Dragonflies, Tortoises, Mice and Elephants: The Internet Zoo

Nevil Brownlee
CAIDA, SDSC, UC San Diego and
The University of Auckland
nevil@caida.org

Atlanta, November 2002

Overview

- Traffic Measurement Methodology
- RTFM and NeTraMet
- Definitions: Streams, Flows, Torrents
- Measuring Stream Lifetimes
- Long-Running Streams
- Analysing traffic at Auckland (UA) and San Diego (UCSD)
  - Byte rates for various traffic kinds
  - Stream and Byte % for various stream lifetimes
  - Long-Running stream histories
  - Stream size vs lifetime
- Conclusion

Traffic Measurement Methodology, RTFM

- Traffic Measurement
  - Active, using probe packets. Requires careful design so as not to perturb the network
  - Passive, using trace files. A large-scale data analysis problem
  - Passive, using real-time data reduction
- Realtime Traffic Flow Measurement (RTFM), RFCs 2720-2724
  - Flows are *bi-directional*
  - Flows have *common values for a set of end-point attributes*
  - Meters: gather flow data
  - Meter Readers: collect flow data
  - Managers: configure meters and meter readers
  - Flows of interest (and data reduction) specified by *rulesets,* written in *SRL*

NeTraMet: Streams, Flows and Torrents

- *NeTraMet:* an open-source implementation of RTFM
  - Implements RTFM attributes (RFCs 2722, 2724)
  - Makes a good platform for realtime experimental measurements ...

- Torrent = *sum of all flows on a link*

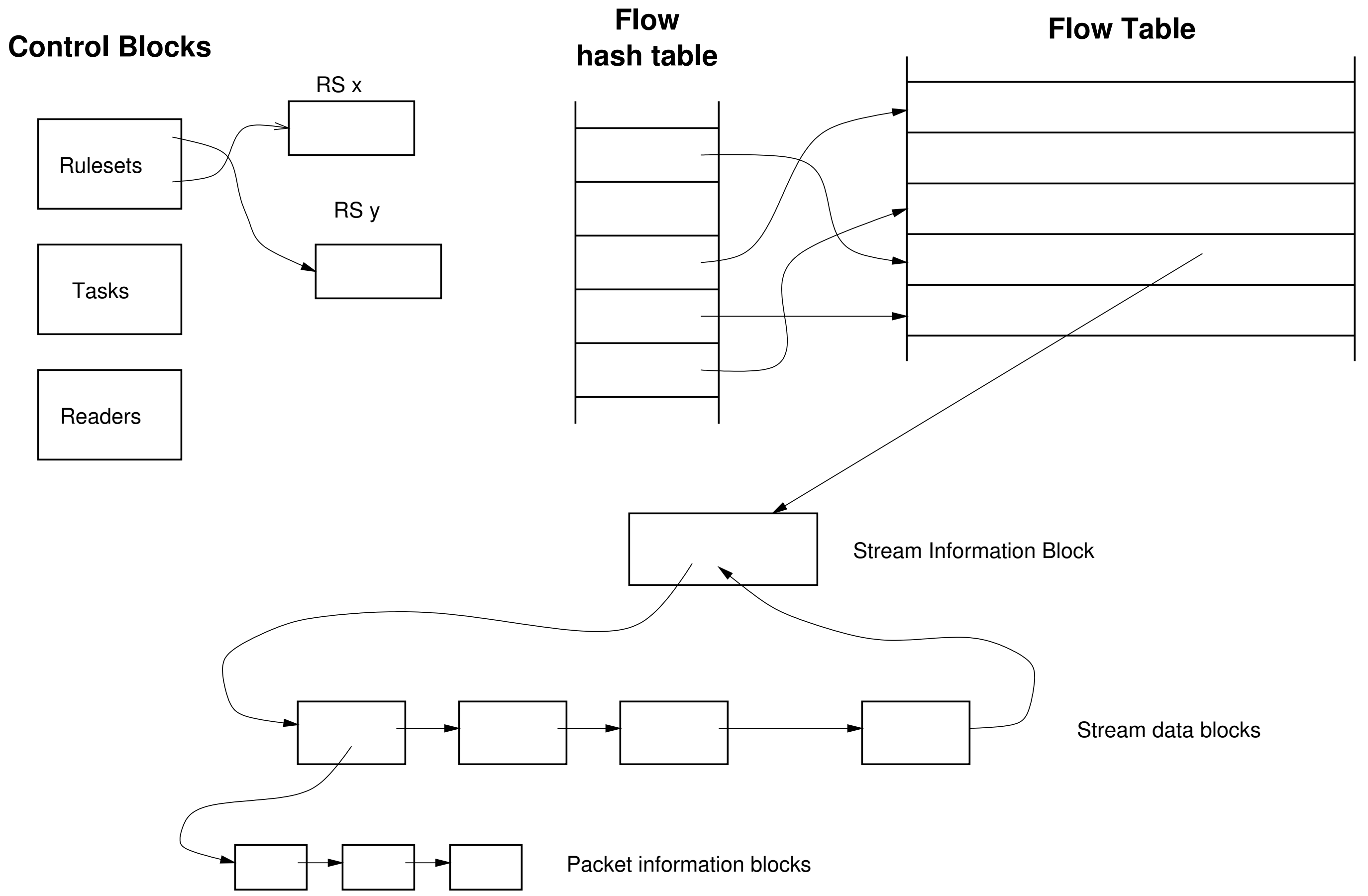
- Stream = *bi-directional IP microflow*
  Streams are individual IP sessions, i.e. pairs of 5-tuples

