The NLANR Network Analysis Infrastructure

A.J. McGregor[a,b], H-W Braun[a], and J.A. Brown[a]
[a]National Laboratory for Applied Network Research (NLANR),
San Diego Supercomputer Center,
10100 Hopkins Drive, San Diego, CA 92186-0505, USA

[b]The University of Waikato,
Private Bag 3107, Hamilton, New Zealand

Keywords: Internet measurement, ATM, TCP/IP

1. INTRODUCTION

The National Laboratory for Applied Network Research (NLANR) is a distributed research and support organisation focused on the high performance connection (HPC) community in the United States. This community is served by two National Science Foundation (NSF) approved high performance research networks. Currently these are the vBNS[1] and Abilene[2] networks.

The Measurement and Operations Analysis Team within NLANR is developing a Network Analysis Infrastructure (NAI). It is intended that this infrastructure will provide both engineering and research support for the HPC community. Specifically the goal of the NAI project is to create an infrastructure that will support measurements and analysis through the collection and publication of raw data, visualisation and analysis of network measurement. Currently the main focus is on:

- passive collection of header traces
- active measurement
- SNMP derived data
- BGP router based data
- presenting the results of analysis to the HPC community

To these ends, there are two well established projects and a number projects that are in the early stages of development. The remainder of this paper reports on these projects and is structured as follows: section 2 gives an overview of the network analysis infrastructure, which supports the other projects and is also made available to support other researchers. This section includes a description of the Cichlid visualisation engine which is used to visualise the data collected in other parts of the project. Section 3 describes NLANR's passive measurement project, OCXmon, this is followed, in section 4, by a description of the Active Measurement Program, AMP. The following section describes some of the other measurement projects that are being developed by NLANR including those based on BGP and SNMP. The paper concludes with a summary of the NAI projects and brief discussion of some of the most important future directions for the project.

2. NAI

There are three methods of collecting data on network behaviour (see figure 1). These are:

- **passive measurement**, where a probe that records network activity is inserted into the network. Most commonly the probe is attached to a link between network nodes and summarises and records information about the traffic flowing on that link.

E-mail: {tonym,hwb,jabrown}@nlanr.net

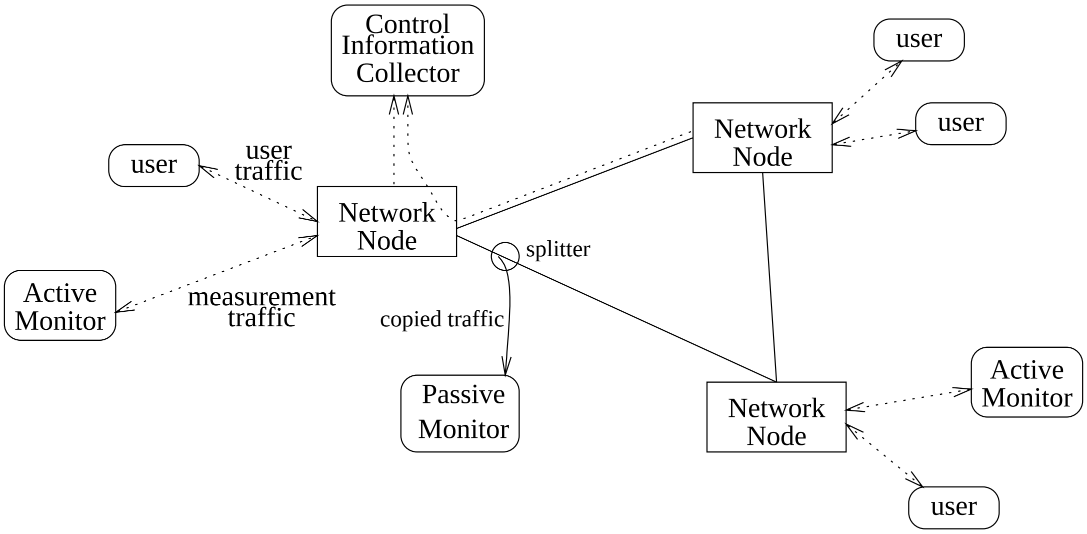

Figure 1. Monitor Types

- **active measurement**, where the behaviour of the network is studied by sending data through the network and observing the results, including the time taken to send the data.
- **control monitoring**, where network control information, such as routing or network management information, is captured and analysed.

