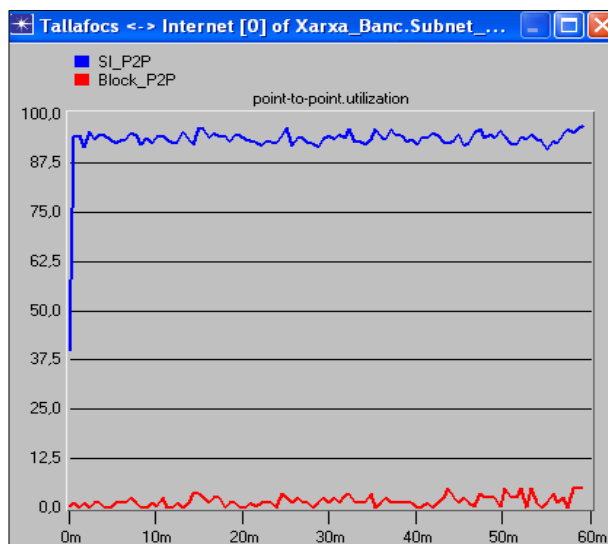


Figura 57 – Gràfic comparativa ús enllaç WAN amb i sense P2P.

Es pot observar com els gràfics mostren una reducció molt important en l'ús de l'enllaç WAN.

Això fa reflexionar al banc i decidir implementar el firewall de manera que bloquegi les comunicacions P2P.

- **Adreça MAC** : (Media Access Control Address) és un identificador físic – un nombre únic al mon de 48 bits – emmagatzemat en fàbrica dins d'una targeta de xarxa o una interfície utilitzada per a assignar globalment adreces úniques en alguns models OSI (capa2) i en la capa física del conjunt de protocols d'Internet.
- **API** : (Application Program Interface) és na Interfície de Programació d'Aplicacions. De fet, es tracta d'un conjunt de funcions, protocols i constants que faciliten la tasca al programador per a desenvolupar aplicacions, car s'ha de limitar a unir totes les peces amb el fi proposat.
- **Capa de xarxa** : també anomenada capa d'Internet, es troba entre la capa d'accés a la xarxa i la capa de transport en el model TCP/IP. Aquesta capa és la que permet als servidors (hosts, en anglès) posar paquets de dades dins d'alguna xarxa i viatjar als seus destins. És a dir, ofereix el servei d'encaminament dels missatges i de traducció de les adreces lògiques en adreces físiques. També efectua el control de la congestió de la xarxa i la reordenació de paquets un cop arribats a la màquina de destí. És aquí on es defineix un format de paquet oficial i un protocol anomenat IP.
- **Capa de transport** : és la part del protocol TCP/IP encarregada de garantir la transmissió de les dades.
- **Compilador** : programa d'ordinador que tradueix un llenguatge informàtic d'alt nivell, com el Visual Basic o el C, per exemple, a un llenguatge de baix nivell que pot llegir la UCP (de l'anglès CPU) d'un ordinador.
- **Depurar** : és el procés d'identificar i corregir errors de programació.
- **Enllaç** : canal de comunicació entre dos nodes adjacents.
- **Enrutador** : (o encaminador) és un dispositiu hardware o software d'interconnexió de xarxes de computadores que opera en la capa3(nivell de xarxa) del model OSI. Interconnecta segments de xarxa o xarxes senceres. Fa passar paquets de dades entre xarxes prenent com a base la informació de la capa de xarxa.
- **Entorn** : conjunt de software per a oferir a l'usuari un ambient amigable i còmode.
- **Event discret** : acció instantània que pot canviar l'estat d'un model dinàmic, estocàstic i discret.