- Flow was defined above
  A flow can comprise few or many streams

- Mice = streams with *few* packets/bytes

- Elephants = streams with *many* bytes/packets

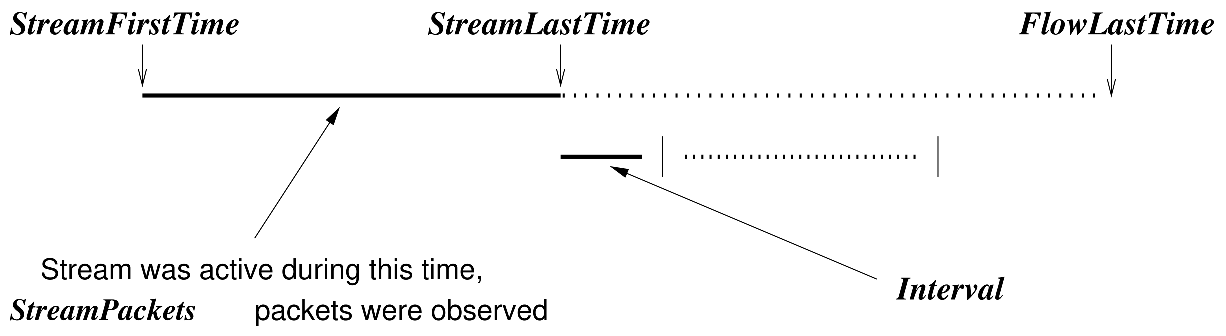
NeTraMet implementation: Data Structures



- Flow Table stores data about flows. Accessed via Flow Hash Table
- Stream data accessed via Stream Info Block (minimises overhead for flows which don't use streams)
  - Stream data blocks chain streams together for each flow
  - A stream may maintain a data queue for packet pair matching

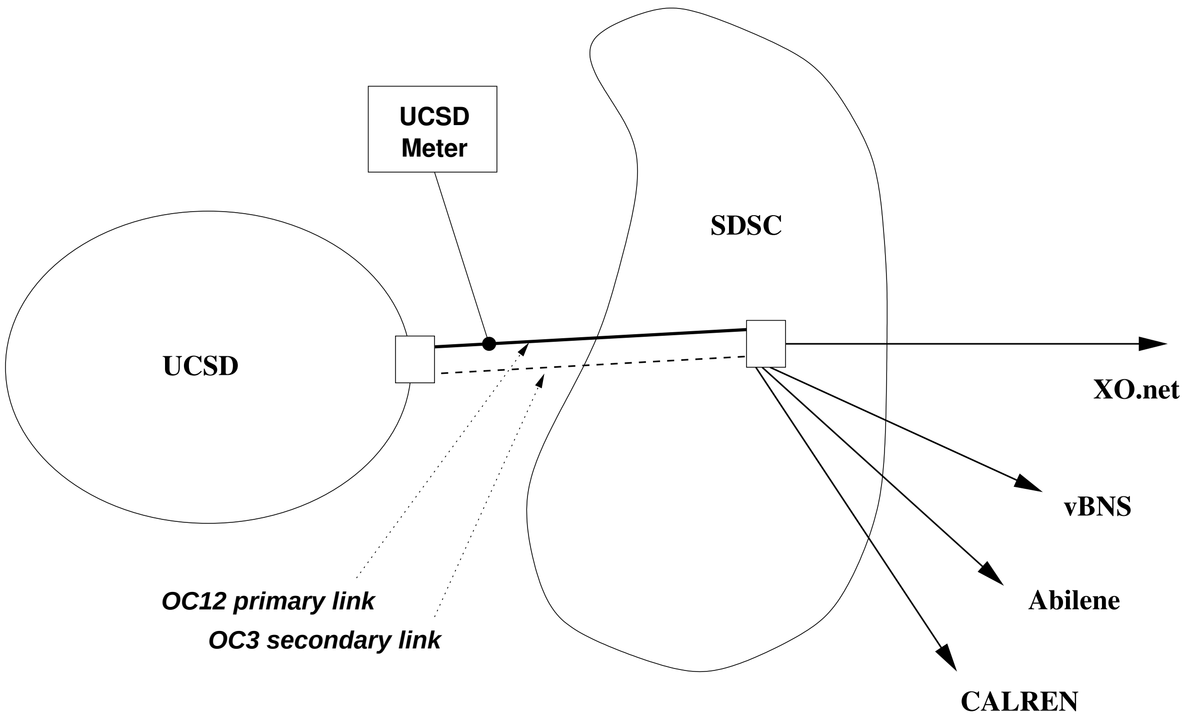
Timeout Algorithms

- For flows, RFC 2720 specifies fixed InactivityTimeout
- Streams are often of short duration
- TCP streams have session state, UDP and ICMP don't
- NeTraMet uses dynamic timeouts for streams



- Stream Timeout Algorithm has two parameters
  - *FixedTime:* stream can't be timed out before this
  - *TimeMultiplier:* specifies minimum space/mark ratio for active streams
  - NeTraMet uses FixedTime = 2s (TCP) or 10s (other protocols), and TimeMultiplier = 10 *(x Interval)*
- Note that *any* dynamic timeout scheme must make assumptions about stream behaviour

