Improving Simulation Credibility Through Open Source Simulations

Tom Henderson
University of Washington
The Boeing Company

Simutools Conference
March, 2008
Talk outline

• A decade of ns-2 network simulations
• *ns* challenges and directions
  – ns-3 project overview
  – Dealing with simulation credibility issues
What is *ns* (or *ns-2*)?

- *ns* is a discrete-event network simulator for Internet systems
  - protocol design, prototyping, multiple levels of abstraction
- *ns* has a companion network animator called *nam*
  - hence, has been called the *nsnam* project
Some ns-2 history

- Dedicated project funding on the simulator itself finished in 2000
  - Key institutions: USC ISI, Berkeley, LBNL, ICIR, PARC, and others
ns-2 Impact

ns is a research community resource

<table>
<thead>
<tr>
<th>Simulators</th>
<th>ns-2</th>
<th>OPNET</th>
<th>QualNet/GloMoSim</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport layer and above</td>
<td>123(75%)</td>
<td>30(18%)</td>
<td>11(7%)</td>
</tr>
<tr>
<td>Network layer</td>
<td>186(70%)</td>
<td>48(18%)</td>
<td>31(12%)</td>
</tr>
<tr>
<td>MAC &amp; PHY layers</td>
<td>114(43%)</td>
<td>96(36%)</td>
<td>55(21%)</td>
</tr>
</tbody>
</table>

Source: Search of ACM Digital Library papers citing simulation, 2001-04

• Other statistics:
  – Over 50% of ACM and IEEE network simulation papers from 2000-2004 cite the use of ns-2
    • Source: ACM Digital Library and IEEEExplore searches

  – 10 Simutools 2008 papers/posters related to ns-2
Still in heavy use...

- over 8000 downloads/month (ns-2 plus ns-allinone), active mailing lists

Statistics: SourceForge project site (http://sourceforge.net/projects/nsnam/)
ns-2 contributed code

- where most ns-2 development now occurs

## Existing core ns-2 capability
- ping, vst, telnet, FTP, multicast FTP, HTTP, probabilistic and trace-driven traffic generators, webcache

## Transport layer
- TCP (many variants), UDP, SCTP, XCF, TPRC, RAP, RTP
- Multicast: PGM, SRM, RLM, PLM

## Network layer
- Unicast: IP, MobileIP, generic dist vector and link state, IPinIP, source routing, Nixvector
- Multicast: SRM, generic centralized
- MANET: AODV, DSR, DSDV, TORA, IMEP

## Link layer
- ARP, HDLC, OAF, MPLS, LDP, Diffserv
- Queuing: DropTail, RED, RIO, WFQ, SRR, Semantic Packet Queue, REM, Priority, VQ
- MACs: CSMA, 802.11b, 802.15.4 (WPAN), satellite Aloha

## Physical layer
- TwoWay, Shadowing, OmniAntennas, EnergyModel, Satellite Repeater

## Support
- Random number generators, tracing, monitors, mathematical support, test suite, animation (nam), error models

## ns-2 contributed code
- NSWEB, Video traffic generator, MPEG generator, BonnTraffic, ProtoLib, AgentJ, SIP, NSIS, ns2voip, Agent/Plant
- TCP PEP, SCPS-TP SNACK, TCP Pacing, DCCP, Simulation Cradle, TCP Westwood, SIMUL, TCP-RH, MFTP, OTERS, TCP Effel
- AODV+, AODV-UU, AOMDV, ns-click, ZRP, iS-IS, CDS, Dynamic Linkstate, DYNAMIC, OLSR, ATM, AntNet, Mobile IPv6, IP micro-mobility, MobileIP, GPSR, RSVP, PGM, PLM, SSM, PUMA, Active Networks
- 802.16, 802.11c HCCA, 802.11e EDCA, 802.11e multirate, UWB DCC-MAC, TDMA DAMA, EURANE, UMTS, GPSR, BlueTooth, 802.11 PCF, 802.11 PSM, MPLS, WFQ schedulers, Bandwidth Broker, CSFQ, BLUE
- ET/SNRT/BER based Phy, IR-UWB

## Emulation
- CANU mobility, BonnMotion mobility, SGB Topology Generators, NSG2, simd, ns2measure, ns-2/akapea-2, yavista, tracegraph, hugim, multistate error model, RPI graphing package, jTrana, GEA,
Skepticism abounds, however

“For years, the community had to rely on simulators, which now seem a little dated, and it’s not clear who was convinced to adopt anything new based on ns2 simulations;”

Nick McKeown, VINI public review, ACM Sigcomm 2006
“...Tragedy of the Commons...”
“...around 50% of the papers appeared to be...bogus...”
“Who has ever validated NS2 code?”
“To be honest, I'm still not sure whether I will use a simulation in a paper.”
“...I will have a hard time accepting or advocating the use of NS-2 or any other simulation tool”
Trends

Many researchers move away from simulations

• Experiments and testbeds (real or virtual) start to be preferred in major conference papers
  – PlanetLab, OneLab, VINI, Emulab, ORBIT, WhyNet, ..

