Proceedings of the

November 2-4, 1987

Internet Engineering Task Force

Edited by
Allison Mankin and Phillip Gross

February 1988

EIGHTH IETF

The MITRE Corporation
Washington C^3I Operations
7525 Colshire Drive
McLean, Virginia 22102

TABLE OF CONTENTS

1.0 Introduction

The Internet Engineering Task Force met November 2 through November 4, 1987 at the National Center for Atmospheric Research (NCAR), in Boulder, Colorado. The meeting was hosted by Don Morris, the head of the NCAR Computer Systems Department. Don did an excellent job of handling all local arrangements, including providing terminals and Internet connectivity for document editing and mail. He also provided superb weather for the days of the meeting.

The basic format of the IETF meeting is:

1) Working Group meetings for the first 1.5 days,

2) Presentations and network reports on Tuesday afternoon and

3) Presentations, Working Group reports and discussion on Wednesday.

The final meeting agenda is presented in Section 3.

Working Group chairs are encouraged to work offline between IETF meetings, the better to fulfill their charter of accomplishing a concrete objective in a relatively short period of time. A number of new Working Groups were started during the November 2-4 meeting. An overview of the current Working Groups is included in the Meeting Notes in Section 4. Reports issued by several Working Groups are reproduced in Section 5.

The IETF's area of focus is short- and mid-range problems in the management, engineering, protocol architecture, and operations of the Internet. The IETF has launched a document series to support its endeavors; at the Boulder meeting, the series was christened IDEAS (Internet Design, Engineering and Analysis noteS). IDEAS are draft documents of the IETF. IDEAS will generally be contributed by IETF Working Groups (or by individuals participating in the IETF) on short- and mid-term issues in network, internetwork and protocol engineering. However, thoughtful papers from any responsible source on any related issue will be considered. The IETF chair is the nominal editor of the series and can be reached by emailing to gross@gateway.mitre.org.

These proceedings were assembled by Allison Mankin, who was also responsible for the main body of the meeting report. Various presenters and Working Group Chairs authored reports in Sections 4 and 5. Individual contributions are noted there.

2.0 IETF Attendees

Name	Organization	Email Address
Barker, Trudy	SRI-NIC	trudy@sri-nic.arpa
Bassett, Britt	NCAR	britt@scdsw1.uca.edu
Berggreen, Art	ACC	art@acc.arpa
Blake, Coleman	MITRE	cblake@gateway.mitre.org
Braun, Hans-Werner	U of Michigan	hwb@mcr.umich.edu
Brooks, Charles	Becam Systems	ceb@dpnt.arpa
Callon, Ross	BBNCC	rcallon@bbn.com
Case, Jeff	Univ of Tenn	case@utkcs2.cs.utk.edu
Catlett, Charlie	NCSA	catlett@newton.ncsa.uiuc.edu
Chiappa, Noel	Proteon/MIT	jnc@xx.lcs.mit.edu
Clark, Pat	Ford Aerospace	paclark@Ford-cos1.arpa
Coggeshall, Bob	CU/Boulder	coggs@boulder.colorado.edu
Crocker, Dave	Wollongong	dcrocker@twg.arpa
Enger, Robert	CONTEL	enger@bluto.scc.com
Fedor, Mark	NYSERNET	fedor@nic.nyser.net
Feinler, Elizabeth	SRI-NIC	feinler@sri-nic.arpa
Gardner, Marianne	BBNCC	mgardner@bbn.com
Hastings, Gene	PSC	hastings@morgul.psc.edu
Hedrick, Charles	Rutgers	hedrick@athus.rutgers.edu
Heker, Sergio	JUNC	heker@junca.csc.org
Hinden, Robert	BBNCC	hinden@bbn.com
Jacobsen, Ole	ACE	ole@csli.stanford.edu
Jacobson, Van	LBL	van@lbl-rtsg.arpa
Karels, Mike	UC Berkeley	karels@berkeley.edu
Korn, Lyndalee	Intermetrics	lkk@inmet.inmet.corn
Krol, Ed	UIUC	krol@uxc.cso.uiuc.edu
Lekashman, John	NASA Ames Research	lekash@orville.nas.nasa.gov
Lottor, Mark	SRI NIC	mkl@sri-nic.arpa
Love, Paul	SDSC	loveep@sds.sdsc.edu
Mamakos, Louis	Univ of MD	louie@trantor.umd.edu
Mankin, Allison	MITRE	mankin@gateway.mitre.org
McCloghrie, Keith	Wollongong	kzm@twg.arpa
Medin, Milo	NASA Ames	medin@orion.arpa
Meehl, Marla	NCAR	marla@windom.vcar.edu
Merritt, Donald	BRL	merritt@brl.arpa
Minnich, Mike	UDEL	mminnich@udel.edu
Montgomery, Doug	NBS	dougm@icst-osi.arpa
Morris, Don	NCAR	morris@scdsu1.ucar.edu
Moy, John	Proteon	jmoy@proteon.com
Mullen, John	CMC	cmcvax!jrm@ucsbcsl.edu
Natalie, Ronald	BRL	ron@rutgers
Partridge, Craig	BBN	craig@bbn.com

Perkins, Drew	CMU	ddp@andrew.cmu.edu
Petry, Mike	Univ of MD	petry@trantor.umd.edu
Ramakrishnan, K.	Digital Equip.Corp.	rama%erlang.dec@decwrl.dec.com
Reschley, Robert	BRL	reschley@brl.arpa
Rodriguez, Jose	Unisys	jrodrig@edu-vax.arpa
Roselinsky, Milt	ACC	milt@acc.arpa
Schoffstall, Marty	RPI/Nysernet	schoff@nisc.nyser.net
Schult, Nancy	Unisys	sdnancys@protolaba.arpa
Stahl, Mary	SRI-NIC	stahl@sri-nic.arpa
Stine, Robert	MITRE	stine@gateway.mitre.org
St. Johns, Michael	DDN PMO	stjohns@sri-nic.arpa
Tontonoz, Jim	DCEC	tontonoz@edn-unix.arpa
Waldbusser, Steve	CMU	swal@andrew.cmu.edu
Wolff, Steve	NSF	steve@note.nsf.gov

3.0 Final Agenda

MONDAY, November 2

- Opening Plenary (local arrangements, Discussion of IETF format, overview of new working groups)
- Working Group meetings convene
    - Open Systems Routing (Hinden, BBN)
    - Short Term Routing, Old Business (Hedrick, Rutgers)
    - Open Systems Internet Operations Center (Case, UTK)
    - Performance and Congestion Control (Stine, Mitre)
    - Open IGP (Petry, UMD)
    - Domain Issues (Lottor/Stahl, SRI-NIC)
    - (Lunch and Breaks scheduled by Chairs)
- Recess at 5:00pm

TUESDAY, November 3

Morning

- Opening Plenary
- Working Group meetings convene
    - Internet Host Requirements (Gross, Mitre)
    - EGP3 (Gardner, BBN)
    - Internet Authentication Protocol (Schoffstall, RPI)
    - InterNICs (Feinler, SRI-NIC)
    - Short-Term Routing, New Business (Hedrick, Rutgers)

Afternoon

- Management/Monitoring Working Group Report (Partridge, BBN)
- SGMP Status and Demonstration (Case, UTK)
- NSFnet Report (Wolff, NSF)
- BBN Report (Hinden/Gardner, BBN)
- Recess at 5:00pm

WEDNESDAY, November 4

Morning

- Opening Plenary
- IP over 802.X (Perkins, CMU)
- Congestion Control Simulation Results (Stine, Mitre)
- Recent Congestion Control Efforts for 4.2/4.3BSD (Van Jacobson, LBL)

Afternoon

- Working Group Reports and Discussion
    - Open Systems Routing (Hinden, BBN)
    - Short Term Routing (Hedrick, Rutgers)
    - InterNICs (Feinler, SRI-NIC)
    - Open Systems Internet Operations Center (Case, UTK)
    - Performance and Congestion Control (Stine, Mitre)
    - Open IGP (Petry, UMD)
    - Internet Host Requirements (Gross, Mitre)
    - Domains (Lottor/Stahl, SRI-NIC)
    - EGP3 (Gardner, BBN)
    - Internet Authentication Protocol (Schoffstall, RPI)
- Concluding Discussion, announce next meeting.
- Adjourn

4.0 Meeting Notes

4.1 Monday, November 2

4.1.1 Working Groups

The first one and a half days were devoted to meetings of the Working Groups. Reports that resulted from these meetings are reproduced in Section 5. A number of new Working Groups had their first meetings during this time. A brief summary of the goals of the current IETF Working Groups follows:

Open Systems Routing
Chair -- Bob Hinden (BBN)
--develop requirements, spec, and design of an interautonomous system routing protocol. Not an EGP fix.

Short Term Routing
Chair -- Chuck Hedrick (Rutgers)
--document RIP, develop administrative measures for the NSFnet technical group.

Open IGP (New)
Chair -- Mike Petry (UMD)
--develop a specificiation for an intra-autonomous system, IS-IS protocol which can replace RIP for the coming 5 years.

EGP3
Chair -- Marianne Gardner (BBN)
--complete specification of new EGP solving short-range problems.

Domain
Chairs -- Mark Lottor, Mary Stahl (SRI-NIC)
--new root server planning.

InterNICs (New)
Chair -- Elizabeth Feinler (SRI-NIC)
--transfer technology from SRI-NIC to new regional NICs, develop a cross-NIC whois service.

NOC Tools (New)
Chair -- Jeff Case (UTK)
--specify and design needed Network Operation Center applications.

Performance/Congestion Control
Chair -- Bob Stine (MITRE)

--define retrofittable fixes to alleviate congestion.

Host/Internet (New)
Chair -- Phill Gross (MITRE)
--draft Host Requirements for Internet Connection RFC

Authentication (New)
Chair -- Marty Schoffstall (RPI)
--facilitate the quick deployment of authentication methods for EGP, IGPs, network management, etc.

4.2 Tuesday, November 3

4.2.1 Management/Monitoring Status Report: Craig Partridge (BBN-NNSC)

Craig Partridge reported on both the GWMON and NETMAN efforts, on the latter standing in for Lee LaBarre who was not present. (See also the report on the NetMan Working Group by Lee LaBarre in Section 5. In the time since the previous IETF, the RFCs on the GWMON High-Level Entity Management System (HEMS) have been published (RFCs 1021-1024). Much of the specification of the system has been tested via implementation experience. Although network management functions should await a strong authentication method, the current 32-bit password on every query is already stronger than the 16-bit password used in HMP. HEMS optional statistics such as the host traffic matrix should not be implemented until strong authentication is possible.

The High-Level Entity Management protocol (HEMP) has been assigned TCP and UDP ports 151. Two independent implementations by gateway vendors (unnamed) have begun. HEMP and HEMS have been presented widely, including to the NBS Standards Workshop.

Craig's implementation of the server and a HEMP route table query client is running. Preliminary measurements show that dumping a remote SUN's route table of ten routes takes under 0.1 second.

The NetMan effort has produced two draft RFCs, one on standards issues by Amatzia Ben-Artzi, and the other an overview by Lee LaBarre. It was initially hoped that a common interface could be defined so that the same management applications could be used whether over HEMS or over the ISO-oriented protocols planned by the NetMan group, but the common interface is proving difficult to define. Though it was hoped that experiences with HEMS (and SGMP) would allow the Internet community to give input to the ISO standards development, there has been a reluctance by those attending ANSI meetings to present the extensions proposed by Internet people.

4.2.2 SGMP Status Report and Demonstration: Jeff Case (UTK)

Jeff Case spoke for the SGMP group, the chief members of which are himself, Marty Schoffstall (RPI, NYSERNET), Mark Fedor (Cornell, NYSERNET), and Chuck Davin (Proteon). The progress towards deployment of SGMP has been rapid, necessarily so in part because of the two-year funding cycle of SURANET. Network statistics and improved operation that can be obtained via SGMP are needed to ensure continued funding of this large regional network.

The SGMP specification has been found to be more powerful than originally expected. One of its greatest strengths is its extensible variable space, devised by Chuck Davin. Interoperable implementations have been demonstrated successfully. The subset of X.409 used for SGMP appears to interoperate correctly.

Two independent SGMP Network Operating Centers (NOC) and four node implementations have been deployed. The number of monitored entities is becoming large and includes sites in SURANET, RPI, NASA and MERIT. The router implementations include Proteon, 4.3 gated, and Kinetics. There is the prospect of an implementation in CMU routers, which may consist of a daemon which answers SGMP queries with information obtained through CMU's rinfo monitoring protocol.

A first version of the RPI implementation is publically available. A number of tools (applications using SGMP) have been implemented and in version 2 will be ported to the MIT X Windows environment.

The University of Tennessee implementation includes the capability to read RIP packets for network information, thus decreasing the query overhead of network monitoring. A demo of the PC-based, color graphics tool netmon was presented between sessions. It used a information from SGMP queries and RIP, both provided by a NOC resident on a SUN workstation. It displayed in realtime the up/down status of gateway interfaces throughout NYSERNET and SURANET.

Management functions have not been implemented yet because of the weakness of SGMP's currently available authentication method, a 1-15 byte password. There is a need to develop alternative methods before SGMP can be used in its fullest capability. [Note: The Authentication Working Group, led by Marty Schoffstall, is focusing in part on SGMP]. General distribution of UT SGMP is being postponed due to concern that too many centers will begin monitoring the same entities. There is a need to plan some controls, and to have most queries processed by a few NOCs. The working group led by Jeff on NOC Tools has started to develop an architecture for this.

SGMP has been assigned UDP port number 153. The RFC specifying the protocol was published (RFC 1028). Jeff summed up SGMP status, "It's here, it's now, it's what's happening; that dog will hunt."

4.2.3 NSFnet Status Report: Steve Wolff (NSF)

Steve Wolff opened this with a rueful point, that the NSFnet has in a year and a half achieved levels of congestion which it took the ARPANET many years to reach. Routing is also a big problem, and he described the RIP infinity problem as currently discouraging some networks from joining the NSFnet. The method worked out by Chuck Hedrick's Short Term Routing Working Group, "handcrafted metric munging," should be implemented globally. The strongest possible recommendation to that effect has been given at the federation meetings.

It was asked if NSF could recommend the removal of some back door connections to further straighten out routing in the NSFNet. This is not possible, since the lines are there because production users find them valuable. In addition, NSF cannot prevent new links being installed unless NSF is funding them.

Traffic analysis of the network is a vital need now, including protocol breakdowns. There are a few statistics available, for instance the amount of user access to the supercomputers which is by network rather than by other means (40% for PSC). These suggest that user congestion could increase rapidly.

New networks joining the NSFnet include MIDNET, which will be used heavily to transport biomedical data. Bitnet is becoming a mid-level component, though its traffic was already part of the scene. The new Bitnet relays (Princeton, CUNY, Cornell) start operation December 1, replacing WISCVM. Some of the upcoming "ultracontinental" NSFNet links are to: Mexico, Chile, Brazil, Puerto Rico (work on the space telescope), France (work on the TP4/ISO IP-TP0/X.25 interconnection problem), England (Level 2 bridge to JANET/EARN), and Germany.

Plans continue as well on the interconnection (not necessarily by IP) of all research agencies. How this, and expansions in general, affect routing is the focus of current NSF- (and NASA) sponsored research.

4.2.4 BBN Status Report: Bob Hinden, Marianne Gardner (BBN)

Bob Hinden started off with a summary of the ARPANET's "very busy October." The networks known to the core passed 300 a month before expected, at the end of September. There are 720 assigned network numbers. GGP and EGP have each been updated to handle up to 400 networks, after which more extensive patches will be required. The graph of the EGP networks reflects data taken at night, and would have higher numbers if the data was collected by day. A new figure available is the mix of network classes.

GGP was upgraded to fragment messages (needed in the high 300s). Both Butterflies and LSI-11's now correctly handle EGP fragmentation and reassembly. Some user sites have had problems with the EGP checksum now that updates are larger. There is a small fix to the Kirton code for this.

A gloomy note is that "an internet of 1000 is not far away. Somewhere in the few thousands, everything will break." It is time to plan what the next limit should be, and to think in terms of policy as well as numbers. Is the goal a size comparable to the telephone network? (The interesting statistic surfaced that members of the IETF have more hosts in their homes than telephones).

Other recent changes included the installation of PSN 7 in the ARPANET and a cutover to the new UDH. Still unresolved is a proposed change to EGP, to have the core include the gateway's update information in the update sent to that gateway. This can be useful for determining the source of problems. BBN may be persuaded to make this change in Butterfly EGP, to take effect when the mailbridges cut over (late spring?).

Marianne Gardner's topics were the tests on BBNNet of PSN7 and the statistics from the routing patch of July.

The PSN release has a lower overhead End to End protocol (the old End to End protocol module is in the release, too, for a smooth transition). The X.25 "munge" has been improved, with piggybacking and aggregation of RRs. PSN 7 was fully installed on the ARPANET by October 17. Testing in BBN's network exposed initial bugs, but this was not testing under any congestion. PSN 7 problems were quickly turned up in the ARPANET, including failure of ECUs with the new software which was fixed. A parameter change was needed in X.25 to insure that hosts were offered buffers.

The statistics on the routing patch show that peak round-trip times were halved and that routing update traffic was significantly decreased. The statistics were actually skewed upward by the figures from the PSNs serving the core EGP servers. The traffic destined for these PSNs is 50-80% EGP. The queues for input to the EGP servers are usually filled to the maximum length. High delays occur long before the EGP peers are blocked by lack of a RFNM. [Note: In response to this problem, Bob Enger of Contel proposed replacing the core gateways CPU boards with 11/73's. He gained a lot of behind the scenes support].

4.3 Wednesday, November 4

4.3.1 IP Over 802.X: Drew Perkins (CMU)

Drew Pearson spoke on IP encapsulations and on the background of Jon Postel's draft RFC. There is a need for technical review of this draft. It should be kept in mind that the goal is interoperable IP and ARP over like link layers, but not necessarily between different link layers. He summarized the most controversial areas of his talk as follows:

As discussed at the IETF meeting, I would like opinions on two things concerning doing IP on IEEE 802 networks.

First, all current 802.x nets have differnt MTU's. 802.3 = 1492, 802.4 > 8k and 802.5 > 2k (actual MTU dependent on a number of factors including number of nodes, speed of light, etc.). Also, a standard ethernet = 1500 (!= 802.3 MTU). We can solve this problem one of two ways. Either we can specify a standard MTU based upon 802.3's low value, thus restricting the size of packets on 802.4 and 802.5 to 1492 bytes, or we can allow different MTU's for each net and deal with the fragmentation problem some other way. With the latter, a stupid host on an ethernet sending full sized packets to a host on a 802.3 net will cause an intermediate gateway to fragment packets into full sized packets and tinygrams. Of course we can say that hosts shouldn't be this stupid and should use the TCP max segsize option or not send packets > 576 bytes. Is this valid? I think so and I think plenty of precedents for this have already been set. Therefore I propose that the MTU's for each type of network should not be administratively restricted.

Second, 802.5 networks provide the sender of a packet an indication of whether or not the packets destination address was recognized and whether or not the frame was copied (because of receiver buffer congestion). The current draft RFC specifies that an address-not-recognized error should be mapped to an ICMP destination unreachable message. It does not specify what to do with a frame-not-copied indication.

There are actually three things that the RFC could specify to do when getting address-not-recognized. First it could specify ignoring it. Second, it could stay as it is, specifying ICMP messages. Third, it could specify that the sender should delete his ARP entry and re-arp for the remote host. For a few reasons, this is an attractive thing to do. It would allow a sender to know immediately if the destination host changed his hardware address (because he replaced a bad piece of hardware or he brought up DECnet or...). Also, it would allow him to know immediately if the first hop bridge died, in the case of an IBM token ring with source routing bridges. Knowing this, he could re-arp to find a backup path.

Of course there are arguments against this scheme. Some people think of this as a layer violation and therefore shun it. Others argue that if there is more than one hop in a source routed path and a bridge other than the first crashes, this won't help you since you only find out about the first hop. Still, I think that it is a good idea and should be the suggested option. So, I propose that the RFC should suggest to do option 3 if possible, else do option 2.

Also, I think that the RFC should be changed to suggest that when getting frame not copied, the sender should attempt to resend the packet some number of times, possibly after some small timeout. This technique has been used quite successfully with proNETs for some time.

4.3.2 Congestion Control Simulation Results: Bob Stine (MITRE)

Bob Stine presented his final conclusions from the very detailed discrete event simulation of TCP connections encountering congestion through a gateway. Interesting insights arose from the validation of the model's outputs for packet delay versus real delay data. Round trip times have been observed to surge upward quickly, then to ramp down in several stages. The simulation delay data only looked like the real data when an event of the gateway shutting down for three seconds was introduced. This behavior of has been observed in gateways to the ARPANET.

The experiments conducted with the simulation required detailed analysis using a combination of rank sum and sequencing methods. Perhaps the clearest prescription from the experiments is for TCP to use a high lower-bound on round trip measurements for its retransmission timer. The high RTT seed was good for average throughput even when packets were dropped.

4.3.3 Recent Congestion Control Efforts for 4BSD: Van Jacobson (LBL)

Van Jacobson spoke on recent experimental results. The following is a summary he provided before the meeting, when it seemed as though teaching conflicts would prevent his attendance.

The most recent change we've made is to implement a congestion avoidance, dynamic window scheme for tcp very much like Raj Jain's DEC-TR-506 proposal. I should explain that because there may be some confusion between the 'DEC bit' in ISO 8473 and the overall congestion avoidance scheme. As I understand it, Jain's scheme has two separate pieces: 1) A method of detecting that congestion exists along the path (the sender's window depending on whether or not congestion is experienced.

We replaced (1) with an estimator that uses lost packets to indicate "congestion experienced". I have several reasons for preferring packet loss as a congestion indicator rather than using a new bit in the packet but the major reason is that the congestion control code can be deployed and started working incrementally and immediately: no modifications need to be made to the gateways (or even the receiving tcp's). Of course, gateway modifications will help the new algorithm (e.g., a gateway algorithm along the lines of fair-queuing or Dave Mill's preemption). But they aren't necessary and they can be done incrementally: large gains in performance could come from just fixing a few bottleneck gateways. (The other nice thing about using packet loss is that the same mechanism that lets a gateway signal a new tcp helps it deal with overload from an old, broken tcp).

I don't think we changed the window algorithm in (2) at all (I'm not sure of this because I haven't received a copy of the DEC report -- I'm basing this on the presentation Raj gave at the Boston IETF meeting): We follow the same multiplicative decrease / additive increase scheme on congestion experienced / not experienced. This isn't an accident. During the Boston presentation, it hit me that this was the only scheme that was guaranteed to converge for an arbitrary, first order linear system (i.e., for an arbitrary traffic distribution and topology) and the optimal control equations follow directly from the equation describing the system (I have since found a couple of references supporting this and I'm sure there are similar proofs in the DEC paper).

The algorithm added one new state variable and four lines of code to TCP (Mike was sanguine about the new code but the new variable hurt -- we're down to two free bytes in the tcpcb). As we currently have the algorithm tuned, it converges to a loss rate of .1 to .5%. I have run a lot of tests looking at fairness, stability and rate of convergence: everything looks great (except fairness -- that's hard to do at the endpoints). For example, I fired up 8 simultaneous ftp's on 8 different machines, each ftp using a 16KB (32 packet) window. All the traffic was fed through our poor Milnet gateway (which would allocate only 16 packets of buffer, total, for all the ftp's since they were all destined for hosts gatewayed by ucbvax). Even though the demand exceeded the gateway capacity by 1600%, all the connections "learned" the available capacity in just 5 round trip times and the loss rate settled down to .5% (the loss rate is due to the algorithm "testing" the path to see if, say, some other connection has closed down and freed up some more bandwidth. You can make the loss arbitrarily small but you increase the time it takes a connection to learn "good news". We thought something around 1% was a good tradeoff between bandwidth lost to retransmissions and bandwidth lost to underestimating the window.)

All the tests have worked so well that we're thinking it's time to put tcp on the back burner and start looking at gateway algorithms. I think fair-queuing, combined with some cleverness in figuring out when to drop packets and which to drop, would be a workable algorithm. But I think we can do things that are a lot simpler: I worry that fair-queuing requires the gateway to know something about the transport protocols (something I think we should avoid since there are several new transport protocols on the horizon and it will be a lot of work to keep gateway implementations current with the protocol mix) and fair queuing requires a lot of state in the gateways (something we should avoid to make the next generation packet switch - the state maintenance adds a lot to the packet processing time and the space used for end-to-end state could probably be better used as packet buffers or routing cache). I have some "random" gateway algorithms that I think would do as good a job for congestion control as fair-queuing but require no state and have negligible per-packet cost. (If my NSF proposal ever makes it through the LBL bureaucracy and out to Steve Wolfe, it asks for funding to simulate, then prototype and test these gateway algorithms.)

That's about all that's been happening here over the past couple of months. Oh, there's one other encouraging note: Keith Sklower at Berkeley has ported all the tcp algorithms (timer stuff, slow start, fast retransmit, dynamic window) to the 4.3bsd XNS Sequenced Packet Protocol implementation. He's just started testing but Friday he reported that the new code improved throughput from a Sun 3/50 XNS client to an (unmodified) Xerox fileserver by 50% -- 16KBS to 24KBS. (I thought this was neat because the algorithms are really intended for a long haul net. It's nice to see them making a big improvement on a local net). Since everything went into SPP pretty easily, it might bode well for applying all this stuff to TP4 (or whatever ISO sticks us with).

5.0 Working Group Reports

This section reproduces the reports on the November 2-3 meetings issued by the working groups (some previously distributed by electronic mail). The NetMan Working Group did not meet at the IETF, but their report of several off-line meetings is included in this proceedings. The Authentication Working Group did meet for the first time at the IETF, but the report included here covers their second meeting in Boston in February.

Reports in this section:

Short Term Routing

NOC Tools and Applications

Performance and Congestion Control

Authentication

NetMan

5.1 Short Term Routing Working Group

Convened and reported by Charles Hedrick (Rutgers)

Participants:

Charles Hedrick	hedrick@rutgers.edu
Sergio Heker	heker@jvnca.csc.org
Mike Minnich	mminnich@udel.edu
Louis Mamakos	louie@trantor.umd.edu
Jeff Forys	forys@boulder.colorado.edu
Mark Fedor	fedor@nic.nyser.net
Bob Coggeshall	coggs@boulder.colorado.edu
Charlie Catlett	catlett@newton.ncsa.uiuc.edu
Jeff Case	case@utkcs2.cs.utk.edu
Ed Krol	krol@uxc.cso.uiuc.edu
Paul Love	loveep@sds.sdsc.edu
Britt Basset	britt@scdsw1.ucar.edu
Ross Callon	rcallon@bbn.com
Hans-Werner Braun	hwb@mcr.umich.edu
Gene Hastings	hastings@morgul.psc.edu
Don Morris	morris@scdsw1.ucar.edu
Steve Wolff	steve@note.nsf.gov

Minutes of the Short Term Routing Working Group Meeting of November 3, 1987.

These minutes are based on notes taken by Jeff Case. As I have been unable to get a machine-readable copy of the originals, I'm typing them again. This will undoubtably result in some editorial comments from me. So you shouldn't hold Jeff Case responsible for the views expressed here.

The meeting began by tabulating a list of problems observed in the NSFnet community, and other issues to be discussed:

EGP backup
X.25 virtual channels running out in Arpanet gateways
X.25 to 1822 host incompatiblities
routing instabilities in NSFnet
SURAnet and NYsernet are seeing routes vanish
some routes are vanishing because RIP metrics are > 16
connections breaking
connections timing out

Major discussions resulted from the EGP backup issue and various routing problems.

5.1.1 EGP BACKUP

Considerable time was devoted to the discussion of EGP backup. The problem is determining how to advertise backup paths to the Arpanet core gateways. As it was presented to us, when a Fuzzball is used as a gateway to the Arpanet, it advertises every network it knows about to the core. This is happening in one or two places, but more sites interpose a VAX running gated between NSFnet and the Arpanet. Gated allows them to control the list of networks to be advertised via EGP. The goals as described in an NSFnet B.O.F. the previous night were as follows:

- we want to make sure that enough gateways advertise each network that failures don't interrupt their access to the Arpanet

- we want to be able to avoid certain links that have performance problems (especially the Linkabit gateway, with a 7200 baud connection)

- some sites have multiple Arpanet connections, and do not want anyone else to provide backup for them, at least not unless all of their sites are down; other sites would prefer to negotiate backup with specific sites.

The primary issue brought to the working group from the B.O.F. was whether it was OK to continue having Fuzzballs advertise all of the NSFnet networks, and if not how much control was necessary.

The problem is exacerbated by the limited number of levels within EGP/GGP, which allow only a two-level hierachy -- 0 and 3 -- for advertisement of paths. If we had three values, there would be no problem: sites would use 0 for their primary connection, N for any specifically negotiated backups, and 2*N as a general fallback to be advertised by all other gateways. However as far as we can tell, only two values can be used. This forces us to choose between being able to designate specific backup sites and having automatic fallback to any gateway.

The problem is made more difficult by several factors:

1. Not all backup routes are equally desirable due to bandwidth, quality, proximity, etc., and

2. The LSI gateways appear to select the Linkabit gateway as the path of choice in case of ties (at value = 3). (No one at BBN can explain why this would happen. I am unable to verify personally that is does, but several people at the meeting claim to have observed it.)

At least one network manager felt very strongly about wanting to be able to control has primary backup gateway(s).

This discussion applies only to EGP, which controls which gateway is used by traffic from the Arpanet to an NSFnet network. There is of course a complementary issue involving how traffic from the NSFnet chooses which gateway to use to get to the Arpanet. This is done by metric computations with the the NSFnet backbone, and normally works out to mean that traffic goes to the "nearest" (in some generalized sense) Arpanet gateway. This behavior was not seen by anyone to be a problem.

Based on all of these considerations, the following agreement was reached:

> It is recommended that no connections which perform uncontrolled advertisement to the Arpanet core of others' (i.e. non-local) routes be allowed between the NSFnet and the Arpanet/Milnet, either directly, or indirectly via regional/consortia networks.

That is, gateways between NSFnet sites and the Arpanet must either obey the third-party rule (they do not advertise networks outside the local AS), or they must have controls on what routes they advertise. No specifications were drawn up for those controls or how they would be used. However it was implied that the controls would be roughly equivalent to those provided by gated. The implication was that each network would be advertised by a few gateways, that specific requests of the network administrator would be taken into account in choosing those gateways, and that otherwise an attempt would be made to make choices based on good network engineering practices. (That is, nearby gateways, and those having high bandwidth connections would be favored.)

There are several ways to implement this recommendation at existing and anticipated sites that would otherwise have no controls. Some possibilities include:

o Linkabit
- determine how to bias the tiebreaking algorithms among the LSI gateways to make the low-bandwidth link the last choice in case of a tie
- cut the link [presumably from the NSFnet core to Linkabit?]
- diddle the Fuzzball software to add a switch such that only local networks are advertised.

o SESQUInet
- advertise only local nets
- acquire funding for additional hardware to implement a filter such as that provided by the gated daemon or equivalent

o Merit
- acquire funding for additional hardware to implement a filter such as that provided by the gated daemon or equivalent

Although the discussion focused on places where the NSFnet backbone meets the Arpanet directly, there is a similar issues at any Arpanet gateway where NSFnet routing information is present. That is, any campus or mid-level network that circulates NSFnet routes in its internal routing table might conceivably end up advertising these routes to the Arpanet core. The recommendations above apply to any such gateway.

5.1.2 ROUTING AMONG THE MID-LEVEL NETWORKS

Many of the problems in the initial list can be traced to problems with routing. Specifically, it appears that INFINITY = 16 in RIP is having an increasingly serious effect. Many of the reports that routes to certain networks come and go appear to be due to this problem. Instabilities in routing appear to be due at least in part to the fact that a single RIP/Hello metric is being run over the entire country. The designers of RIP did not intend it to be used for such a large network, and do not consider the protocol to be stable in such a use.

After considerable discussion, it was recommended that mid-level networks immediately begin implementation of schemes that segment routing. Routing information exchanged among the mid-level networks, and between them and the NSFnet backbone would be primarily reachability, not metrics. This was referred to variously as "fallback routing" or "autonomous system-style routing". I will be remailing notes from the July IETF meeting, where an attempt was made to work out the implications of this in somewhat more detail.

It was recommended that Suranet be worked on first, although it appears that BARnet will also using a similar strategy when it is finally connected to the NSFnet. (Suranet is suggested because it appears to have the most serious problem, probably because of its relatively large diameter.)

Ed Krol agreed to put this issue on the agenda of the Federation meeting scheduled for November 18, in Pittsburgh.

The working group wishes to be clear that we see the routing reorganization described here as only a stop-gap. It is obvious that new routing protocols are needed. Thus we see the activities of the IGP and inter-AS routing groups in the IETF as quite important. In particular, we would like to make sure that the permanent NSFnet management team, when in place, is charged with the responsibility of finding and implementing better routing mechanisms. However we think it will be at least a year before new protocols can be developed and deployed. We are already seeing dead bodies on the floor. So we believe it is essential to move to autonomous system style routing immediately. It appears that most long-term solutions are going to use the distinction between an IGP and an inter-AS protocol, so this reorganization will be useful preparation in any case.

5.1.3 DISCUSSION OF THE X.25 VIRTUAL CIRCUIT PROBLEM

The root of the problem is that the popularly used X.25 hardware/ software runs out of resources for virtual connections. The current systems are limited to 64 open virtual connections between Arpanet (net 10) host-host pairs and, at times, more have been required. This resource limitation has been particularly severe at PSC.

Part of the problem appears to be that unused connections are not closed and scavanged.

No action was taken nor recommendations formed, as it is believed that efforts are in progress between the vendor and PSC.

5.1.4 THE FUTURE OF THE SHORT TERM ROUTING WORKING GROUP

It was agreed that the group meet at least one more time to review the status of the implementation of the recommendations and their subsequent effects.

It was also decided to create a mailing list to discuss items related to the morning's discussions.

I had some concerns about possible overlap between this group and an NSFnet B.O.F. chaired by Ed Krol. At least this time, the NSFnet B.O.F. was directed towards more directly operational issues, whereas this group looked at system-wide routing issues. It is still possible that these two groups might merge over time. It is important to have a group that can take an overall look at how the technology is working out, and suggest changes.

5.2 Performance/CC Working Group

Convened and reported by Bob Stine (MITRE)

Participants:

Berrggreen, Art	art@acc.arpa
Blake, Coleman	cblake@gateway.mitre.org
Chiappa, Noel	jnc@xx.lcs.mit.edu
Coggeshall, Roy	coggs@boulder.colorado.edu
Jacobsen, Ole	ole@csli.stanford.edu
Mankin, Allison	mankin@gateway.mitre.org
Merrit, Don	Merritt@brl.arpa
Minnich, Mike	mminnich@UDEL.EDU
Mullen, John	cmcvax!jrm@ucsbcsl.ucsb.edu
Partridge, Craig	craig@bbn.com
Ramakrishnan, K.K.	rama%erlang.dec@decwrl.dec.com
Schult, Nancy	nls@oahu.mcl.unisys.com
Stine, Robert	stine@gateway.mitre.org
Tam, Kok	tam@UWOVAX.bitnet
Wolff, Stephen	steve@note.nsf.gov

Summary of the 2 Nov 87 Meeting of the Congestion Control Working Group

The goal of the congestion control working group is to produce a white paper recommending quick fixes for improved Internet performance. By definition, a quick fix is one which:

1. improves performance,
2. can be retrofitted into host or gateway protocol implementations, and
3. allows interoperation with "unfixed" implementations.

In the 2 Nov meeting, several candidate congestion control techniques of this type were discussed. This paper summarizes the major points disussed at that meeting.

Parentheses are used to flag afterthoughts of the author. Comments and nonprofane suggestions are welcome.

There was agreement that several fixes should be recommended. Other approaches were regarded as requiring more study before decisions to deploy them. In addition, several schemes were discussed that would require protocol modifications, and hence are beyond the scope of this working group. Also, a long-term requirement for the development of a distributed, adaptive mechanism for Internet resource allocation was noted.