Metering setup: DNS Server Performance



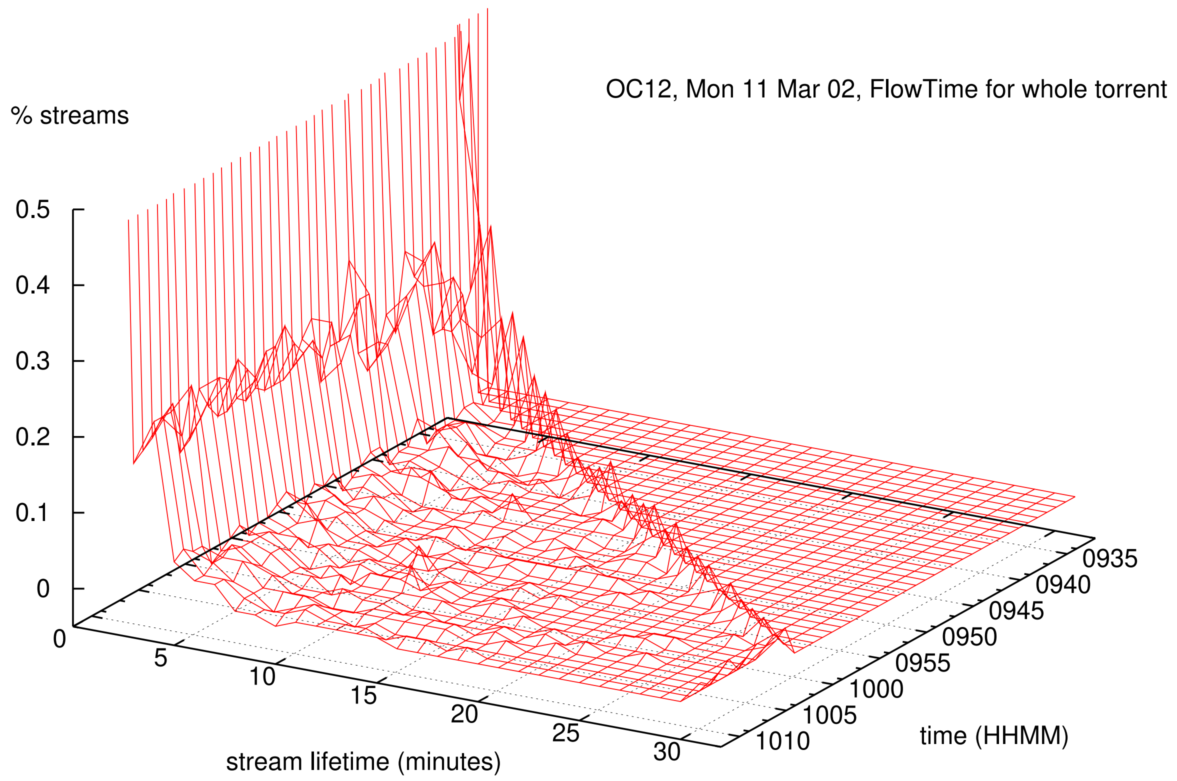
- Observe DNS packets leaving and returning to UCSD
- Use optical splitter to tap OC12 link (stealing 5% of the light)
- Use NeTraMet meter to pick out DNS packets
  - Look for DNS request and corresponding response, difference in timestamps gives DNS resolution time
  - Count packets to get request rate
  - Count requests with no response to get loss rate

RTFM Rulesets, Packet Matching

- A ruleset configures an RTFM meter to gather flow data. It must
  - Specify flows of interest
  - Determine each flow's direction (which endpoint is the source?)
  - Specify any required data reduction, e.g. computed or distribution-valued attributes to be saved
- Example ruleset (in SRL) builds stream Lifetime distributions

```
if SourceInterface == 1 {   # dag0 is the UCSD Tx interface
   if SourceTransType == TCP {
      if DestTransAddress == WWW || SourceTransAddress == WWW
         store FlowKind := 3;
      else store FlowKind := 2;
      }
   else if SourceTransType == UDP
      store FlowKind := 1;
   else {
      store FlowKind := 5;
      count;
      }
   save FlowTime = 30.0.0!0 & 1.6.60!1800;  # 1 .. 30 minutes
      # 30 buckets, no pair matching, linear, us x 10**6
   count;
   }
```

OC12 stream lifetimes, first attempt

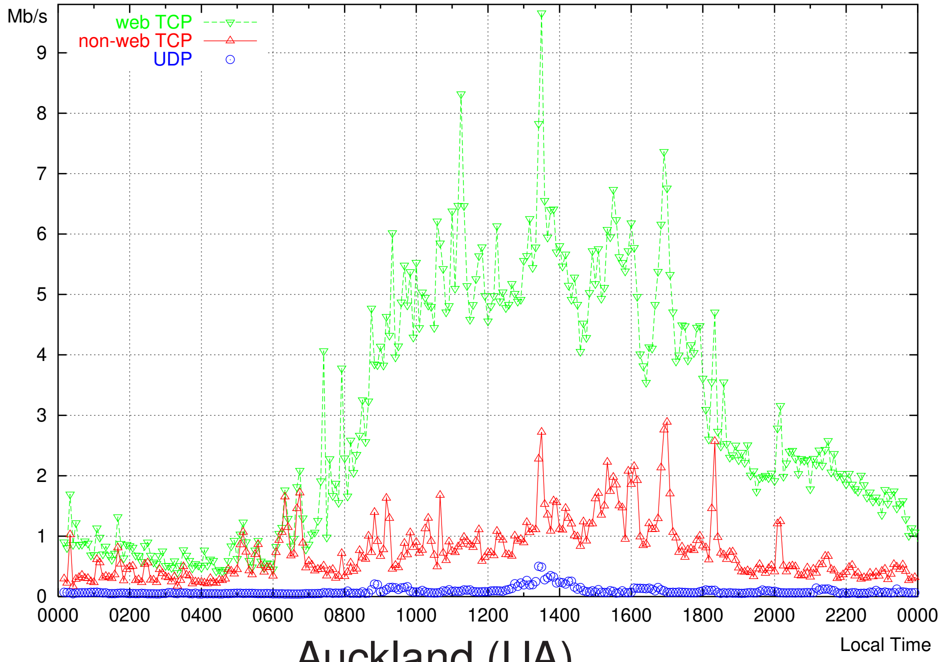


- Plot shows lifetimes for 40 consecutive minutes from ruleset startup
- During first 30 minutes max lifetime increased, reaching 30th bin
- Rightmost edge of plot is *overflow* bin; few streams continue to time out after 30 minutes
- Nearly 97% of all streams on this link lasted only one minute or less
- fewer than 1% of the streams lasted more than five minutes