Yet simulation tools proliferate

• ns-2, OMNET++, NetSim, NCTUns, QualNet, OPNET, SSFNet, yans, GTNetS, GloMoSim, OSA, JiST/SWANS, cnet, simscript, Traffic, Shunra VE, Extend, INES, J-Sim, HEGONS, Narses, 3LS, NeuroGrid, P2PSim, PeerSim, ONE, ..
Challenges for ns (and simulators)

- Align with how research is now conducted
- Improve credibility

Can ns-3 help with these problems?
What is ns-3?

An open source project building a new network simulator to replace ns-2
Relationship to ns-2

ns-3 is *not* an extension of ns-2

- does not have an OTcl API
  - C++ wrapped by Python
- synthesis of yans, ns-2, GTNetS simulators, and new software
  - example ns-2 models so far: random variables, error models, OLSR
- guts of simulator are completely replaced
- new visualizers are in works
ns-3 people

• NSF PIs:
  – Tom Henderson, Sumit Roy (University of Washington), George Riley (Georgia Tech.), Sally Floyd (ICIR)

• Associated Team: INRIA Sophia Antipolis, Planete group
  – Walid Dabbous, Mathieu Lacage (software lead)

• Developers: Raj Bhattacharjea, Gustavo Carneiro, Craig Dowell, Joseph Kopena, Emmanuelle Laprise
ns-3 priorities

• Aid the serious network researcher
  – Flexible low-level API
  – Software reuse
  – Modularity
  – Scalability
  – Current models

• Ease educational use via higher-level APIs and scripts

• Open source development model and community participation
Challenges for ns (and simulators)

• Align with how research is now conducted

• Improve credibility
Why simulate?

• Field tests are expensive
  – Food, lodging, equipment rental, labor, etc.
• Experiments (especially wireless) can be hard to reproduce
• Collaboration

For these reasons, simulation is vital part of our work.
My simulation requirements

• Reduce time, when possible, developing complicated protocols
  – e.g. Open Shortest Path First (OSPF)
• Need to validate results in field testing
• Interim step often desired: Emulation
• Align with common interfaces; e.g.
  – pcap tracing (tcpdump)
  – ns-2 mobility scenarios
  – topology generators
Example project: IMUNES

- Integrated Multiprotocol Network Emulator/ Simulator
  - Leverages FreeBSD netgraph and lightweight stack emulation
  - [http://www.tel.fer.hr/imunes/](http://www.tel.fer.hr/imunes/)
Example: OSPF-MANET project (Boeing)

Write new code once, run in many environments
• (simulation) Quagga OSPFv3 ported to GTNetS
• (emulation) Quagga running on IMUNES
• (experiments) Quagga running on madwifi
  – http://hipserver.mct.phantomworks.org/ietf/ospf
My ns-3 priorities

• Integration with testbeds and virtual machines
  – emulation modes

• Use of real code, where possible
• Align ns-3 to be more faithful representation of real computers
  – sockets API
  – packets are serialized
  – packet sockets
  – alignment with Linux architecture
  – multiple network interfaces
ns-3 goals for emulation

1) ns-3 interconnects virtual machines

2) testbeds interconnect ns-3 stacks
ns-3 and research priorities

In summary, *make it easier to move from simulation to emulation to experiments*

- Align with popular interfaces
- Support use of real code
- Develop emulation capabilities
Other software improvements

• Better modularity
• (Optional) Python interface
• Flexible tracing framework
• Powerful logging (debugging)
• In-line documentation (Doxygen)
• (Future plan): Distributed simulations
Challenges for ns (and simulators)

• Align with how research is now conducted

• Improve credibility
Background

[1] “Why We STILL Don’t Know How to Simulate Networks”
   – Mostafa Ammar, Georgia Institute of Technology, Annual Simulation Symposium 2005

   – Sally Floyd, WNS2 Workshop Keynote, October 2006
Background (cont.)