5.2.1 Recommendations

There was general agreement that the following congestion fixes be recommended:

5.2.1.1 RTO values. For system stability, RTO timers must increase exponentially. However, if connections are to be maintained across lossy nets, the Maximum Segment Lifetime (MSL) must be large enough so that several retransmissions can occur without causing the connection to abort. It is recommended that the MSL be application configurable.

5.2.1.2 RTT estimation. TCP's algorithm for RTT estimation is a cause of wasted resources on the Internet. The white paper produce by the Congestion Control Working group will point to several papers (by Mills, Partridge, Zhang, and Jacobson) which cite the deficiencies of exponential smoothing and offer alternative algorithms. At a minimum, host administrators must guarantee that the seed for the SRTT algorithm is reasonably high.

Despite the deficiencies of the exponential smoothing algorithm, ad hoc experimentation with RTT algorithms is strongly discouraged.

5.2.1.3 Small packet avoidance. TCP implementations should attempt to avoid the proliferation of tinygrams. Withholding acks, however, is not a good means of effecting this policy. Withholding acks would interact poorly with Van Jacobson's slow start algorithm. Also, a bug has been seen in which hosts with very large windows never receive enough data to trigger an ack.

5.2.1.4 Van Jacobson's algorithms. Van Jacobson's recent developments - use of mean deviation for estimating RTT, slow start, fast retransmisson, and dynamic window sizing - look very promising. Individuals who have implemented them report very good results. Before endorsing these methods, members of the IETF will have the opportunity to thoroughly test the performance of these algorithms: Mike Karels has developed a beta release of bsd 4.3 tcp which includes them.

Van's dynamic window adjustment is similar to that of the Jain, Ramakrishnan, and Chiu "DEC-bit" scheme: windows are increased incrementally, but decreased multiplicatively. K.K. Ramakrishnan noted that using dropped packets to signify a congested state allows a system to reach a suboptimal state.

(The goal of the DEC-bit scheme is to keep a network operating near its optimal load; it is a congestion avoidance technique, rather than a congestion control technique).

(It is particularly desirable that results be obtained for the performance of Van's algorithms in support of interactive applications, since, to the best of my knowledge, most tests have studied the impact of Van's algorithms on large file transfers. Also, results should be obtained on the performance of these algorithms across lossy nets, since Van has pointed out that their performance may be less optimal than that of TCP with fixed windows if a high percentage of packets are dropped.)

(Barring disappointing results from Mike's tcp, the white paper produced by this working group will explain Van's algorithms, and recommend their use in TCP implementations.)

The slow start algorithm achieves the same function, and is thought to be superior, to the Nagle algorithm for window adjustment, in that the slow start scheme is explicitly nonlinear in traffic reduction.

5.2.1.5 Random dropping. When a gateway must drop a packet, selecting the packet at random is preferable to dropping the most recent in. The major reason for this approach is to curtail hosts that are overloading gateways: using random dropping, the source that has contributed the greatest amount of traffic has the highest probability of having one of its packets dropped. This policy is not necessarily unfair to high volume hosts; in effect, it treats all sessions on an equal basis. Also, it would be simple to implement and inexpensive to perform. (Furthermore, Van's adaptive windowing scheme works better with random dropping.)

5.2.1.6 Source Quench messages. It is recognized that Source Quench messages are not perfect, but they are available and can be useful for congestion control. Several principles should be followed in their use:

1. quenches should be sent before overflow occurs.
2. the rate at which quenches are sent to a particular source should be controlled.
3. there should be different triggers for quenching and ceasing to quench; a hysteresis is desired.

The question of determining which host to quench is unsolved. If, however, sources which are spuriously retransmitting can be detected, then their traffic should be preferentially discarded. (Van remarked that he suspects that Source Quenches have the undesirable effect of bunching traffic, and as a result causing a net increase in segment retransmissions).

5.2.1.7 IP Fragmentation/Efficient Packet Size. Fragmentation is extremely wasteful of gateway resources, and must be avoided. However, because much network processing has a per packet cost, efficiency is increased if packets are as large as the least MTU of the subnets they traverse. Discussion was curtailed when it was noted that the issue had been authoritatively explored is explored in the '87 SIGCOMM Proceedings article, "Fragmentation Considered Harmful," by Mogul and Kant.

(MTU negotiation would require a mod to IP, and so is beyond the scope of a quick fix. Based on the article by Mogul and Kant, I'd recommend the following:

1. Above all, IP should attempt to avoid fragmentation.
2. For packets greater than 576, the "don't fragment" bit should be set. If an ICMP "fragment needed" message is received, then packet length should be reduced.

3. Hosts that will frequently send large volumes of data to a given destination can probe for the minimum MTU in a path by step 2 above. Results should be cached, though changing routes will date the information.
4. For hosts that will not often maintain long connections, the appropriate policy is to keep packet length no more than 576 for traffic destined for other nets.)

5.2.2 For Further Study

Of the following congestion control schemes, it was generally agreed that deployment would be premature:

5.2.2.1 SQuID. There is great concern over the impact of Source Quench Introduced Delay (SQuID), especially on its potential for poor interaction with transport layer reliability schemes. SQuID should be strictly regarded as a topic for research. Vendors are strongly discouraged from including SQuID in operational IP releases.

5.2.2.2 . Source Quench Induced Retransmission (SQuIRT) is a proposed practice of retransmitting TCP segments whose associated datagrams have triggered ICMP Source Quench messages. Research would be required to judge its effectiveness.

5.2.2.3 DEC congestion avoidance. The Congestion Avoidance scheme of Jain, Ramakrishnan, and Chiu could be implemented by us of an IP option. It would be interesting to have experimental results on the use of this algorithm. It is premature to recommend its adoption. However, several of the principles of the scheme are worth considering in their own right, in particular, the calculation of average arrival rates, and the implementation of resource allocation policies ("fairness"). If a gateway is using the Jain/Ramakrishnan/Chiu scheme in an environment with uncooperating hosts, it must be prepared to penalized traffic from these hosts.

5.2.2.4 Fair Queuing. There is concern that Fair Queuing effects too egalitarian an allocation of gateway services, and so would have the disadvantage of punishing legitimate high-volume traffic sources (e.g., mail relays, name servers, etc.). In other words, mail relays and name servers perhaps deserve more than an equal share of gateway bandwidth. Another criticism of fair queuing is its gateway processing requirements. However, fair queuing is useful for protection against abusive hosts.

(In addition, it provides very quick feedback (in increased RTT) for traffic sources that are offering traffic at a higher rate than the gateway can process. Also, fair queuing is useful for "evening out" traffic loads over time.)

More research should be performed before fair queuing can be whole-heartedly endorsed. (However, it merits serious scrutiny. Note that fair queuing could be used in conjunction with random dropping.)

5.2.2.5 Scheduling. There was some discussion on introducing scheduling disciplines in gateways (e.g., a policy of giving preference to interactive traffic). There was concern that mail and name-server traffic would be adversely affected if interactive traffic were too aggressively promoted. It was noted that this has been a thoroughly researched topic in application to operating systems. However, policy decisions determining which traffic should be preferred must be reached before scheduling techniques should be installed.

5.2.2.6 Circuit oriented service. It was noted that it would be preferable for a minimum level of throughput to be guaranteed for a TCP connection. The thought is that it would be better for several users to have adequate service than for many users to have inadequate service. One means of implementing a connection oriented service would be for gateways to cache connection IDs (source and destination address, and port numbers), and to respond with an ICMP "Host Unreachable" message if the number of connections is exceeded. A connection oriented scheme would require preemption, and also the ability to timeout inactive connections.

This proposal is controversial enough that it should be regarded strictly as a research topic.

(Soft circuits have also been proposed as a method for reducing processing overhead in routers; such schemes would be of questionable effectiveness if deployed in boxes that don't perform timer interrupts efficiently.)

5.2.2.7 Selective Retransmission. It is conceivable that selective acknowledgement and retransmission of TCP packets could be implemented as an upwardly compatible TCP option. The introduction of a new TCP option, however, is not within the scope of a quick fix.

5.2.2.8 Firewalls. It was noted that gateways must have means of protecting networks from abusive hosts. One suggestion was the use of use of a gateway-to-gateway ICMP telling a host's entry gateway to throttle a given host. The notion of a "squelch host" option, however, was regarded with some trepidation.

(Protecting nets form overly verbose hosts would seem to require, at a minimum, measuring the rate at which hosts are offering traffic. In "Congestion Avoidance in Computer Networks With a Connectionless Network Layer," Jain, Ramakrishnan, and Chiu offer a means of calculating average queue lengths; this technique could be applied on a source or application basis.)

5.2.2.9 Purging duplicate packets. This is a probably bad idea that wouldn't die. If a TCP implementation is spuriously retransmitting segments, then a gateway might have several identical packets from it. In that case, dropping the duplicate packets would seem to assist in lowering congestion. The major problems with this proposal are:

1. the overhead would be high (people scream about the overhead fair queuing would require; bookkeeping on a per packet basis

would seem much worse).
2. TCP does not necessarily maintain consistent segment boundaries when segments are retransmitted.

A solution to the second problem was discussed, viz., for the IP IDs and segment boundaries of retransmitted packets to be the same as those of the original packets. It was noted that this would require major modifications to most TCP implementations.

(If a gateway developer really wanted to implement the purging of duplicate packets, it would probably be simpler to peek into datagrams as far as the TCP sequence number and length field to detect duplicates. It has not been established that the performance gains from purging duplicate packets would justify the processing cost.)

5.2.3 Internet Resource Allocation

There is a large measure of agreement on techniques for improving the effectiveness of gateway operation. For example, Source Quench messages should be sent prior to buffer overflow, since waiting for overflow allows the network to become too congested. However, there is an unsolved problem concerning the allocation of resources. For example, if congestion occurs, which host should be quenched?

There is a requirement for a gateway resource allocation algorithm to be developed. It should:

1. allocate resources based on a stated policy of an Internet governing body (or bodies), and have the ability to reflect changes in this policy (note that this requires a policy!!).
2. implement the allocation "dynamically" (i.e., in a demand-based manner. Resources unneeded by preferred hosts should be available for other hosts).

Clearly, this problem is not solvable in the short term.

(Hence, perhaps a short term solution would be to attempt to provide most efficient service on an egalitarian basis - attempting to give all hosts an equal share of Internet resources - and then hack exceptions to this policy if necessary services (mail, name service) are unable to function adequately.)

(Doesn't this topic belong with the network management group?)

5.3 NOC Tools Working Group Report

Convened and reported by Jeff Case (UTK)

Participants:

Keith McCloghrie
The Wollongong Group
kzm@twg.arpa

Steve Waldbusser
Carnegie Mellon University
sw01@adndrew.cmu.edu

Charles E. Brooks
Becam Systems
CEB@DDNT.ARPA

Paul Love
San Diego Supercomputer Center
Loveep@sds.sdsc.edu

Jon Rochlis
MIT
jon@bitsy.mit.edu

Charlie Catlett
National Center for Supercomputer Applications
University of Illinois
catlett@newton.ncsa.uiuc.edu

Ron Natalie
Rutgers University
ron@rutgers.edu

Don Morris
NCAR
morris@scdsw1.ucar.edu

Drew Perkins
Carnegie Mellon University
ddp@andrew.cmu.edu

Marty Schoffstall
RPI/NYSERnet
schoff@nisc.nyser.net

Craig Partridge
NNSC/BBN
craig@nnsc.nsf.net

Jeff Case(Chair)
University of Tennessee
case@utkcs2.cs.utk.edu

5.3.1 Goals and Scope:

This being the first meeting of the working group, much of the meeting was spent defining the goals and objectives of the working group.

The general scope of the effort will be to identify:

1) the duties and activities of NOC personnel including the questions they need to answer, problems they need to solve, and reports they need to generate;

2) the information they need to accomplish #1, above;

3) the data that are needed to produce the information in #2, above;

4) the sources of the data in #3, above;

5) the tools and applications needed to process those data; and

6) architectures for the development of those tools and applications.

The meeting began with a discussion of the tasks that a network operations center performs and they were combined into three broad categories:

- collection
- distribution
- display

The characteristics of tools needed by a NOC included:

- appropriate tools for various skill levels
    - (operator, beginner, expert)
- appropriate tools for various tasks
    - monitoring (fault detection)
    - firefighting
    - control (bypass and repair)

There was general consensus with the thesis that network monitoring and control is a multi-dimensional problem, including:

- product specific/protocol specific
- boundaries:
    - economic
    - administrative
    - political
    - trust
- skill level of operator

help desk
network operator
network systems programmer
network design engineer
network manager

task

user troubleshooting
firefighting
routine monitoring
capacity planning/engineering
configuration/change management

Consensus was reached regarding the several aspects of the focus of the WG's effort.

Monitoring versus Control: The consensus was that the group should tackle both but place a priority on monitoring.

Scope: The consensus was that end-to-end monitoring and control is essential and should be the scope of our deliberations. To limit discussions to only some entities, such as gateways, was deemed to be inappropriate. The entities to be monitored would be those which could be reached by an IP based monitoring protocol such as HEMS, SGMP, HMP, RWHO, ICMP plus those which can be reached via another protocol (such as DEC's MOP) indirectly through an IP proxy agent.

The group began to identify some of the questions that a network operations center must answer, the reports which need to be generated, and the problems which need to be solved.

Why doesn't it work?
Why can't I get there?
Is it working?

How much traffic is there?
  what is the nature of the traffic?
  capacity planning
  performance

Intermittent problems
  can you tell me more about this?
  intense monitoring

Uptime report
  MTTR
  Problem analysis

What is the problem?
- automatic network map
- wire walker
- How to display
- Collect the data
- Point of view

5.3.2 Model/Architecture:

The model for network management was considered. The conclusion was that any given monitored entity is likely to be of interest to multiple monitoring stations and that it may be desirable in some situations to have a NOC server "front" for the device by answering redundant requests for network monitoring data. The resulting model was that, in the general case, any monitored entity might be monitored and/or controlled by zero or many NOC's (primary and backup or national, regional, campus, etc) and zero or many monitoring stations. Similarly, a NOC may serve data to and from zero or many monitoring stations and a monitoring station may interact with zero or many NOCs and zero or many gateways.

<see figures in Section 6>

It was deemed to be beyond the scope of the group to create or rework any existing protocols for the traffic between the monitored entities and the monitoring center(s).

It was thought desirable to consider using the same protocol between the display stations and monitoring centers as is used between the monitoring centers and the monitored entities.

The architecture of the internals of a likely NOC server implementing this model is shown on the following figure.

<see figures in Section 6>

At that point in the discussion Craig Partridge joined the WG to answer questions about the architecture and its compatibility with the HEMS/HEMP protocol suite (there were enough SGMP developers in the meeting to address the issue for that protocol.) There were no apparent conflicts between the model/architecture and the protocols.

It was observed that the architecture was very similar to a very simplified version of BBN's Automated Network Management (ANM) efforts. ANM uses the NMP protocol between the NOC server and the monitoring / display station(s).

5.3.3 Future WG Activities:

It was decided to interact via a mailing list to be established and to meet at the next IETF, if not before.

5.4 Authentication Working Group

Convened by Martin Schoffstall (NYSERNET) Reported by Chuck Davin (Proteon)

A meeting to discuss authentication mechanisms in the context of network monitoring and control was held at BBN on 4 February 1988. Present were Chuck Davin, Phill Gross, Steve Kent, John Rochlis, and Mike St. Johns. Absent, due to weather, was Marty Schoffstall.

Discussion centered at any given moment on one of the four topics below.
(1) Requirements for Authentication Mechanisms in the Internet

In the most general terms, it was agreed that any desirable authentication scheme has the following characteristics:

(i) It supports authentication of data origin.
(ii) It supports protection of data integrity.
(iii) It optionally supports confidential exchange of information.

Four problem areas were identified as being of concern to those present.

(a) Authenticating network monitoring and control exchanges among gateways and one or more monitoring centers

(b) Authenticating exchanges of routing information among gateways

(c) Authenticating exchanges among components of the Domain System

(d) Authenticating users of remote resources in a way that does not involve plaintext transmission of user passwords

It was generally agreed that a single mechanism that addresses all of these problem areas is highly desirable.

It was generally agreed that the feasibility of a single solution that addresses all of the identified problem areas is closely tied to the computational cost of that solution.

It was generally agreed that the integrity of the Domain

Name System was not required to realize a generally usable authentication mechanism.

It was generally agreed that the problem of authenticating users of remote resources almost is certainly solved by any scheme that addresses the other problem areas.

It was understood by all present that the problem of authenticating a human user is quite different from the the problem of authenticating a process that acts on behalf of a human user. It was generally agreed that the latter problem is of most immediate concern.

(2) Computational Costs of Authentication

Because the computational cost of an authentication mechanism largely determines its applicability in a particular problem area, some attempt was made to quantify the traffic requiring authentication.

BBN staff provided relevant figures for core gateways:

In times of relatively stable network topology, about 30 % of core traffic is "control" traffic (i.e. addressed either from or to a gateway -- HMP, ICMP, GGP, and EGP). In times of relatively volatile topology, about 50 % of core traffic is control traffic.

A typical core gateway passes about 500K pkts/day. Average observed packet size is about 100 bytes.

It was assumed that control traffic does not vary much with network usage (although it does vary with topological instability).

From these figures, it was estimated that a gateway handles about 3 control packets / second on a fairly regular basis.

In this context, it was asserted that an 8Mhz 68000 microprocessor can perform the DES algorithm in software at a rate of 64 Kbits/second. DES Mulitbus boards built by BBN using AMD chips can perform the DES algorithm at 1.3 Mbits / second. It was also asserted that computing RSA encryption in software is feasible.

The number of independent sessions that must be supported is another number useful in assessing the cost of particular authentication schemes.

In the network management problem area, it was estimated that a gateway might support SGMP interactions with 25-30 monitoring entities, although the number of distinct sessions requiring authentication might be a much smaller number (2-3).

It was estimated that each DDN gateway interacts with 2-3 monitoring entities.

In the routing exchange problem area, it was estimated that each core gateway exchanges routing information with about 400 other parties. Non-core gateways typically peer with 2-4 other parties. If exchanges between both EGP peers and other members of the local AS are counted, then the average number of routing exchange peers for core and non-core gateways considered together is perhaps 100.

(3) Gateway-Network Binding Problem

The problem of authenticating exchanges of routing information among gateways was considered in connection with the more complicated problem of authenticating advertisements of particular networks by particular gateways.

It was generally agreed that, given an authentication scheme that addressed the former problem, the latter problem could be solved by associations between networks and advertising gateways that are realized independently of the actual authentication mechanism.

(4) Discussion of Various Authentication Schemes

Each of a number of authentication schemes was discussed with respect to

- (a) the time-frame in which it could be implemented
- (b) the marketability of required encryption mechanisms
- (c) the number and frequency of required packet exchanges
- (d) required clock synchronization (if any)
- (e) support for multiple administrative domains

J. Rochlis distributed documents describing the Kerberos authentication scheme and reported on its salient features:

- (a) a release available in 1-2 months

under terms similar to that for
the X distribution

(b) 3-party authentication scheme
(c) Loosely synchronized clocks (delta 5 min)
(d) Session keys good for extended periods
(e) Hierarchical name space

It was generally agreed that the least compromising way to use Kerberos across multiple administrative domains involved the identification of the Kerberos name space with the Domain System name space.

S. Kent presented a scheme related to the "certificate" mechanism in X.500. This scheme was quite attractive, but the time-frame in which it could be realized remains unclear.

Other schemes discussed briefly included Vice, Visa, and XNS.

Vendor concerns regarding patent rights and international distribution were discussed.

Methods of exploiting any particular authentication scheme to satisfy the requirements articulated at the meeting were not discussed.

A meeting for further discussion of outstanding issues was scheduled.

5.5 NETMAN Working Group Activities

Reported by Lee LaBarre (MITRE)

The NETMAN working group has adopted the ISO model and protocols for management of TCP/IP based networks. This approach will facilitate the transition from TCP/IP based components to ISO based components and management of networks that contain both types of components during the transition. It also builds on the many years of effort expended within ISO and IEEE in developing management standards.

Once the decision was made to align with the ISO model and protocols for network management, work was concentrated in four basic areas:

- Define a mapping from the ISO management protocol (CMIP) and associated ISO application layer protocols onto the TCP stack of protocols. This was accomplished by development of a "thin" presentation layer protocol which offers ISO presentation kernel services and maps onto TCP and UDP. We are indebted to Marshall Rose of Wollongong ("father of ISODE") for this excellent work, as documented in the draft RFC "ISO Presentation Services on top of TCP/IP-based internets".

- Define the manner of identifying management information in a manner which is consistent with ISO. We have agreed on a management information tree for the Internet application layers and below. The tree is described in the draft RFCBBBBB Implementors Agreements on Network Management.

- Define the structure of management information for data transfer using ASN.1, e.g., counters, guages, thresholds, tables, etc. This will be documented in a separate RFC, but for now it is contained in RFCBBBB to facilitate discussion in the group.

- Define the management information for the transport, network, data link, physical and application (FTP, Telnet, SMTP) layers; and define management information peculiar to individual classes of systems (routers, bridges, endsystems, etc.). This will be documented in individual RFCs as appropriate, but meanwhile portions (ASN.1 descriptions) will reside in the draft RFCBBBB to facilitate the work of the group.

The current structure of the NETMAN document set is:

- Implementor's agreements (RFCBBBB, editor Lee Lee LaBarre) which reference the following documents:
    - Management Overview: concepts and Architecture (RFCAAAA, editor Amatzia Ben-Artzi)
    - ISO ACSE
    - ISO ROSE
    - Marshall Rose Presentation (RFCXXXX)
    - ISO CMIS
    - ISO CMIP
    - Transport (TCP/UDP) Management Information (RFCTTTT)
    - Network (IP, ICMP, etc.) Management Information (RFCNNNN)
    - IEEE 802.2 and 802.3 Layer Management documents and ASN.1 syntax (RFCIEEE)
    - SMI and Tree (RFCSMI)

Initial text of RFCTTTT, RFCNNNN, RFCIEEE, and RFCSMI will be kept in draft RFCBBBB and moved to separate documents as appropriate.

A draft implementor's agreements document is in progress (Draft RCBBBB). Agreement has been reached on:

- Protocol Architecture
- use of ISO ACSE,
- use of ISO ROSE,
- "pseudo" presentation protocol and its use,
- format of RFCBBBB,
- the Management Information Tree
- CMIS services to be supported,
- CMIP syntax, semantics, and parameter options,
- Functional classes for managers and agents

6.0 Presentation Slides

This section contains the slides for the following presentations made at the November 2-4, 1987 IETF meeting:

- Management/Monitoring Status Report (Partridge, BBN-NNSC)
- SGMP Status Report and Demonstration (Case, UTK)
- NSFnet Status Report (Wolff, NSF)
- BBN Status Report (Hinden/Gardner, BBN)
- IP over 802.X (Perkins, CMU)
- Congestion Control Simulation Results (Stine, MITRE)
- Recent Congestion Control Efforts for 4.2/4.3BSD (Van Jacobson, LBL)
- Network Operating Center Tools Working Group (Case, UTK)
- InterNICs Working Group (Feinler, SRI-NIC)
- Domain Working Group (Lottor, SRI-NIC)
- EGP3 Working Group (Gardner, BBN)

Management/Monitoring Status Report

Craig Partridge, BBN-NNSC

Report On Work of Gateway Monitoring Working Gp and Net Management Working Group

- Status At Last Meeting
- GWMON: Making HEMS More Real
- NetMan: Defining Management Interface
- GWMON + NetMan: Trying To Consolidate Efforts
- Upcoming Plenary

Status At Last Meeting

- GWMON Group: Initial HEMS RFCs were finished at Last Meeting and Implementations Were Being Planned.
- NetMan Group: Decided Not To Try to Roll Their Own Protocol But Instead To Devise A Network Management Interface Which Could Be Used With Any Protocol (CMIP and HEMS, etc).

HEMS In Brief

- A Query-Response Protocol.
- To Get Information An Application Sends A "Database" Query To A Remote Agent.
- This Query Is Processed And The Results Sent Back.
- Database Is An Abstract Representation Of Device (Values In Database Are Abstractions Of Features Of Device).
- Writing Database = Control Operations (Side-Effects OK). Reading Database = Monitoring.

GWMON: Making HEMS More Real (Filling In Holes)

- Port Number Assigned: #151 (TCP and UDP)
- RFCs Issued: RFCs 1021-1024.
- Article To Be Published In *IEEE Network*
- Presentations To Various Groups (NBS Standards Meeting, etc).

GWMON: Making HEMS More Real (Implementations)

- At Least Two Underway. Mine Is A 4.3BSD Version.
- Initial Experiments With 4.3BSD Version Suggest That It Is Very Efficient. (<< 1 second to dump routing table).
- License In the Works For A Free Distribution.
- A Few Small Wording Problems In RFCs, But No Major Problems.

NetMan: Defining Management Interface

- A Couple Of RFCs Developed
- One Tries To Define Scope Of Effort
- Other Attempts To Define The Common Management Interface

GWMON + NetMan: Trying To Consolidate Efforts

- Concerns About Whether NetMan Interface Can Be Mapped Into HEMS.
- NetMan Interface Heavily Influenced by CMIS/CMIP. HEMS Has Some Features That CMIS/CMIP Doesn't. Should Interface Expand?
- CMIS/CMIP Has Features That HEMS Would Treat As No-ops (Such As Negotiation Of Facilities). Some NetMan Members Disturbed About No-ops.

Upcoming Plenary

- At Interoperability Conference (On Tutorial Day). Felt To Be Best Place For High Visibility With Vendors.
- Presentations By Various Groups (Expect GWMON, NETMAN and SGMP).
- Possible Demos.

SGMP Status Report and Demonstration

Jeff Case, University of Tennessee, Knoxville

SGMP

(SIMPLE GATEWAY MONITORING PROTOCOL)

STATUS REPORT

Jeffrey D. Case

University of Tennessee

Computing Center and

Department of Computer Science

case@utkcs2.cs.utk.edu

2.

RFC Authors:

Chuck Davin
Proteon, Incorporated

Jeff Case
University of Tennessee at Knoxville

Mark Fedor
Cornell University/NYSERnet

Martin Schoffstall
Rensselaer Polytechnic Institute

3.

OUTLINE

Philosophy in Brief

Protocol in Brief

Implementation/Deployment Status

Description of Demonstration of NETMON:
An SGMP Application

Future Directions

What We've Learned

For Further Information

Questions

4.

PHILOSOPHY IN BRIEF

- Simplicity
- Driven by the need for rapid deployment
- Place demands on NOCs, not gateways
- Sensitivity to gateway implementations (performance, memory size)
- Extendible protocol variable space for:
  - more variables
  - vendor specific variables
  - hosts
  - unanticipated needs
- Implementation variability

5.

THE SGMP IN BRIEF
SUMMARY

UDP-based -- adaptable to other transports

Retrieval of individual variables by name

Limited number of unsolicited trap messages

ASN.1 subset data representation (integers and octet strings only)

Creative approach to protocol variable space management problem

---

6.

THE SGMP IN BRIEF
LAYERING

SGMP session	SGMP session
authentication	
UDP	
IP	

7.

THE SGMP IN BRIEF
SERVICES

Get request/get response

Set request/set response

A few traps
- boot
- link failure
- authentication
- EGP neighbor loss

Multiple sessions

Hooks for authentication

8.

THE SGMP IN BRIEF
PROTOCOL VARIABLE SPACE MANAGEMENT PROBLEM

Variable Space is conceptually a tree with named edges

Variables are at the leaves of the tree

Name for an individual variable is the concatenation of edge names along the path from the root to the leaf

For a given node of the tree, its edges are ordered lexicographically from left to right according to name

9.

THE SGMP IN BRIEF
VARIABLE NAMING CONVENTIONS

Symbolic representation of variable names -- used by humans

Numerical representation of variable names -- used on the network

Example:

The variable whose name is represented symbolically as "GW_version_rev" might be represented numerically as

01 01 02

10.

IMPLEMENTATION/DEPLOYMENT STATUS

Known implementations underway:

4 gateway/IS/host implementations (SGMP server)

2 Network Operations Center (NOC) implementation efforts (SGMP client)

Known development groups:

Proteon

Cornell/NYSERnet

Carnegie Mellon University

Rensselaer Polytechnic Institute

University of Tennessee at Knoxville

11.

IMPLEMENTATION/DEPLOYMENT STATUS

Proteon

Proteon p4200 gateway implementation working in Version 7.4 -- now in Beta test

Cornell/NYSERnet

Gateway implementation in gated gateway daemon by Mark Fedor

Carnegie Mellon University

Kinetics Gateway

May add to CMU router code

12.

IMPLEMENTATION/DEPLOYMENT STATUS
Rensselaer Polytechnic Institute

Version 1 completed including manual pages:

Library with ASN.1 parsing and generation

Six applications:

SGMPASK
SGMPLOOKUP
SGMPQUERY
SGMPROUTE
SGMPTRAPD
SGMPWATCH

13.

IMPLEMENTATION/DEPLOYMENT STATUS
Rensselaer Polytechnic Institute (cont'd)

Current efforts:

Enhance the code and improve portability

More applications

SGMPXPERFMON

monitors the traffic on every interface of a single gateway

has adaptive learning mode to learn about gateway

in/out bytes and packets

SGMPCONFIG

determine the configuration of a given gateway

SGMPLOOP

walk a pair of network addresses and look for routing loops

14.

IMPLEMENTATION/DEPLOYMENT STATUS
University of Tennessee at Knoxville

Library with ASN.1 parsing and generation

client half

- 4.3 BSD
- ACIS 4.2 (IBM PC/RT)
- Ultrix 1.2 & 2.0
- MSDOS MIT/CMU/etc

server half

- 4.3 BSD
- Ultrix 1.2 & 2.0

Applications

- SGMP QUERY
- GETVAR
- GETMANY
- NETMON -- being demonstrated at IETF

15.

DESCRIPTION OF DEMONSTRATION OF NETMON:
AN SGMP APPLICATION

Purpose:

real-time graphical display of
network node and link reachability
status

illustrate/test/evaluate SGMP

Networks:

geographical map of SURAnet
logical map of NYSERnet
live data

Platform:

IBM PC with EGA, MIT/CMU PC/IP

Features:

node and link states shown in one of six colors
alarms
event logging to disk

Condition:

please provide feedback to help
guide further development to make
it most useful in your environment

Thanks:

to Don Morris for providing facilities

16.

FUTURE DIRECTIONS

1. NETMON Evolution

integration with other NOC tools

real time traffic monitoring

finer diagnostics
  additional data from gateways
    - interface status
    - neighbor status
  static information, such as:
    -contacts
    -circuit numbers
    -phone numbers

2. Additional Tools

dynamically "learn" the topology --
perhaps tied to NETMON

wire walker -- broken ping
chaser / loop detector

search for the source of a route

generate traffic reports

others

3. Port to Additional Platforms

X-windows
Sun
uVAX2000
VMS

17.

FUTURE DIRECTIONS (cont'd)

4. Distributed Architecture

   SGMP data collector/event logger

   multiple SGMP applications displays communicating with SGMP collector/logger

5. Additional Servers

   expand host instrumentation

   extend to additional host/OS configurations

6. Additional Work On The Protocol Itself

   implementing what has been defined but not yet implemented

   management (SETS)

   real authentication

   more variables

18.

WHAT WE'VE LEARNED

ASN.1/X.409 parsing is not impossible

ASN.1/X.409 constructs that pertain to multiple protocol layers are difficult to implement

Easily extensible protocols are easier to specify and standardize than those that are not

Geographically true maps do not work

19.

FOR FURTHER INFORMATION

Mailing list:

simple-umon@nic.nyser.net

(simple-umon-request@nic.nyser.net for admin)

Proposed RFC available via anonymous FTP at:

nic.nyser.net in pub/simple-mon.rfc

(copy at sri-nic.arpa in <IETF> is no longer accurate)

NSFnet Status Report

Steve Wolff, NSF

NSFNET

If congestion & delay are the price of success . . .

Routing is a problem; STRWG recommendations to be implemented.

We need traffic analysis

NCSA ~ 20-30%

PSC ~ 40%

New networks —

* MIDNET

* BITNET

SDSC retermination

New ultra-continental connections -

* Mexico
* Chile
* Brazil
* Puerto Rico
* France
* England
* Germany

From elias@tcgould.TN.CORNELL.EDU Wed Nov 4 11:16:28 1987
Received: from SCDSW1.UCAR.EDU by WINDOM.UCAR.EDU (3.2/4.7) id AA06894; Wed, 4 No
Received: from tcgould.TN.CORNELL.EDU by SCDSW1.UCAR.EDU (2.0/4.7) id AA03808; We
Date: Wed, 4 Nov 87 13:12:28 EST
From: elias@tcgould.tn.cornell.edu (Doug Elias)
Received: by tcgould.TN.CORNELL.EDU (5.54/1.2-Cornell-Theory-Center)
id AA16291; Wed, 4 Nov 87 13:12:28 EST
Message-Id: <8711041812.AA16291@tcgould.TN.CORNELL.EDU>
To: morris@scdsw1.ucar.edu
Subject: Backbone Traffic Reports
Status: R

i'm sending you 6 weeks worth of data, Sep thru Oct, plus the summary data for both months. i'm missing the last week of Sep, and the last few days of Oct (but not from the summary data).