The three approaches have different focuses and it is often the case that more than one approach is required to develop an understanding of some aspect of a network's behaviour. Because passive measurement only observes the behaviour of a network at a point without adding to or modify the data carried by the network it has no impact the behaviour of the network. A very detailed understanding of the behaviour at the point of measurement can be developed but it is difficult to gain an understanding of the network as a whole, or the end-to-end behaviour of the network. Passive measurement is often used to measure traffic characteristics, such as a breakdown of traffic by type or destination.

Because active measurement involves sending real traffic into the network it lends itself to measuring parameters that reflect the service the network is offering to its users, including parameters like round trip time and packet loss. However the traffic inserted into the network may alter the behaviour of the network. This is particularly true for those parameters, such as available throughput, that are difficult to measure without sending significant amounts of traffic.

Monitoring the control information of the network provides a ready source of information about those aspects of the networks behaviour which are described by data transfered as part of normal network operation. Parameters like link utilisation or route stability may be collected this way. Assuming that the researcher can gain access to these flows (which can sometimes be administrately difficult) and that they include the information of interest, it may still be difficult to verify the accuracy of the information because its collection and transfer is outside the control of the researcher.

The NLANR NAI infrastructure supports all three types of network monitoring as shown in figure 2. The colours in the diagram indicate the sensitivity of the data, red being the most sensitive and green indicating data that is made publicly available via the WWW. Newly collected data is stored on the machine `nai.nlanr.net` (160 GB, 256MB memory, dual 450MHz PII, FreeBSD) where local analysis and encoding of the data occurs. Encoded data is copied to `moat.nlanr.net` (160 GB, 256MB memory, dual 450MHz PII, FreeBSD) for WWW publication. Researchers who need access to large quantities of data may be granted use of one or more of the compute engines (18GB, 256MB memory, 450MHz PII, FreeBSD or Linux) in the infrastructure.

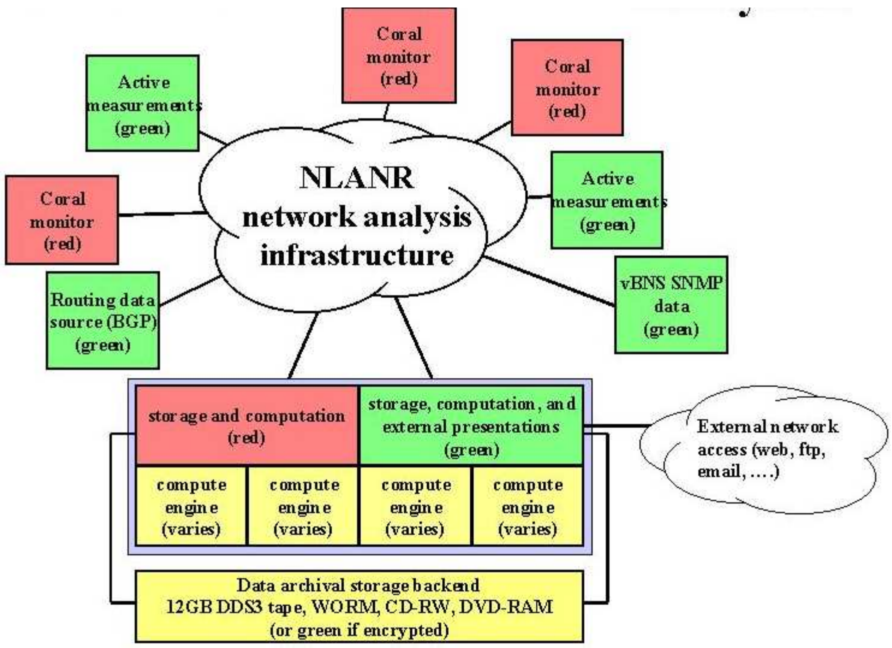

Figure 2. NAI Architecture

2.1. Cichlid

It is easy to collect large amounts of data from high speed networks. Even once some analysis has been done there are often large numbers of metrics and graphs. In the AMP project, described in section 4 below, for example, measurements are taken between more than 7100 pairs of monitors. It is not possible for a human to examine the graphs for all paths regularly. Instead some form of aggregation, that highlights abnormalities, is needed. Visualisation tools that portray network information in a graphical form can fill this role.

