Network Experiment Programming Interface (NEPI)

Alina Quereilhac  Martin H. Ferrari
Thierry Turletti   Walid Dabbous
Mathieu Lacage
Objective Scenario
Objective Scenario

- Simulated links/networks
- ns-3 simulation models
- Realtime scheduler
- Tap device
- Light weight Virtual Machines
- Linux Network Namespaces
Objective Scenario

Simulated links/networks
ns-3 simulation models ✓
Realtime scheduler ✓
Tap device ✓
Simulated links/networks
ns-3 simulation models ✓
Realtime scheduler ✓
Tap device ✓

Light weight Virtual Machines
Linux Network Namespaces ✓
Easy Deployment

Problem
▶ Tap/VM creation and setup
▶ Coherent IP address assignment across simulation and VMs
▶ Coherent IP forwarding tables across simulation and VMs

Solution
▶ Automate everything

BUT
▶ Need global view of experiment topology
Easy Deployment

Problem

Tap/VM creation and setup
Easy Deployment

Problem

- Tap/VM creation and setup
- Coherent IP address assignment across simulation and VMs

Solution

- Automate everything

BUT

- Need global view of experiment topology
Easy Deployment

Problem

- Tap/VM creation and setup
- Coherent IP address assignment across simulation and VMs
- Coherent IP forwarding tables across simulation and VMs
Easy Deployment

Problem

- Tap/VM creation and setup
- Coherent IP address assignment across simulation and VMs
- Coherent IP forwarding tables across simulation and VMs

Solution

- Automate everything
Easy Deployment

Problem

- Tap/VM creation and setup
- Coherent IP address assignment across simulation and VMs
- Coherent IP forwarding tables across simulation and VMs

Solution

- Automate everything

BUT

- Need global view of experiment topology
Related Work

- Emulab Tcl: ad hoc, hard to generalize
- OMF Ruby: unclear how to extend it to model complex topologies
- OMNeT++ NED: hard to ensure correctness
- SSF DML: hard to ensure correctness
- Geni RSPEC: hard to ensure correctness
- Geni Omnispec: hard to ensure correctness
Emulab Tcl: ad hoc, hard to generalize
Emulab Tcl: ad hoc, hard to generalize
OMF Ruby: unclear how to extend it to model complex topologies
Related Work

Emulab Tcl: ad hoc, hard to generalize
OMF Ruby: unclear how to extend it to model complex topologies
OMNeT++ NED: hard to ensure correctness
Related Work

Emulab Tcl: ad hoc, hard to generalize
OMF Ruby: unclear how to extend it to model complex topologies
OMNeT++ NED: hard to ensure correctness
SSF DML: hard to ensure correctness
Related Work

Emulab Tcl: ad hoc, hard to generalize
OMF Ruby: unclear how to extend it to model complex topologies
OMNeT++ NED: hard to ensure correctness
SSF DML: hard to ensure correctness
Geni RSPEC: hard to ensure correctness
Emulab Tcl: ad hoc, hard to generalize
OMF Ruby: unclear how to extend it to model complex topologies
OMNeT++ NED: hard to ensure correctness
SSF DML: hard to ensure correctness
Geni RSPEC: hard to ensure correctness
Geni Omnispec: hard to ensure correctness
Functional unit / Box

Example:
IP stack
TCP stack
Ethernet card
Functional unit / Box
Attributes

Example:
IP checksum
IP forwarding
NEPI Object Model

Functional unit / Box
Attributes
Trace sources

Example:
Out packets
In packets
Functional unit / Box
Attributes
Trace sources
Connectors

- IP
- dev
- node
- Ethernet
Functional unit / Box
Attributes
Trace sources
Connectors
Allowed connections

![Diagram of NEPI Object Model](image)
Functional unit / Box
Attributes
Trace sources
Connectors
Allowed connections
Hierarchical
Conclusion

Well chosen Object Model

GUI

More details: http://yans.pl.sophia.inria.fr/trac/nepi
Conclusion

Well chosen Object Model

- GUI
- Expressive

More details: http://yans.pl.sophia.inria.fr/trac/nepi

Planète (INRIA)
Conclusion

Well chosen Object Model
  GUI
  Expressive
  Simple coherent API

More details: http://yans.pl.sophia.inria.fr/trac/nepi

Planète (INRIA)
Conclusion

Well chosen Object Model
  GUI
  Expressive
  Simple coherent API
  Connection checking

More details: http://yans.pl.sophia.inria.fr/trac/nepi
Conclusion

Well chosen Object Model
  GUI
  Expressive
  Simple coherent API
  Connection checking
  Automates deployment thanks to global topology knowledge

More details: http://yans.pl.sophia.inria.fr/trac/nepi

Planète (INRIA) NEPI GEC9 10 / 15
Conclusion

Well chosen Object Model

GUI

Expressive

Simple coherent API

Connection checking

Automates deployment thanks to global topology knowledge

More details: http://yans.pl.sophia.inria.fr/trac/nepi
Future work

Support SFA/rspec
Support ...?
Martin H. Ferrari, twice über Intern: implementation, ns-3 backend, NetNs
Alina Quereilhac, Engineer: bugfixing, cleanup, refactoring
Thierry Turletti: First user
Walid Dabbous: Initiated this work
Email: mathieu.lacage@inria.fr
Cannot use DNS:
Some backends do not support it
Hard to deploy
IP endpoints

Cannot use DNS:
  Some backends do not support it
  Hard to deploy

Instead:
  Resolve endpoint names to IP/port before deployment
Currently stupid algorithm:

Allocate consecutive IP addresses

$n$ entries per forwarding table
Currently stupid algorithm:
Allocate consecutive IP addresses
\( n \) entries per forwarding table

Planned:

*Automatic IP Address Assignment on Network Topologies*, by John Byers, Jay Lepreau, Jonathon Duerig and Robert Ricci