The pkts going out to the local area nets is found in the "DQ0-output" columns:

NSFNET TRAFFIC REPORT Period: 9/7 - 9/13, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	23202879	16597819
Output	23529085	16717052
In+Out	46731964	33314871

Grand 80046835

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	4.57	3.63		8.20
JvNC	2.80	4.65		7.45
Ether	7.48	6.72		14.20
Totals	14.85	15.00		
			%SITE	29.85
Cornell				
NCAR	1.98	1.86		3.83
JvNC	2.46	1.68		4.15
SURA	1.36	1.13		2.49
Ether	0.97	2.24		3.21
Totals	6.77	6.91		
			%SITE	13.68
JvNC				
Cornell	1.64	2.52		4.16
PSC	4.61	2.85		7.46
Ether	4.81	5.79		10.60
Totals	11.06	11.16		
			%SITE	22.22
NCAR				
Cornell	1.81	2.03		3.84
UIUC	1.52	1.54		3.06

SDSC	0.20	0.27		0.47
Ether	2.95	2.92		5.88
Totals	6.49	6.76		
			%SITE	13.25

SDSC				
NCAR	0.22	0.25		0.47
UIUC	0.36	0.44		0.80
Ether	0.51	0.55		1.06
Totals	1.09	1.24		
			%SITE	2.33

UIUC				
NCAR	1.49	1.56		3.05
SDSC	0.39	0.40		0.79
PSC	3.57	4.59		8.16
Ether	4.01	2.67		6.68
Totals	9.46	9.21		
			%SITE	18.67

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		3657154	55.72	2906773	44.28	6563927	27.47
JvNC		2244161	37.61	3723196	62.39	5967357	24.97
DQ0		5988558	52.69	5377118	47.31	11365676	47.56
Subtotal		11889873		12007087			
		%site	49.75	%site	50.25		
Total	23896960	%Grand	29.85				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		1582363	51.55	1487136	48.45	3069499	28.04
JvNC		1972891	59.41	1347958	40.59	3320849	30.34
SURA		1087050	54.60	903874	45.40	1990924	18.19
DQ0		773551	30.15	1792386	69.85	2565937	23.44
Subtotal		5415855		5531354			
		%site	49.47	%site	50.53		
Total	10947209	%Grand	13.68				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		1313079	39.46	2014317	60.54	3327396	18.71
PSC		3687371	61.74	2284710	38.26	5972081	33.58
DQ0		3852931	45.40	4634055	54.60	8486986	47.72
Subtotal		8853381		8933082			
		%site	49.78	%site	50.22		
Total	17786463	%Grand	22.22				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		1451234	47.22	1621956	52.78	3073190	28.98
UIUC		1217206	49.69	1232241	50.31	2449447	23.09
SDSC		162234	42.85	216414	57.15	378648	3.57

DQ0		2364513	50.26	2340478	49.74	4704991	44.36
Subtotal		5195187		5411089			
		%site	48.98	%site	51.02		
Total	10606276	%Grand	13.25				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		177641	47.13	199286	52.87	376927	20.20
UIUC		287701	44.81	354309	55.19	642010	34.40
DQ0		409906	48.38	437341	51.62	847247	45.40
Subtotal		875248		990936			
		%site	46.90	%site	53.10		
Total	1866184	%Grand	2.33				

UIUC		input	%device	output	%device	subtotal	%site
NCAR		1192172	48.89	1246079	51.11	2438251	16.32
SDSC		312033	49.44	319050	50.56	631083	4.22
PSC		2858589	43.77	3671786	56.23	6530375	43.70
DQ0		3208360	60.04	2135674	39.96	5344034	35.76
Subtotal		7571154		7372589			
		%site	50.66	%site	49.34		
Total	14943743	%Grand	18.67				

NSFNET TRAFFIC REPORT Period: 9/14 - 9/20, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	24323148	17429083
Output	24884514	17341817
In+Out	49207662	34770900
Grand	83978562	

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	2.91	2.82		5.73
JvNC	3.59	6.26		9.85
Ether	8.11	5.70		13.82
Totals	14.60	14.79		
			%SITE	29.39
Cornell				
NCAR	2.02	1.93		3.96
JvNC	2.51	1.95		4.46
SURA	0.77	0.85		1.62
Ether	1.01	1.71		2.72
Totals	6.32	6.43		
			%SITE	12.75
JvNC				

Cornell	1.93	2.59		4.52
PSC	6.21	3.63		9.85
Ether	5.66	7.70		13.36
Totals	13.81	13.92		
			%SITE	27.73
NCAR				
Cornell	1.83	2.00		3.83
UIUC	1.57	1.46		3.03
SDSC	0.23	0.46		0.69
Ether	2.84	2.83		5.67
Totals	6.47	6.75		
			%SITE	13.22
SDSC				
NCAR	0.43	0.28		0.71
UIUC	0.35	0.41		0.76
Ether	0.49	0.70		1.18
Totals	1.26	1.39		
			%SITE	2.65
UIUC				
NCAR	1.46	1.65		3.11
SDSC	0.37	0.39		0.76
PSC	2.78	2.95		5.73
Ether	2.64	2.01		4.66
Totals	7.26	7.00		
			%SITE	14.26

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		2440634	50.72	2371762	49.28	4812396	19.50
JvNC		3010920	36.42	5256939	63.58	8267859	33.50
DQ0		6812886	58.72	4789040	41.28	11601926	47.01
Subtotal		12264440		12417741			
		%site	49.69	%site	50.31		
Total	24682181	%Grand	29.39				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		1699321	51.16	1622188	48.84	3321509	31.02
JvNC		2108421	56.31	1636220	43.69	3744641	34.97
SURA		648612	47.75	709668	52.25	1358280	12.68
DQ0		848909	37.17	1434814	62.83	2283723	21.33
Subtotal		5305263		5402890			
		%site	49.54	%site	50.46		
Total	10708153	%Grand	12.75				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		1621190	42.74	2171718	57.26	3792908	16.29
PSC		5218799	63.10	3051553	36.90	8270352	35.52
DQ0		4755058	42.37	6467943	57.63	11223001	48.20
Subtotal		11595047		11691214			

		%site	49.79	%site	50.21	
Total	23286261	%Grand	27.73			

NCAR	input	%device	output	%device	subtotal	%site
Cornell	1539729	47.85	1678046	52.15	3217775	28.98
UIUC	1317641	51.76	1228261	48.24	2545902	22.93
SDSC	194668	33.38	388463	66.62	583131	5.25
DQ0	2384028	50.11	2373444	49.89	4757472	42.84
Subtotal	5436066		5668214			
	%site	48.95	%site	51.05		

Total 11104280 %Grand 13.22

SDSC	input	%device	output	%device	subtotal	%site
NCAR	357787	60.40	234591	39.60	592378	26.63
UIUC	291685	45.64	347446	54.36	639131	28.73
DQ0	407538	41.03	585840	58.97	993378	44.65
Subtotal	1057010		1167877			
	%site	47.51	%site	52.49		

Total 2224887 %Grand 2.65

UIUC	input	%device	output	%device	subtotal	%site
NCAR	1225807	46.94	1385745	53.06	2611552	21.81
SDSC	311099	48.67	328167	51.33	639266	5.34
PSC	2336835	48.58	2473747	51.42	4810582	40.18
DQ0	2220664	56.77	1690736	43.23	3911400	32.67
Subtotal	6094405		5878395			
	%site	50.90	%site	49.10		

Total 11972800 %Grand 14.26

NSFNET TRAFFIC REPORT Period: 9/21 - 9/27, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	24519953	17816680
Output	25241774	17420398
In+Out	49761727	35237078

Grand 84998805

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	2.41	2.35		4.76
JvNC	2.65	6.50		9.15
Ether	7.68	4.01		11.69
Totals	12.73	12.86		
			%SITE	25.60

Cornell

NCAR				
JvNC	3.83	2.50		6.33
SURA	2.02	2.58		4.60
Ether	0.99	1.43		2.42
Totals	1.33	1.66		2.99
	8.18	8.17		
			%SITE	16.35
JvNC				
Cornell	2.54	2.07		4.60
PSC	6.49	2.72		9.21
Ether	5.14	9.43		14.57
Totals	14.16	14.22		
			%SITE	28.38
NCAR				
Cornell	1.65	2.91		4.56
UIUC	1.44	1.44		2.88
SDSC	0.26	0.33		0.58
Ether	3.51	2.41		5.92
Totals	6.85	7.09		
			%SITE	13.94
SDSC				
NCAR	0.29	0.31		0.60
UIUC	0.29	0.31		0.60
Ether	0.54	0.63		1.18
Totals	1.13	1.25		
			%SITE	2.37
UIUC				
NCAR	0.91	0.98		1.89
SDSC	0.78	0.84		1.62
PSC	2.30	2.44		4.74
Ether	2.76	2.35		5.11
Totals	6.75	6.61		
			%SITE	13.36

Site PacketSummary

		input	%device	output	%device	subtotal	%site
PSC							
UIUC		2045737	50.58	1998848	49.42	4044585	18.59
JvNC		2253597	28.98	5521884	71.02	7775481	35.74
DQ0		6525262	65.67	3411103	34.33	9936365	45.67
Subtotal		10824596		10931835			
		%site	49.75	%site	50.25		
Total	21756431	%Grand	25.60				

	input	%device	output	%device	subtotal	%site
Cornell						
NCAR	3256780	60.52	2124564	39.48	5381344	38.73
JvNC	1717597	43.93	2192542	56.07	3910139	28.14
SURA	843020	40.93	1216804	59.07	2059824	14.83
DQ0	1134236	44.62	1407543	55.38	2541779	18.30
Subtotal	6951633		6941453			
	%site	50.04	%site	49.96		

Total	13893086	%Grand	16.35

JvNC	input	%device	output	%device	subtotal	%site
Cornell	2157732	55.14	1755381	44.86	3913113	16.22
PSC	5514216	70.46	2311509	29.54	7825725	32.44
DQ0	4367230	35.26	8019343	64.74	12386573	51.34
Subtotal	12039178		12086233			
	%site	49.90	%site	50.10		

Total	24125411	%Grand	28.38

NCAR	input	%device	output	%device	subtotal	%site
Cornell	1402644	36.22	2469899	63.78	3872543	32.68
UIUC	1221758	49.87	1227960	50.13	2449718	20.67
SDSC	218665	44.02	278106	55.98	496771	4.19
DQ0	2983138	59.31	2046781	40.69	5029919	42.45
Subtotal	5826205		6022746			
	%site	49.17	%site	50.83		

Total	11848951	%Grand	13.94

SDSC	input	%device	output	%device	subtotal	%site
NCAR	247415	48.80	259550	51.20	506965	25.11
UIUC	248352	48.48	263960	51.52	512312	25.38
DQ0	462191	46.25	537109	53.75	999300	49.51
Subtotal	957958		1060619			
	%site	47.46	%site	52.54		

Total	2018577	%Grand	2.37

UIUC	input	%device	output	%device	subtotal	%site
NCAR	770767	48.05	833480	51.95	1604247	14.13
SDSC	666685	48.34	712448	51.66	1379133	12.14
PSC	1954988	48.51	2074839	51.49	4029827	35.49
DQ0	2344623	53.98	1998519	46.02	4343142	38.24
Subtotal	5737063		5619286			
	%site	50.52	%site	49.48		

Total	11356349	%Grand	13.36

NSFNET TRAFFIC REPORT Period: Sept., 1987

Total Traffic Figures

	Between Sites	Ethernet
Input	102868099	71756593
Output	102197912	70629065
In+Out	205066011	142385658

Grand 347451669

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	3.43	2.91		6.33
JvNC	2.88	5.19		8.07
Ether	7.17	5.41		12.58
Totals	13.48	13.51		
			%SITE	26.98
Cornell				
NCAR	2.59	2.11		4.70
JvNC	2.49	2.23		4.73
SURA	1.28	1.08		2.35
Ether	1.24	2.10		3.34
Totals	7.60	7.52		
			%SITE	15.12
JvNC				
Cornell	2.25	2.56		4.81
PSC	5.20	2.89		8.09
Ether	5.10	7.08		12.18
Totals	12.55	12.52		
			%SITE	25.07
NCAR				
Cornell	1.87	2.31		4.19
UIUC	1.57	1.65		3.21
SDSC	0.28	0.36		0.64
Ether	3.42	2.75		6.17
Totals	7.14	7.07		
			%SITE	14.20
SDSC				
NCAR	0.37	0.29		0.66
UIUC	0.40	0.42		0.82
Ether	0.57	0.66		1.22
Totals	1.33	1.37		
			%SITE	2.70
UIUC				
NCAR	1.66	1.57		3.23
SDSC	0.44	0.42		0.86
PSC	2.90	3.42		6.33
Ether	3.15	2.34		5.49
Totals	8.16	7.76		
			%SITE	15.91

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		11909414	54.11	10100897	45.89	22010311	23.48
JvNC		10010625	35.70	18033243	64.30	28043868	29.91
DQ0		24908428	56.99	18795196	43.01	43703624	46.61
Subtotal		46828467		46929336			
		%site	49.95	%site	50.05		
Total	93757803	%Grand	26.98				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		9009203	55.16	7323115	44.84	16332318	31.09
JvNC		8655155	52.72	7761984	47.28	16417139	31.25
SURA		4432988	54.25	3737751	45.75	8170739	15.55
DQ0		4319083	37.20	7292474	62.80	11611557	22.10
Subtotal		26416429		26115324			
		%site	50.29	%site	49.71		
Total	52531753	%Grand	15.12				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		7805052	46.74	8894058	53.26	16699110	19.17
PSC		18069977	64.29	10036342	35.71	28106319	32.26
DQ0		17731701	41.90	24582477	58.10	42314178	48.57
Subtotal		43606730		43512877			
		%site	50.05	%site	49.95		
Total	87119607	%Grand	25.07				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		6503851	44.72	8040320	55.28	14544171	29.47
UIUC		5438526	48.74	5720528	51.26	11159054	22.61
SDSC		984680	44.32	1237214	55.68	2221894	4.50
DQ0		11866532	55.38	9559357	44.62	21425889	43.42
Subtotal		24793589		24557419			
		%site	50.24	%site	49.76		
Total	49351008	%Grand	14.20				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		1281932	55.62	1022926	44.38	2304858	24.53
UIUC		1372764	48.41	1463198	51.59	2835962	30.18
DQ0		1978691	46.50	2276736	53.50	4255427	45.29
Subtotal		4633387		4762860			
		%site	49.31	%site	50.69		
Total	9396247	%Grand	2.70				

UIUC		input	%device	output	%device	subtotal	%site
NCAR		5763555	51.32	5467760	48.68	11231315	20.31
SDSC		1542619	51.34	1462127	48.66	3004746	5.43
PSC		10087758	45.89	11896449	54.11	21984207	39.76
DQ0		10952158	57.42	8122825	42.58	19074983	34.50
Subtotal		28346090		26949161			
		%site	51.26	%site	48.74		
Total	55295251	%Grand	15.91				

NSFNET TRAFFIC REPORT Period: Oct 5 - 11, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	12917851	8653704
Output	12753228	8062881
In+Out	25671079	16716585

Grand 42387664

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	2.82	3.31		6.13
JvNC	4.22	6.22		10.44
Ether	7.96	5.57		13.53
Totals	15.00	15.10		
			%SITE	30.10
Cornell				
NCAR	3.39	2.55		5.95
JvNC	2.91	2.11		5.02
SURA	0.97	2.07		3.04
Ether	1.71	2.10		3.81
Totals	8.99	8.83		
			%SITE	17.81
JvNC				
Cornell	2.11	2.91		5.02
PSC	6.22	4.17		10.39
Ether	5.73	6.76		12.50
Totals	14.06	13.85		
			%SITE	27.91
NCAR				
Cornell	0.94	1.32		2.26
UIUC	0.82	0.57		1.39
SDSC	0.17	0.18		0.34
Ether	1.43	1.28		2.71
Totals	3.35	3.35		
			%SITE	6.71
SDSC				
NCAR	0.39	0.38		0.77
UIUC	0.82	0.48		1.30
Ether	0.54	0.58		1.12
Totals	1.75	1.43		
			%SITE	3.19
UIUC				
NCAR	1.14	0.52		1.66
SDSC	0.97	1.10		2.07
PSC	2.58	2.20		4.78
Ether	3.04	2.73		5.77
Totals	7.73	6.55		
			%SITE	14.29

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		1195878	46.02	1402514	53.98	2598392	20.37
JvNC		1789083	40.43	2635867	59.57	4424950	34.68
DQ0		3373818	58.83	2360970	41.17	5734788	44.95
Subtotal		6358779		6399351			
		%site	49.84	%site	50.16		
Total	12758130	%Grand	30.10				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		1438168	57.06	1082305	42.94	2520473	33.38
JvNC		1233988	57.98	894354	42.02	2128342	28.19
SURA		411346	31.94	876460	68.06	1287806	17.06
DQ0		725863	44.97	888083	55.03	1613946	21.38
Subtotal		3809365		3741202			
		%site	50.45	%site	49.55		
Total	7550567	%Grand	17.81				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		894221	42.02	1234113	57.98	2128334	17.99
PSC		2635524	59.83	1769500	40.17	4405024	37.23
DQ0		2430195	45.88	2867021	54.12	5297216	44.78
Subtotal		5959940		5870634			
		%site	50.38	%site	49.62		
Total	11830574	%Grand	27.91				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		399227	41.61	560199	58.39	959426	33.76
UIUC		347961	58.92	242620	41.08	590581	20.78
SDSC		70079	48.53	74320	51.47	144399	5.08
DQ0		604421	52.66	543397	47.34	1147818	40.38
Subtotal		1421688		1420536			
		%site	50.02	%site	49.98		
Total	2842224	%Grand	6.71				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		166585	51.16	159026	48.84	325611	24.10
UIUC		346561	63.06	203012	36.94	549573	40.68
DQ0		230636	48.46	245252	51.54	475888	35.22
Subtotal		743782		607290			
		%site	55.05	%site	44.95		
Total	1351072	%Grand	3.19				

UIUC		input	%device	output	%device	subtotal	%site
NCAR		483272	68.53	221937	31.47	705209	11.65
SDSC		412757	47.05	464468	52.95	877225	14.49
PSC		1093201	53.97	932533	46.03	2025734	33.46
DQ0		1288771	52.67	1158158	47.33	2446929	40.41
Subtotal		3278001		2777096			

		%site	54.14	%site	45.86
Total	6055097	%Grand	14.29		

NSFNET TRAFFIC REPORT Period: Oct 12 - 18, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	25930052	13526069
Output	27916281	12872957
In+Out	53846333	26399026

Grand 80245359

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	2.12	1.73		3.85
JvNC	2.55	4.40		6.95
Ether	4.67	2.60		7.27
Totals	9.35	8.73		
			%SITE	18.07
Cornell				
NCAR	4.53	3.97		8.50
JvNC	4.31	3.82		8.13
SURA	2.74	3.43		6.17
Ether	1.24	1.57		2.80
Totals	12.81	12.79		
			%SITE	25.60
JvNC				
Cornell	3.85	4.34		8.20
PSC	4.70	2.76		7.47
Ether	5.70	7.07		12.77
Totals	14.26	14.17		
			%SITE	28.43
NCAR				
Cornell	2.29	2.54		4.83
UIUC	0.77	0.69		1.45
SDSC	0.24	0.25		0.49
Ether	2.47	2.39		4.86
Totals	5.77	5.86		
			%SITE	11.64
SDSC				
NCAR	0.38	0.40		0.79
UIUC	0.74	0.31		1.05
Ether	0.52	0.55		1.08
Totals	1.65	1.26		
			%SITE	2.91
UIUC				
NCAR	1.07	1.38		2.45

SDSC	0.29	3.32	3.61
PSC	1.71	1.46	3.17
Ether	2.26	1.87	4.12
Totals	5.33	8.02	
		%SITE	13.35

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		1704512	55.13	1387064	44.87	3091576	21.32
JvNC		2046959	36.69	3531690	63.31	5578649	38.47
DQ0		3747576	64.27	2083565	35.73	5831141	40.21
Subtotal		7499047		7002319			
		%site	51.71	%site	48.29		
Total	14501366	%Grand	18.07				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		3636903	53.32	3184298	46.68	6821201	33.21
JvNC		3454648	52.97	3067175	47.03	6521823	31.75
SURA		2197226	44.39	2752118	55.61	4949344	24.09
DQ0		992525	44.11	1257466	55.89	2249991	10.95
Subtotal		10281302		10261057			
		%site	50.05	%site	49.95		
Total	20542359	%Grand	25.60				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		3092284	47.01	3485571	52.99	6577855	28.83
PSC		3774971	63.00	2217125	37.00	5992096	26.26
DQ0		4575408	44.66	5669810	55.34	10245218	44.91
Subtotal		11442663		11372506			
		%site	50.15	%site	49.85		
Total	22815169	%Grand	28.43				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		1841085	47.48	2036670	52.52	3877755	41.53
UIUC		614908	52.74	551114	47.26	1166022	12.49
SDSC		196385	49.91	197100	50.09	393485	4.21
DQ0		1979704	50.76	1920232	49.24	3899936	41.77
Subtotal		4632082		4705116			
		%site	49.61	%site	50.39		
Total	9337198	%Grand	11.64				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		308004	48.85	322565	51.15	630569	26.99
UIUC		594561	70.69	246486	29.31	841047	36.00
DQ0		419731	48.54	444904	51.46	864635	37.01
Subtotal		1322296		1013955			
		%site	56.60	%site	43.40		

Total	2336251	%Grand	2.91

UIUC	input	%device	output	%device	subtotal	%site
NCAR	860211	43.77	1104916	56.23	1965127	18.34
SDSC	233466	8.06	2664452	91.94	2897918	27.05
PSC	1373929	54.05	1167937	45.95	2541866	23.73
DQ0	1811125	54.75	1496980	45.25	3308105	30.88
Subtotal	4278731		6434285			
	%site	39.94	%site	60.06		

Total	10713016	%Grand	13.35

NSFNET TRAFFIC REPORT Period: Oct 19 - 25, '87

Total Traffic Figures

	Between Sites	Ethernet
Input	31266376	20729098
Output	29942302	21212807
In+Out	61208678	41941905

Grand 103150583

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	0.81	1.73		2.53
JvNC	2.06	3.59		5.66
Ether	4.46	2.04		6.50
Totals	7.33	7.36		
			%SITE	14.69
Cornell				
NCAR	3.76	2.50		6.26
JvNC	2.37	2.22		4.59
SURA	1.73	2.69		4.41
Ether	1.49	1.93		3.41
Totals	9.35	9.33		
			%SITE	18.68
JvNC				
Cornell	5.32	5.72		11.04
PSC	3.71	2.11		5.82
Ether	8.69	11.57		20.26
Totals	17.73	19.39		
			%SITE	37.13
NCAR				
Cornell	2.41	3.61		6.02
UIUC	1.57	0.69		2.26
SDSC	0.44	0.44		0.88
Ether	3.22	3.01		6.23
Totals	7.64	7.75		
			%SITE	15.39

SDSC				
NCAR	0.46	0.46		0.93
UIUC	0.72	0.35		1.07
Ether	0.47	0.50		0.96
Totals	1.65	1.31		
			%SITE	2.96

UIUC				
NCAR	2.03	1.68		3.71
SDSC	1.12	0.37		1.49
PSC	1.78	0.87		2.66
Ether	1.77	1.52		3.29
Totals	6.71	4.45		
			%SITE	11.15

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		832257	31.83	1782461	68.17	2614718	17.25
JvNC		2129647	36.50	3705719	63.50	5835366	38.50
DQ0		4602385	68.64	2103029	31.36	6705414	44.24
Subtotal		7564289		7591209			
		%site	49.91	%site	50.09		
Total	15155498	%Grand	14.69				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		3879471	60.05	2581257	39.95	6460728	33.53
JvNC		2444595	51.68	2285984	48.32	4730579	24.55
SURA		1783434	39.16	2770380	60.84	4553814	23.64
DQ0		1535149	43.60	1985848	56.40	3520997	18.28
Subtotal		9642649		9623469			
		%site	50.05	%site	49.95		
Total	19266118	%Grand	18.68				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		5492760	48.23	5895152	51.77	11387912	29.74
PSC		3831821	63.78	2175970	36.22	6007791	15.69
DQ0		8968203	42.91	11933086	57.09	20901289	54.58
Subtotal		18292784		20004208			
		%site	47.77	%site	52.23		
Total	38296992	%Grand	37.13				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		2486395	40.04	3723692	59.96	6210087	39.13
UIUC		1620942	69.55	709727	30.45	2330669	14.69
SDSC		451747	50.01	451568	49.99	903315	5.69
DQ0		3318938	51.65	3107464	48.35	6426402	40.49
Subtotal		7878022		7992451			
		%site	49.64	%site	50.36		
Total	15870473	%Grand	15.39				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		477060	49.91	478746	50.09	955806	31.28
UIUC		743702	67.11	364535	32.89	1108237	36.27
DQ0		479811	48.41	511339	51.59	991150	32.44
Subtotal		1700573		1354620			
		%site	55.66	%site	44.34		
Total	3055193	%Grand	2.96				

UIUC		input	%device	output	%device	subtotal	%site
NCAR		2094486	54.69	1735144	45.31	3829630	33.28
SDSC		1158172	75.27	380561	24.73	1538733	13.37
PSC		1839887	67.12	901406	32.88	2741293	23.82
DQ0		1824612	53.72	1572041	46.28	3396653	29.52
Subtotal		6917157		4589152			
		%site	60.12	%site	39.88		
Total	11506309	%Grand	11.15				

NSFNET TRAFFIC REPORT Period: Oct., 1987

Total Traffic Figures

	Between Sites	Ethernet
Input	93861977	60959185
Output	93447006	57320761
In+Out	187308983	118279946

Grand 305588929

Site Traffic Percentages of Grand

	%INPUT	%OUTPUT		%LINK
PSC				
UIUC	1.39	2.14		3.52
JvNC	2.70	4.41		7.11
Ether	6.39	3.51		9.90
Totals	10.48	10.06		
			%SITE	20.54
Cornell				
NCAR	3.98	2.94		6.92
JvNC	3.05	2.69		5.74
SURA	2.04	3.01		5.04
Ether	1.43	1.81		3.24
Totals	10.50	10.45		
			%SITE	20.95
JvNC				
Cornell	3.75	4.20		7.94
PSC	4.43	2.71		7.13
Ether	6.43	8.19		14.62
Totals	14.60	15.09		
			%SITE	29.69

NCAR				
Cornell	2.21	3.06		5.27
UIUC	1.41	0.79		2.20
SDSC	0.37	0.35		0.72
Ether	2.75	2.63		5.38
Totals	6.75	6.82		
			%SITE	13.57
SDSC				
NCAR	0.43	0.46		0.89
UIUC	0.76	0.41		1.18
Ether	0.58	0.59		1.17
Totals	1.77	1.46		
			%SITE	3.23
UIUC				
NCAR	1.41	1.67		3.07
SDSC	0.76	0.44		1.20
PSC	2.02	1.31		3.34
Ether	2.37	2.03		4.40
Totals	6.57	5.45		
			%SITE	12.02

Site PacketSummary

PSC		input	%device	output	%device	subtotal	%site
UIUC		4240983	39.37	6530893	60.63	10771876	17.16
JvNC		8241272	37.92	13490709	62.08	21731981	34.63
DQ0		19536067	64.56	10722176	35.44	30258243	48.21
Subtotal		32018322		30743778			
		%site	51.02	%site	48.98		
Total	62762100	%Grand	20.54				

Cornell		input	%device	output	%device	subtotal	%site
NCAR		12160178	57.51	8984312	42.49	21144490	33.03
JvNC		9333521	53.17	8219244	46.83	17552765	27.42
SURA		6223473	40.38	9189775	59.62	15413248	24.08
DQ0		4354764	43.99	5543740	56.01	9898504	15.46
Subtotal		32071936		31937071			
		%site	50.11	%site	49.89		
Total	64009007	%Grand	20.95				

JvNC		input	%device	output	%device	subtotal	%site
Cornell		11456468	47.19	12820890	52.81	24277358	26.76
PSC		13525955	62.06	8270149	37.94	21796104	24.02
DQ0		19640072	43.97	25024857	56.03	44664929	49.22
Subtotal		44622495		46115896			
		%site	49.18	%site	50.82		
Total	90738391	%Grand	29.69				

NCAR		input	%device	output	%device	subtotal	%site
Cornell		6757847	41.94	9357019	58.06	16114866	38.86
UIUC		4308737	64.15	2407939	35.85	6716676	16.19
SDSC		1135303	51.62	1063935	48.38	2199238	5.30
DQ0		8416122	51.18	8027089	48.82	16443211	39.65
Subtotal		20618009		20855982			
		%site	49.71	%site	50.29		
Total	41473991	%Grand	13.57				

SDSC		input	%device	output	%device	subtotal	%site
NCAR		1314789	48.28	1408566	51.72	2723355	27.56
UIUC		2337656	65.10	1253077	34.90	3590733	36.34
DQ0		1762965	49.43	1803766	50.57	3566731	36.10
Subtotal		5415410		4465409			
		%site	54.81	%site	45.19		
Total	9880819	%Grand	3.23				

UIUC		input	%device	output	%device	subtotal	%site
NCAR		4300723	45.78	5094248	54.22	9394971	25.58
SDSC		2337656	63.49	1344466	36.51	3682122	10.03
PSC		6187416	60.67	4011784	39.33	10199200	27.77
DQ0		7249195	53.90	6199133	46.10	13448328	36.62
Subtotal		20074990		16649631			
		%site	54.66	%site	45.34		
Total	36724621	%Grand	12.02				

BBN Status Report

Bob Hinden, Marianne Gardner, BBN

Gateway Congestion Control Simulation (cont.)

- Model limitations
  - Subnet model is highly abstract.
  - Congestion in long-haul net not modeled.
  - Traffic from long-haul net to LAN hosts not modeled.
  - Non-TCP traffic not modeled.

INTERNET STATUS

Robert M. Hinden

BBN Communications Corporation

CURRENT INTERNET

- Current Internet

  ~ 313 Operational Networks
  ~ 720 Assigned Networks

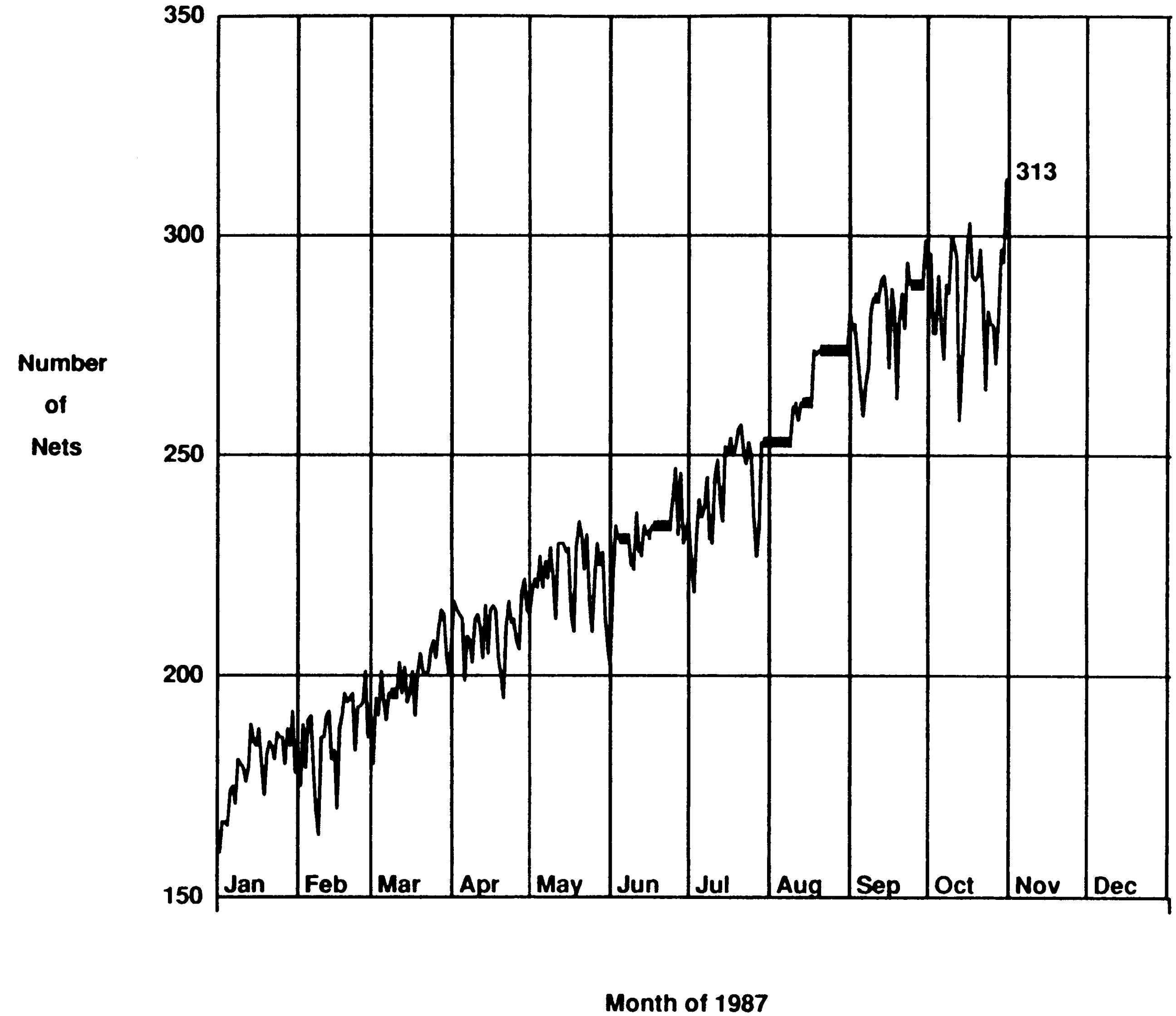
- LSI-11 Gateway

  - 30 Operational

- Butterfly Gateway

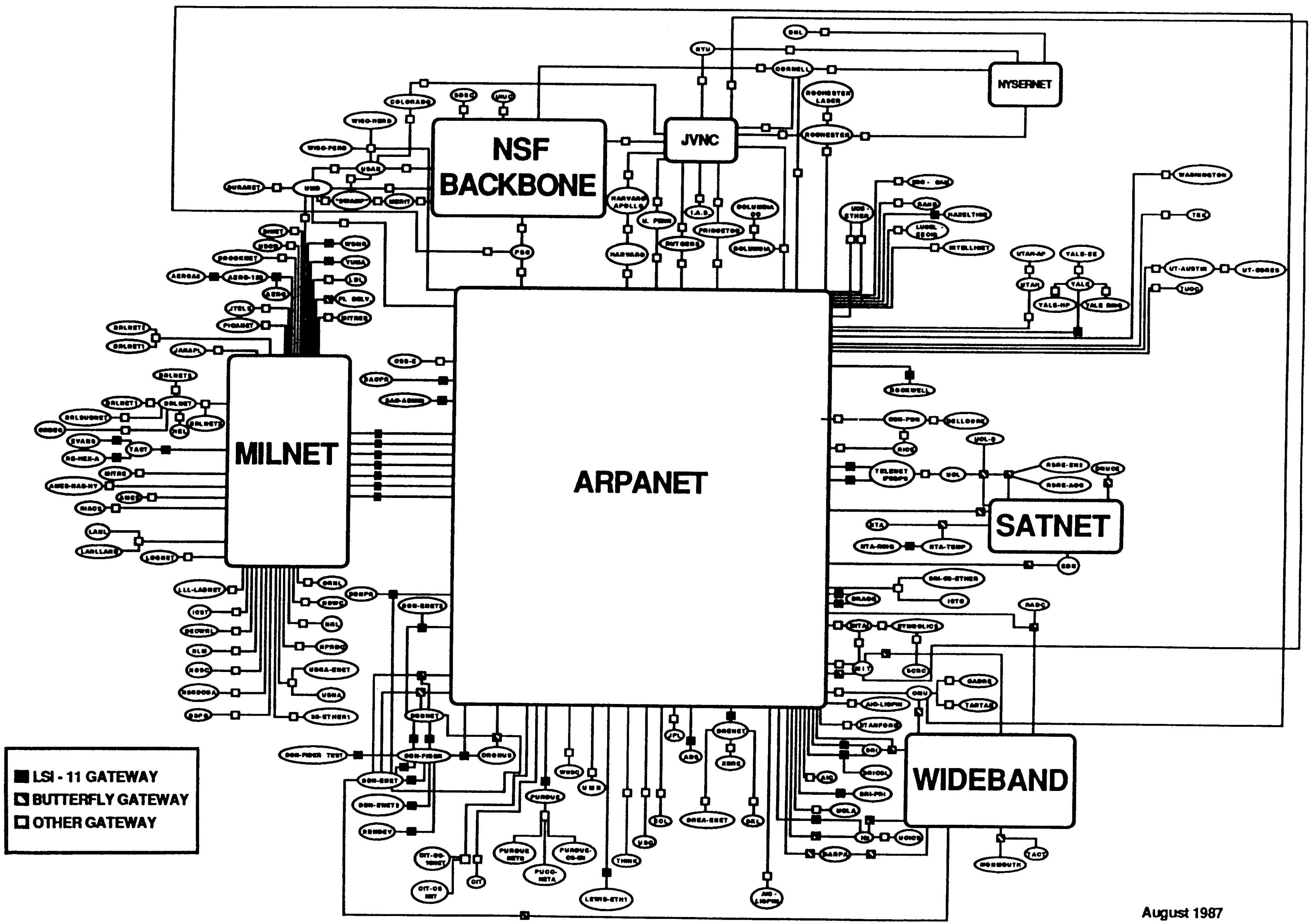
  - 25 Operational

OPERATIONAL NETWORKS



10/30/87

BBN Communications Corporation



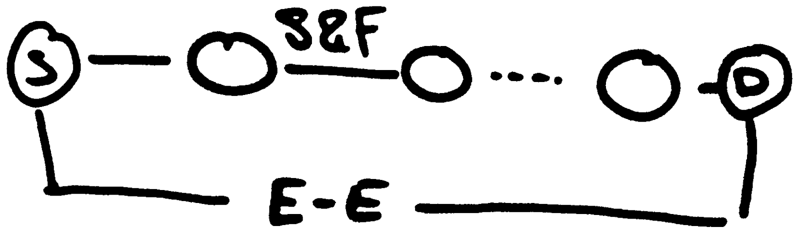
August 1987

BBN Communications Corporation

RECENT CHANGES

- PSN 7 Installation
- LSI-11 Gateways
  - Upgraded to 400 Networks
  - EGP/GGP Fragmentation/Reassembly
- Mailbridge Gateways
  - Upgraded to 400 Networks
  - GGP Fragmentation/Reassembly
- Butterfly Gateways
  - EGP Fragmentation/Reassembly
- TACACS User Database Host (UDH) Cutover