The Cichlid tool (named after the species of fish with the same name, some of which change colour) was developed at NLANR to allow visualisation of a wide range of network data. Cichlid is a distributed, animated, display tool for bar charts and 'vertex/edge' graphs.

Cichlid is implemented as one or more data servers which provide raw data through TCP connections to a client which renders the data into a visualisation. By splitting the program this way the data server(s) may be on one or more remote systems, possibly the systems where the data is collected or stored. The client, on the other hand, should be run on the human interface system with the display.

Figure 3 shows some example snapshots of Cichlid displays. Still images do not convey the full utility of the tool. The reader is challenged to imagine the animation of these displays. Alternatively preprepared animations can be found at the NLANR/MOAT web site.[3]

The display in figure 3 with the smoothed surface uses similar data to the adjacent bar chart but Cichlid has rendered the surface using NURBs. We believe this rendering aids quick assimilation of the nature of network performance by allowing well understood analogies (for example a 'network terrain') and by removing many of the extraneous features present in the bar chart.

Among other applications, Cichlid servers have been developed for generating AS and traffic matrices from OCXmon data and for visualising RTT and loss for AMP measurements. Cichlid is implemented in C, using Mesa, a free OpenGL implementation. Cichlid will compile under Windows95 and various Unix platforms. It is available from the NLANR web site under a standard University of California free software licence.

3. OCXMON

The NLANR passive measurement project utilizes OCXmon and FDDI monitors. At the time of writing, 11 OC3/ATM monitors, two OC12/ATM monitors, and one FDDI monitor have been deployed, predominantly at

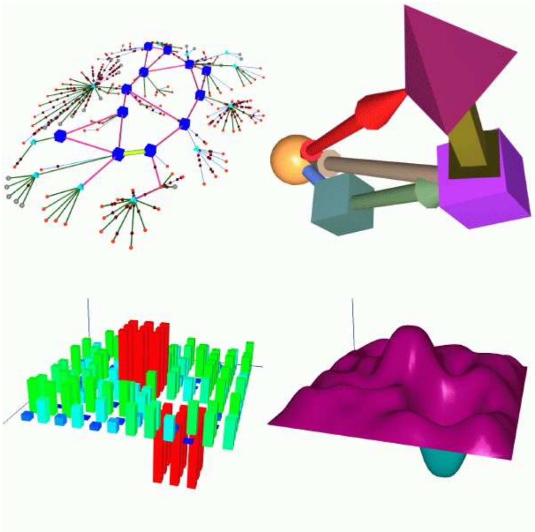

Figure 3. Example Cichlid Displays

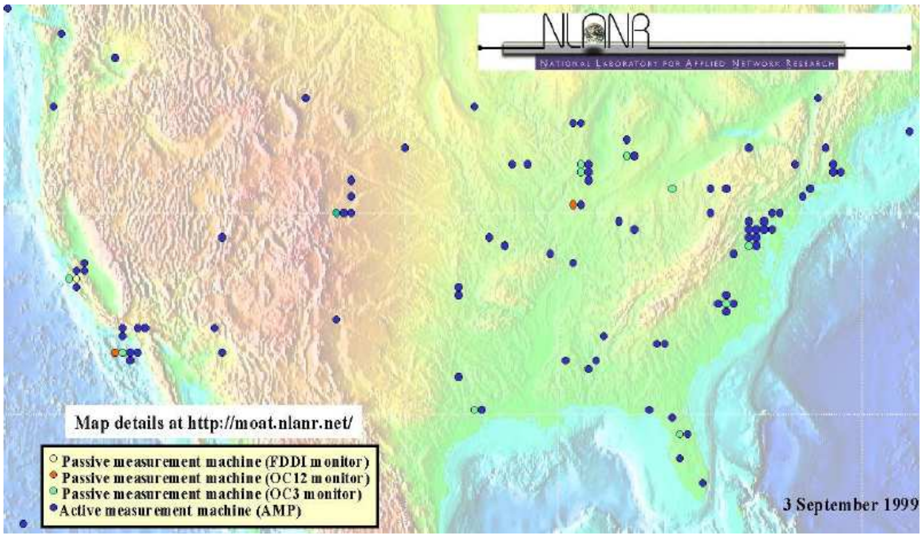

Figure 4. Monitor Sites

HPC sites. The deployment sites are indicated on the map shown in figure 4 by a yellow, red or green dot, depending on the type of interface. Deployment of approximately another 25 monitors, several in collaboration with the Internet2's Abilene network, is currently being planned.