- **GUI** : (Interfície gràfica d'usuari) creada pel programadors durant la fase de desenvolupament del programa per facilitar-ne el seu ús.
- **Hipòtesi** : proposició de la que es parteix per a comprovar la veracitat d'una tesis mitjançant arguments vàlids.
- **Kernel** : (o nucli) és la part fonamental d'un sistema operatiu. Es el software responsable de facilitar als diferents programes els accessos al hardware, gestionant els recursos a través de crides al sistema.
- **Lan** : (Local Area Network (Xarxa d'Àrea Local), i es tracta d'una xarxa de comunicacions utilitzada per una organització a través d'una distancia limitada la qual permet als usuaris la compartició d'informació i recursos. A la xarxa s'hi poden connectar dispositius de tot tipus com computadores, terminals, perifèrics, etc.
- **Llavor** : valor inicial per a la generació de nombres aleatoris.
- **Llibreria** : (o biblioteca) és un conjunt de procediments i funcions (subprogrames) agrupats en un o mes arxius amb la finalitat de ser aprofitades per altres programes.
- **Model** : és una idealització de la realitat utilitzada per a plantejar un problema, generalment des de el punt de vista matemàtic. Es una representació conceptual d'un procés o sistema, amb la finalitat d'analitzar la seva natura o comprovar hipòtesis, i permetre una millor comprensió del fenomen real al qual el model representa.
- **Nombre Aleatori** : és el resultat d'una variable a l'atzar especificada per una distribució. Quan no s'especifica cap distribució, se suposa que utilitza una distribució uniforme contínua en el interval $[0,1)$.
- **Programació Orientació a objectes** : (POO o OOP) és una metodologia de disseny de software i un paradigma de programació que defineix els programes en termes de classes d'objectes", objectes que son entitats que combinen estat (es a dir, dades) i comportament (procediments o mètodes). Aquests mètodes estan pensats per a fer els programes i mòduls més fàcils de mantenir, escriure i reutilitzar.
- **Protocol de xarxa** : (o protocol de comunicació) és el conjunt de regles que controlen la seqüència de missatges que ocorren durant una comunicació entre entitats que formen una xarxa.

- **Servidor** : Computadora en la que s'executa una o mes aplicacions que realitzen alguna tasca en benefici d'altres aplicacions allotjades en altres computadores anomenats clients.
- **Simulació** : és l'experimentació amb un model d'una hipòtesi de treball.
- **Sistema** : entitat material formada per parts organitzades i els seus components, que interactuen entre sí de manera que les propietats del conjunts no poden deduir-se totalment de les propietats de les parts.
- **Sistema de cues** : consisteix en un o diversos servidors que donen servei a un o mes usuaris que accedeixen al sistema. El procés d'arribades el regula una font generadora d'usuaris i les arribades seran de forma aleatòria. Aquesta font generadora d'usuaris pot ser finita o infinita.
- **Switch** : (commutador) és un dispositiu d'interconnexió de xarxes de computadores que opera en la capa 2(nivell d'enllaç de dades) del modes OSI. Interconnecta un o mes segments de xarxa, passant dades d'una xarxa a l'altre, d'acord amb una adreça de destí MAC dels datagrames de xarxa.
- **Tallafocs** : (firewall en anglès), és un element de hardware o software utilitzat per a prevenir alguns tipus de comunicacions prohibides per la política de la xarxa, les quals es fonamenten en les necessitats de l'usuari.
- **Topologia de xarxa** : tipus d'enllaç o cablejat que interconnecta els diferents nodes d'una xarxa. Un node donat pot tenir enllaços a un o més dels altres nodes. La topologia de la xarxa es determina només per la configuració de les connexions entre nodes; així que forma part de la teoria de grafs. Les distàncies entre nodes, les interconnexions físiques, les velocitats de transmissió i/o tipus de senyals no tenen importància a l'hora de definir la topologia. Si una xarxa utilitza diferents tipus de topologies s'anomena mixta.
- **Xarxa** : es un conjunt de dues o més computadores o dispositius connectats entre sí i que comparteixen informació, recursos, serveis, etc.
- **Xarxa P2P** : xarxa que no té clients i servidors fixes, sinó una sèrie de nodes que es comporten a la vegada com a clients i servidors dels altres nodes de la xarxa.
- **Wan** : (Wide Area Network) és una xarxa d'àrea ample. Son construïdes per a organitzacions o empreses particulars i d'altres per proveïdors d'Internet (ISP) per a proveir de connexió als seus clients.