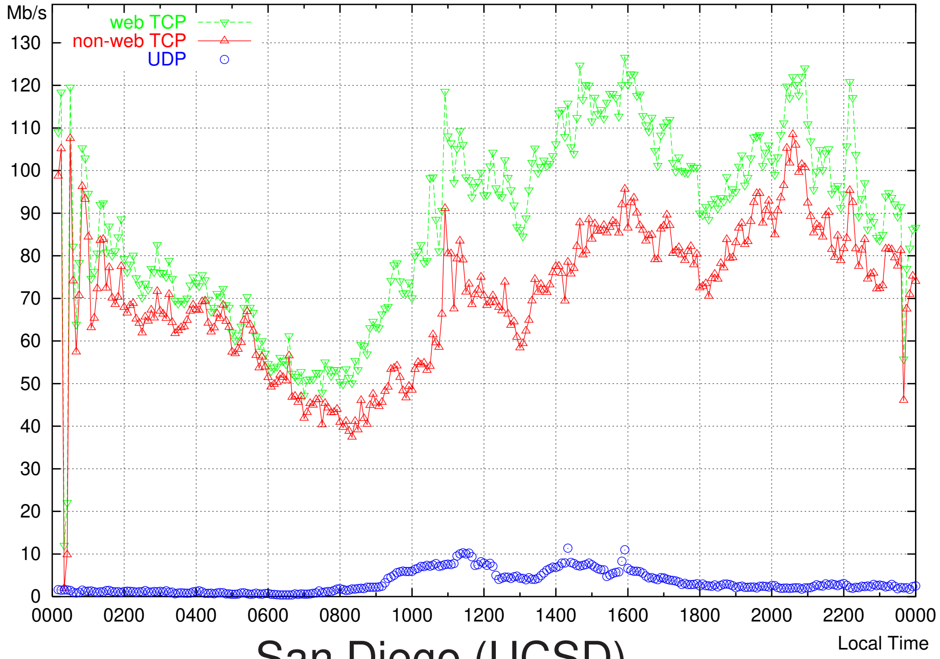
Stream Lifetimes revisited

- NeTraMet (as used above) only updates stream distributions when a stream times out, therefore it couldn't see long-running streams
- NeTraMet modified to observe *long-running* streams
  - Specify a threshold lifetime, LRT seconds
  - Make periodic snapshots of streams running than LRT
  - Now we can observe all the streams in a torrent
- New experiment:
  - Observe streams at UCSD and Auckland, compare their traffic
  - Try various LRT values; 15 minutes works well
  - Examine behaviour of long-running streams
- We recognise *lifetime* as a stream metric, and define
  - *Dragonflies* = streams lasting less than 2s
  - *Tortoises* = streams lasting longer than 15m

Traffic summary plots



Auckland (UA)



San Diego (UCSD)

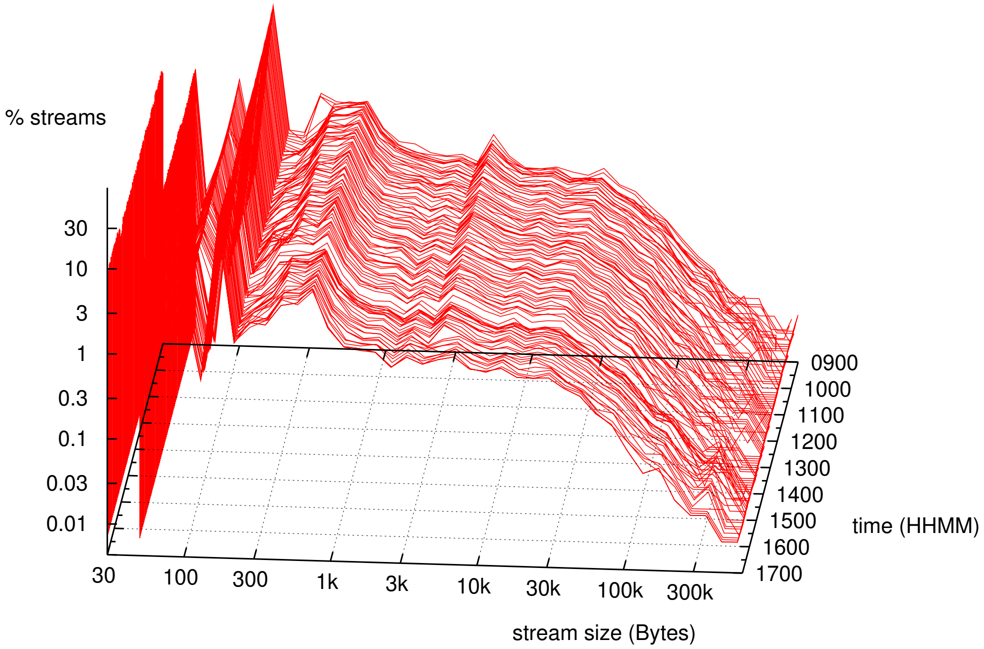
- Cumulative rate (kb/s) for various traffic kinds vs time of day (HHMM) at Auckland (UA) and San Diego (UCSD) for 24 hours from midnight local time on Wed, 12 Jun 02, median values for 5m intervals
- Both sites have about the same amount of web traffic
- UCSD has much more UDP and non-web TCP than Auckland
- Auckland charges its users for each MB sent or received