Release 7



E-E maintains connections
obtains reassembly resources
Reassembles

Rel.7 New E-E

- better compatibility with X.25
- piggy-backed RRs
- aggregated RRs

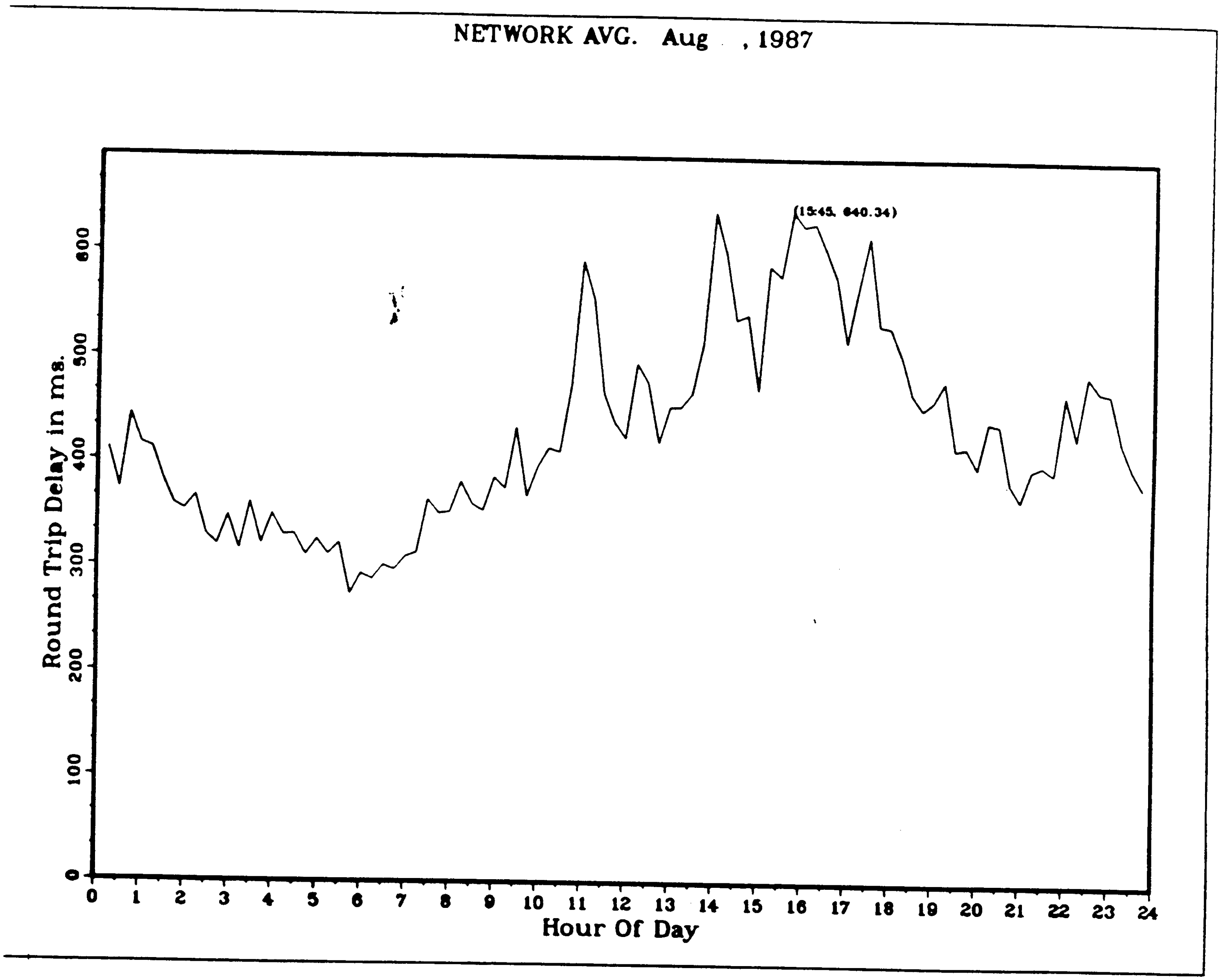
Has been running in BBN net

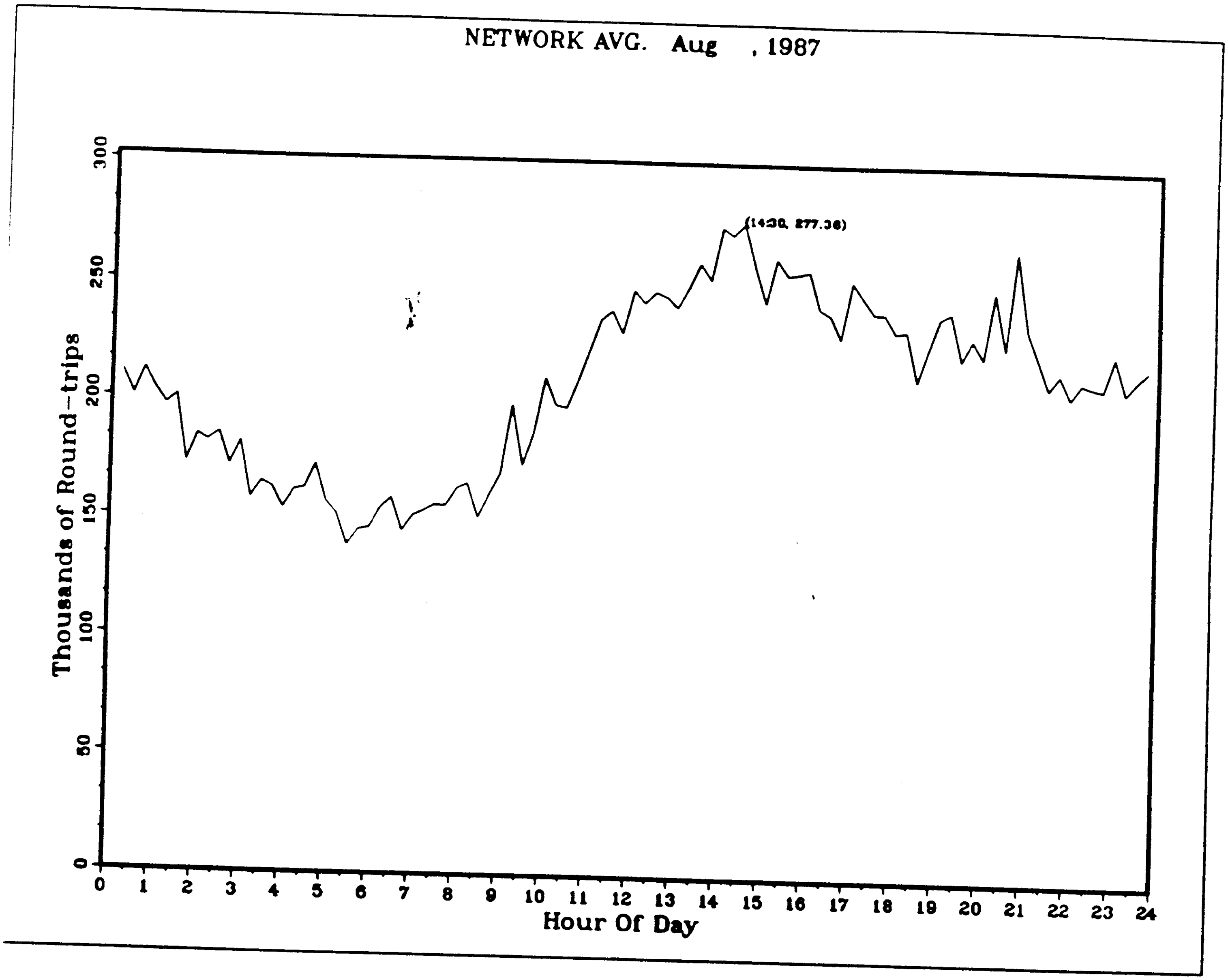
Installed in ARPAnet 10/17/87

week-end ~~tests~~ to new EE
new ~~E-E~~ & old E-E coresident
can ~~run~~ either one

ARPANET

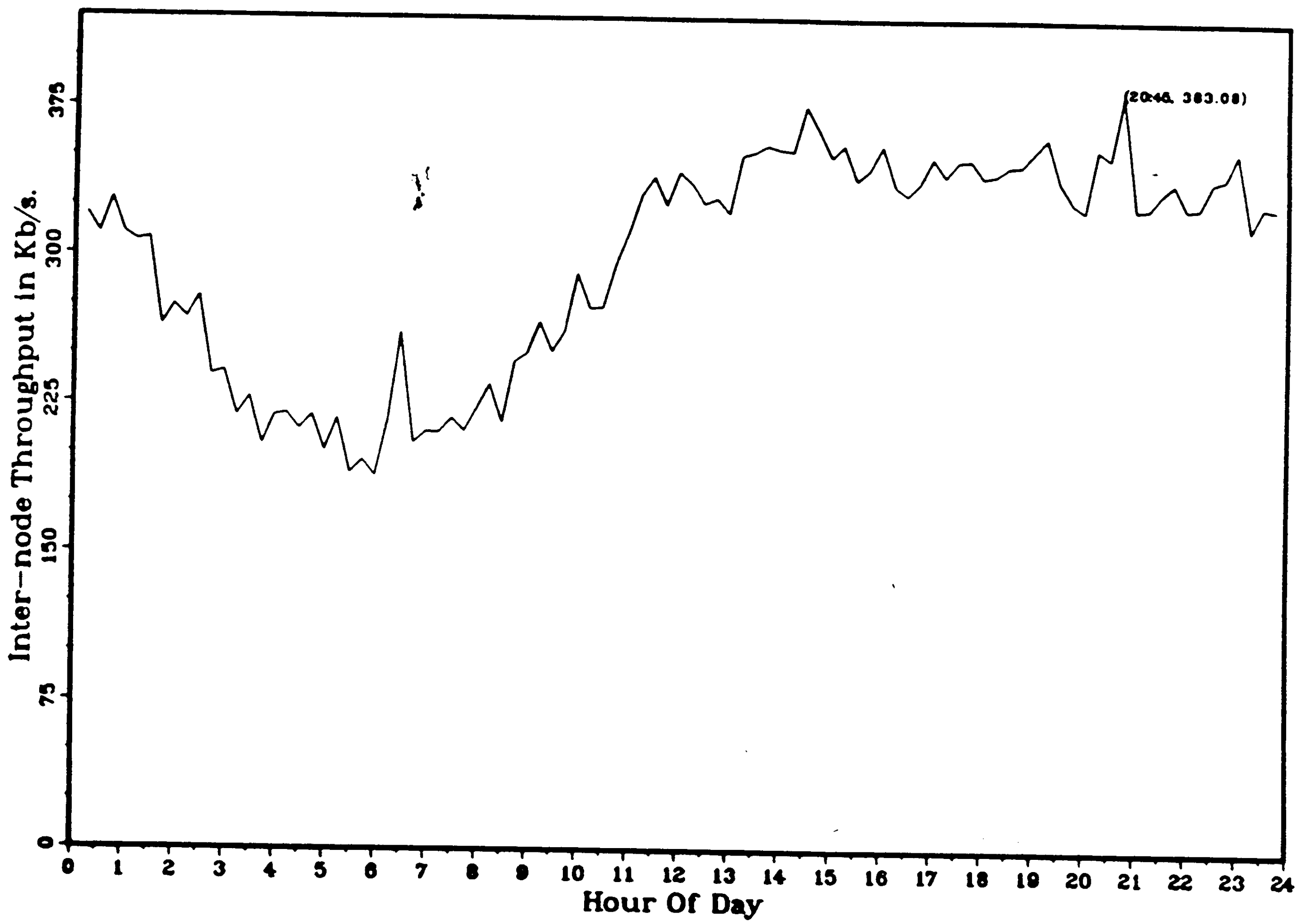
	May 87	Aug 87	Change (%)
Peak-hour			
Internode traffic (kb/s)	366	414	+13
Round-trip delay (ms)	635	339	−47
Internode Actual Path (hops/msg)	4.91	3.70	−25
Internode Minimum Path (hops/msg)	3.67	3.24	−12
Ratio (Actual to Minimum)	1.33	1.14	*
Routing updates per node per sec	.046	.038	−17
Week-long			
Internode traffic (kb/s)	262	300	+15
Round-trip delay (ms)	503	441	−12
Internode Actual Path (hops/msg)	5.37	4.09	−24
Internode Minimum Path (hops/msg)	3.96	3.39	−14
Ratio (Actual to Minimum)	1.36	1.21	**
Routng updates per node per sec	.036	.032	−11


NETWORK AVG. Aug , 1987
(15:45, 640.34)
Round Trip Delay in ms.
0
100
200
300
400
500
600
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour Of Day

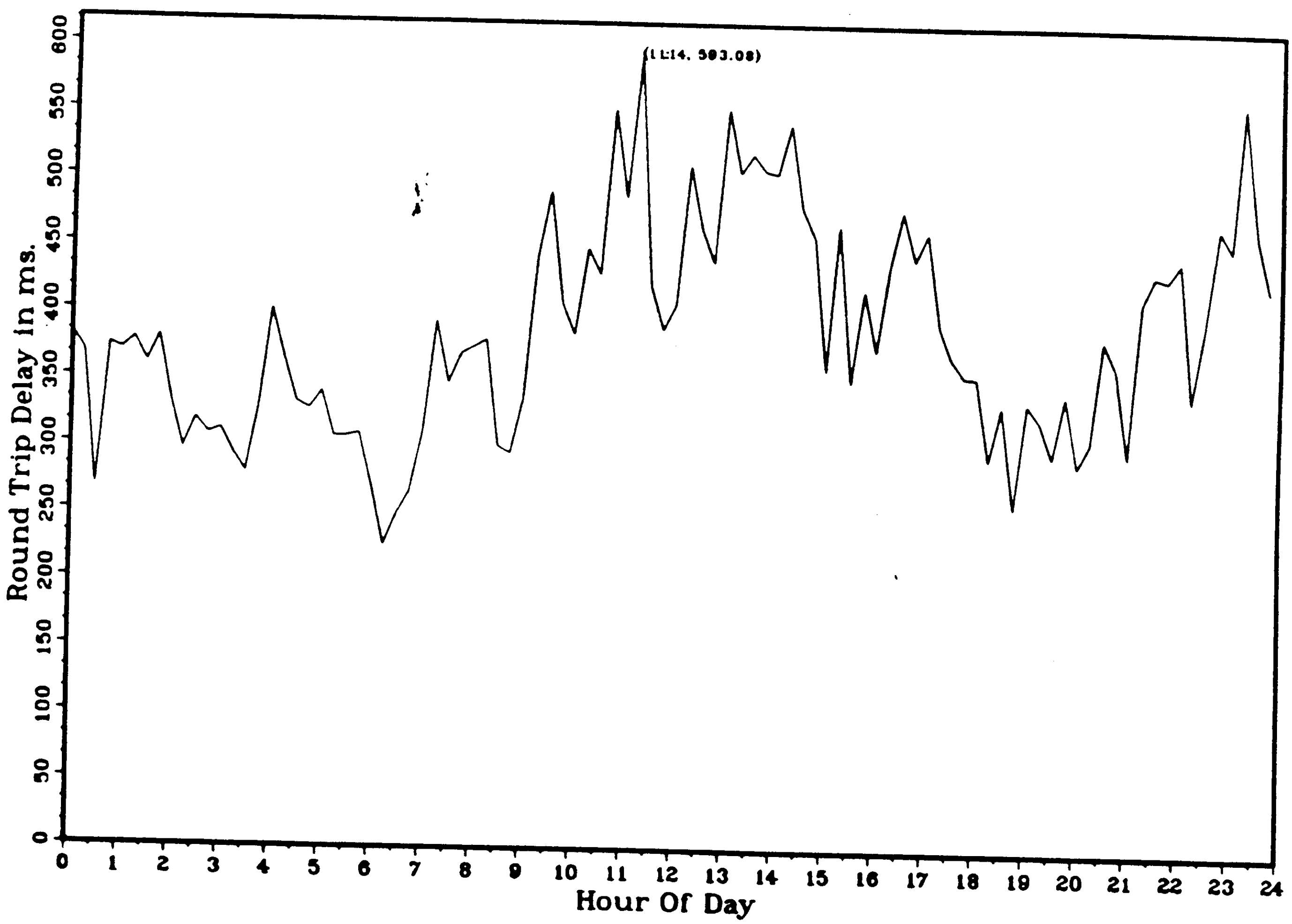


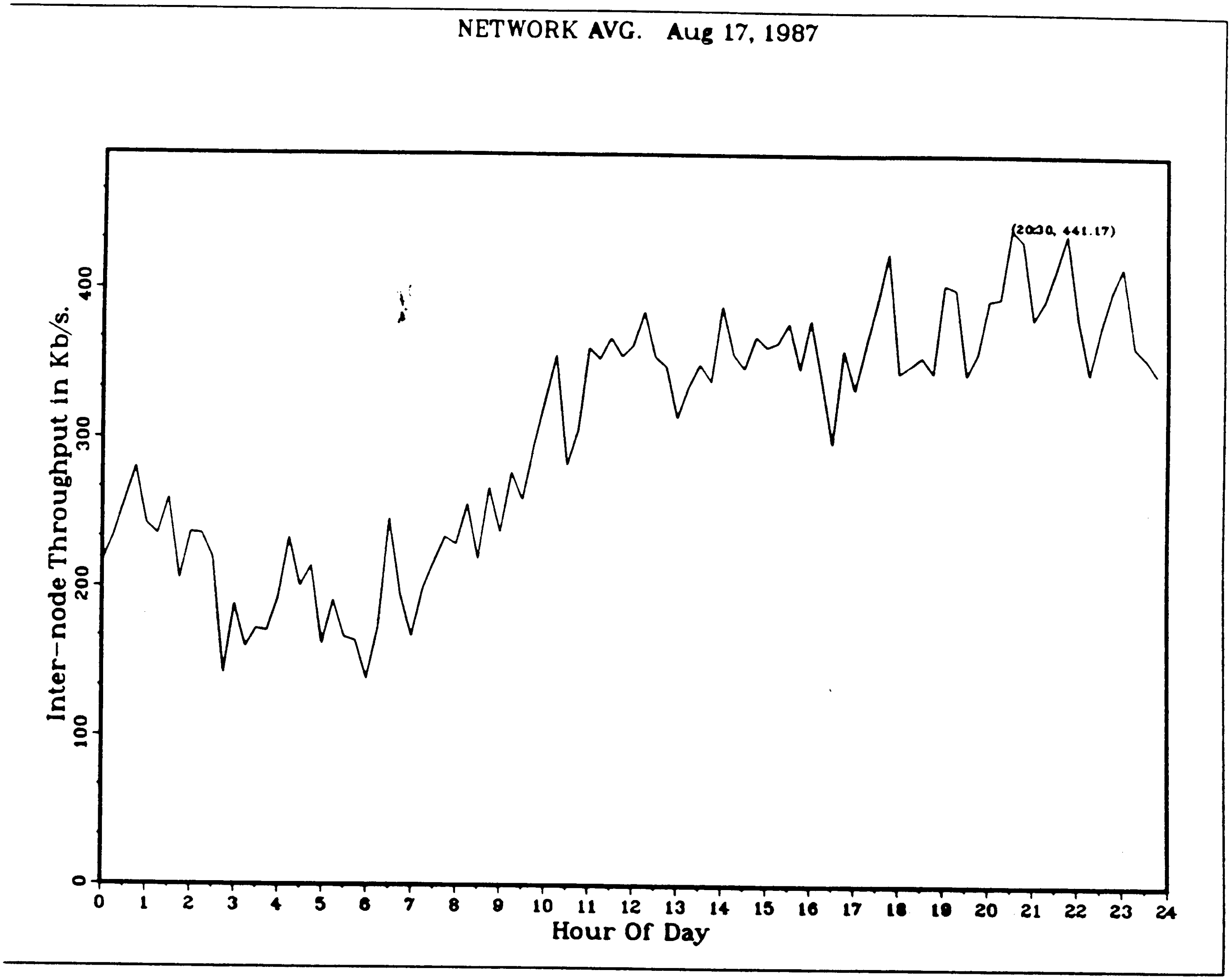
NETWORK AVG. Aug , 1987
Thousands of Round-trips
Hour Of Day
(14:30, 277.36)
0
50
100
150
200
250
300
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

NETWORK AVG. Aug , 1987

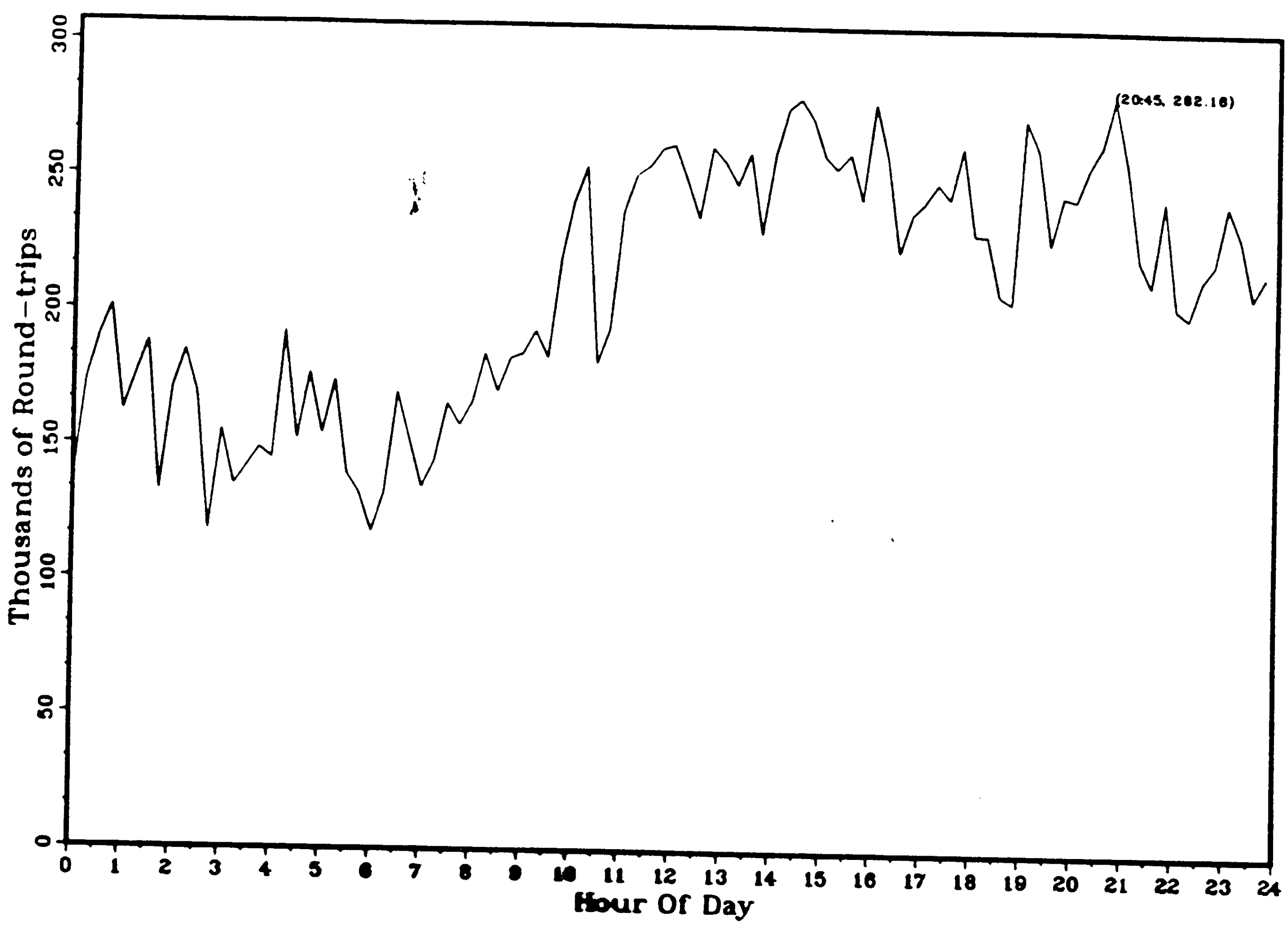


NETWORK AVG. Aug 17, 1987




NETWORK AVG. Aug 17, 1987
(2030, 441.17)
Inter-node Throughput in Kb/s.
0
100
200
300
400
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour Of Day

NETWORK AVG. Aug 17, 1987



IP over 802.X

Drew Perkins, CMU

RFC - Draft - IP on IEEE 802

J. Postel - ISI
J. Reynolds - ISI

Reviewers
Drew Perkins - CMU
Jacob Rekhter - IBM

Obsoletes RFC-948, assigned #'s, et

Goal - Specify IP & ARP so that use is consistent between implementations on particul 802.x network. Not necessari between 802.x and 802.y, x!=y

802.3 - CSMA/CD

1 Mb/sec to 20 Mb/sec

<data rate><medium type><max seg length>

10 BASE 5 - 10 Mb/sec, Baseband, 500 m /segment

7 octets	- preamble (10101010....)
1	- Start Frame Delimiter (10101011)
2 or 6	- Destination Address
2 or 6	- Source Address
2	- Length
n	- LLC Data
	- PAD
4	- Frame Check Sequence (CRC)

(64-1518)

max Frame Size = 1518 octets
min Frame Size = 64 octets

Addresses

I/G	U/L	46 bits

or

I/G	15 bits

802.4 - Token - Passing Bus

3 types specified

Phase Contiguous FSK
- Omnidirectional Bus
- 1 Mb/s
- Manchester encoding

Phase Coherent FSK
- Omni directional Bus
- 5 Mb/s & 10 Mb/s
- Direct Encoding

Multilevel Duobinary AM/PSK
- Directional Bus w/ headend repeater
- 1 Mb/s, 5 Mb/s & 10 Mb/s

octets	
7=1	- preamble (>= 2 μs)
1	- Start Delimiter (NN0NN000)
1	- Frame Control (type of frame MAC or LLC
2 or 6	- Destination
2 or 6	- Source
n	- Data
4	- FCS
1	- End Delimiter (NN1NN1IE)

Ø ≤ SIZE ≤ 8191

16 or 48 bit addresses

I/G bit of source = Ø

I = Intermediate Frame Bit

= 1, more frames follow

E = Error-Detected Bit

set to 1 by repeater detecting error

802.5 - Token Ring

1Mb/s & 4 Mb/s

Manchester Encoding

4 symbols - 0, 1, J, K

J, K code violations

1 octet	-	Starting Delimiter (JK0JK000)
1	-	Access Control (priority, reservat
1	-	Frame Control (Frame type, MAC or LLC)
2 or 6	-	Destination Address
2 or 6	-	Source
n	-	INFO
4	-	FCS
1	-	Ending Delimiter (JK1JK1IE
1	-	Frame Status (ACrr ACrr)

A = Address Recogni

C = Frame Copied

Addresses same format

802.2 Logical Link Control

Type 1 - data-link-connectionless service

Type 2 - data-link-connection-oriente[d] service, basically HDLC ABM (X.25 LAPB)

Class I - Type 1 only

Class II - Type 1 & Type 2

1 octet - DSAP

1 - SSAP

1 or 2 - Control (N(S), N(R), P/F, functi[on])

n - Information

Type 1

Commands	Responses	
UI		
XID	XID	(class query)
Test	Test	(ping)

Type 2

Commands	Responses		
I	I	SABME	UA
RR	RR	DISC	DM
RNR	RNR		FRMR
REJ	REJ		

SNAP - SubNetwork Access Protocol

LLC DSAP/SSAP = 170

3 octets - Product Id / Org Code

Ø = Blue Book Ethernet

2 octets - Ethernet type field

2048 = IP

2054 = ARP

802.x Summary

802.3

Preamble	SFD	DA	SA	Lng	Data	Pad	FCS
7	1	2,6	2,6	2	n		4

802.4

Preamble	SD	FC	DA	SA	Data	FCS	ED
≥1	1	1	2,6	2,6	n	4	

802.5

SD	AC	FC	DA	SA	Data	FCS	ED	FS
1	1	1	2,6	2,6	n	4	1	1

Addresses:

DA

I/G	U/L	addr
1	1	46

or

I/G	addr
1	15

SA

Ø	U/L	addr
1	1	46

or

Ø	addr
1	15

802.2

DSAP	SSAP	C+I
1	1	1 or 2

SNAP

org code (3)	×2

IBM style Source Routing

RII - Routing Information Indicat

uses I/G bit of Source Address

RI - Routing Information

2 to 18 octets

2 octets - Routing Control Field

B	LB	r	Length	D	rrr
1	1	1	5	1	7

B = Broadcast bit (independent of all stations

LB = Limited Broadcast

r = reserved

Length = size of RI including RC

D = Direction bit

2*N octets - Segment #'s

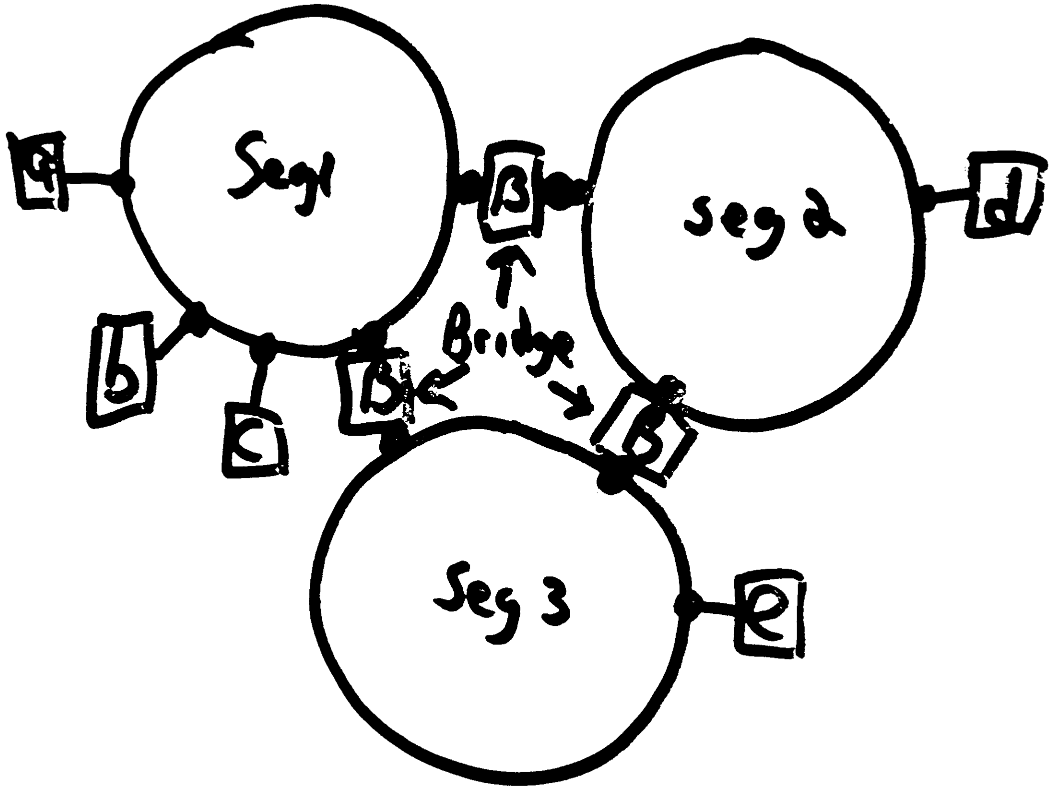
"advantages"

- multiple paths for load splitting
- "automatic" rerouting on bridge failure

references

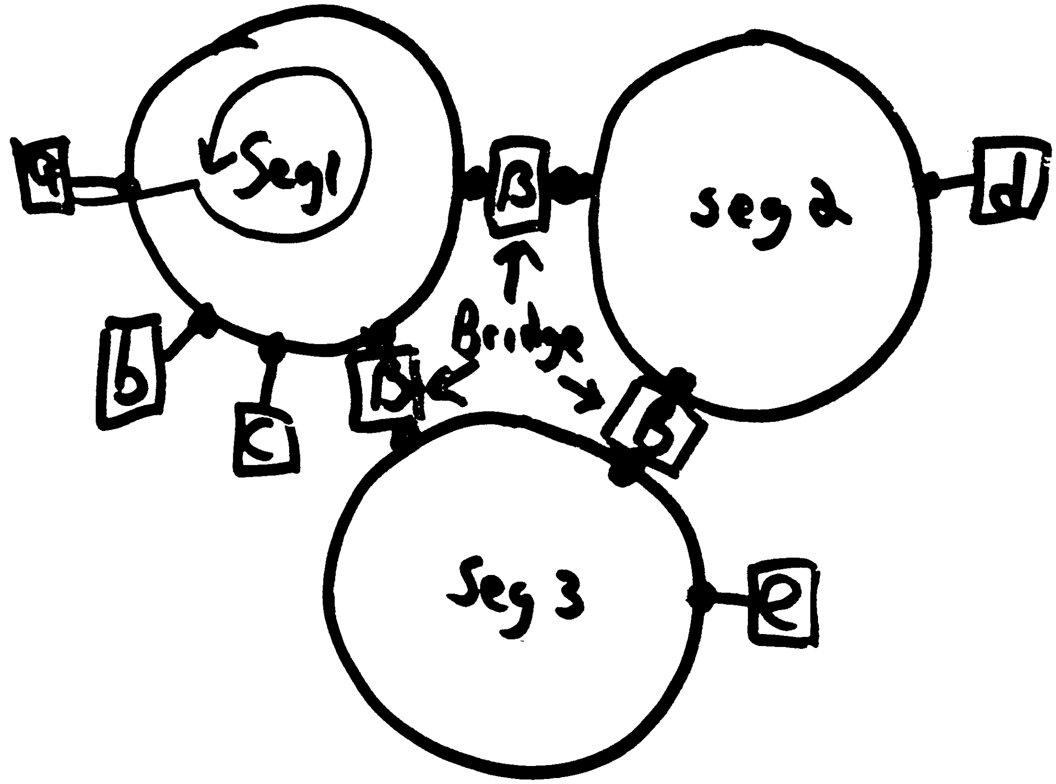
Farber
Sunshine
Forss
Saltzer, Reed & Clark