7 – BIBLIOGRAFIA

- Banks, J., et al (1999) : "Introduction to Simulation". Proceedings of the 1999 Winter Simulation Conference P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, eds.
- Banks, J., et al (1998) : "Simulation, Principles, Methodology, Advances, Applications, and Practice". Jerry Banks Editions.
- Banks, J., et al (2001) : "Discrete-event system simulation". Upper Saddle River, Prentice-Hall.
- Banks, J., et al (2004) : "Discrete-Event System Simulation". Editorial Prentice Hall.
- Cao, R. (2002) : "Simulación. Métodos y Aplicaciones". Editorial Netbiblo.
- Carson, J. S. (2003): "Modeling and Simulation". Proceedings of the 2003 Winter Simulation Conference, ed. S.Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, 7-13. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Guasch, M.A., et al (2002) : "Modelado y Simulación. Aplicación a procesos logísticos de fabricación y servicio". Edicions UPC.
- Harrington, HJ (1991): "Business Process Improvement". MacGraw-Hill.
- 5807: 1985 - International Organization for Standardization (ISO) : <http://www.iso.org>
- Jensen, K. (1997): "Colored Petri Nets". Vol. 1. Springer-Verlag.
- Jeruchim, M.C., et al (2000): "Simulation of Communication Systems: modeling, methodology and techniques." Editorial Plenum US.
- Kendall, G.G. (1953): "Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of Imbedded Markov Chains". Annals of Mathematical Statistics 24, pp. 338-354.
- Law, A.& Kelton, D.W. (2000) : "Simulation modeling and analysis" . Editorial McGraw-Hill Science/Engineering/Math.
- Ortner, W & Sary, C (1998): "Virtualization of Organizations : Consequences for Workflow Modeling". Proceedings of the 32nd Hawaii international Conference on System Sciences.
- Overstreet, C.M. (1982): "Model Specification of Computer Science, Virginia Tech, Blacksburg, VA.
- Petri, C.A. (1966) "Kommunikation mit Automaten, Bonn: Institut für Instrumentelle Mathematic, Schriften des IIM Nr.2, Second Edition, New York: Griffiss Air Force database, Technical report RADC-TR-65-377, vol. 1.
- Prat, A., et al (1997): "Métodos Estadísticos. Control y mejora de la calidad". Edicions UPC.
- Sadiku, M., et al (2000): "Simulation of LANs". Editorial CRC Press. ISBN 0849324734
- Sauer, C.S., et al (1981): "Computer Systems Performance Modeling". Editorial Prentice Hall.

- Sargent, R.G., et al (1999): "Validation and Verification of Simulation Models". Proceedings of the 1999 Winter Simulation Conference. P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, eds.
- WSC : <http://www.informs-cs.org/wscpapers.html>

Enllaços web

- Association of computer machineri: <http://www.acm.org/>
- ACM SIGSIM, the ACM Special Interest Group on Simulation:
<http://www.acm.org/sigsim/home/ACMSIGSIMHome.htm>
- ACM Transactions on Modeling and Computer Simulation (TOMACS):
<http://www.linklings.net/tomacs/>
- IEEE Communications System Integration and Modeling Committee (CSIM):
<http://www.dcc.unicamp.br/%7Enfonseca/csim/>
- INFORMS College on Simulation: <http://www.informs-cs.org/>
- Journal of Statistical Computation and Simulation:
<http://www.tandf.co.uk/journals/titles/00949655.asp>
- OMNET++ <http://www.omnetpp.org/>
- OPNET <http://www.opnet.com/>
- QUALNET <http://www.qualnet.com/>
- Society for Modeling & Simulation International: <http://www.scs.org/>
- Telecommunication Systems (Modeling, Analysis, Design and Management):
[http://www.springerlink.com/\(1pym0f55slvw0ayc5l4alyzw\)/app/home/journal.asp](http://www.springerlink.com/(1pym0f55slvw0ayc5l4alyzw)/app/home/journal.asp)
- Winter Simulation Conference: <http://www.wintersim.org/>

8 – ANNEXES

8.1 – ANNEX 1 – Llibreries de models d'ONMET++

Xarxes TCP/IP	
INET Framework	Models per a IP, TCP, UDP, PPP, Ethernet, MPLS amb LDP i RSVP-TE, i d'altres protocols. Web site en CTIE. Inclou alguns suports per a simulacions mòbils i wireless, la majoria basades en codi MF
IPv6Suite	Desenvolupat en CTIE, Monash University, Melbourne. Suporta models IPv6, i disposa de models PPP, Ethernet i 802.1. Aquests models de simulació funcionen amb els següents RFCs : 2373, 2374, 2460, 2461, 2462, 2463, 2464, 2472, 2473. També amb : <ul style="list-style-type: none"> * Mobility Support in IPv6 Revision 24 * Hierarchical Mobile IPv6 Mobility Management (HMIPv6) revision 2 * Optimistic Duplicate Address Detection Revision 0 * Fast Solicited Router Advertisements Revision 4
Protsim	Utilitzat per a l'ensenyament en TU Ilmenau i TU Berlin

Protocols LAN/MAN	
Ethernet	Disponibile en CTIE. Forma part també dels models de INET framework
FDDI	FDDI i Token Ring estan inclosos com a exemples de simulació. El model FDDI està bastant detallat, el model Token Ring és relativament simple

Protocols per a ad-hoc frameworks i wireless	
Mobility Framework (MF)	Creat per TKN, TU Berlin. MF inclou un model 802.11 (ad-hoc mode només).
AODV framework	Escrit per Nicola Concer en Bologna, per a xarxes ad-hoc. Implementa el protocol AODV i bastants models mòbils mes.
NesCT	L'eina NesCT del projecte EYES WSN de UT Twente permet executar aplicacions TinyOS directament en les simulacions.