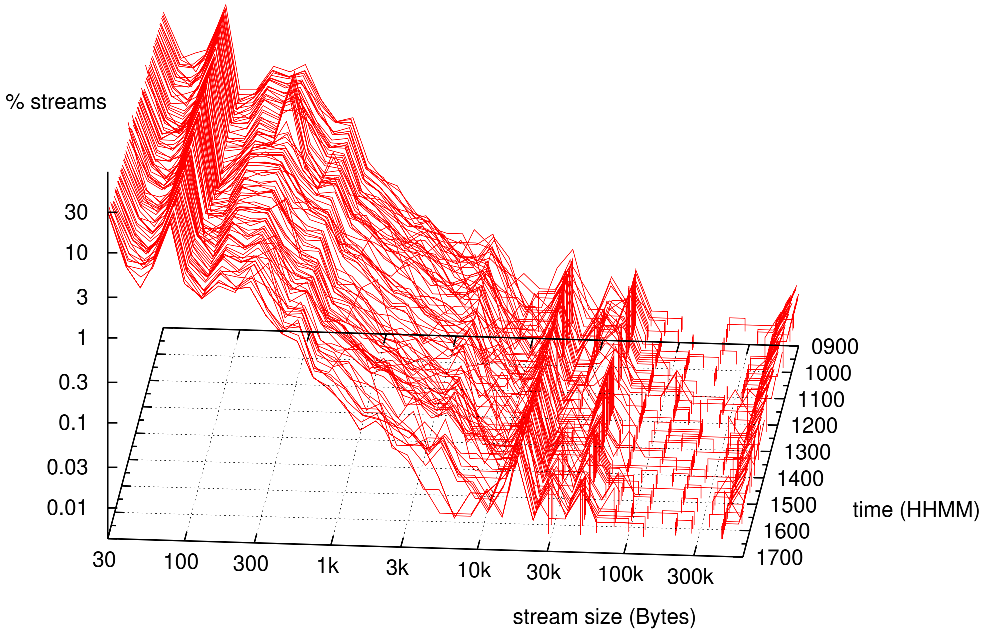
Stream Size plots

Size distributions at 5m intervals

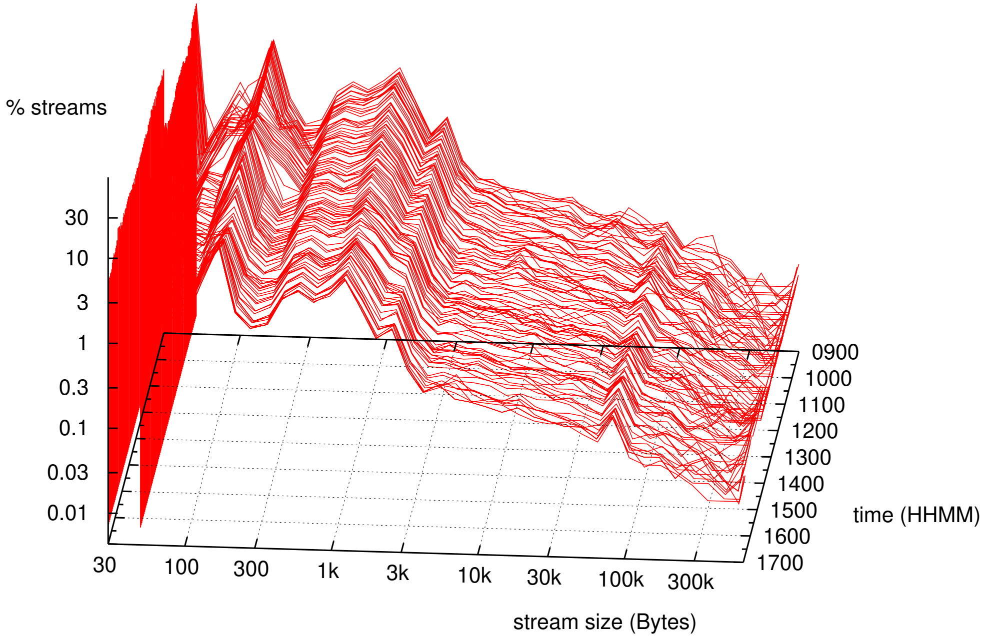
UCSD 9 a.m. to 5 p.m. local time

Wednesday, 12 June 2002



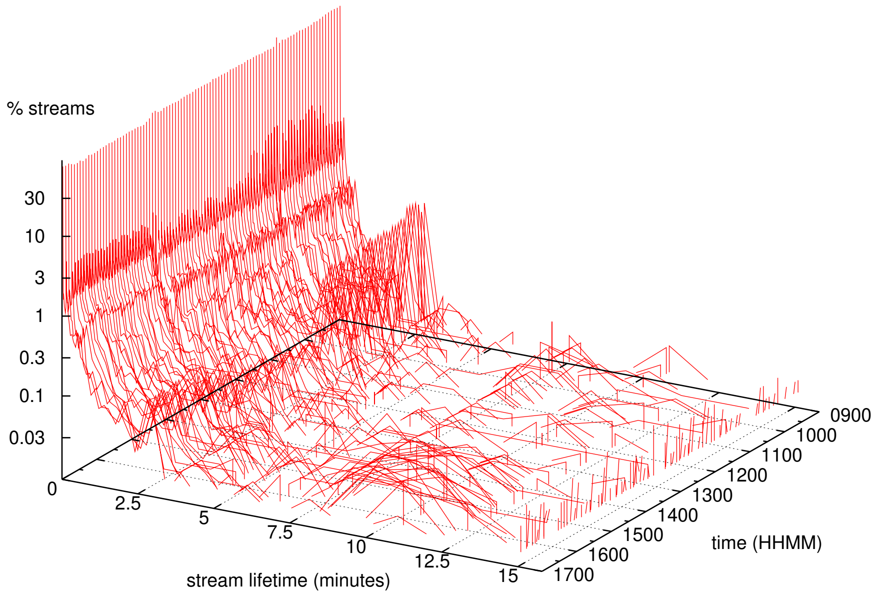


UDP streams

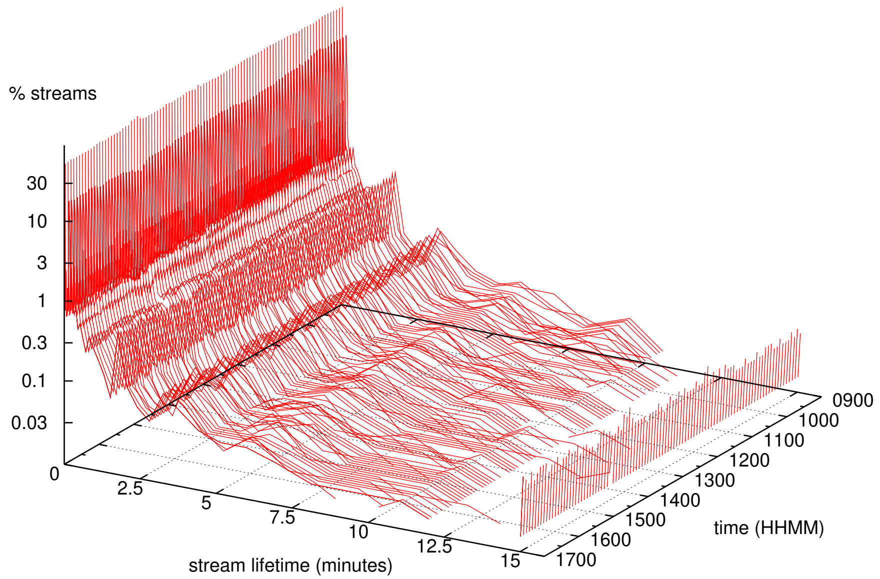


Web/Non-web TCP streams

Stream lifetime plots



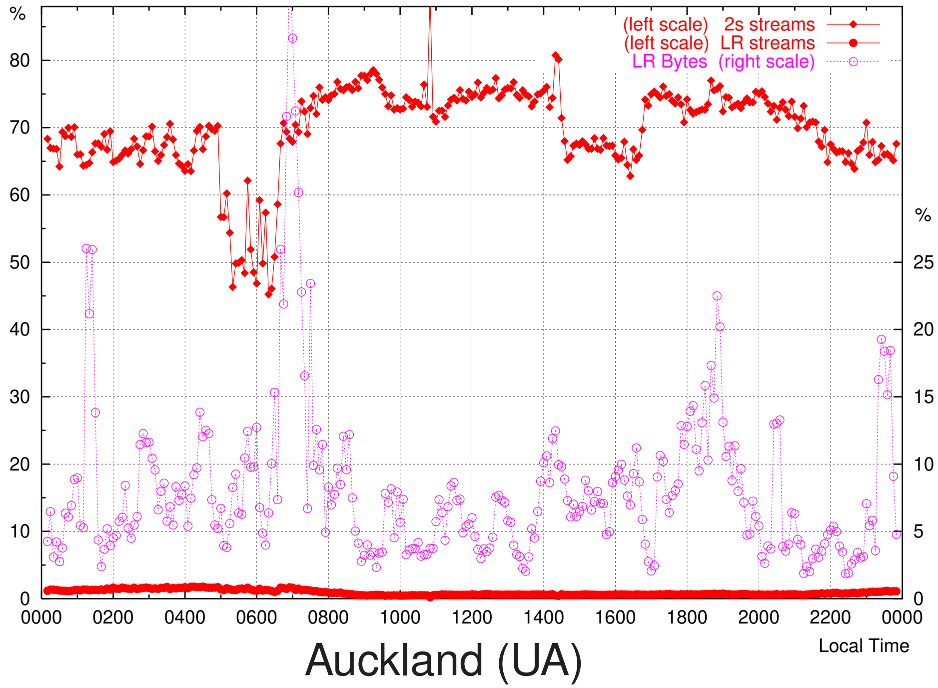
Auckland (UA)