Source Routing Examples



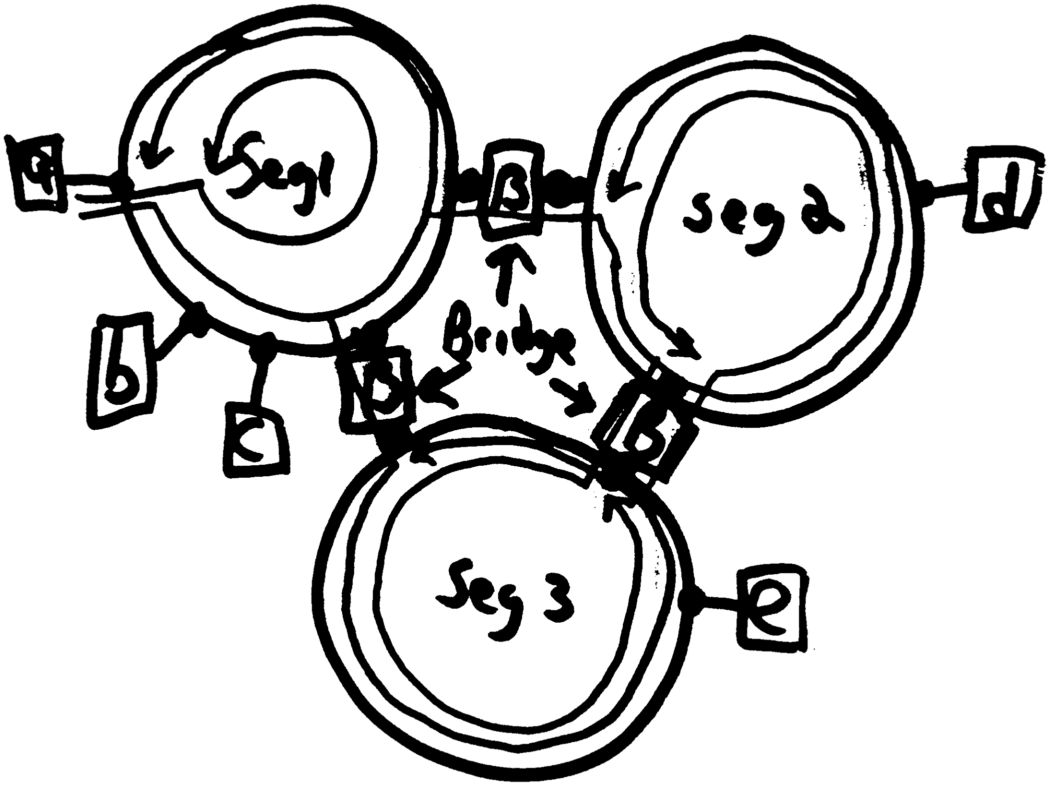
DA SA RC Recipients

Source Routing Examples



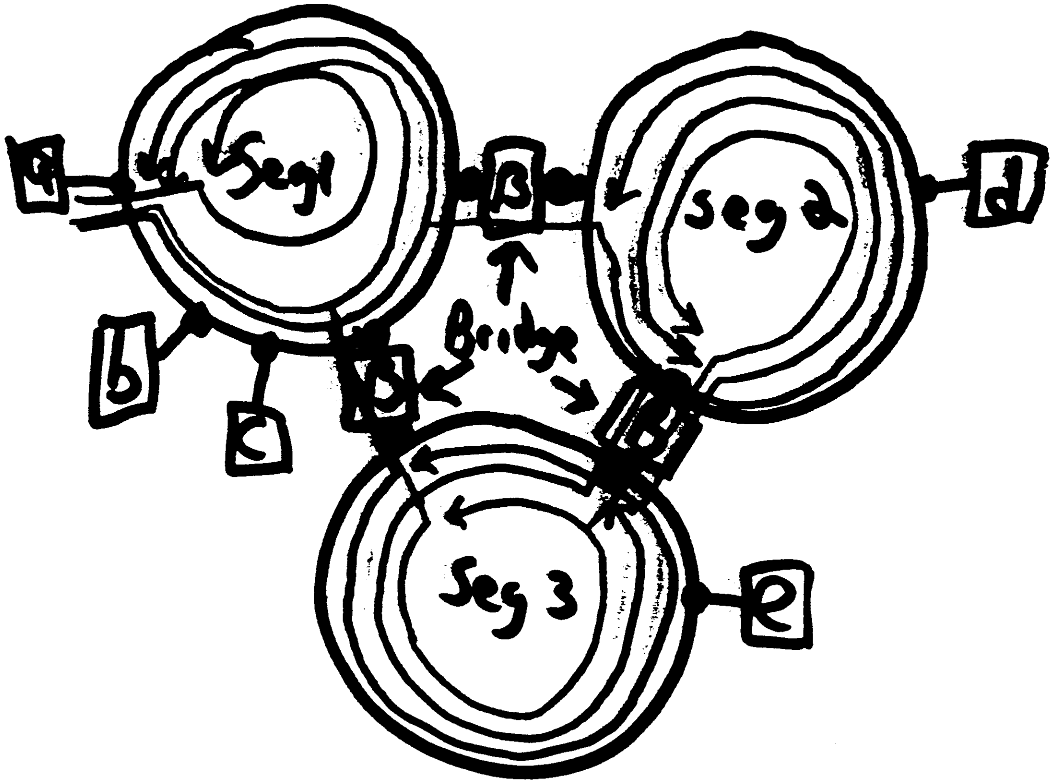
DA	SA	RC	Recipients
a	a	–	a
b	a	–	b
*	a	–	a, b, c

Source Routing Examples



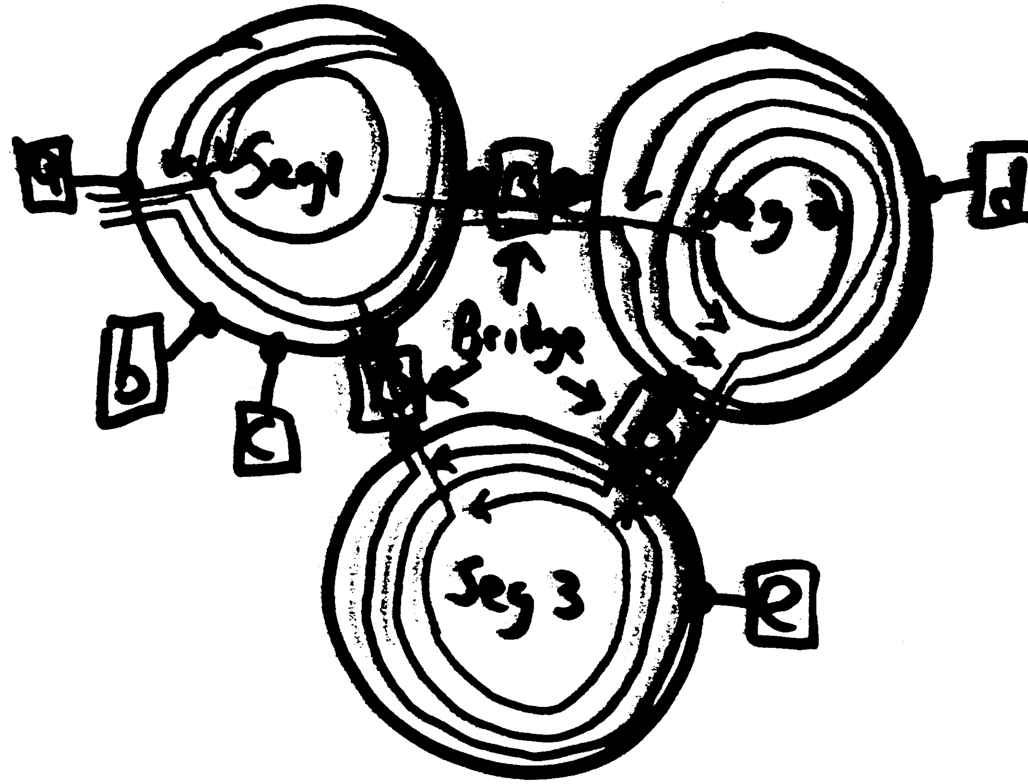
DA	SA	RC	Recipients
a	a	–	a
b	a	–	b
*	a	–	a, b, c
d	RI+a	B	d, d
e	RI+a	B	e, e
c	RI+a	B	c
*	RI+a	B	a, b, c, d, d, e, e

Source Routing Examples



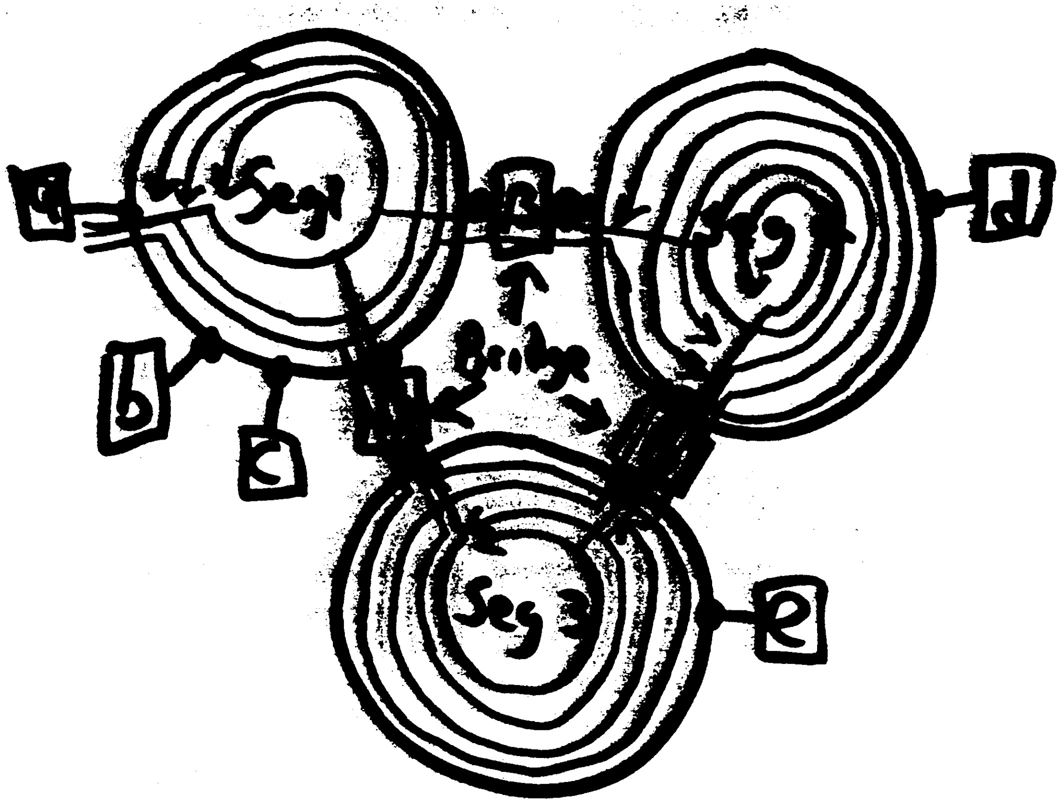
DA	SA	RC	Recipients
a	a	–	a
b	a	–	b
*	a	–	a, b, c
d	RI+a	B	d, d
e	RI+a	B	e, e
c	RI+a	B	c
*	RI+a	B	a, b, c, d, d, e, e
c	RI+a	B+LB	c
d	RI+a	B+LB	d
*	RI+a	B+LB	a, b, c, d, e

Source Routing Examples



DA	SA	RC	Recipients
a	a	–	a
b	a	–	b
*	a	–	a, b, c
d	RI+a	B	d, d
e	RI+a	B	e, e
c	RI+a	B	c
*	RI+a	B	a, b, c, d, d, e, e
c	RI+a	B+LB	c
d	RI+a	B+LB	d
*	RI+a	B+LB	a, b, c, d, e
d	RI+a	2, 1, 2	d

Source Routing Examples



DA	SA	RC	Recipients
a	a	–	a
b	a	–	b
*	a	–	a,b,c
d	RI+a	B	d,d
e	RI+a	B	e,e
c	RI+a	B	c
*	RI+a	B	a,b,c,d,d,e,e
c	RI+a	B+LB	c
d	RI+a	B+LB	d
*	RI+a	B+LB	a,b,c,d,e
d	RI+a	a,1,a	d
d	RI+a	a,b,a,a	d

RFC Decisions

- Uses 802.2 type 1 communicatio only. Type 2 may be used among consenting hosts
- Same hardware type used for all 802.x in ARP (6).
- ARP hardware address length used to differentiate 16 & 48 bit addresses
- IP & ARP broadcasts use all-stations address of all-one's.
- Trailers may be used between consenting systems.
- Uses 802.2 Unnumbered Informatio (UI) packets.

- Same MTU's used across all 802.x for compatibility. Is this good ???
  Since 802.3 MTU == 1492, so does 8024, 802.5

- 802.5 implementations may decide to implement source routes or not do them. However, both implementations should still interoperate in the case of a single ring (no intermediate bridge)

To do that:

- all implementations are required to accept ARP & IP broadcasts with no RIF (RII = 0) and packets with empty RIF (only RC field with length = 2)

- Implementations which do not support source routes should gracefully ignore packets with non-empty RIF's.

- Implementations which do support it must be prepared to receive multiple copies of broadcasts, but may decide between multiple ARP replies however they wish.

~~Other~~

- IBM bridges may have MTU configured between 508 & >8K octets
- RIF information is logically distinct from ARP table. Whether to store it there anyway is implementation decision
- 802.5 multicasts are useless for IP multicast since current hardware limits you to 1 address.
- 802.5 Frame Not Copied bit should be used to retransmit frame some # of times
- Address Not Recognized mapped to ICM destination unreachable.??? ARP cache

Congestion Control Simulation Results

Bob Stine, MITRE

Congestion Control Simulation Results

Robert Stine

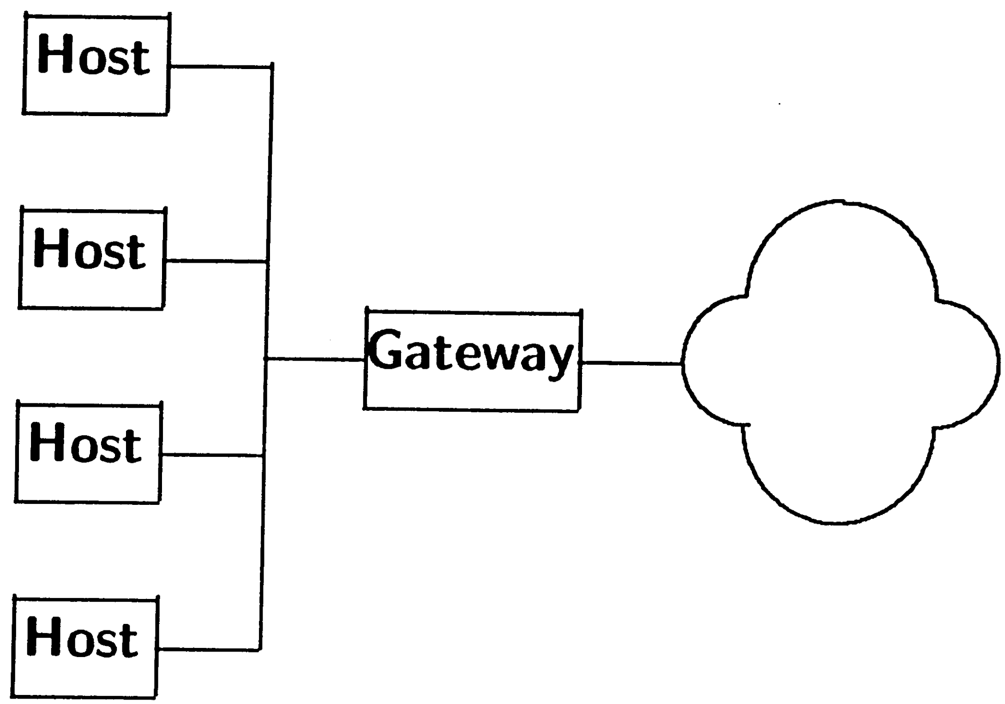
MITRE

Gateway Congestion Control Simulation

- Detailed model of TCP connections.
- Underlying Assumptions:
  - Dominant congestion-inducing effects in stub topology:
    - Dropped packets at gateway.
    - Gateway queuing delays at slow interface.
  - Negligible effects in stub topology:
    - LAN contention.
    - Traffic from long-haul net to LAN.
    - Non-TCP traffic.

Simulation Model

- Model detailed at ES/IS level, abstract past gateway.



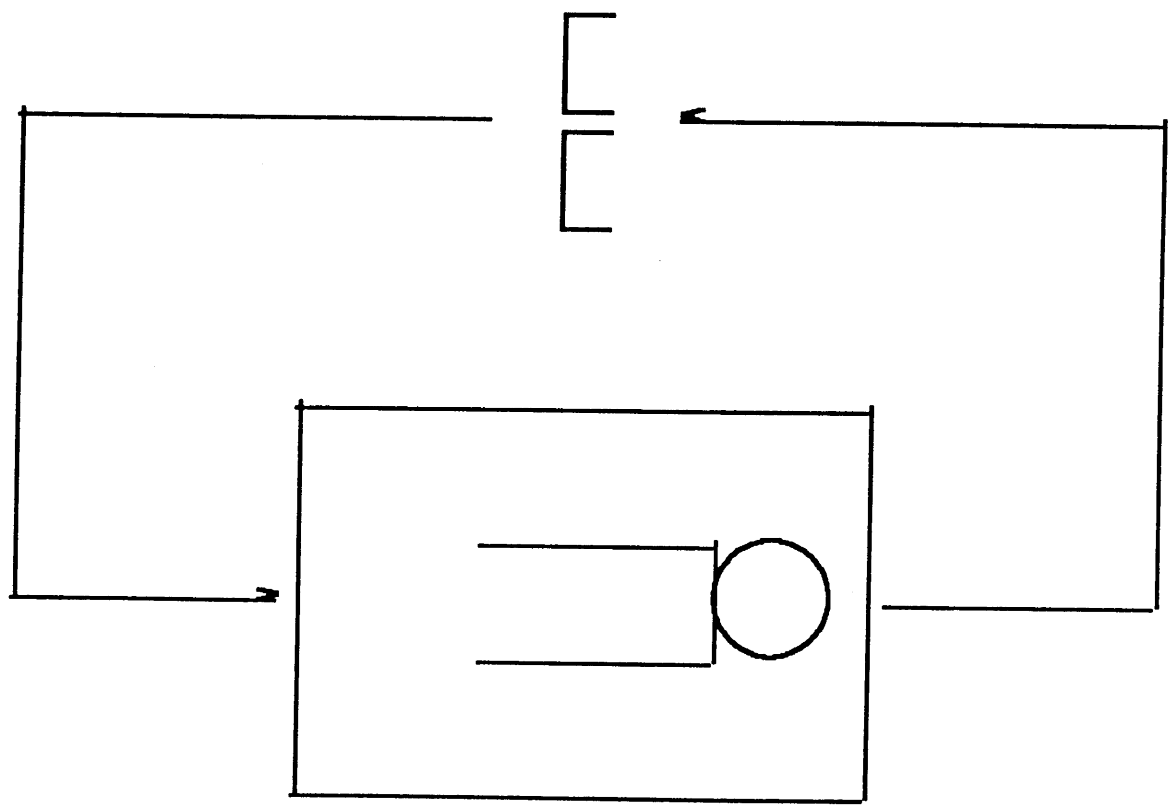
Validation by Analytical Models

- Closed queuing network with finite population:
  - $mT \sim K - mD$
- Throughput through end-to-end windows:
  - $r = \min[W/d, 1/X]$

Closed Queuing Network Approximation

- $mT \sim K - mD$
  - m: service rate.
  - T: average service time.
  - K: population.
  - D: time between requests (inverse of rate of generation).
- see H. Kobayashi,
  Modeling and Analysis:
  An Introduction to System Performance Methodology

A Closed Queuing Network with Finite Population



Recasting Approximation for Gateway, Bulk FTPs

- For stable bulk data transfers, $GW \sim K/m - D$
  - GW: average queuing, processing delay at GW
  - K: Total packets in flight
  - m: Baud rate of long haul interface
  - D: RTT component not at GW (host processing, etc.)

Gateway Delays, Predicted vs. Simulated

Parm values	Predicted	Simulated
K/m=4.8, D=.0112	4.7888	4.7195
K/m=2.4, D=.0112	2.3888	2.3724
K/m=2.4, D=.5008	1.8992	1.8897
K/m=1.2, D=.0176	1.1824	1.1822

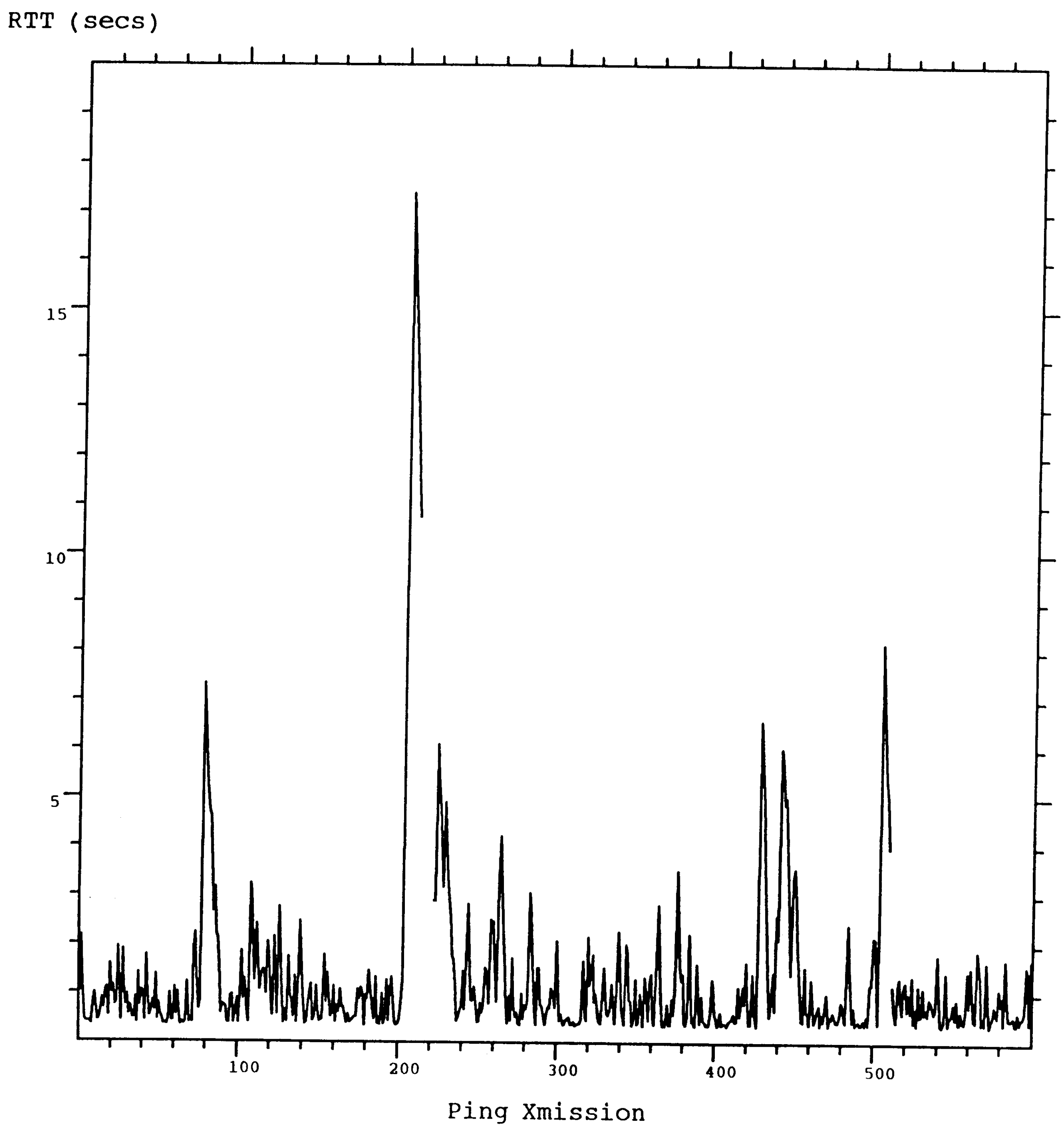
Throughput through End-to-End Windows

- $r = \min[W/d, 1/X]$
  - W: packets per window
  - d: RTT delay
  - X: Transmission time for a single packet
- see D. Bertsekas, R. Gallager, *Data Networks*

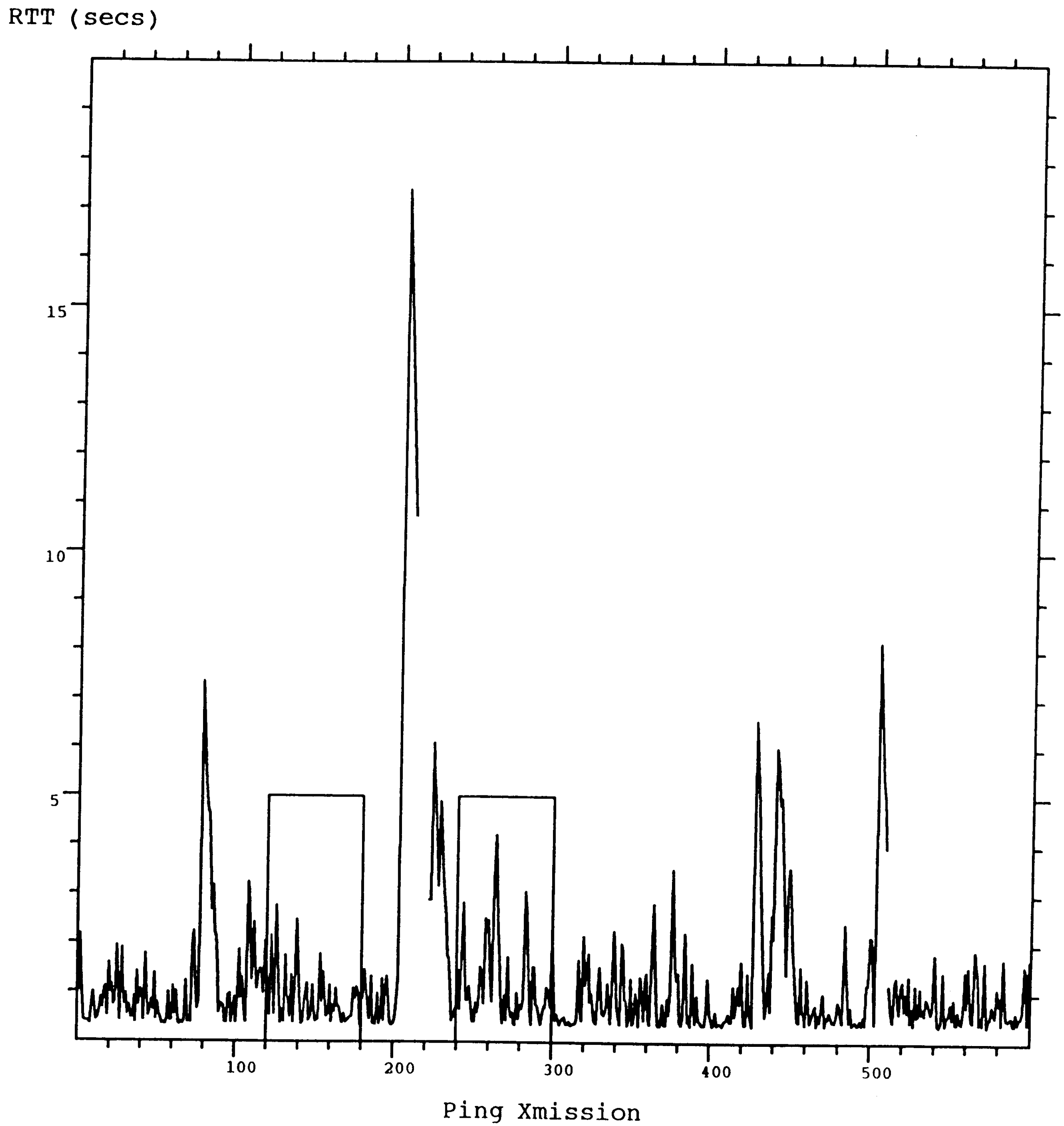
TCP Throughput, Predicted vs. Simulated

Parm values	Predicted	Simulated
W=5, d=.10005	49.98	49.87
W=5, d=.1204	41.53	41.46
W=5, d=.2204	22.69	22.67
W=5, d=.3204	15.61	15.60
W=10,d=.2211	45.23	45.04
W=10,d=.3211	31.14	31.06