An OCXmon monitor is a rack mountable PC running the FreeBSD or Linux operating system. In addition to the PC components (400MHz PII, 128MB RAM, 6GB SCSI disk) two measurement cards are installed in the PC and an optical splitter is used to connect the monitor to an OC3 (155mbps) or OC12 (622mbps) optical link. DS3, FDDI and electrical interfaces are also available or being experimented with. There are five types of measurement card available for OCXmon. These are:

- Fore ATM OC3 cards
- Applied Telecom ATM OC12

- DAG ATM/POS OC3/12/DS3 cards (developed by the University of Waikato WAND group[4]).
- Any FreeBSD supported FDDI card (we have used DEC DEFPA cards)
- Any 10/100 Ethernet card supported by FreeBSD.

OCXmon machines have two measurement cards installed in each monitor so that they can capture traffic in both directions of a full duplex connection. The total cost for an OC3mon based on the Fore measurement cards is under US$5000. Further details of the hardware are available at the project web page.[3] An Applied Telecom based OC12mon costs in the order of US$18,000. DAG cards are not currently commercially available but it is expected that an OC3/12 monitor based on DAG cards will cost around US$9,000.

Control software (originally developed at MCI by Joel Apisdorf in the case of the Fore and Applied Telecom cards) is downloaded to the measurement card to control the capturing of traffic. Using different versions of this software the ATM based monitors can be configured to capture different parts of the traffic they see. When an IP packet is sent over an ATM connection it is broken into 48 byte pieces to be sent in cells. Assuming the most common approach to breaking the packet into cells is being used (AAL5) the last cell of the packet is marked to indicate the packet is complete. The next cell (on the same virtual circuit) will be the first cell of the next packet. If there are no optional headers in use the 48 bytes of data in the first cell include the IP and TCP or UDP header but very little, if any, user data.

Traces of just the first cell from each packet are captured from each monitor several times per day. After sanitisation to protect the identity of the users, by means of encoding the IP addresses, the data is published via the WWW.

Currently, every three hours 90 seconds worth of header trace data is collected at each monitor. The start time within the hour is pseudo-random to avoid the risk of distortion created by regular network events. All monitors start at the same pseudo-random time so that events that are detected at different monitors may be synchronised.

Following the trace collection several network analysis routines are run, and the results posted in the data cube.[5]

A day's traces from a single site vary in size from a few tens of megabytes to over a gigabyte depending on the speed of the link being monitored and level of traffic. The disk space available in the NAI system allows about a months traces to be kept on line. Older traces are retired to tape but can be made available if they are needed for specific research needs. It is interesting to note that, so far, there have been few requests for historical data while there are many hits per day on the current trace files. We surmise that this is indicative of the state of networking research. The high rate of change of the Internet means that there are more important questions about the behaviour of today's networks than network researchers can answer. Consequently there is little research capacity available to investigate trends in historical data.

The OC3mon trace file format is shown in figure 5. Most of the fields in the cell entry are direct copies of the data in the captured cell. The first three fields are exceptions. The first two fields together record the time at which the cell was captured. Because the time stamp is added by the measurement card it is close to the wire time, relative to the time the card was last reset. Shortly after the measurement is started both cards are reset to synchronise the timestamps between the cards. Because most cards have separate clocks there may be a small amount of drift between the clocks over the period of the measurement. The DAG cards, which use a somewhat different file format, have a facility that allows a synchronising signal between the cards if very accurate synchronisation is required between the cards. The other additional header field is used to check for overload conditions where the monitor is unable to keep up with the workload.