Consensus project	El projecte Consensus (TU Delft i Univ.of Twente, the Netherlands) ha creat els paquets Positif i MAC Simulator, usats en xarxes wireless i publicacions de capes MAC.
802.11	També desenvolupat per a IPv6Suite.
Hiperlan/2	Escrit per Imanol Martinez Coca.
SensorSimulator	De Louisiana State University.

Altres models de simulació	
FreeBSD TCP/IP stack	Inclou ICMP, ARP, sockets i Ethernet frames.
Ant-net	El model implementa els algorismes d'encaminament basats en Ant (AntNet-CL i AntNet-CO) .Proposat per proposed G. Di Caro i M. Dorigo.
Video interface	Aquest codi està en fase beta de desenvolupament.
Swarming P2P protocol	
Centralized Network	Basat en Hiperlan/2
RTP	Protocol Real Time Transport, models RTP v2 (RFC 1889), basats en el IP-Suite.

8.2 – ANNEX 2 – Lliberies de models de QUALNET

	Llibreria estàndard	Llibreria IPv6
Aplicacions	CBR, FTP, HTTP, Lookup, MCBR, TCPLIB, Traffic gen, Traffic trace, VBR, VOIP (H323, RTP, RCTP, H225).	CBR

	Llibreria QoS
Quality of Service	Diffserv (Traffic, Classifier, per hop behaviors), QOSPF, RED, SCFQ, WFQ

	Llibreria estàndard	Llibreria QoS	Llibreria IPv6
Transport	TCP, TCP Lite, TCP (New Reno, Reno, Sack, Tahoe), UDP.	ECN	UDP

	Llibreria estàndard	Llibreria QoS	Llibreria MANET
Multicast	DVMRP, MOSPF, Static.	PIM-DM	ODMRP

	Llibreria estàndard	Llibreria QoS	Llibreria IPv6
Queueing	FIFO	RED, RIO, WRED	FIFO
Scheduling	Strict priority	CBQ, Round robin, SCFQ, WFQ, WRR.	Strict priority

	Llibreria estàndard	Llibreria MANET	Llibreria IPv6
Routing	Acces list, AODV, BGP v4, Distributed Bellman Fordm, EIGRP, HSRP, ICMP, IGMP, IGRP, OSPF v2, RIP v1, RIP v2, Static.	DSR, Fisheye, IARP, IERP, LANMAR, LAR, OLSR, STAR, ZRP.	Static
MAC	802.3 (wired, bus), Abstract, witch, Aloha, CSMA, IEEE802.11(DCF, PFC), Detailed Switch (GARP, GVRP), Detailed Switch VLAN, Spaning tree TDMA	MACA	802.3 (wired, bus), Detailed Switch (GARP, GVRP, VLAN, Spaning tree

	Llibreria estàndard	Llibreria MANET
MPLS	MPLS (LDP, RSVP-TE)	
Physical	802.3 (Wired, bus), Abstract satellite, Abstract radio, IEEE802.11a, IEEE802.11b, Point to point link, Switched ethernet.	
Antenna	Omni directional antenna	Switched beam antenna Steerable beam antenna
Path loss		ITM Path Loss Matrix
Shadowing	Free space, 2 Ray	Longnormal
Fading		Ricean, Rauleigh
Modulation		BPSK, DPSK, DQPSK, GMSK, Turbo coding / FEC
Terrain		DEM
Mobility	Trace files, Random waypoint	Group mobility
Devices	Routers, Switches	

8.3 – ANNEX 3 – Lliberies de models d'OPNET

(en blau aquells mòduls i característiques que no estan incloses en la llibreria de models estàndards i s'han d'adquirir apart)