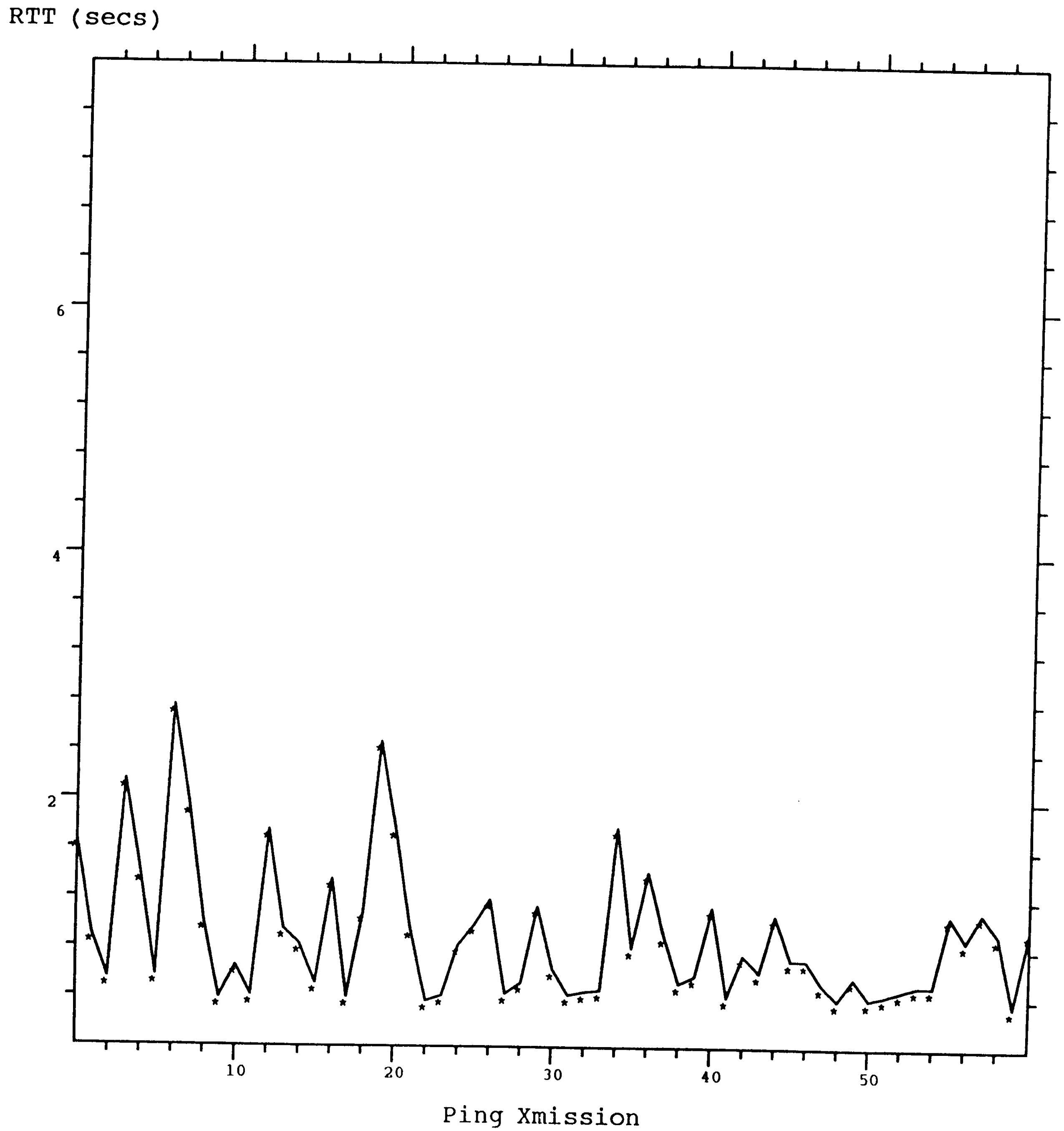
RTT, Pings to NRL-AIC.ARPA



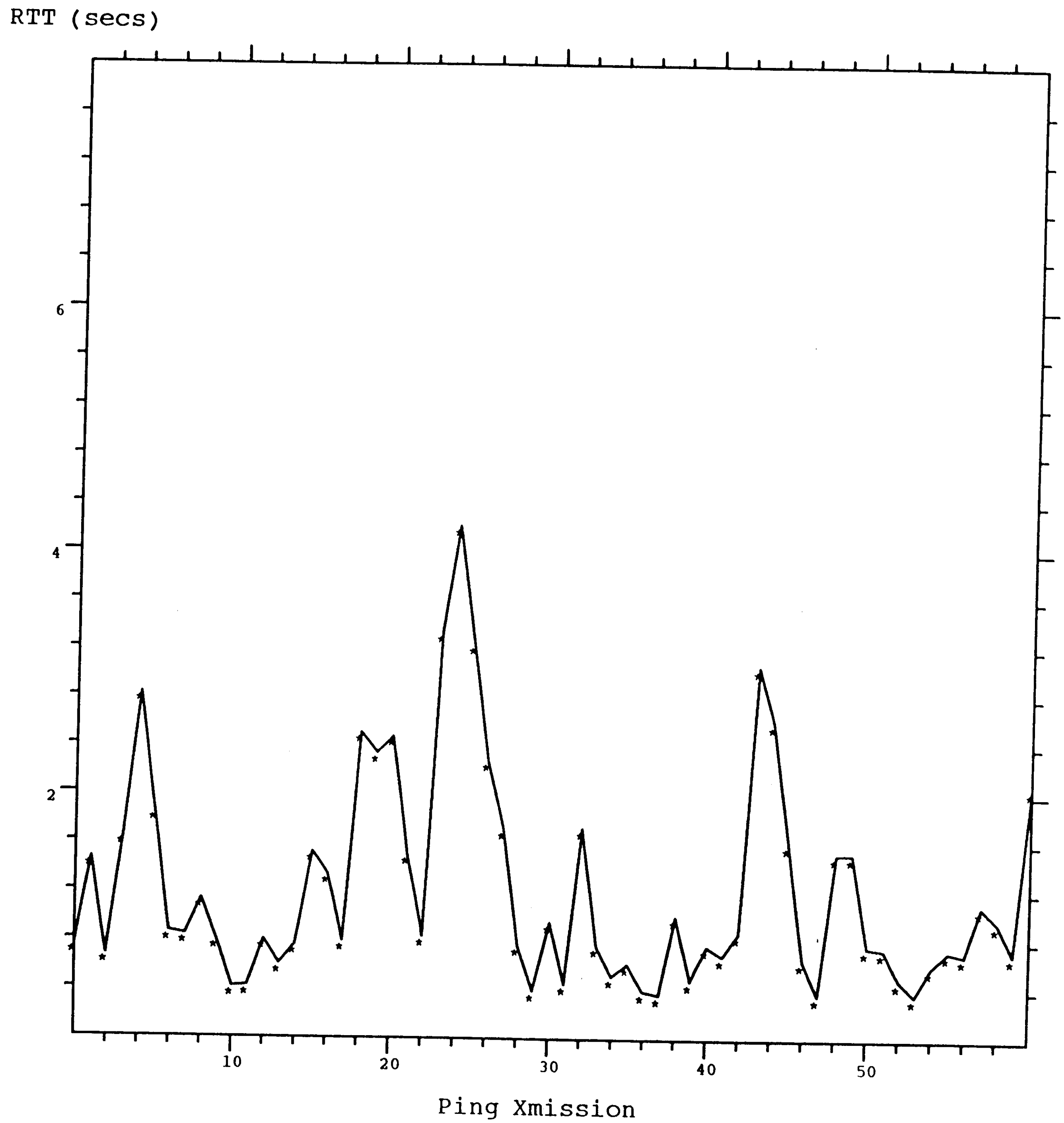
RTT, Pings to NRL-AIC.ARPA



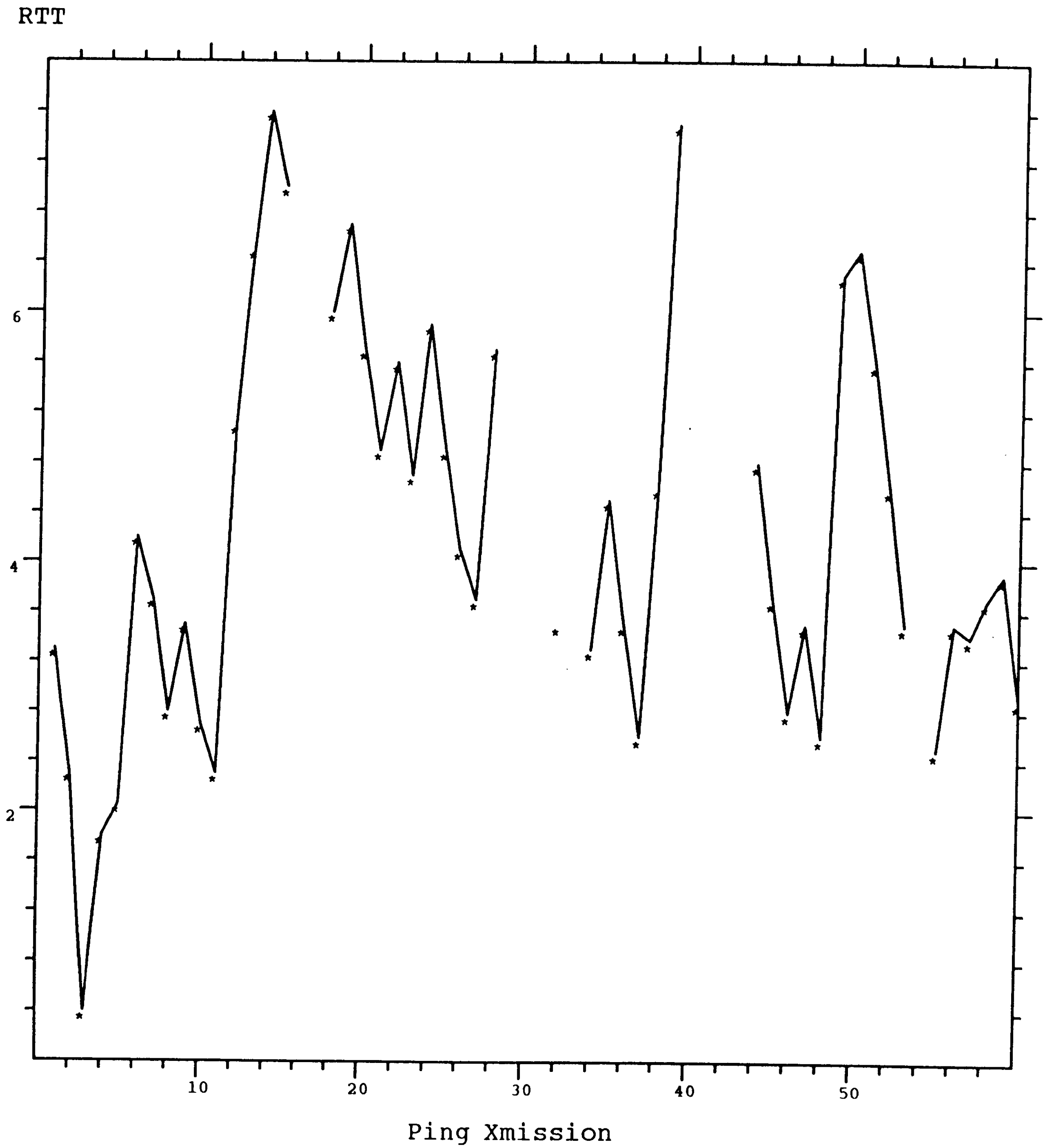
Detail 1, RTT of Pings to NRL-AIC



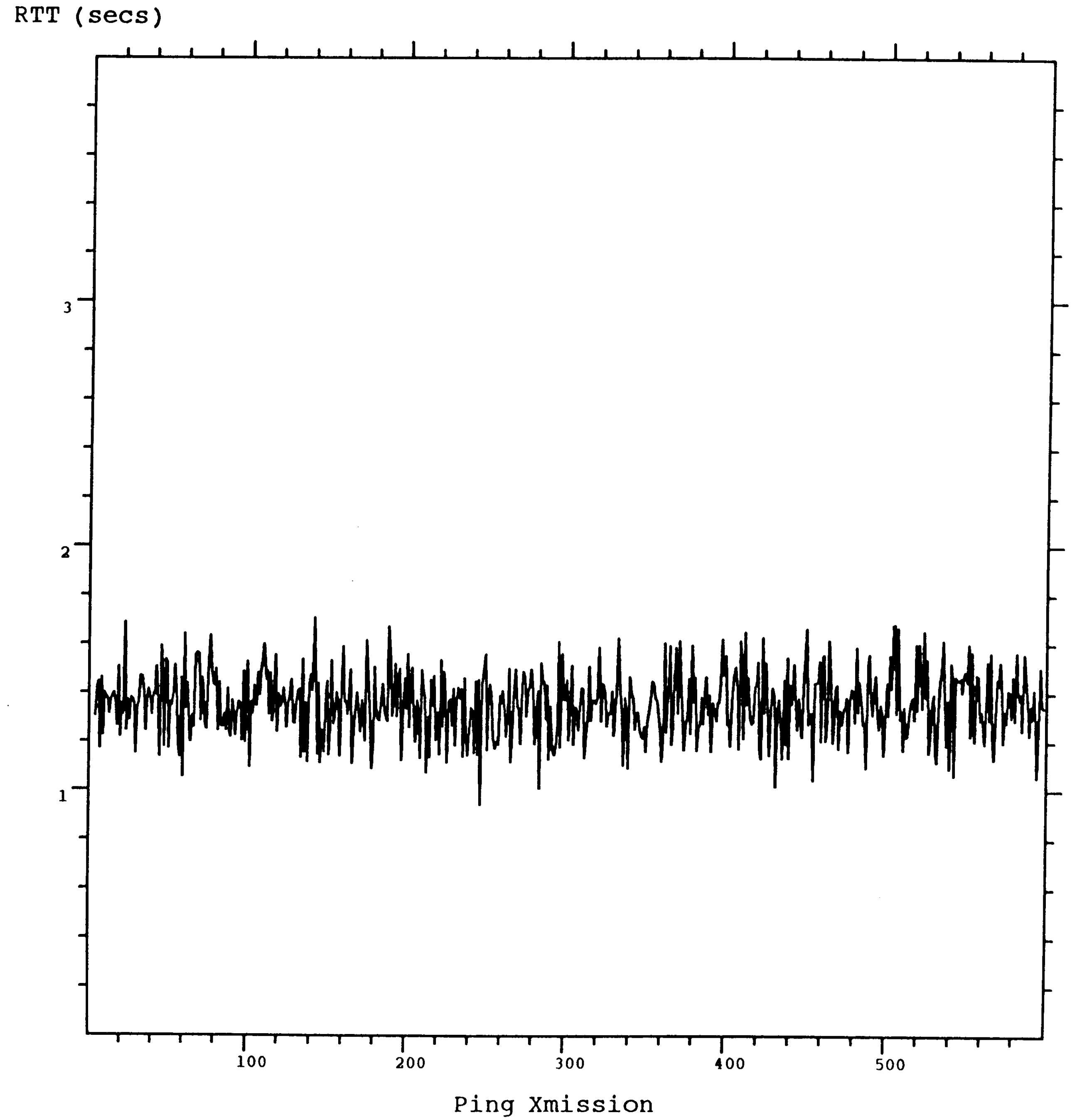
Detail 2, RTT of Pings to NRL-AIC



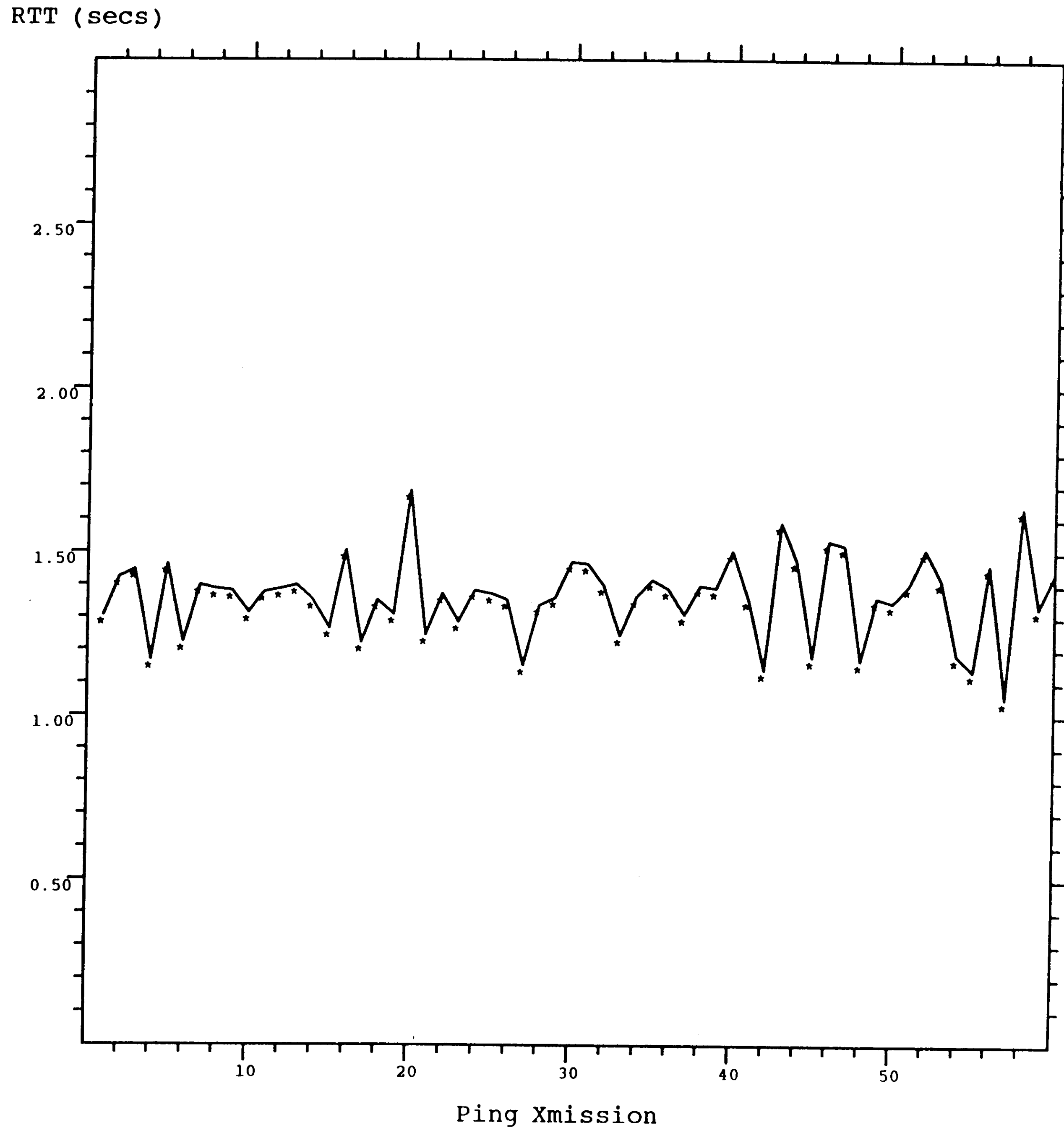
Dec 3, 1986: RTSG to Monet RTT



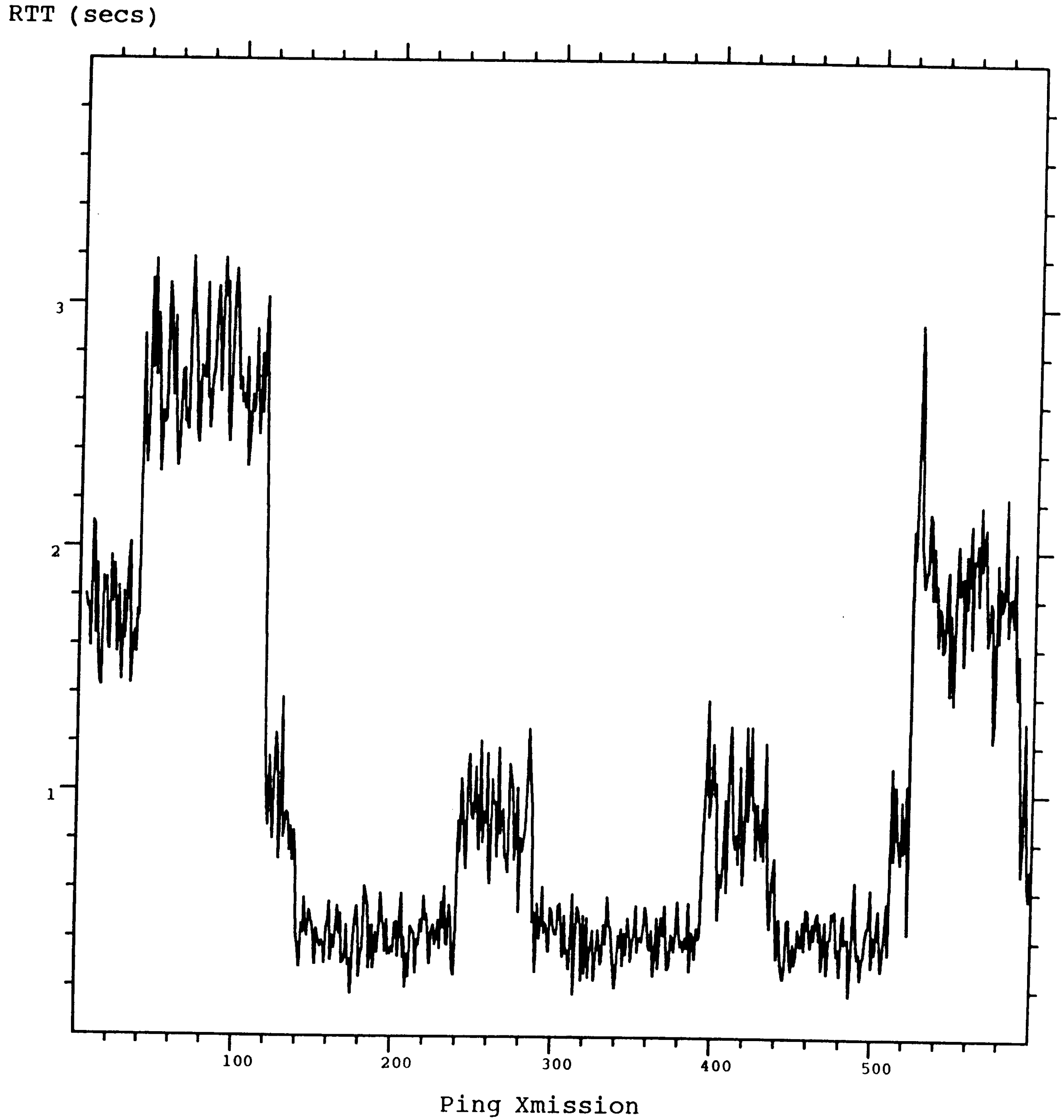
Simulated Ping RTTs with Well-behaved TCP Connections



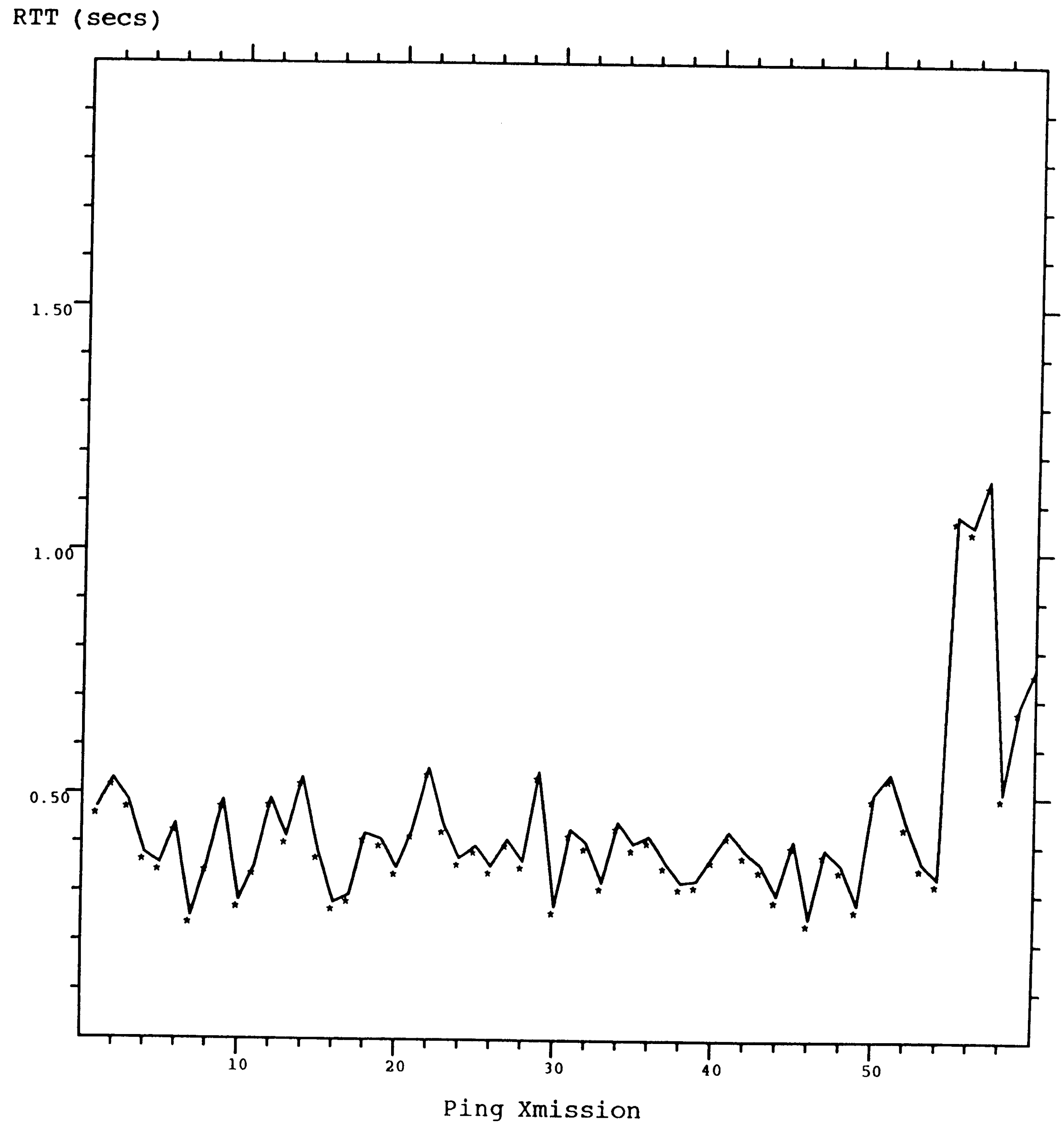
Detail, Simulated Pings, No Retransmissions



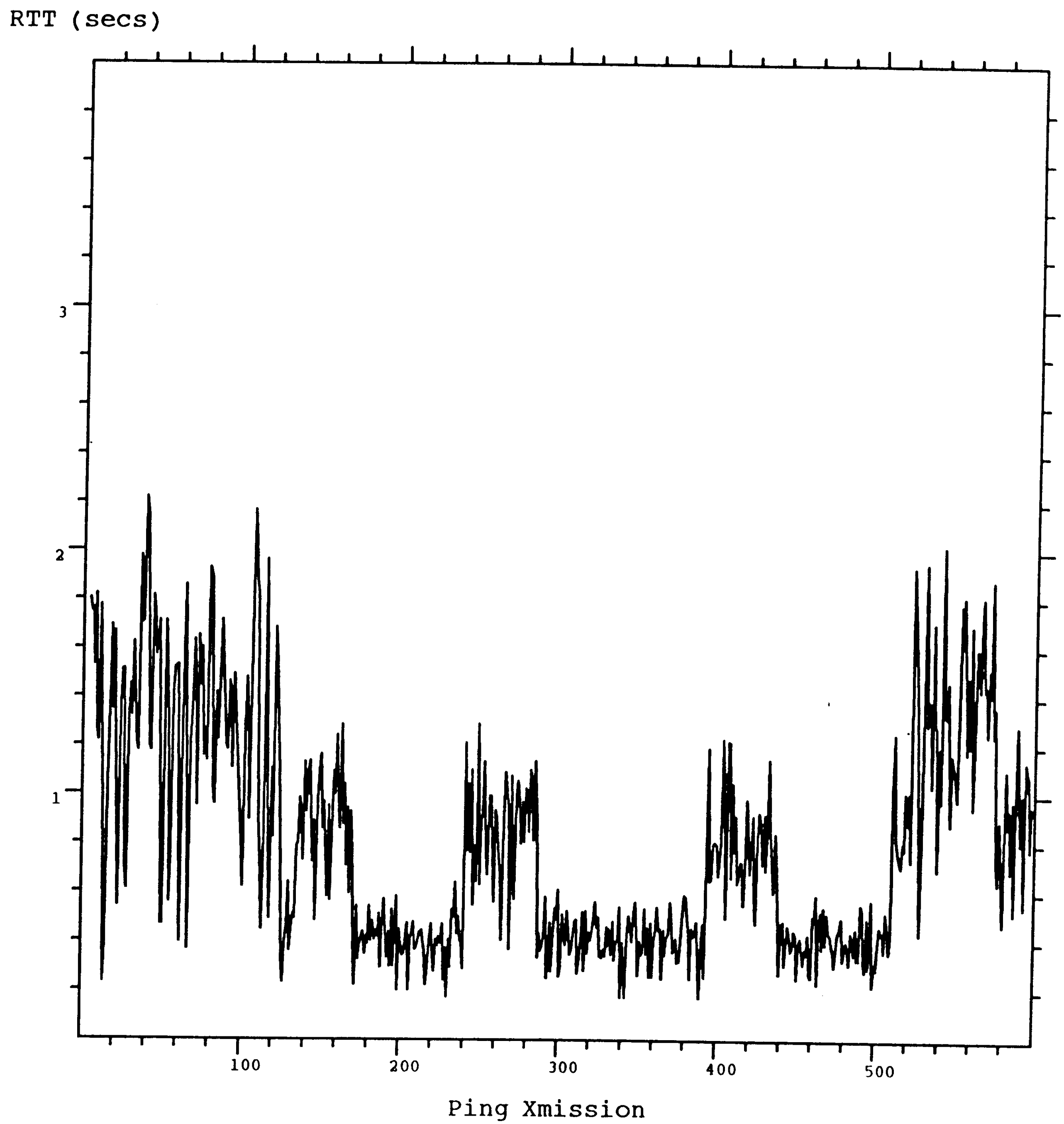
Simulated Ping RTTs, Spurious Retransmissions



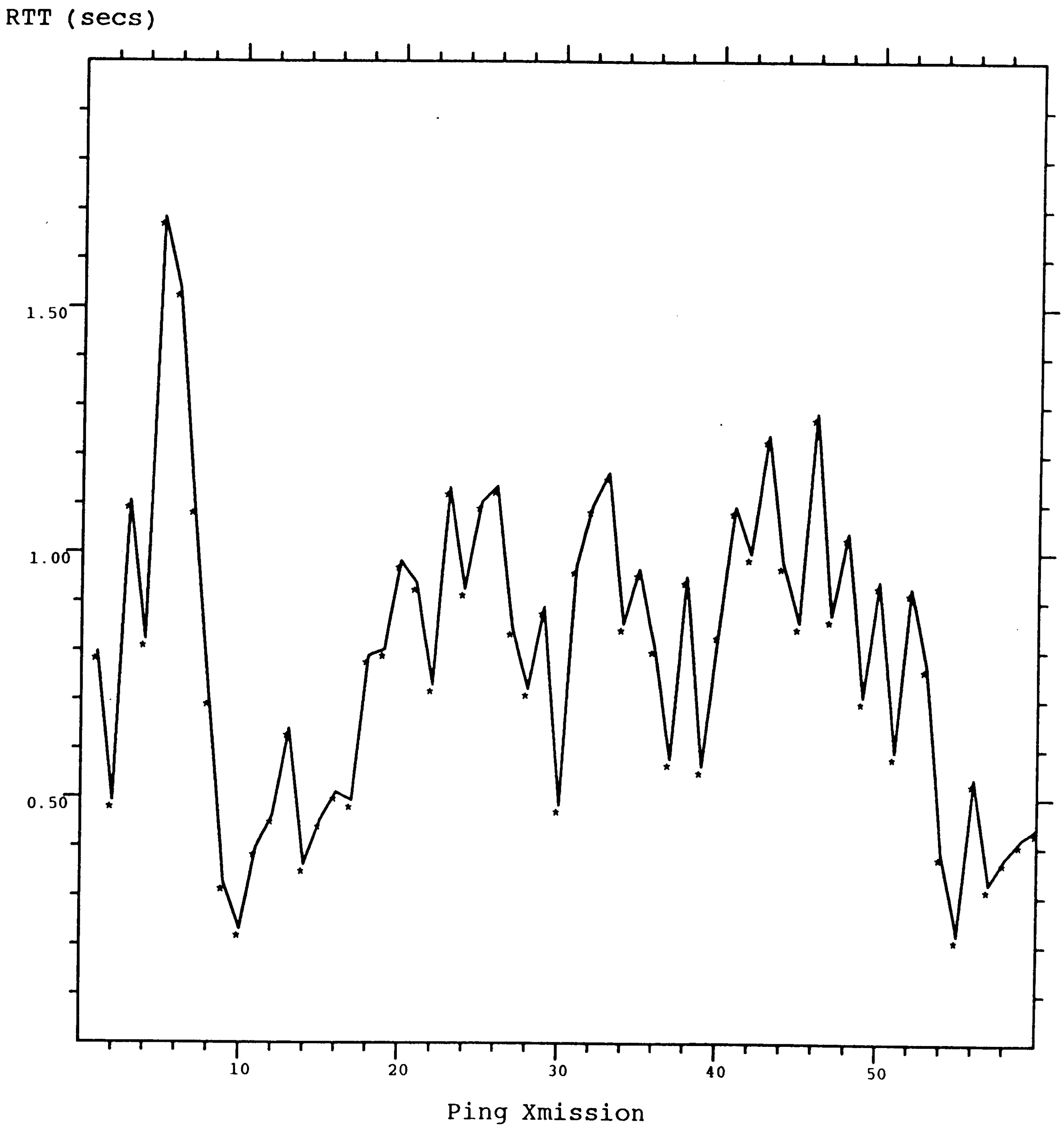
Detail, Simulated Pings, Spurious Retransmissions



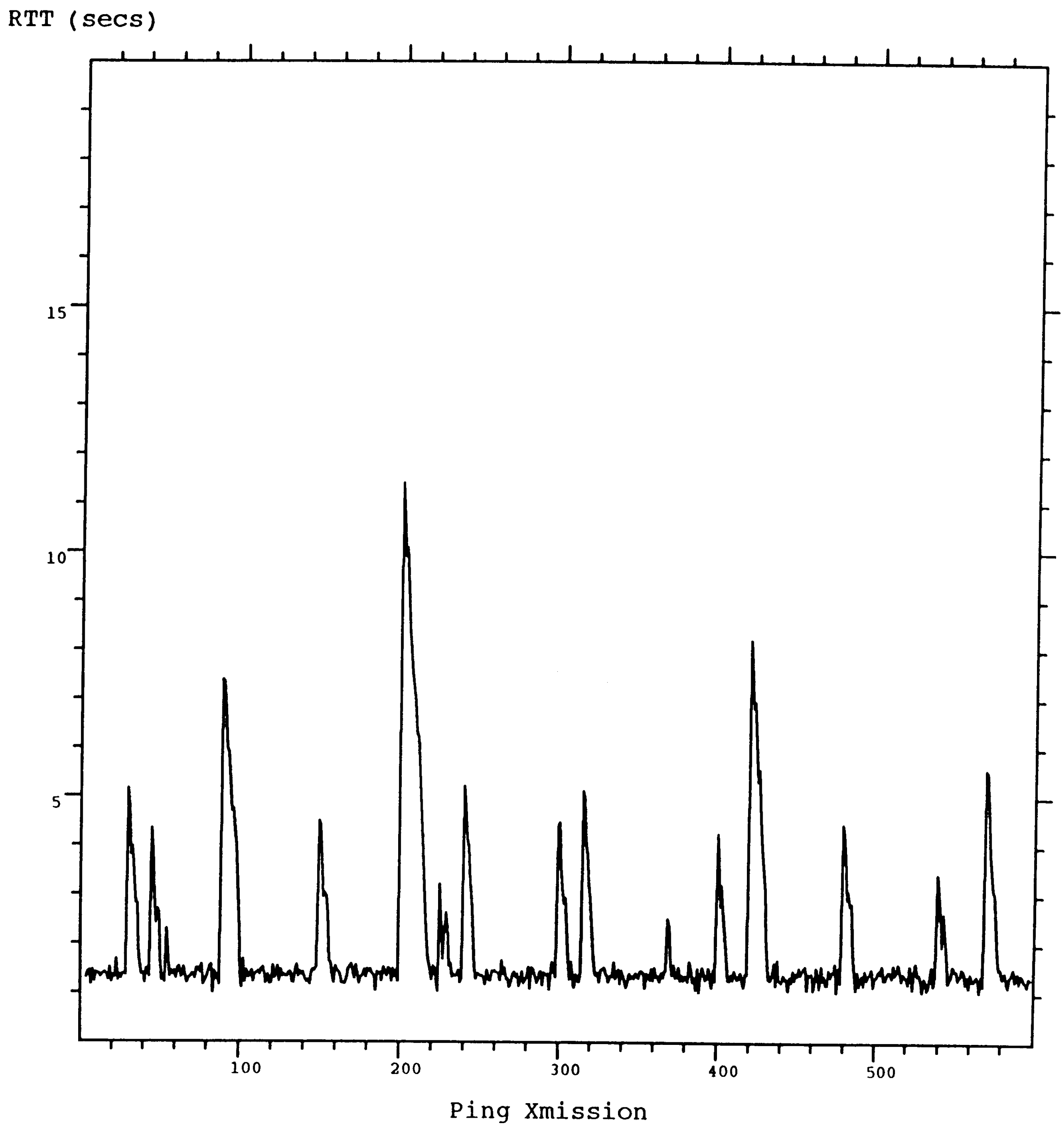
Simulated Ping RTTs with Gateway Buffer Overflow



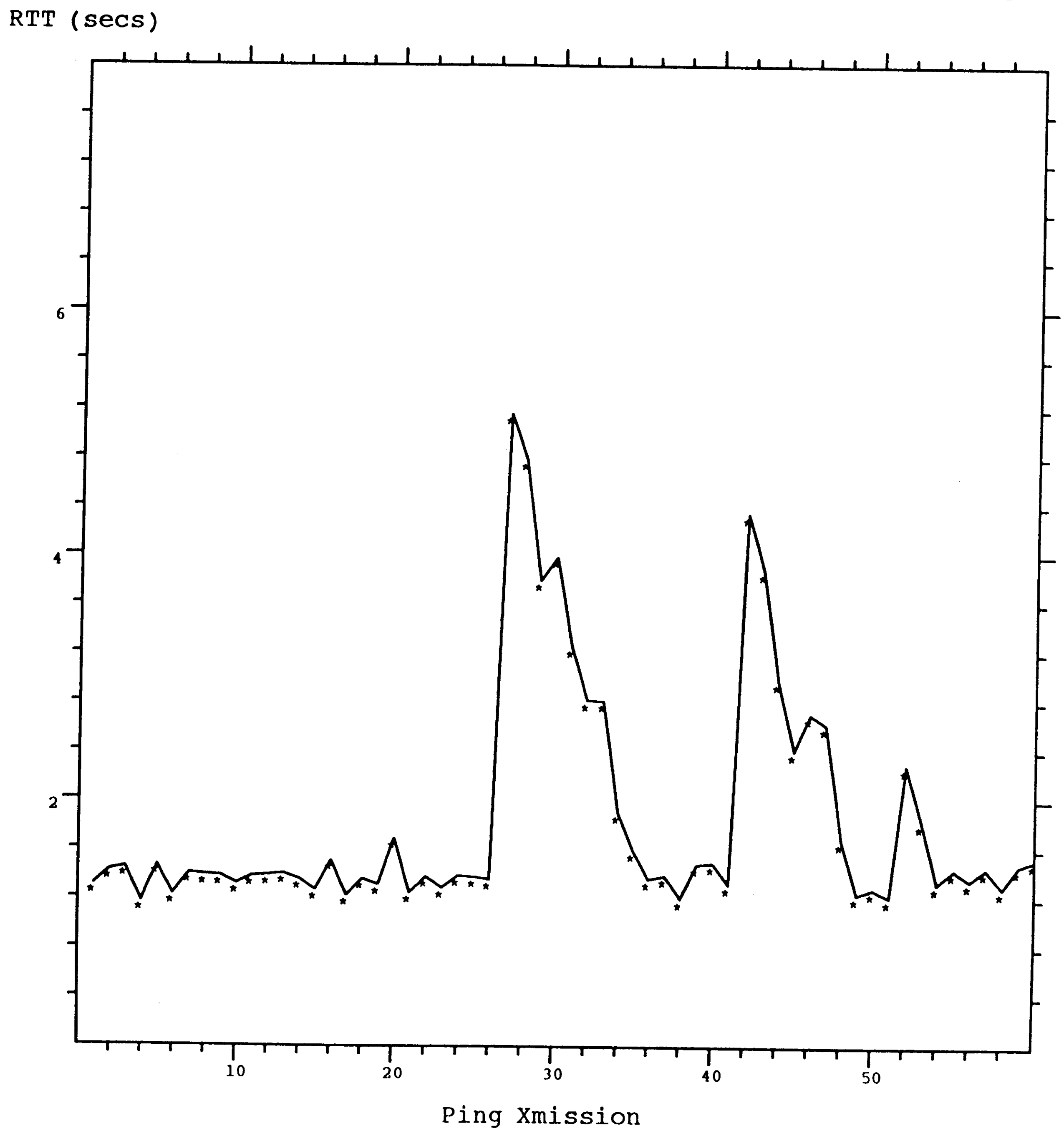
Detail, Simulated Pings, Buffer Overflow



Simulated Ping RTTs, Exogenous Delays



Detail, Simulated Pings, Exogenous Delays



Empirical Validation

- "Brute force" required to model RTT excursions.
- Conclusion: RTT excursions not caused by
  - Stable, well-behaved TCP connections,
  - Spurious retransmissions,
  - Gateway buffer overflow.
- N.B., Excursions are seen on non-gateway traffic.

Use of the Simulation: Congestion Control Experiments

- Investigated bulk data transfers.
- Techniques, TCP parms varied, under constant loads.
- End-to-end load: data passed to TCP for transmission.
  - Each bulk connection to "transmit" an equal number of segments.
  - Load level set by adjusting
    - Rate of connection generation,
    - Segments per connection.
- Output: end-to-end throughput and mean delay.

Congestion Control Experiments: Traffic Profiles

- Low, spiked, and high levels.
  - Ten stochastically different loads at each level, to emmulate multiple runs.
- High level: each connection sends 250 segments.
  Low, spiked: each connection sends 150 segments.
  - Spike caused by incresed frequency of connection generation.
- Values for low level found by trial and error:
  - Goal: With no congestion control, reach stable delay and drop few, if any, packets.

Congestion Control Techniques Evaluated

- Fair Queuing:
  - Round-robin service, per traffic source.
- Source Quench Introduced Delay (SQuID):
  - Hosts slow rate of IP transmissions.
- Retransmit Timeout (RTO) Backoff:
  - Adjustment to retransmit timer if retransmission occurs.
- Nagle Windowing (NW):
  - Connections reduce of packets "in flight" in response to Source Quench.