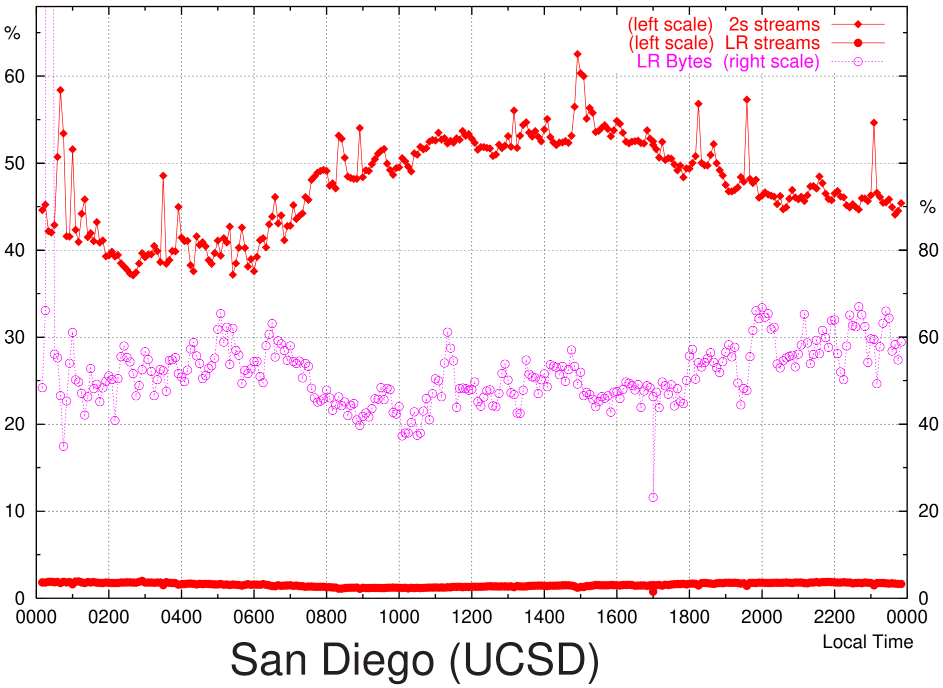


San Diego (UCSD)

- Percent streams vs lifetime (minutes) for eight hours, 9 a.m. to 5 p.m. local time on Wed, 12 Jun 02
- Note that although the link data rates differ the shape of the distribution is similar for both sites, and to our earlier lifetime plot (above)
- This plot is taken from the middle of three days' trace, hence it doesn't show the 'startup' behaviour we saw above