   – Muessig, Laack, and Wrobleski, U.S. Naval Air Warfare Center, August 2001
Criteria for Credibility

• Repeatable
• Unbiased
• Realistic Scenarios
• Statistically Sound
• Model Accuracy
• Results Accuracy (Validation)
• Data Accuracy
• Usability

from [3]

from [4]
Repeatability

• Identify simulator, version, operating system, parameters, etc.

• Make code and configuration scripts available to the community
  – Yet, **0 out of 84** ACM Mobihoc MANET simulation papers (2000-2004) referenced publicly available code (from [3])
Repeatability in ns-3

• We will host your code/scripts in a number of possible ways
  – Contribute your code to the ns-3 core
    • src/contrib directory, or main tree
  – Contribute unmaintained code or scripts to our repository
  – Contributed Code page (wiki)
• Simulation output that dumps pertinent configuration data to an output file (planned)
Unbiased

- Initialization bias
- Pseudo-Random Number Generator issues
- Use a variety of scenarios

- Much of this is up to the researcher to get right
  - Note: ns-3 inherits ns-2's combined multiple recursive generator from Pierre L'Ecuyer
Realistic scenarios and conditions

- Multiple scenarios tested
- Simulator defaults are reasonable
- Derived parameters are reasonable
- Appropriate levels of abstraction used
Statistically sound

- Metric collection
- Generating sufficient runs
- Avoid biases (above)
- Data processing

- In ns-3:
  - Flexible means to collect metrics
  - Lean on other projects who have contributed frameworks for this to ns-2
Model accuracy

• “error-free-ness” of software and models

• ns-3 goals here:
  – Support real code where possible
  – Open source models
  – Regardless, we need people or groups to develop and maintain good models
Open source simulations

• “Given enough eyeballs, all bugs are shallow”
  – Eric Raymond, “The Cathedral and the Bazaar”

• ns-3 needs ways to certify models, too
  – capture level of community acceptance
  – publication lists, cross-reference
  – need to identify maintainers, or state the absence of a maintainer
  – validation techniques
Results accuracy

- validation against other simulators
- validation against expert opinion
- validation against (good) test data
ns-3 and validation

- exploit tracing framework to validate events or statistics
- code coverage tests (in regression suites)
- unit tests, valgrind
- calibrate against testbeds
Example: ORBIT collaboration

- Planned use of Rutgers WINLAB ORBIT radio grid to validate ns-3 wifi models
Usability

• not “ease of use” so much as “avoidance of misuse”
  – training and tutorials
  – responsive mailing lists
  – extensive documentation
  – configuration management

• NSF project for ns-3 funds some of these activities
Other activities to improve credibility

• Transport Modeling Research Group (TMRG)
• Discussions on IRTF work to produce a “Simulation Best Practices” document
• Reviewing community raises the bar on paper/thesis acceptance
• (Your ideas wanted!)
Summary

- Learn from good and bad examples of simulation research, to produce credible simulations
- Consider open source (or publishing of models and scripts) to be integral part of your research
- Please give back to the simulators that you use
Closing remarks on ns-3 (March 2008)

ns-3 is in a pre-alpha state
• monthly development releases
• APIs being finalized
• emphasis has been on setting the architecture
• new users should expect rough edges
• many opportunities to work on the core models
ns-3 status (March 2008)

What others are already using ns-3 for:

• wifi-based simulations of OLSR and other MANET routing
• MANET routing (SMF and unicast protocols)
• OntoNet: Scalable Knowledge Based Networking" by Joe Kopena and Boon Thau Loo (UPenn)
ns-3 roadmap (2008)

near term (through June)
• finalize and release simulation core (April/May)
  – core APIs
• ns-3.1 complete release (June timeframe)
  – add Internet and Device models
  – add validation framework
  – some higher-level topology/scenario APIs
ns-3 roadmap (2008)

planned for later this year
• emulation modes
• statistics
• support for real code
• additional ns-2 porting/integration
• distributed simulation
• visualization

We're looking for more early adopters and users
Resources

Web site:
http://www.nsnam.org

Mailing list:
http://mailman.isi.edu/mailman/listinfo/ns-developers

Tutorial:
http://www.nsnam.org/docs/tutorial/tutorial.html

Code server:
http://code.nsnam.org

Wiki:
http://www.nsnam.org/wiki/index.php/Main_Page
Acknowledgments

• This talk builds on lots of ideas/suggestions collected over time, and discussion with ns-3 team

• Tom Henderson is supported by NSF CNS-0551686 (University of Washington)