Parameters Varied

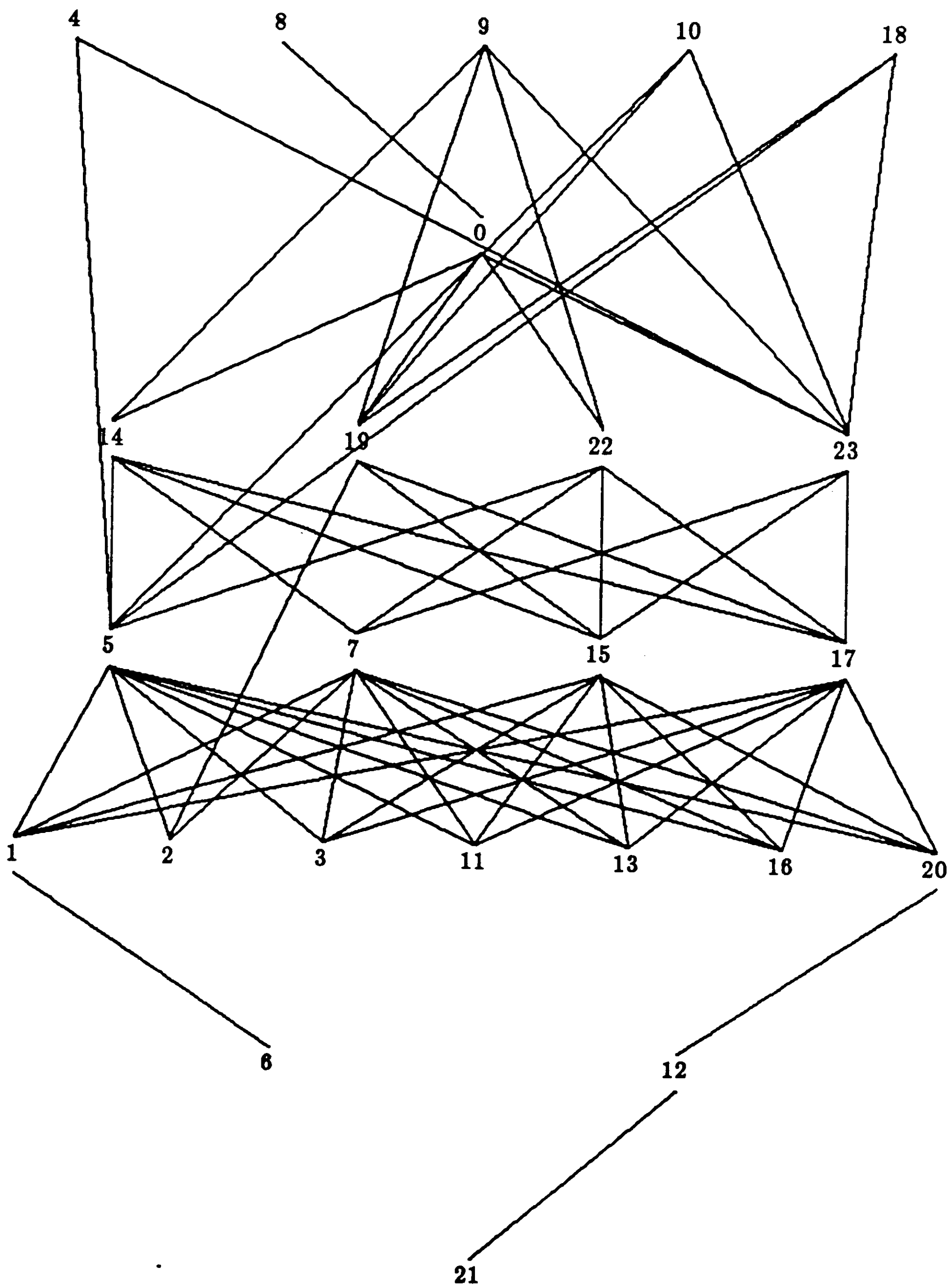
- SRTT seed and RTO lower bound
- Standard SRTT algorithm used:

$$\text{SRTT} = \alpha\text{SRTT} + (1 - \alpha)\ \text{RTT}$$
$$\text{RTO} = \min(\text{Ubound}, \max(\text{Lbound}, \beta\text{RTT}))$$

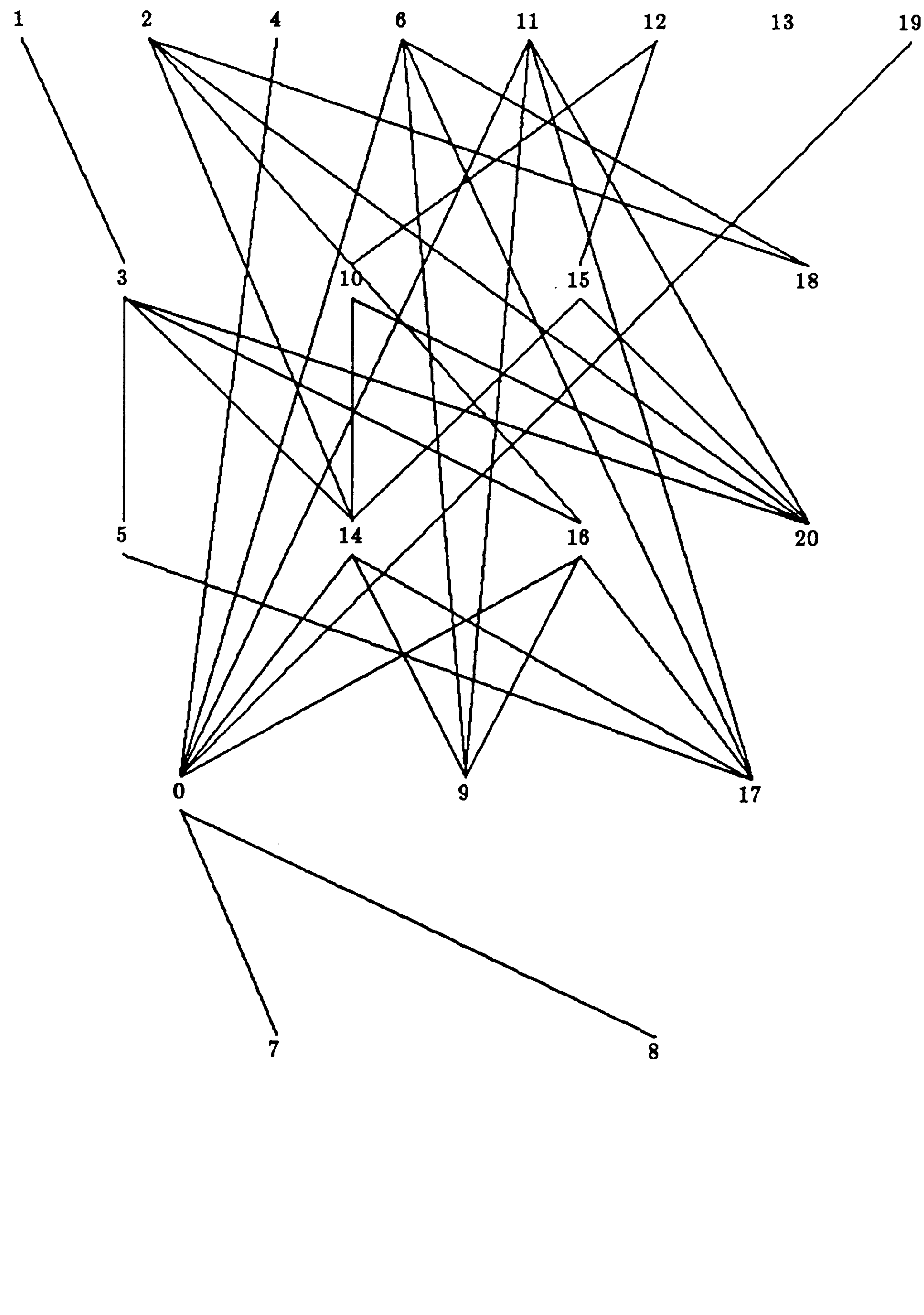
Analyzing Simulation Results

- Lattice of parameter settings, order based on dominance.
  - One element dominates another if its output value is not less than the other's, for each of the 10 loads.
- Mann-Whitney U-test (rank sum) applied to upper and lower bounds of lattice.
- "Minimum Cover" techniques used to characterize results (espresso).

Delay, Low Traffic



Throughput, Low Traffic



Least Average Delay, Low Traffic

- Lower bound of lattice includes 34 of 64 minterms, Mann-Whitney does not discriminate.
- For either queuing discipline:
  - SQ, SQuID, high seed, Nagle Windowing (NW).
- For FIFO Queuing:
  - No SQ, high seed.
  - SQ, high seed, and SQuID or NW.
- For Fair Queuing:
  - No SQ and high seed,
  - SQ, high seed, and SQuID or NW.

Worst Delay, Low Traffic

- Upper bound of lattice includes 7 minterms.
  With Mann-Whitney, 2 minterms distinguished:
- FIFO queuing, SQ, SQuID, both low seed and low bound, with or without NW.

Average Delay, Spiked Load

- Lower bound of lattice includes 3 of 64 minterms.
- Mann-Whitney strongly discrinated betweem these, $\alpha = 0.001$.
- Least delay with Fair Queuing, SQ, SQuID, RTO Backoff, high seed, high RTO bound, with or without NW.
- Upper bound of lattice a single case:
  FIFO queuing, SQ, SQuid, but no backoff,
  Both seed and RTO bound low, and no NW.

Average Delay, High Traffic

- Lower bound of lattice includes 30 of 64 minterms, Mann-Whitney does not discriminate between them.
- Upper bound of lattice includes only 2 minterms. Mann-Whitney does not differentiate: $\alpha \sim 0.21$
- FIFO queuing, no backoff, both low seed and low bound, with either:
  - No SQ, or
  - SQ and SQuID, but no NW.

Throughput, Low Traffic

- Upper bound of lattice includes 38 of 64 minterms, Mann-Whitney does not discriminate: least $\alpha \sim 0.29$.
- Lower bound of lattice includes 7 minterms, Mann-Whitney distinguishes two worst cases:
  - FIFO Queuing, SQ, SQuID, no backoff, Both RTO bound and SRTT seed low.

Throughput, Spiked Load

- Upper bound of lattice includes 31 of 64 minterms,
  Mann-Whitney does not discriminate: $\alpha \sim 0.15$.

- Lower bound of lattice includes only 2 minterms,
  Which Mann-Whitney does not differentiate: $\alpha \sim 0.25$.

- Worst throughput with FIFO Queuing, SQ, SQuID,
  No backoff,
  Both RTO bound, and seed low (same as for low traffic).

Throughput, High Traffic

- Upper bound of lattice includes 36 of 64 minterms, Mann-Whitney does not discriminate: least $\alpha \sim 0.24$.
- Lower bound of lattice includes 7 minterms, Mann-Whitney strongly discriminates two cases
- FIFO Queuing, SQ, SQuID, no backoff, both RTO bound, and seed low (same as for low and spiked traffic).

Conclusions

- Even with "low" traffic, high seed helps delay and throughput.
- SQuID with no backoff, low seed and low bound is bad for throughput.
- In some cases, more is not better for congestion control.

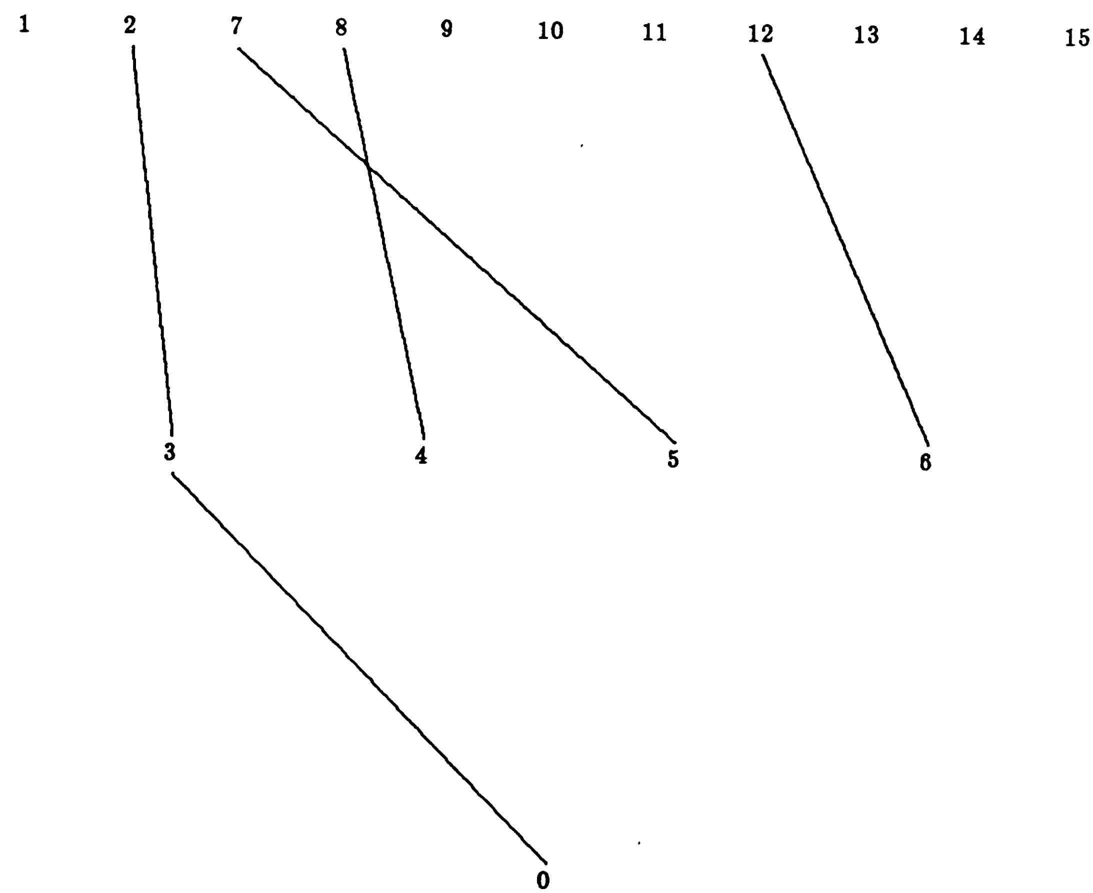
Narrowing Focus

- Motivation:
  - Underestimates for SRTT seed a known blunder.
  - Forced high RTO bounds unlikely to proliferate.
- Area of interest:
  - Performance with low bound, high seed, or vice-versa.

Narrow Focus

- No conclusions in low traffic scenarios:
  - No significant difference among mixed seed and bounds, for delay or throughput.
- Same lack of discrimination for high traffic.
- However, spiked load is perhaps most important.

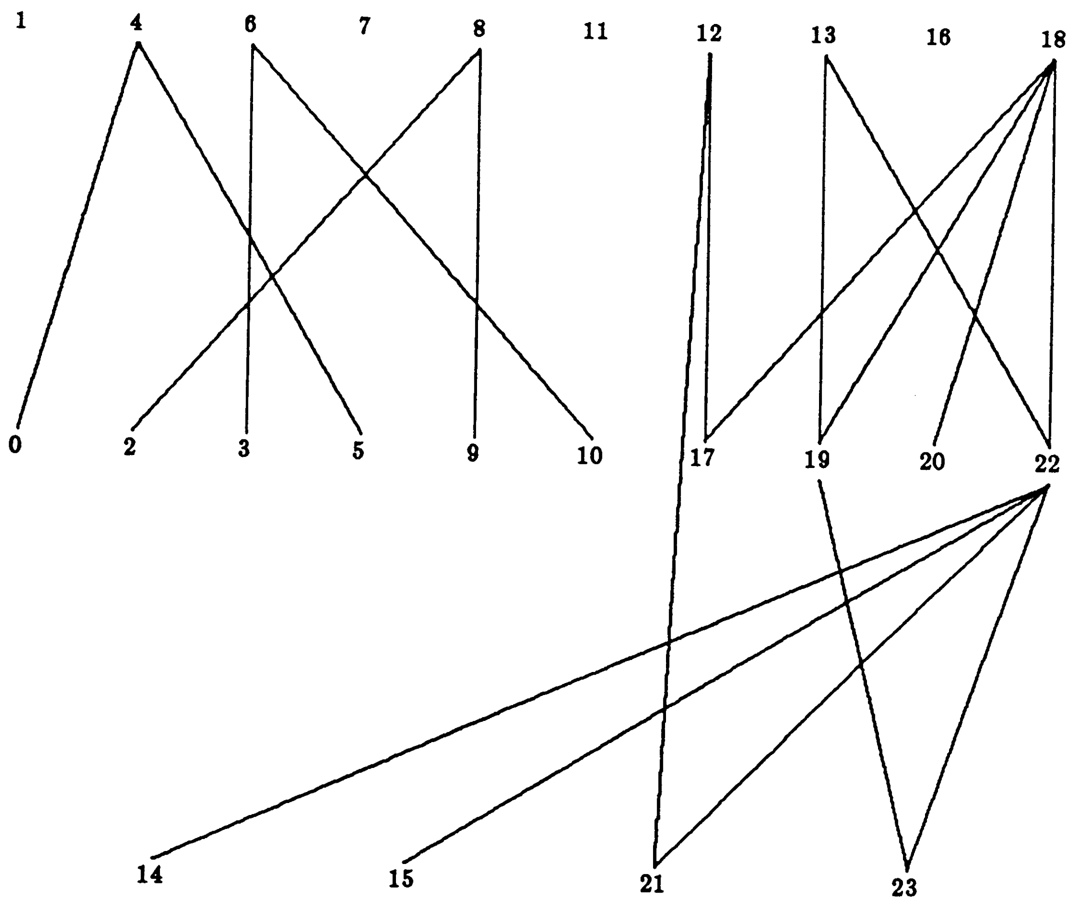
Select cases, Thruput, high traffic



Narrow Focus: Least Delay, Spiked Load

- Lower bounds include 16 equivalence groups, Some are also upper bounds!
- Mann-Whitney discriminates 4 best equivalence groups, six distinct minterms:
- Least Delay with Fair Queuing and backoff, with either
  - No SQ or
  - SQ and NW.

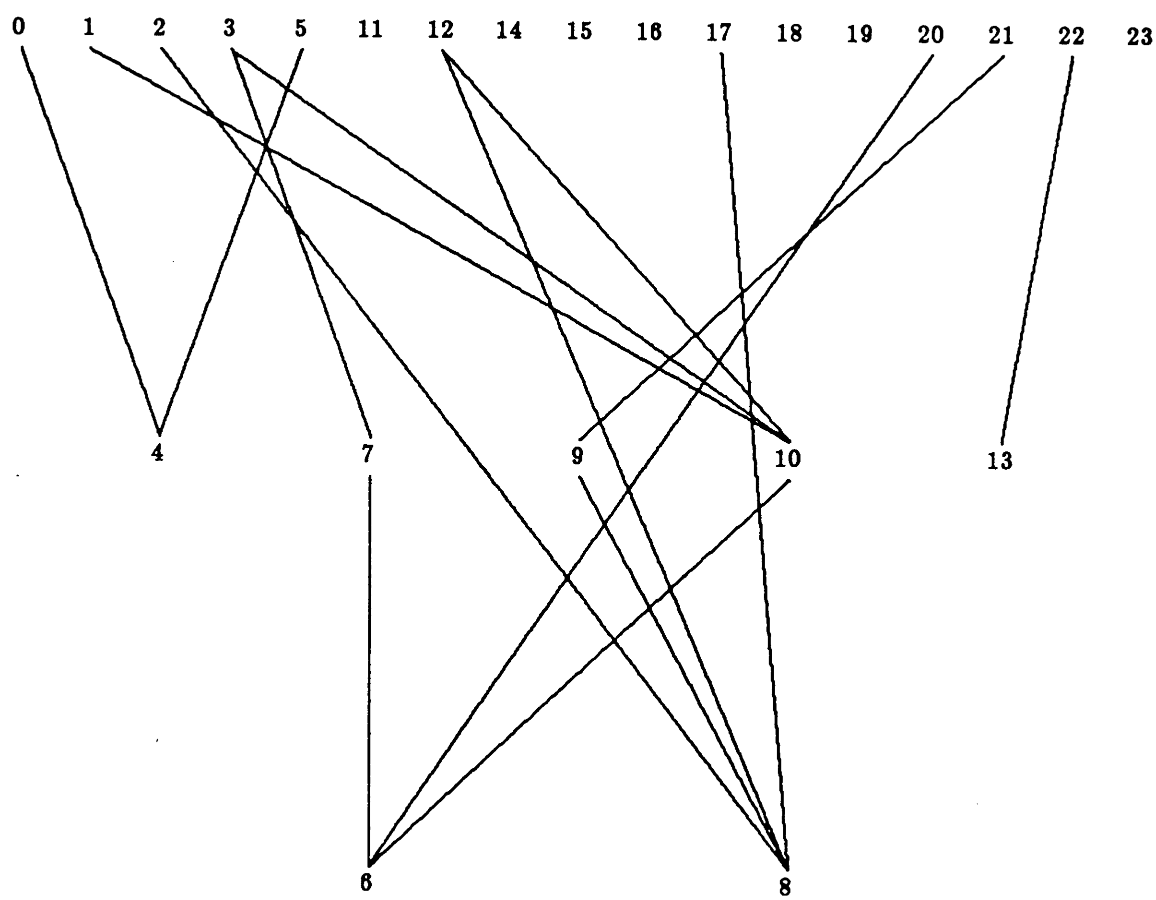
Delay, selected cases, spiked load



Narrow Focus: Greatest Delay, Spiked Load

- Upper bounds include 13 minterms, Mann-Whitney fails to discriminate.
-
- Since lattice is disconnected, assumption that lower bound identifies worst performance is dubious.

Throughput, Spiked Load, Selected Minterms



Narrow Focus: Worst Throughput, Spiked Load

- Lower bounds include 11 equivalence groups, 14 out of 32 minterms.
- Mann-Whitney does not discriminate between groups.
- As with delay, lattice is disconnected.

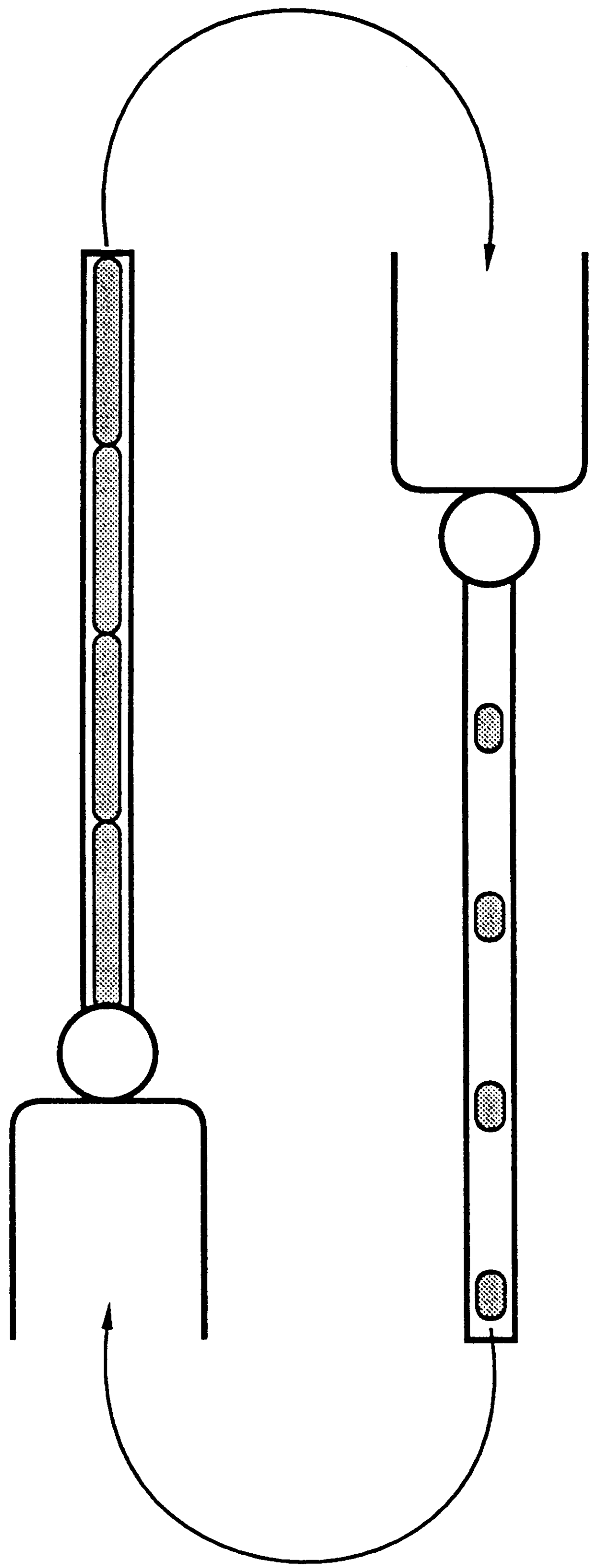
Conclusion

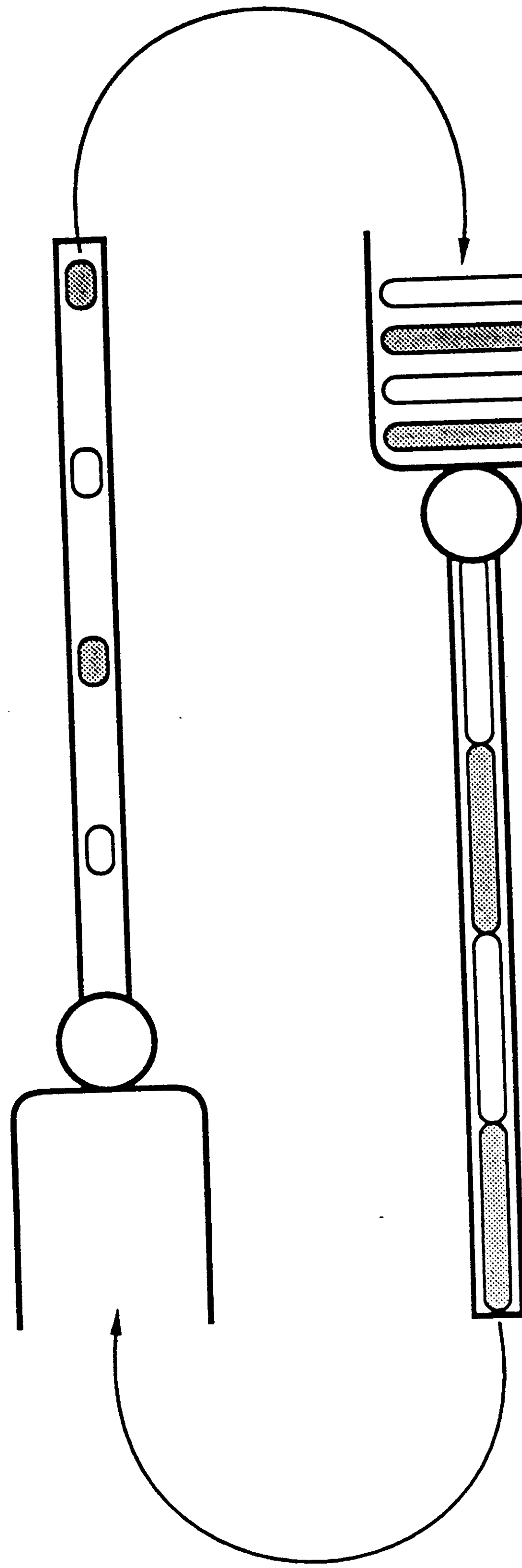
- Overall performance best with Fair Queuing, backoff, SQ, and NW.
- Delay worst with SQuID and no backoff.

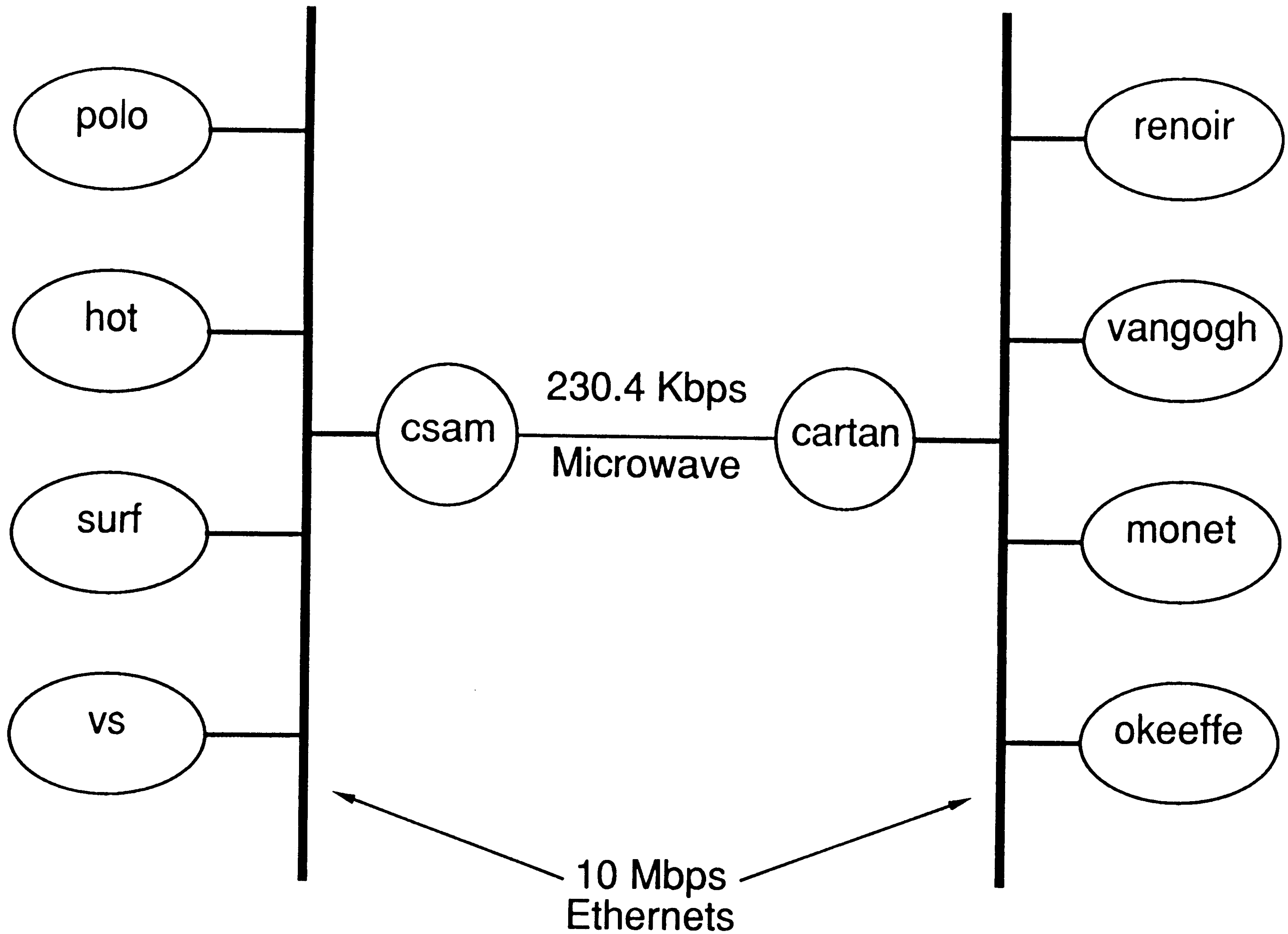
Recent Congestion Control Efforts for 4.2/4.3BSD

Van Jacobson, Lawrence Berkeley Labs

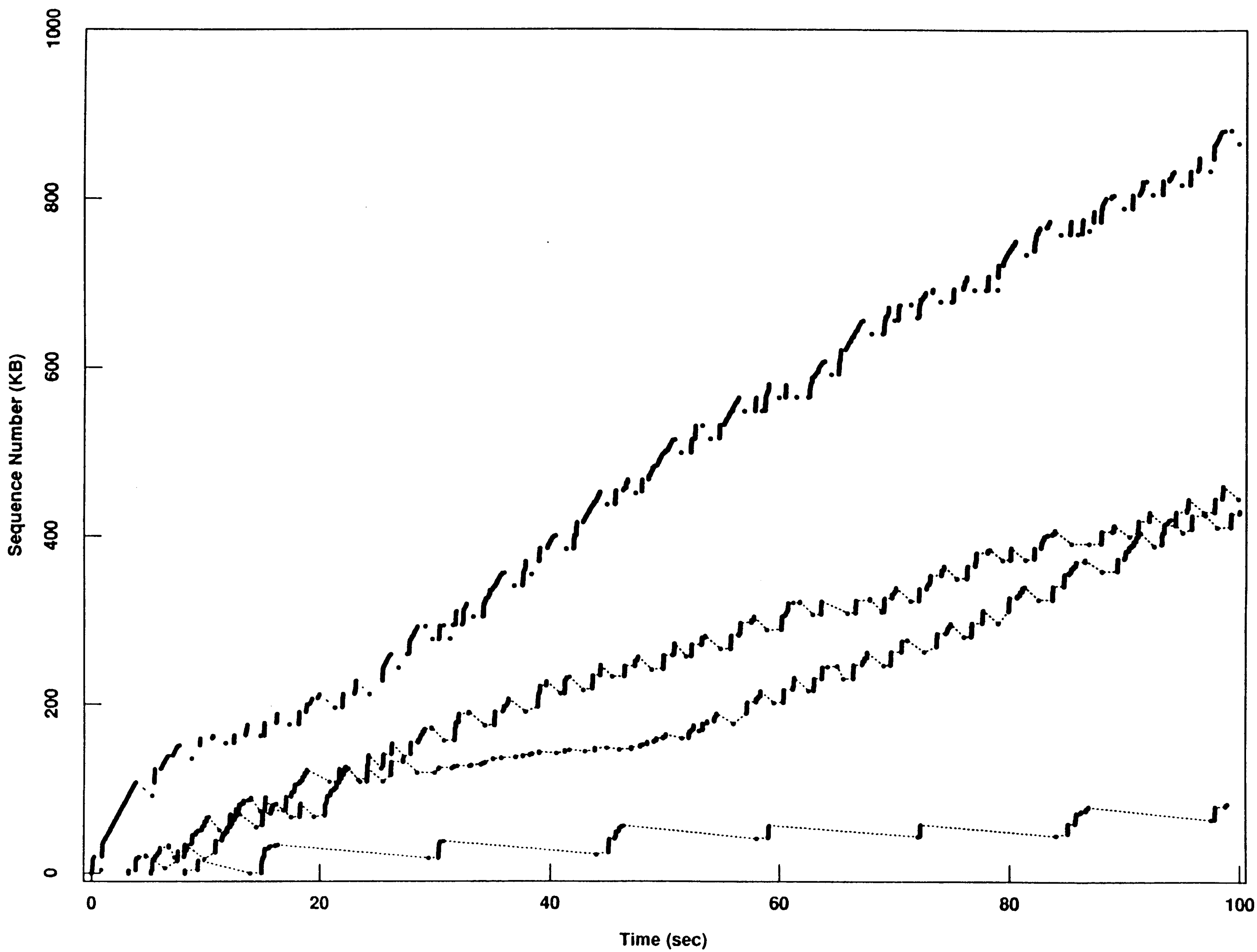
- Better statistics.
- Retransmit timer based on RTT mean and variance.
- Exponential retransmit backoff.
- Phil Karn's clamped retransmit backoff.
- Slow start.
- Better receiver ack policy.
- Dynamic window sizing based on congestion.
- Fast Retransmit.