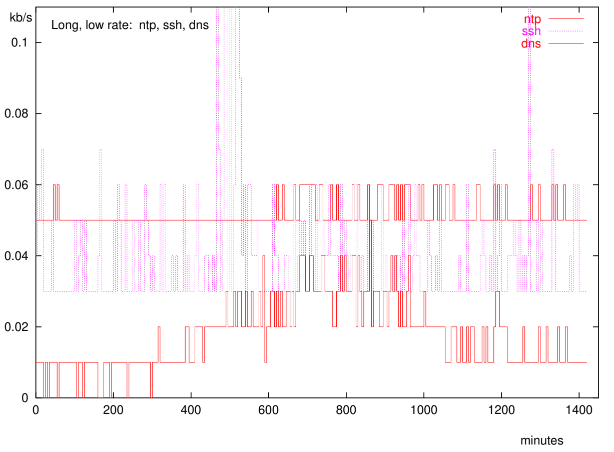
Stream and Byte percentage plots



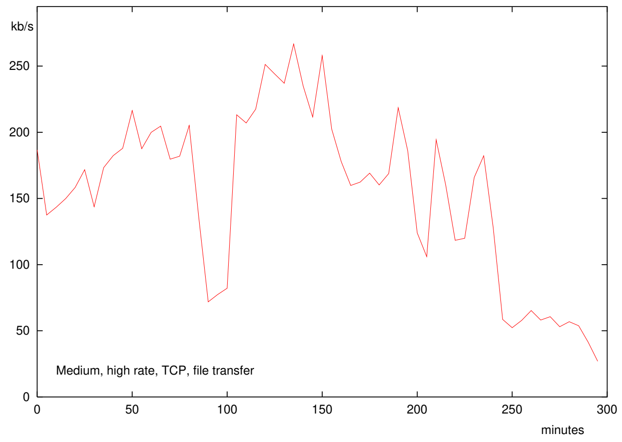


- Stream and Byte % vs time, median values for 5m intervals vs time of day (HHMM) at Auckland (UA) and San Diego (UCSD)
  - % Short and Long-Running streams (red symbols, left-hand axes)
  - % Bytes in LR streams (magenta symbols, right-hand axes)
- Note that:
  - 40% to 70% of streams are *dragonflies* (red diamonds), lasting 2 seconds or less.
  - About 1.5% of streams are *tortoises* (red dots), i.e. LR streams lasting more than 15 minutes. However, they contribute 5% to 50% of a torrent's bytes (magenta circles).

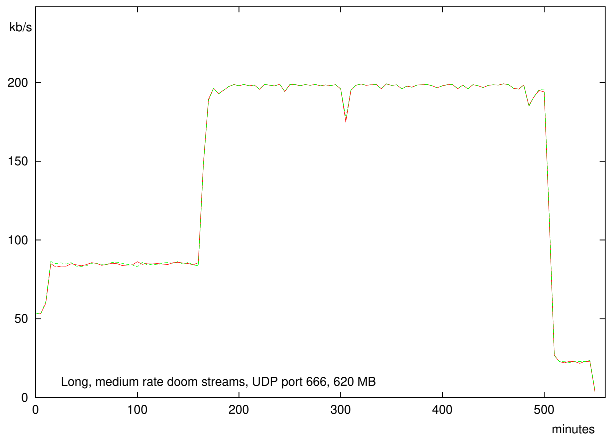
Long-Running (tortoise) stream behaviour



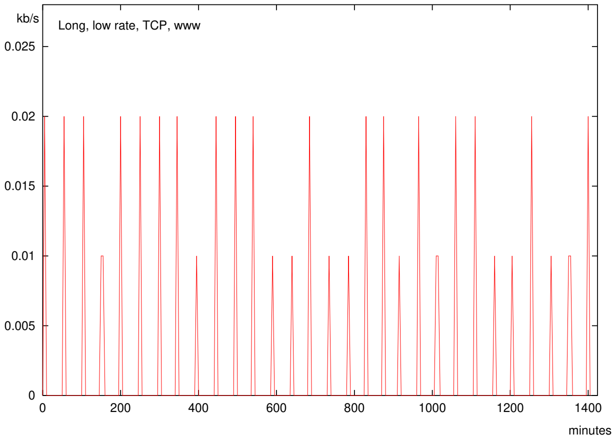
(a) UCSD, long, low rate: NTP, SSH, DNS



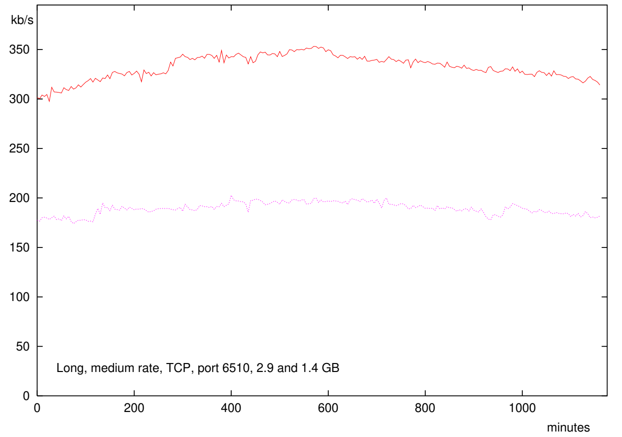
(c) UA, medium, high rate: TCP file transfer



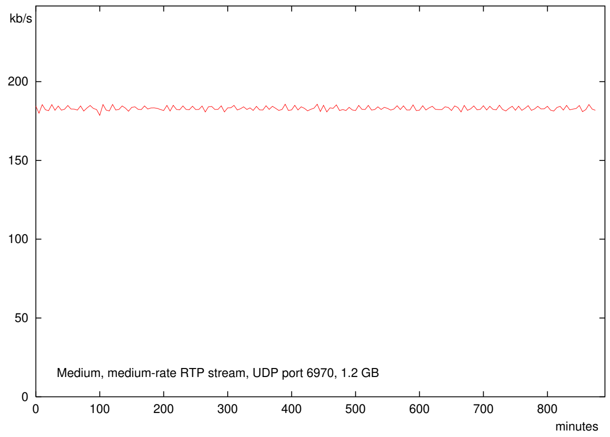
(e) UCSD, long, medium rate: doom streams, UDP