After the measurement period some analysis programs are run on the captured files and the file and the analysis results are transfered to the central NAI machines. The sanitisation and publication of the traces is also done at the central machines.

The analysis that is done on the traces includes:

- bidirectional transaction analysis

  Statistics are generated on transactions where a transaction is a bidirectional sequence of packets with the same IP source, IP destination, IP protocol and port numbers (if appropriate). Transactions terminate when no matching packets are sent for 8 seconds.

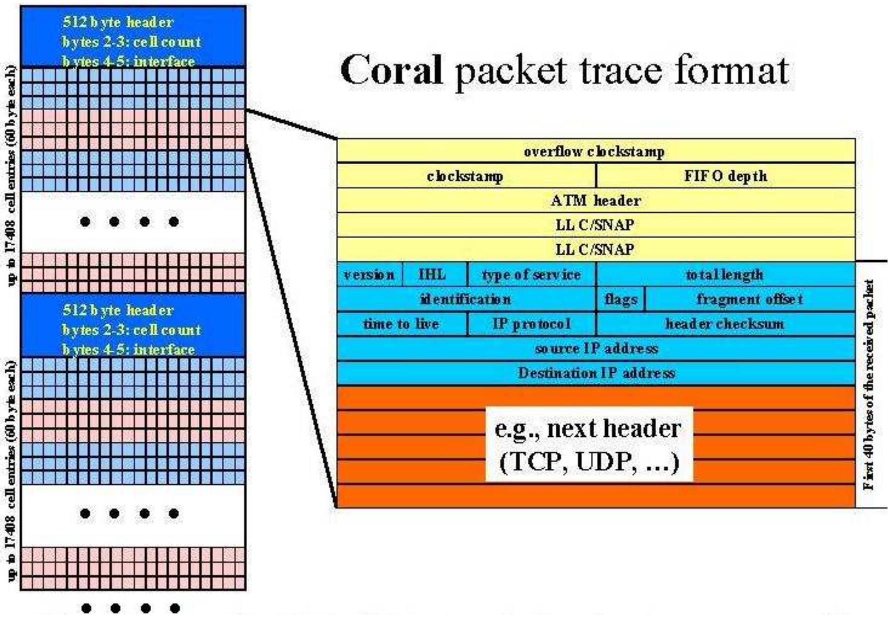

Figure 5. Trace Format

The results include statistics on individual transactions, aggregated across all transactions and a summary of the transactions with the highest throughput.

- flow analysis

  Similar statistics are also generated on flows where a flow is one (unidirectional) half of a transaction.

More details and example output from the analysis can be found on the WWW.[5]

Other analysis programs are available through CAIDA's[6] CoralReef software. These include flows analysis, packet size and frequency histograms, packet size run lengths, protocol and port break down, host and autonomous system matrices, type of service breakdown and ASCII dumping of packets.

4. AMP

AMP is NLANR's active measurement project. The focus is on making site to site measurements of round trip time (RTT), packet loss, topology and throughput across the National Science Foundation (NSF) approved HPC networks. At the time of writing around 85 monitors are deployed at NSF HPC awardee sites. This number is currently increasing by about 10 per month. The location of the monitors is shown by the blue dots in the site map (figure 4).

Each of these monitors sends a single ICMP packet to each of the others every minute and records the time to (or absence of) the reply. In addition, every 10 minutes the route to each other monitor is recorded using traceroute. Throughput tests can also be run between any pair of the monitors using a web based throughput test request. (Throughput tests are only run on demand because of the high cost, in terms of network traffic, of running these tests.) The following throughput tests are available:

- Bulk TCP data transfer
- Bulk UDP data transfer
- ping -F
- treno

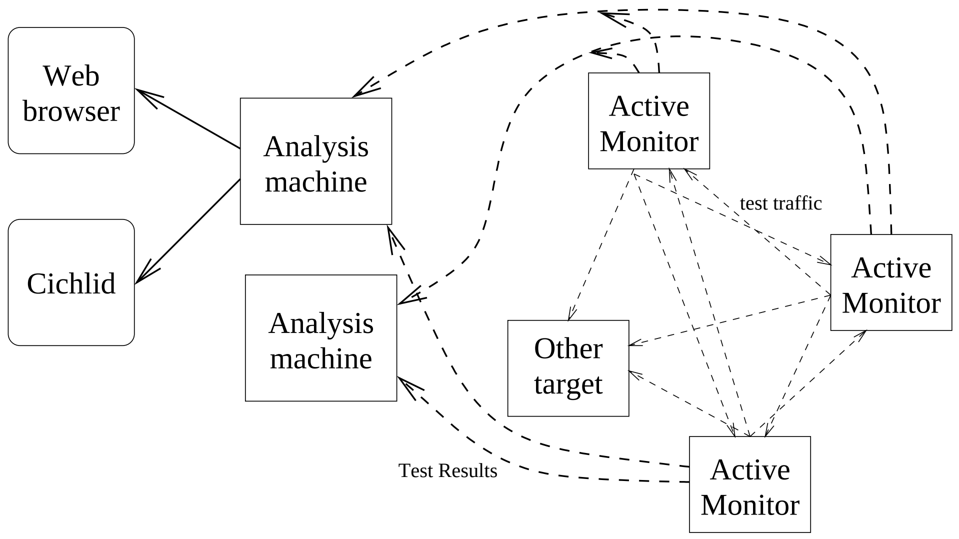

Figure 6. Amp Architecture

The data collected is sent to the central AMP site, at the San Diego Supercomputer Center, for processing and publication via the WWW. To improve robustness, two central machines are used. The data is sent to each central machine independently by each monitor. This arrangement is shown in figure 6. Currently one of the central machines is configured as the primary web site (`amp.nlanr.net`) and the other as its hot standby (`volt.nlanr.net`). If there is a failure the identity of these two machines can be swapped to allow the web pages and data analysis to be continued. In the near future we plan to share the web serving and analysis workload between the two central machines under a common DNS name.

Because AMP measurements are continuous there is no natural end time at which to send the collected data from the measurement machines to the central servers. In addition we want the system to have a near to real time nature with the data in the web pages for today being current to within a few minutes. This is important if the system is to be used as a diagnosis tool. To achieve these ends we have developed an active mirror system. This operates much like the daily mirror used on many FTP sites except that file changes are reflected on the mirror site more quickly. When a monitor is started it opens a TCP connection to each of the central machines. It then watches the last-modified date on the files in its directory tree and when a file is updated the changes are sent to each of the central machines. The process is fault tolerant so that if a central machine or a monitor fails, when it recovers all machines will be brought up to date. In addition to keeping the central sites current, this approach avoids a peaky transfer load that could overwhelm the central servers or disturb the measurements being taken by the monitor.

Access to the central data is through a web page listing the monitor sites as hyperlinks. Once a link is selected a table of the RTT and loss from that site to all the other sites is supplied. Again the site names are hyperlinks. If a site from the table is selected, RTT and loss data for that pair of sites is displayed as a year-to-date graph and a set of weekly graphs for all weeks this year. Further hyperlinks allow selection of a detailed display of any day, including the RTT by time of day and as a frequency distribution. The route data can be displayed in a tabular form (like the output of traceroute) or as a graphical plot using the Otter tool from CAIDA.[6] These displays are best understood by visiting the AMP web site, which is linked off the NLANR home page.[7]

There are a large number of AMP monitors and consequently a very large number of pairs of monitors. Data is collected on the path between every pair of monitors and there are web pages for each pair. As described in section 2 this creates problems for people looking for interesting events in the data. The Cichlid visualisation tool (see section 2.1) is used as one way to address this problem. See also the "automatic event detection" section in the planned developments described below.

In the 10 months since the first AMP deployments, most of the resources of the project have been consumed by the deployment of machines and the human and software infrastructure required to manage a large network of remote machines. We are more than half way to our goal of an AMP machine at every NSF HPC awardee site. As the pressure to deploy eases more resources will become available to develop the system further. Some of the developments planned for the AMP project are:

- **Automatic Event Detection.** It would be useful to allow an interested party to register to receive notification of interesting events on some or all paths. It is difficult to determine what is an interesting event in the presence of the natural differences between RTT graphs. These differences include different base levels of RTT and the presence of large spikes in some data. A suitable method would adjust itself to the normal RTT level and not be triggered by brief changes in the RTT.