	Protocols Technologies : Estàndard / mòduls
Aplicacions	ACE, ACE Whiteboard, CBR, Database, Email, FTP, HTTP, Remote Login / TELNET, Self-Similar Traffic (RPG), Print, SIP, VBR, Voice, VoIP – SIP, RTP, Video, Custom.
Transport	NCP, TCP Tahoe, TCP Reno, TCP New Reno, TCP ECN, TCP SACK, UDP.
Routing	BGP, OSPF, OSPFv3, EIGRP, ISIS, RIP, RIPng, IGRP, Static Routes, Distance Vector (for ATM Networks), PNNI (for ATM Networks), Dijkstra's Algorithm (API).
Multicasting	IGMP / Static Membership, PIM-SM, Static RP, Auto RP / Bootstrap.
Network	HSRP, IPv4, IPv6, IPX, Mobile IPv4, Mobile IPv6, RSVP.
Mpls	MPLS, CSPF, Signaling (LDP / RSVP-TE), IGP Extensions (OSPF-TE / ISIS-TE), Fast Reroute, VPN.
Quality of service	CAR / Policing, CQ, DWFQ / CBWFQ, DWRR / MDRR/ MWRR, FIFO, LLQ with Rate Limit, Marking, PQ, TOS / DSCP - based classification, WRED/ RED.
Mac	Aloha, ATM (incl. IMA), CSMA, CSMA/CA, CSMA/CD, DOCSIS, DPT / SRP, Ethernet (coax, 10baseT, 100baseT), FDDI, Fiber Channel , Frame Relay, Gigabit Ethernet (10Gbps, 1000baseX), LANE, LAPB, Link Aggregation (incl. EtherChannel), SNA, Spanning Tree, Token Ring, VLAN, X.25.
Tdm	Circuit Switching.
Physical	DSL, ISDN, SONET, Custom Link, Models (OPNET Pipeline Stages).
3d visualization	3DNV (based on MAK Technologies), 3DNV Logger (based on MAK Technologies).

	Wireless : Standard / Modules
Protocols / technologies	802.11a, 802.11b, 802.11e, 802.11g, 802.16 (model development consortium), AODV, DSR, OLSR, SMART MAC, TORA, UMTS .
Custom modulations	BPSK, CCK, DPSK, FSK, GMSK, MSK, PSK, QAM, QPSK.
Mobility	Arbitrary trajectories, HLA mobility updates, Random Waypoint.
Propagation	Free Space, Longley-Rice, TIREM.
Terrain	DEM, DTED.
Custom antenna patterns	Antenna Pattern – 9 vendor specific w/ pattern viewer and editor, Antenna conversion from 2D definitions to 3D, Antenna APIs for complex antenna patterns, Isotropic, UMTS Cell, UMTS Sector.
Satellite capability	Amplifier Bandwidth, Amplifier Gain, Amplifier Noise Figure, Antenna APIs for complex antenna patterns, Antenna conversion from 2D definitions to 3D, Antenna Pattern – 9 standard vendor models, w/ pattern viewer and editor, Channel Characteristics (Bandwidth, Data Rate, Frequency, Unlimited Partitions, Signal Power, Spreading Code), Modulation Schemes – 19 standard, w/ editor for customization, Orbital Trajectory – 6 degrees, Orbital Viewing, Satellite Phase, Satellite Start / Stop time, STK Integration – Orbital Data.

	Fabricants de dispositius : Standard / Modules
Routers & switches	3Com, ADC, Alteon, Ascend (Lucent), Avici, Bay Networks (Nortel), Brocade, Cabletron, Cisco, Coyote Point, Equipe, Extreme, F5, Fore (Marconi), Foundry, HP, Juniper, Lucent, McData, Motorola, NEC, NET, Newbridge (Alcatel), Nortel, Radware, Any custom model using Device Creator.
Servers	AMD, Compaq, Dell, HP, IBM, Intel, Sun.
Mainframes	IBM zSeries
Antenna	Antel, Dapa, Ems.
Firewalls	Cisco PIX, Checkpoint.
Dual msfc	Cisco.

	Integració amb productes de tercers : Standard / Modules
Topology	ATM ASCII Files, Cisco WAN Manager, Cisco Works, Device Configuration Files, HP Network Node Manager, Lucent Naviscore SMARTS In-Charge, XML (no additional module req.).
Traffic	ASCII file (no additional module req.), Cisco NetFlow Collector, CFlowd, Fluke Networks OptiView Console, NAI Distributed Sniffer NetScout nGenius, NavisXtend, Spreadsheet (no additional module req.).
Loads	ASCII file (no additional module req.), Concord eHealth-Network, HP Openview Performance Insight, InfoVista, MRTG, XML (with topology import).
Servers	BMC Patrol Perform, Concord System Edge, HP Openview, Performance Agent, NetIQAppManager, MicrosoftPerfmon XML.
Mainframes	IBM RMF Spreadsheet Reporter, AS/MXG, XML.
Satellite	AGI STK.
External interfaces	Cosim (used for HITL), ETS API (External Tool Support), HLA.