Evolving NAM

Mark Handley, ISI North East

VINT Retreat, July 1997

Outline

- **Current Status**
- **Near-term plans**
- **Longer-term plans**
- **Weird and wacky ideas**

Current Status:

Imaging Model

- **Experimented with two tk-based variants of nam**
  - **concluded that tk canvas performance is insufficient even without using Tcl.**
- **Hence nam 1.0 is based on nam 0.9a**
- **Imaging model is xlib based:**
  - **Each frame a pixmap for the entire image is cleared and the whole image is redrawn from scratch. The entire image is then "put" to the x-server.**
  - **For current animations, performance is limited mostly by the size of the image.**
  - **Clearly some optimisations are possible here should they be needed.**

Current status:

Tracefile

- **nam 0.9 used a positional format for the tracefile**
  - **this is too inflexible and inextensible.**
- **nam 1.0 uses an entirely new tracefile format using unix/tcl style parameters.**
  - **allows incremental addition of functionality to nam without obsoleting earlier nam-1.0 tracefiles.**
- **design goal: tracefile should be capable of being parsed either forwards or backward**
  - **important for backing up in very long animations and for use from a pipe.**
  - **means adding extra state to events and adding extra events in some cases (eg: hop event needs a corresponding receive event)**

Current status:

New Tracefile Format

```
+ -t0.0366 -s2 -d7 -pack -e40 -i3 -a10
- -t0.0366 -s2 -d7 -pack -e40 -i3 -a10
h -t0.0366 -s2 -d7 -pack -e40 -i3 -a10
r -t0.0379653 -s4 -d9 -pack -e40 -i2 -a20
+ -t0.0379653 -s9 -d8 -pack -e40 -i2 -a20
- -t0.0379653 -s9 -d8 -pack -e40 -i2 -a20
h -t0.0379653 -s9 -d8 -pack -e40 -i2 -a20
R -t0.038 -s9 -d8 -g24 -miif -p* -T2.0
a -t0.1 -s3 -d8 -n"TCP 21"
f -t0.1 -s3 -a"TCP 21" -ncwin -Tv -v1000
+ -t0.140632 -s7 -d6 -pack -e40 -i23 -a11
- -t0.140632 -s7 -d6 -pack -e40 -i23 -a11
h -t0.140632 -s7 -d6 -pack -e40 -i23 -a11
- -t0.142221 -s8 -d9 -ptcp -e1000 -i17 -a20
h -t0.142221 -s8 -d9 -ptcp -e1000 -i17 -a20
l -t0.145293 -s8 -d9 -SDOWN
n -t0.145293 -s8 -SDOWN
R -t0.145293 -s8 -d9 -g24 -moif -p* -T1.892707 -x
```

Current Status:

Animation Objects

nam 0.9a supported “node”, “link”, “queue” and “packet”.

- **these are fine for packet level animation but insufficient for examination of protocol state.**

nam 1.0 introduces “agent”, “feature” and “route”

Current Status:

Animation Objects

- ***agent***
  - **roughly similar to an agent in ns**
  - **associated with either a node or an interface**
  - **container for "features"**
- ***feature***
  - **a protocol state item (basically a variable!)**
  - **three types at present:**

    variable **(a string like Tcl variables)**

    list **(like Tcl lists, but tells nam to display it differently)**

    timer **(numeric, can count up, down, or be stopped)**
- ***route***
  - **multicast or unicast routing table entry**
  - **associated with an interface**
  - **can have negative caches, (S,G) and (*,G) routes**

Current Status:

Other New Features

- **Nodes and Links can have state:**
  - **“up”, “down”**
  - **only affects their display, not an associated packets, links or agents.**
- **Packets can now be dropped from links**
  - **previously only queue drops were possible**
- **Animations can play backwards**
  - **except drops don’t work properly backwards because there is no “hit the floor” event!**
- **All objects can have “monitors” set on them to continually examine their state.**

Near Term Plans:

Animation Features

- **Currently screen distance is proportional to delay.**
  - **this causes a problem when animating LANs or satelite hops in the same animation as terestrial WANs.**
  - **we need to add a mechanism to selectively break this proportionality when required.**
    - » **hence packets will move at different speeds and get compressed or expanded as appropriate.**
- **Need a way to visualise LANs**
  - **current thoughts are to do this as a virtual node so LANs will tend to look like stars.**
  - **other topologies make it more difficult to indicate who the sender is and add odd false delay implications.**

Near Term Plans:

Animation Features: Interactivity

- **need ways to turn on or off the display of certain agents and routes.**
- **want to be able to interactively control the coloring and labelling of packets.**
- **want to be able to lay out simulation topologies more easily using nam in interactive mode, simulate them in ns, and view them again in nam.**
- **auto-layout would be nice for animations that originate in ns**
  - **ns has enough information to give the delays and bandwidths to nam directly. It doesn't know about directions though.**

Near Term Plans:

Animation Features: Queues

- **nam queues can now go forwards and backwards, but they still model FIFO queues.**
  - **although we can drop other than the tail, but only when playing forwards! Need a queue position in the drop event when playing backwards: TBD**
- **how to model non-FIFO queues?**
  1. **model a class-based or priority queue as muliple queues?**
  2. **have tracefile primitives for queue re-ordering?**

Near Term Plans:

Finding Interesting Events

This can be hard in nam

- **Addition of annotations and being able to provide an annotation index so you can jump to an annotation would help significantly.**
- **Being able to display the state of an object across the animation timeline would also help**
  - **should be possible to graph all sorts of properties from within nam through the entire timeline to help you identify when interesting events occur.**

Longer Term Plans:

Animation Features

Support interactively switching between many different views of the same animation:

- **Time-sequence plots of individual connections**
- **load/loss shown on the network topology instead of packets**
- **zooming into parts of complex animations**
- **Tim Shepards TCP plots**
- **???**

Allow scriptable object animation

- **TCL fragments to define how to display an object (particularly an agent) in various plots.**
- **Eg: display graphically SRM agents' request and response timers.**

Longer Term: not plans

- **Right now I have no plan to move away from the Xlib imaging model for animations.**
  - **Probably use TK canvas for static images such as graphs**
- **No plan to merge with ns**
  - **But it would be good to be able to drive nam from a pipe directly from ns.**

Weird and Wacky?

- **3D visualisation?**
  - **what for?**
- **You tell me!!!**