  There is a statistical method, called a process chart, which is used in process control to look for unusual situations in a manufacturing system. These conditions might indicate a machine approaching a breakdown condition. We have borrowed this technique and made some changes to allow it to be used to detect unusual situations in the RTT data.

  In this modified version of a process chart a window of data with its most advanced edge on the most recent data is maintained. The upper and lower 1 percentile of the window is calculated. If a number of successive points lie outside this range, a significant event is said to have occurred. The number of events, the percentile range and size of the window can be varied to change the way the algorithm performs, adjusting the sensitivity and rate of adaptation to new conditions of the algorithm.

  While this algorithm has been implemented and appears to give valuable results it is not practical to implement as it stands over a large number of paths in real time. Currently we are investigating how the algorithm can be simplified but retain its usefulness as an identifier of interesting events.

- **Light Weight Throughput Test**

  There is a great need for a light weight throughput test so that the available bandwidth of a path can be tested without sending large amounts of data. This is a difficult problem to solve in the general case where nothing is known about the connection. However, in the case of long term measurements of a path that does not change route often (which is the case with the networks the AMP project is measuring) changes can be understood against a well understood value of a parameter. We plan to revisit techniques such as packet pairs and different sized packets and calibrate them against known situations to see if their accuracy can be improved when more information is known about the network.

- **The IP Measurement Protocol (IPMP)**

  ICMP has a number of weaknesses as a measurement protocol. These include its general purpose nature, which makes it harder for a measurement system to process, its inability to carry time stamps and route information from intermediate or measured systems and its ability to be used as a naive denial of service attack. This last point is perhaps the most important because it causes many sites to restrict or totally block ICMP. In addition to the negative effects this has on network behaviour it means that special effort is required to persuade sites to admit ICMP to the measurement host(s).

  We have developed a proposal for an alternative to ICMP called IPMP[8] which we plan to use in the AMP project. IPMP is customised to active measurement and has many simplifications that make it easier to process. It also contains facilities to carry timestamps and path information that make it easy to add a time stamp at the kernel level or at an intermediate router. Finally, because IPMP is very easy to process, it could be implemented in the forwarding path of a router and does not provide any new opportunities for denial of service attacks, beyond those inherent in any IP packet.

5. OTHER PROJECTS

Two areas of interest to us which we have started to investigate, but which require significant further work, are the use of SNMP and BGP data for network monitoring.

In addition to the passive and active measurements, the NAI infrastructure also collects BGP information from a vBNS router and daily full routing table snapshots from a route server facility at the University of Oregon. The University of Oregon server has BGP sessions with many routers globally. MCI is also providing the full set of daily vBNS SNMP data to NLANR.

Current investigations of the BGP data focuses on both the long term evolution of the usage of the routing and addressing space, as well as short term churn in the HPC environment

6. SUMMARY AND FUTURE DIRECTIONS

We are interested in developing a good understanding of the behaviour of high-performance education and research networks with a particular focus on the networks and campuses supported and approved by the National Science Foundation. Major developments include wide deployment of both active and passive probes with more deployment planned over the next months.

We recognise the importance of analysis of the collected data and expect to shortly move to a new phase of our development, with a stronger emphasis on extending the current set of analysis programs. However we do not believe NLANR can meet all the needs of the HPC networking community in this area. We hope to support other researchers through the NAI infrastructure. This system coordinates collected data and makes both the data, and some computing resources, available to other researchers.

We invite collaborations with other organisations. Several institutions formally or informally already collaborate with us, or use our data to undertake research on their own. We would like to foster more HPC network research and particularly encourage faculty and graduate students to consider the ample opportunities for thesis work based upon the data we collect.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. ANI-9807479.

Some of the work described here was developed by students working with NLANR/MOAT including Todd Hansen, Ryan Kassel and Neil Cotofana.

REFERENCES

1. http://www.vbns.net/.
2. http://www.internet2.edu/abilene/.
3. http://moat.nlanr.net/.
4. http://atm.cs.waikato.ac.nz/wand/.
5. http://moat.nlanr.net/Datacube.
6. http://www.caida.org/.
7. http://www.nlanr.net/.
8. http://www.nlanr.net/ActMon/IPMP/.