(b) UA, long, low rate TCP, ‘refreshed’ web page



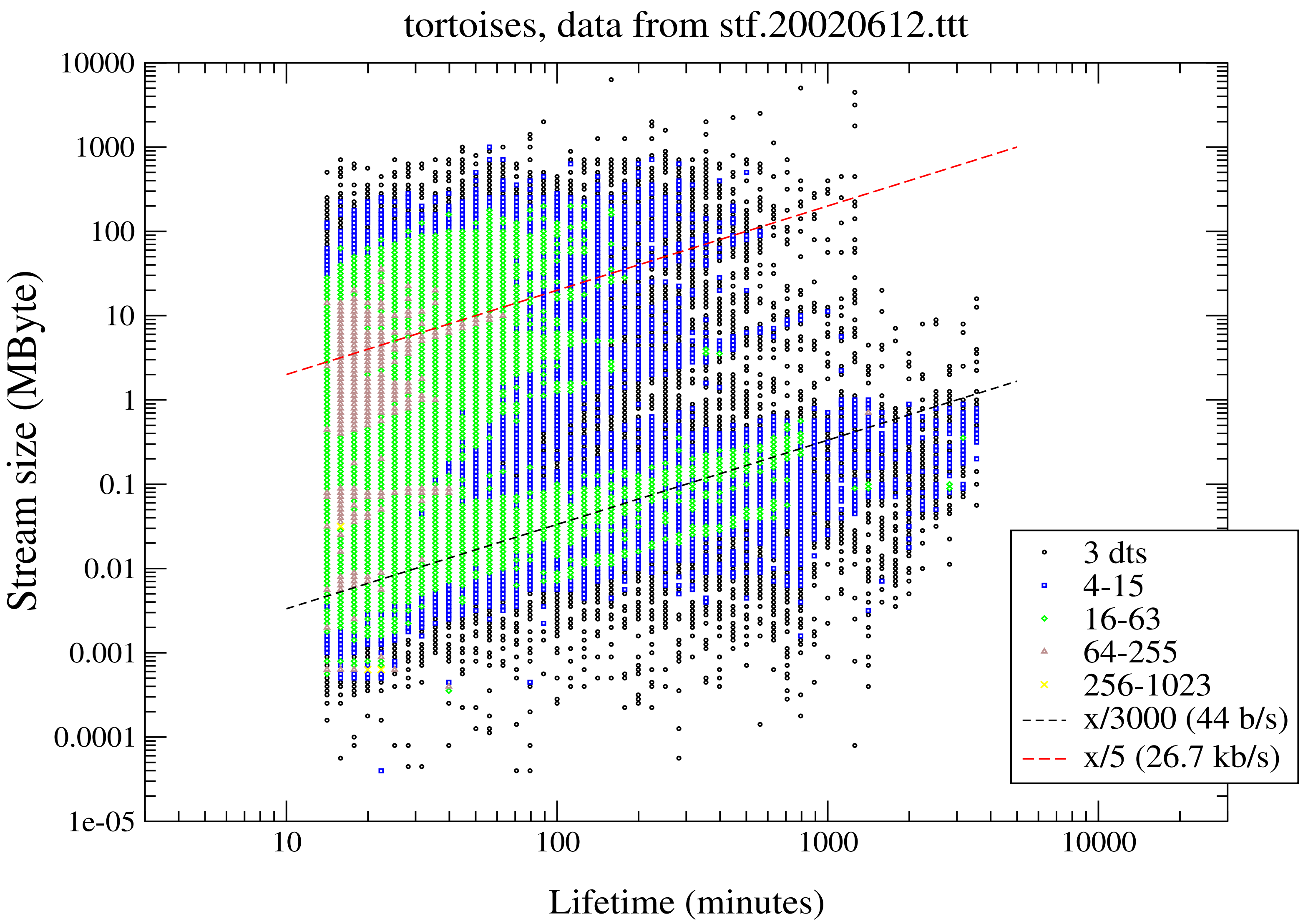
(d) UCSD, long, medium rate: port 6510, TCP



(f) UCSD, medium, medium-rate: RTP stream, UDP

- Bit rate (median kb/s for 5m intervals) vs elapsed time (minutes)

Stream Size vs Lifetime



- Contour plot, data from UCSD OC12 link, Wednesday, 12 June 02
  - Maybe there are some preferred values (data rates)
  - Size and lifetime don't seem strongly correlated

Conclusion

- NeTraMet is a good platform for stream-based measurements, since it produces size and lifetime distributions in near real time
- Stream sizes are increasing, reflecting the increase in network and computer system performance. Web objects up to 50 kB are now common
- Non-web TCP is increasing rapidly (UCSD vs UA), changing the overall traffic mix
- Stream lifetime distributions are similar at UA and UCSD
  - At least 45% of streams are dragonflies $< 2s$
  - 98% of streams last $<= 15m$
  - The remaining 2% are tortoises, and can carry 50% to 60% of the bytes on a link
- Stream size (mice..elephant) and lifetime (dragonfly..tortoise) don't seem strongly correlated

Further Information ..

- CAIDA, papers by author
  `http://www.caida.org/outreach/papers/byauthor/index.xml`
- Nevil's home page
  `http:/www.caida.org/˜nevil`
- NeTraMet distribution site
  `http://www.auckland.ac.nz/net/NeTraMet`

- Acknowledgements
  - Thanks to Andre Broido for his help in producing the contour plot