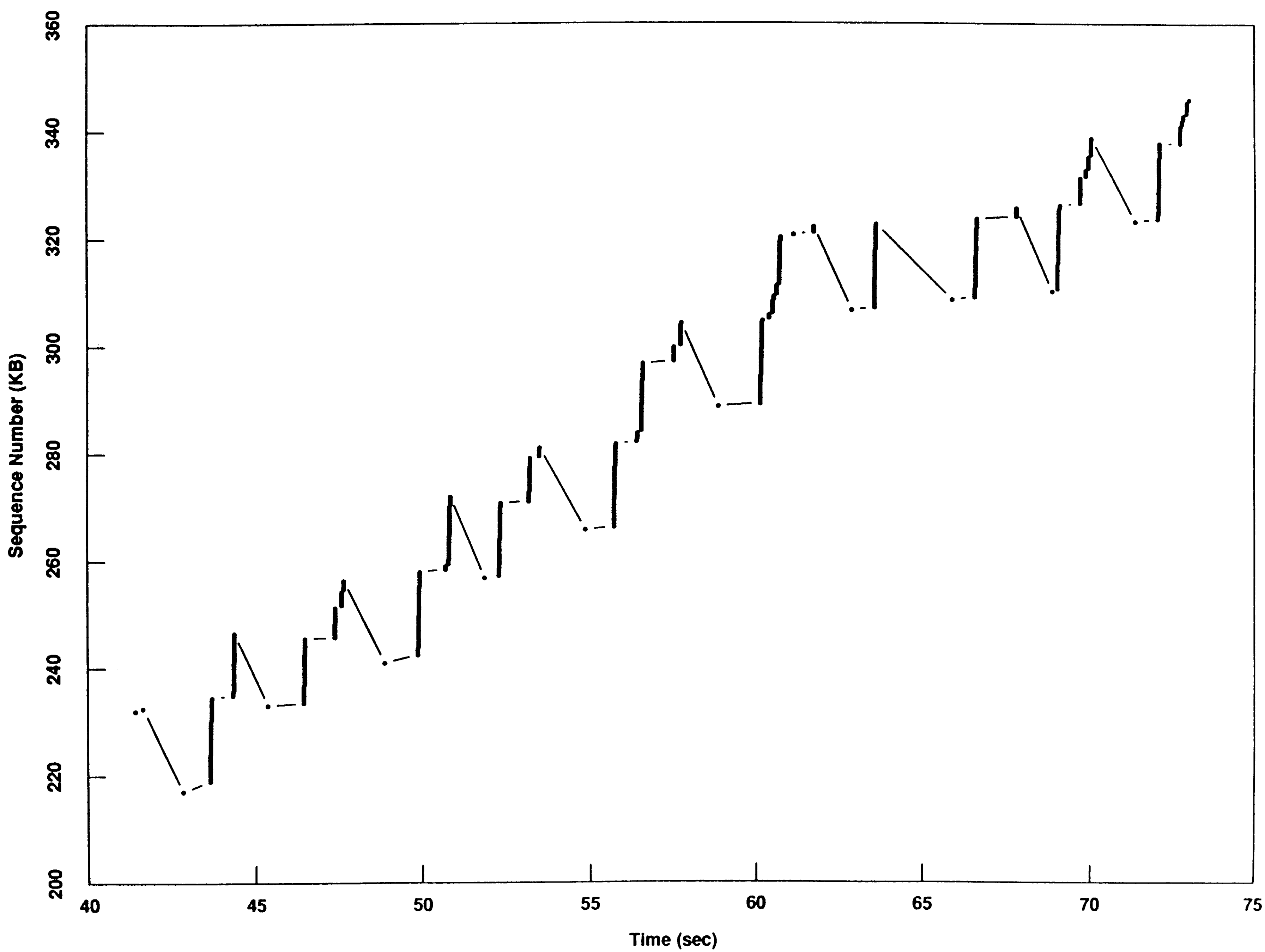




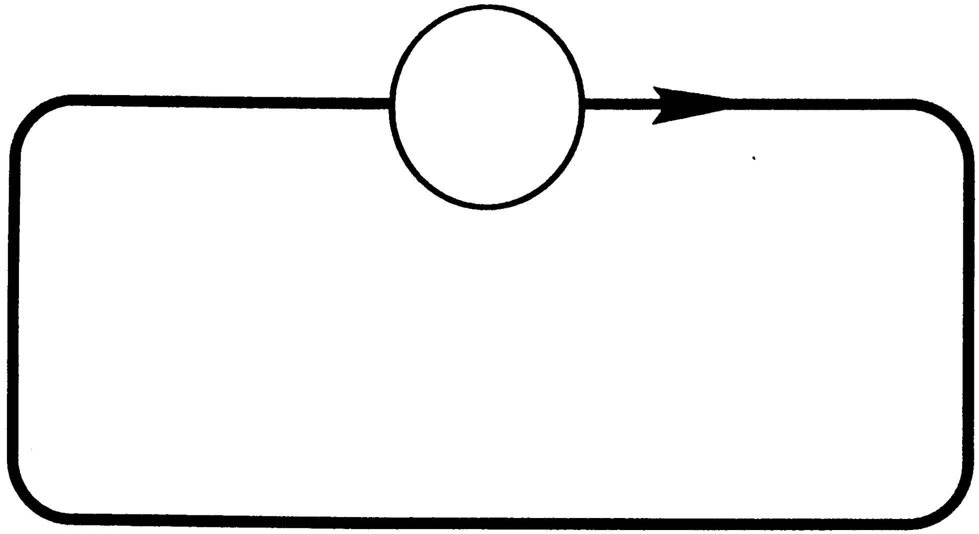

polo
hot
surf
vs
csam
230.4 Kbps
Microwave
cartan
renoir
vangogh
monet
okeeffe
10 Mbps
Ethernets

Stability Tests - Original 4.3 TCP



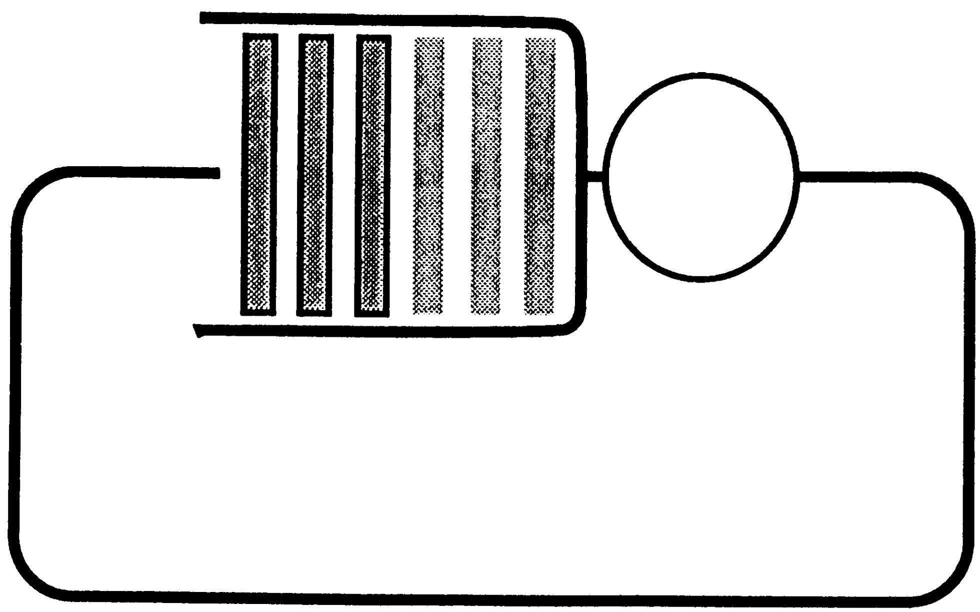


Sequence Number (KB)
360
340
320
300
280
260
240
220
200
40
45
50
55
60
65
70
75
Time (sec)



$$R_i = \alpha$$

$$R^* = \alpha$$



$$R_i = \alpha + \beta R_{i-1}$$

$$R^* = \frac{\alpha}{1-\beta}$$

Adding "Slow Start" to TCP

Add "congestion window" `cwnd` to tcp connection state.

On Retransmit timeout:

```
cwnd = maxseg;
```

When new data acked:

```
cwnd += maxseg;
```

When checking if output possible:

```
win = MIN(cwnd, snd_wnd);
```

Adding Dynamic Window Sizing to "Slow Start"

Add "loss threshhold" `thresh` to tcp connection state.

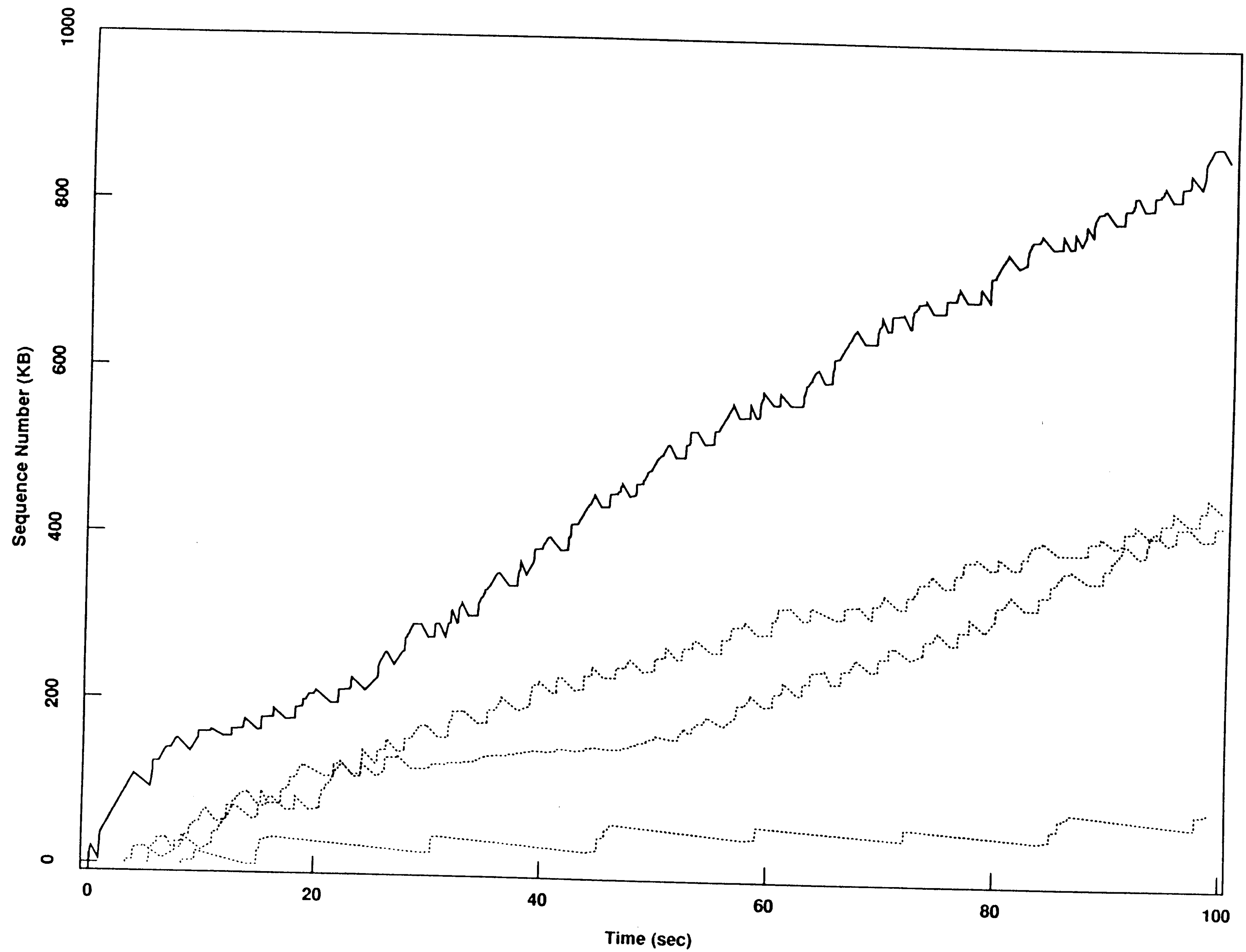
On Retransmit timeout:

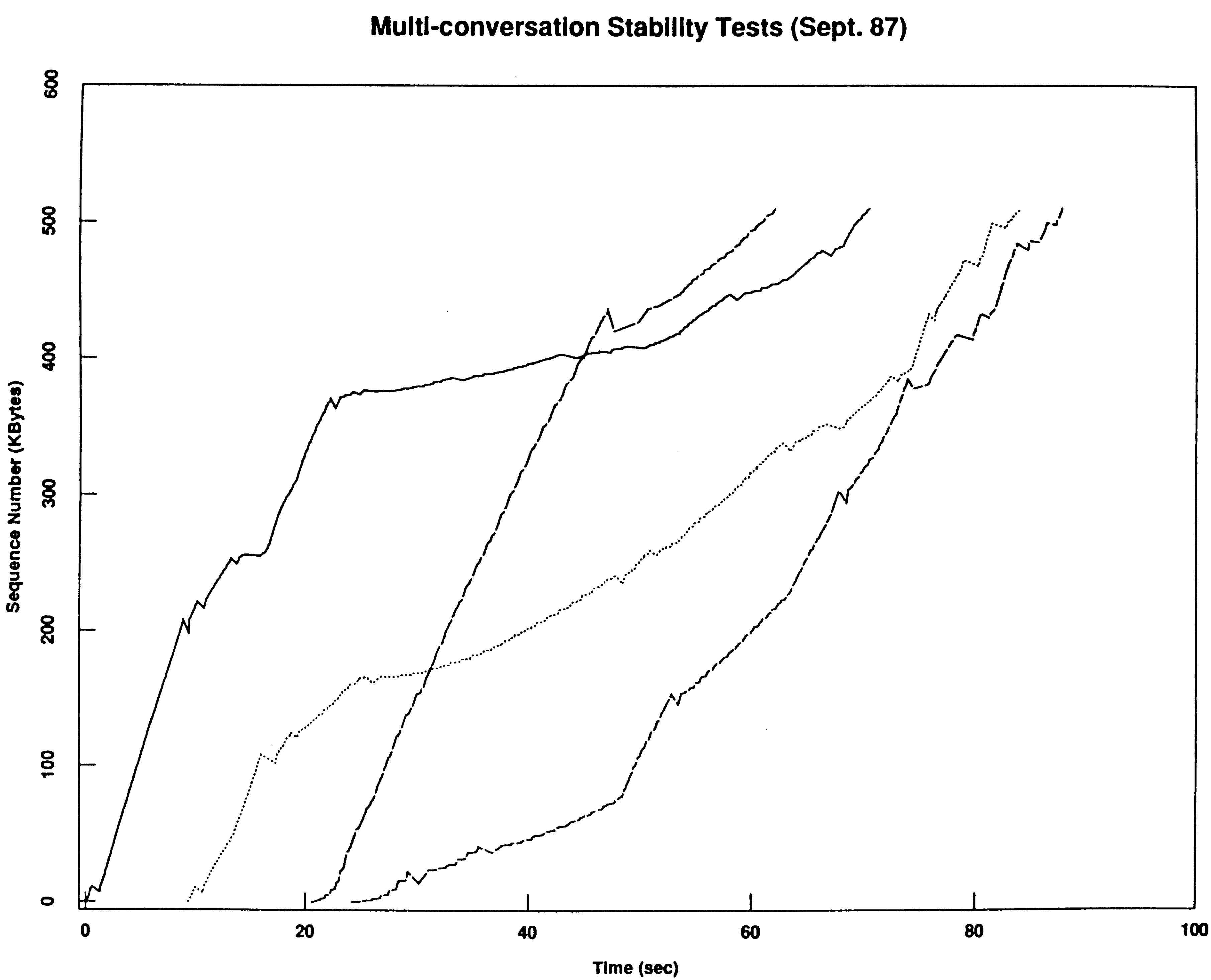
```
thresh = MIN(cwnd,snd_wnd)/2;
cwnd = maxseg;
```

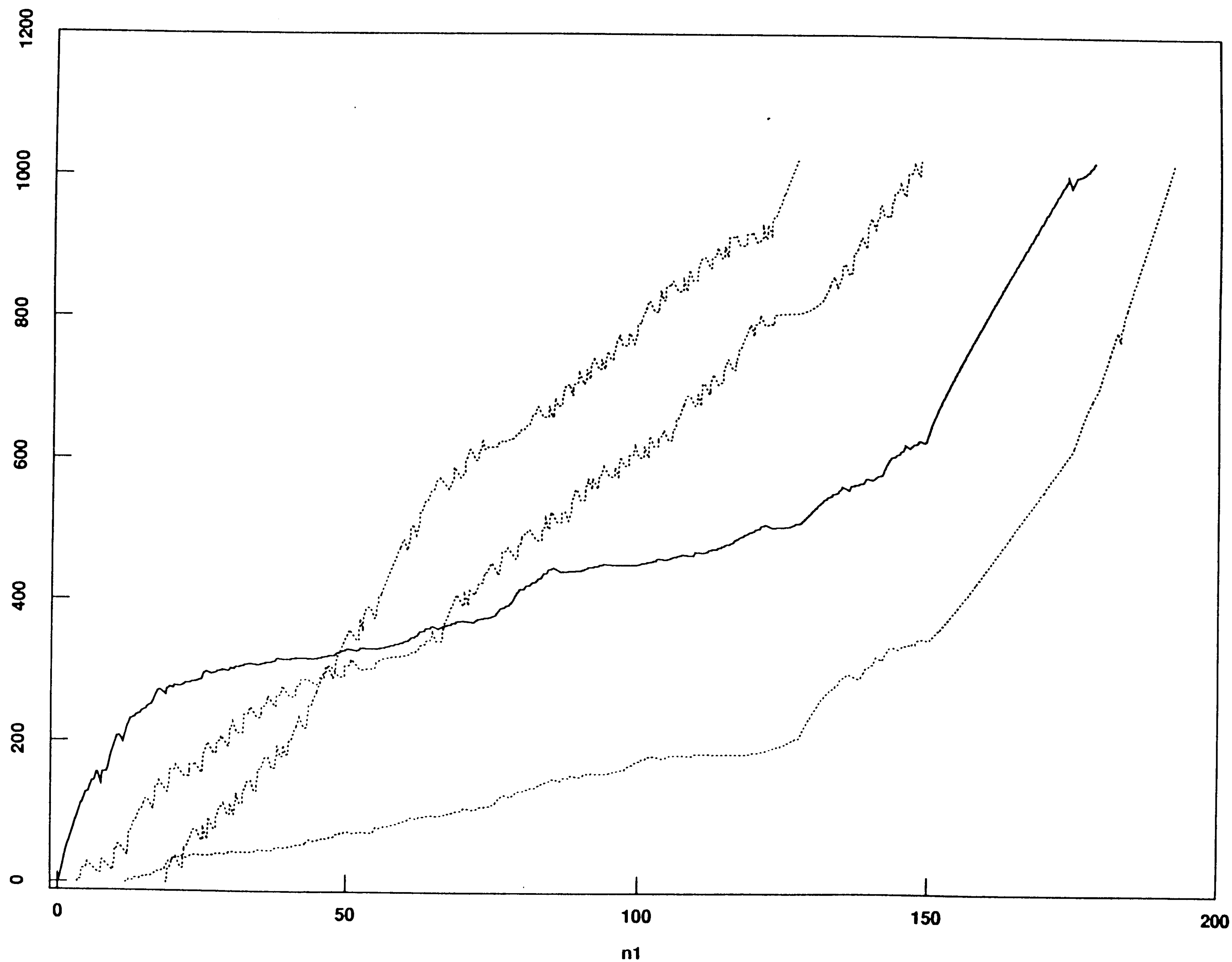
When new data acked:

```
if (cwnd < thresh)
    cwnd += maxseg;
else
    cwnd += maxseg*maxseg/cwnd;
```

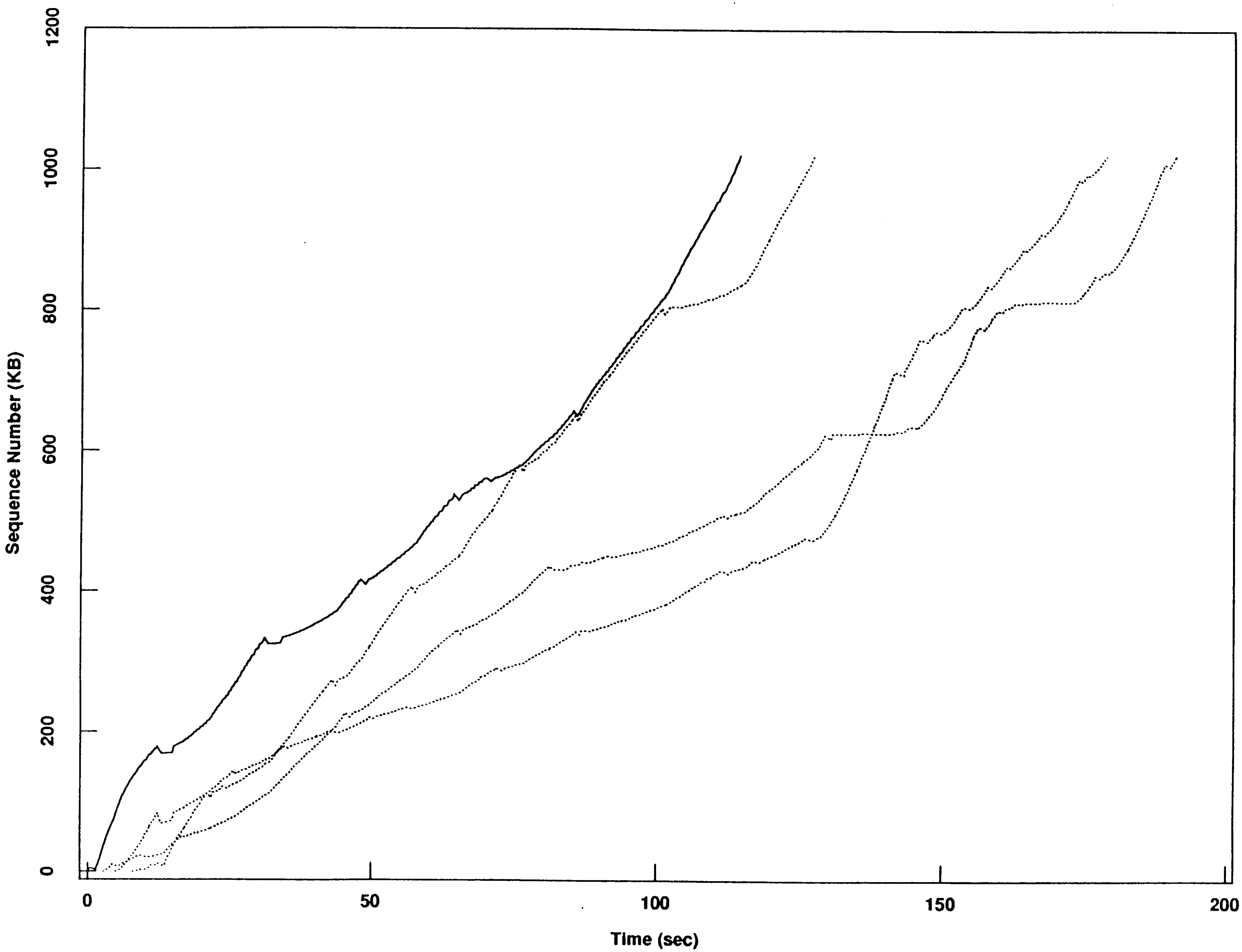
Stability Tests - Original 4.3 TCP




Multi-conversation Stability Tests (Sept. 87)
Sequence Number (KBytes)
0
100
200
300
400
500
600
0
20
40
60
80
100
Time (sec)


0
200
400
600
800
1000
1200
0
50
100
150
200
n1

Stability Tests: New-New + New-Old



Network Operating Center Tools Working Group

Jeff Case, University of Tennessee, Knoxville

NOC TOOLS W.G.

Charge:

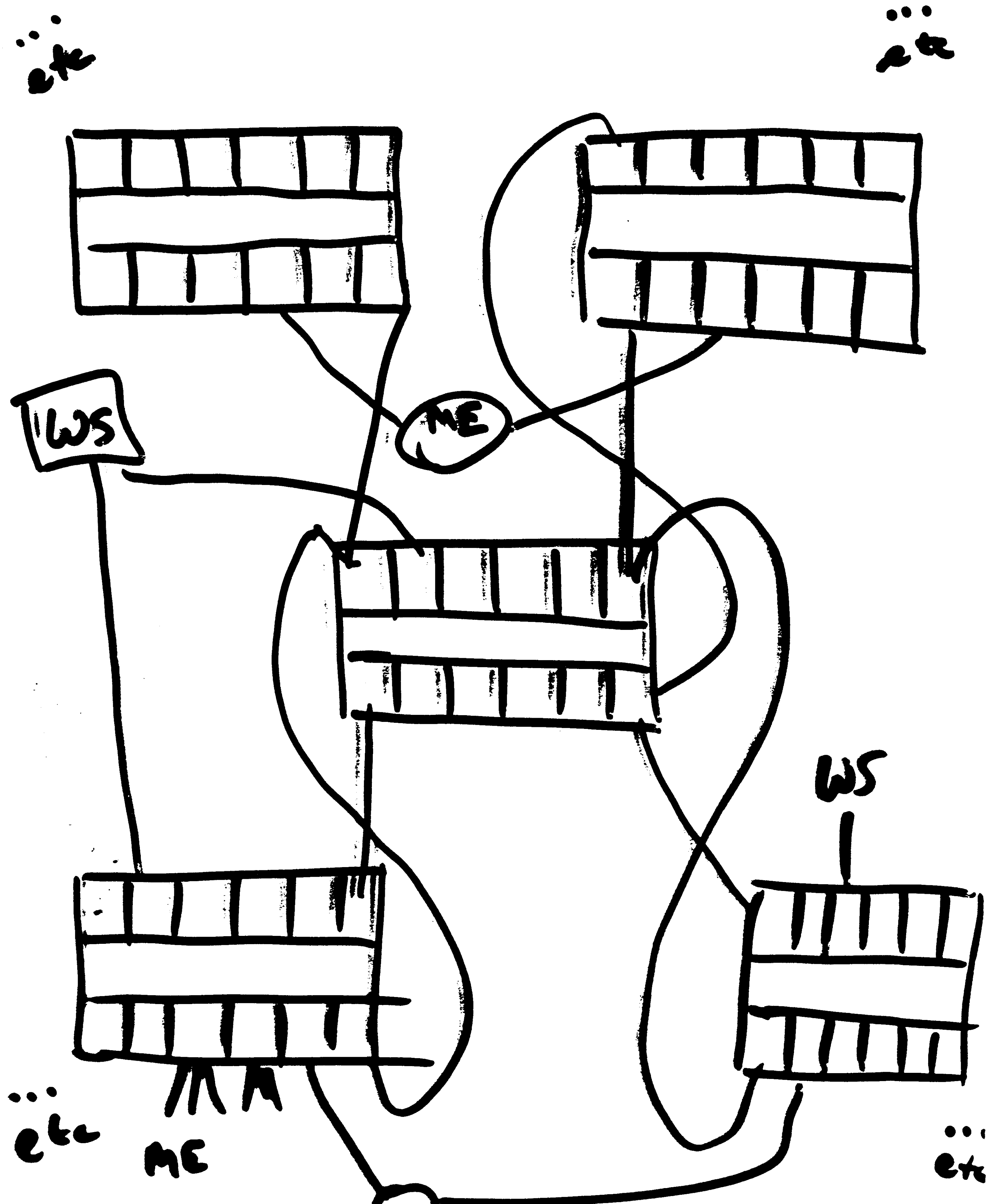
- Define + design needed NOC network monitoring + control applications
- Prepare + test some rapid prototypes
- Focus to be on tools
  - NOT skills
  - NOT resources

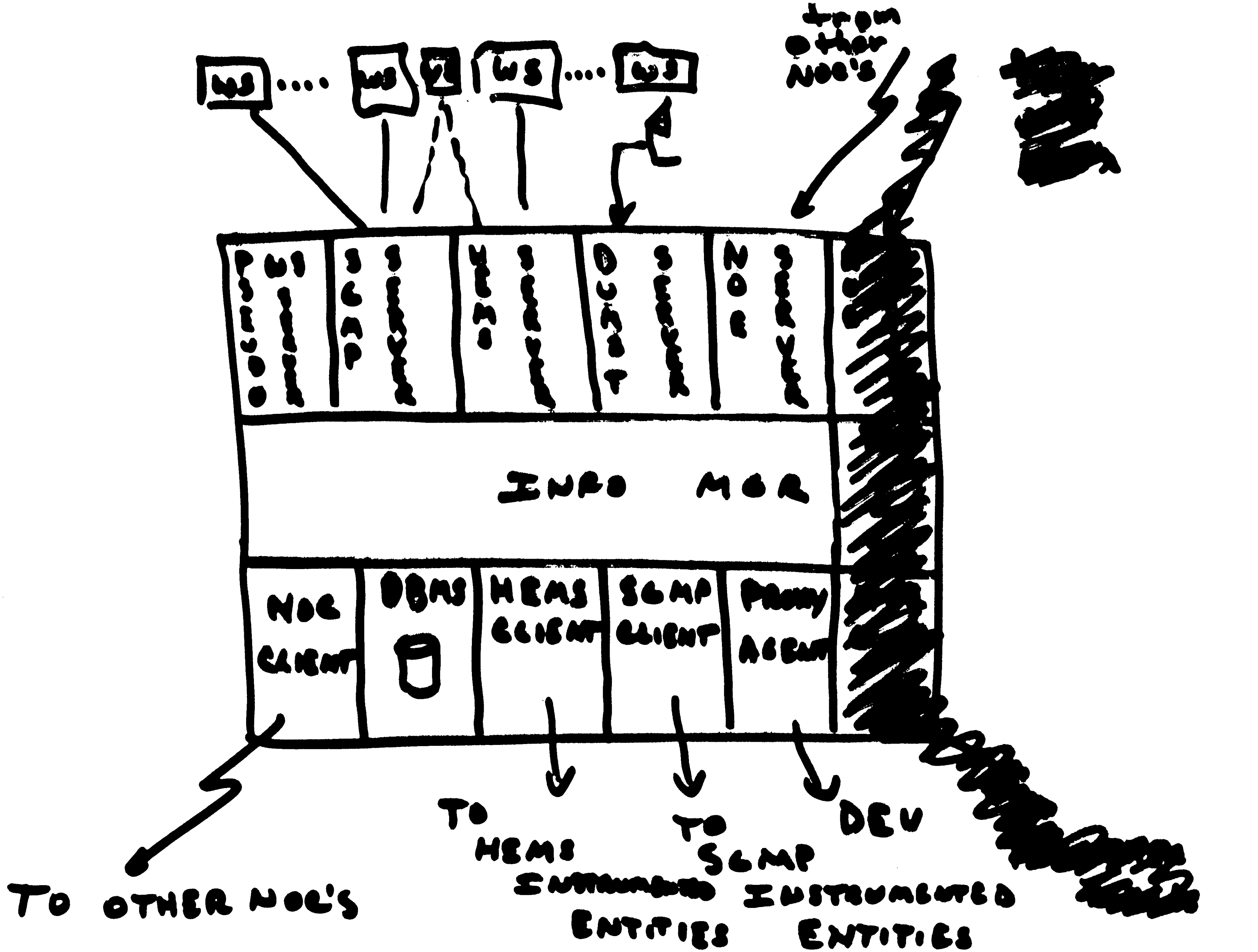
Scope:

- monitoring <u>and</u> control
- end to end
- IP
- Proxy via IP

Activities:

- Mailing list
- future meetings


... etc
... etc
WS
ME
WS
... etc
ME
... etc


WS
WS
WS
WS
from other NOC's
PSEUDO WS SERVER
SGMP SERVER
HEMS SERVER
NOC SERVER
INFO MGR
NOC CLIENT
DBMS
HEMS CLIENT
SGMP CLIENT
PROXY AGENT
TO OTHER NOC's
TO HEMS INSTRUMENTED ENTITIES
TO SGMP INSTRUMENTED ENTITIES
DEV

InterNICs Working Group

Elizabeth Feinler, SRI-NIC

INTERNICS

SRI International

GENERAL PURPOSE:

- TO DEFINE THE ROLE OF NETWORK INFORMATION CENTERS IN AN INTERNET ENVIRONMENT
- TO SHARE KNOWLEDGE AND RESOURCES
- TO DEVELOP TOOLS, STDS, PROCEDURES FOR EXCHANGE OF INFORMATION BETWEEN NICS

IETF PURPOSE:

- TO ASSIST WITH INFORMATION FLOW
- TO ALLEVIATE INFO CONGESTION
- TO DEVELOP INTERNET INFO EXCHANGE SERVICES

WHAT HAS HAPPENED SO FAR

SRI International

1985	-	!%@ AT SRI, CA
APR 87	-	DDN NIC PERSONNEL MET WITH NSFNET NIC REPS AT BBN
	-	METANICS MTG SRI WASHINGTON
SUMMER 87	-	BEGAN WORK ON NIC HANDBOOK TEMPLATE
OCT 87	-	JOINT DSAB/INTERNICS MTG, SRI
	-	INTERNICS MTG AT EDUCOM, LA
	-	IETF MTG, BOULDER

CURRENT ACTIVITIES

- PREPARING HANDBOOK OF NICS INCLUDING STD DATA TEMPLATE
- FEEDBACK DSAB PROTOCOL
- DEVELOPING AN ARCHITECTURE FOR A DISTRIBUTED WHOIS WHICH INCLUDES THE NICS
- POLLING USERS FOR IDEAS
- DESIGN DOCUMENT, FEASIBILITY, OPTIONS
- IMPLEMENTATIONS

HIGHLIGHTS LA MEETING

- DSAB PROTOCOL GOOD START;
  - NOT COMPLETE
  - ADMINISTRATIVELY NAIVE
  - DOESN'T AGREE WITH NBS/OSI
- GROUP AGREED TO ATTEMPT TO IMPLEMENT INTERNIC WHOIS SERVICE(S) USING EXISTING PROTOCOLS
  - "WHITE PAGES" FIRST BECAUSE WELL-KNOWN, REASONABLY STANDARDIZED DATA
  - "YELLOW PAGES" LATER
- BASIC "ATOM" OR TEMPLATE DESCRIBING AN INDIVIDUAL SHOULD BE IN CONTROL OF INDIVIDUAL
  - INDIVIDUAL CAN DELEGATE CONTROL TO ORGANIZATION

HIGHLIGHTS (CONT)

- SYSTEM SHOULD BE ATTRIBUTE-BASED
- NOT SURE WHETHER ITS A GOOD IDEA TO COMBINE "HARD" DATA (MUST BE THERE TO OPERATE, eg HOST NAMES) WITH "SOFT" DATA, e.g. NICNAMES OF PEOPLE
  - ADMINISTRATION, USE, ETC, <u>VERY</u> DIFFERENT
  - MOST ATTEMPTS HAVEN'T WORKED
- NEED WAY TO USE INFO ALREADY IN EXISTANCE WITHIN ORGS
- MUST RESPECT INDIVIDUAL AND ORG'S PRIVACY AND RIGHTS

HIGHLIGHT'S (CON'T)

- ORGANIZATIONS FEED INFO TO NICS
  - IF NO ORG, INDIVS FEED INFO TO NICS
  - CAN ALSO USE INFO WITHIN ORG
  - CONSISTENT PROCEDURES, STANDARDS, DATA ELEMENTS
- NICS COMMUNICATE WITH EACH OTHER
  - GENERAL NICS
  - SPECIALIZED NICS
  - EACH GENERAL NIC KNOWS ABOUT OTHER NICS

HIGHLIGHTS (CONT'D)

- MUST BE ABLE TO HANDLE DIVERSE NEEDS, e.g.
  - BITNET (CENTRALIZED)
  - uucp (TOTALLY DECENTRALIZED)
- WANT REDUNDANCY AT HIGH LEVELS
  - FEW WRITERS
  - MANY READERS
  - SIMPLE BUT EFFECTIVE DATA BASE ADMINISTRATION

FUTURE

WHOIS

- AGREEMENT ON SCOPE
- DESIGN INSTRUMENT FOR FEEDBACK
- POLL USERS, OTHERS - DEC 87
- SUMMARIZE FEEDBACK - FEB 88
- WHITE PAPER ON TECHNICAL FEASIBILITY, OPTIONS, FEATURES - SPRING 88
- DESIGN DATA STRUCTURE, FURTHER DEVELOP PROTOCOL - SPRING 88
- BEGIN IMPLEMENTATIONS - SUMMER 88
- PROTOCOL, BASIC DATA STRUCTURE, ADMINISTRATIVE GUIDELINES, SOFTWARE TOOLS - FALL/WINTER 88

FUTURE

HANDBOOK OF NICS

- TEMPLATE REVISION - DEC 87
- TEMPLATE RETURNED - JAN 88
- 1ST HANDBOOK - SPRING 88

OTHER ACTIVITIES

- EXCHANGE INFO ON POCS
  EACH OTHER'S SERVICES
- ASSIST NEW NICS
- SHARE DATA/INFO
- TRY NOT TO REINVENT WHEELS

...... IN SHORT

NETWORK!

Domain Working Group

Mark Lottor, SRI-NIC

Domain Working Group

- identify and fix current problems
- work on Milnet domain transition
- look at future extensions to domain system

HOSTS.TXT

	Oct 1986	Oct 1987
Hosts	3,295	5,235
Networks	524	736

Current host table size = 525,000 bytes
3 years ago = 120,000 bytes

Current Domain System Size

Top-level domains =	25	
2nd-level domains =	380	
Hosts still in .ARPA =	2210	
		149 (net 10)
		1219 (net 26)
		842 (others)
Hosts in.COM =	382	
Hosts in .EDU =	2328	
Hosts in .GOV =	155	
Hosts in .IL =	1	
Hosts in .MIL =	116	
Hosts in .NET =	15	
Hosts in .ORG =	18	
Hosts in .UK =	10	

Root Servers		
Server	**Status**	**Networks**
SRI-NIC.ARPA	up	arpanet, milnet
A.ISI.EDU	up	milnet
BRL-AOS.ARPA	up	milnet
C.ISI.EDU	going away	arpanet
GUNTER-ADAM.ARPA	new	milnet
NS.NASA.GOV	new	milnet
C.NYSER.NET	new	nysernet
TERP.UMD.EDU	new	arpanet

Bind

- Mike Karels contracted to do work
- Domain Working Group to provide list of problems, changes, and mil specific items
- Made list of current problems to fix

  ignore zone file errors

  lots of problems with zone transfers

  add negative caching

Milnet Domain Transition

- formalize naming proposal (NIC + StJohns)

  A.MIL army
  AF.MIL air force
  N.MIL navy
  CG.MIL coast guard
  MC.MIL marine corps

  DCA.MIL
  DARPA.MIL
  DDN.MIL

- announce in DDN Mgt. Bulletin that mil hosts can change name

- all hosts out of .ARPA (one year)

- provide domain servers (NIC at first)

RFC's

Queued

Domain Transition Overview

Domain Administrators Guide

Domain Server Operations Guide

Domain Names - Concepts and Facilities
(obsoletes RFC 882)

Domain Names - Implementation and Specification
(obsoletes RFC 883)

Future

Responsable Person RR

Mailbox RR usage

EGP3 Working Group

Marianne Gardner, BBN

EGP, version 3

Features
- incremental Rtg updates
- version negotiation
- Polls replace hellos
- active / passive stays
- does <u>not</u> fix tree topology

goal
reduce overhead

Status
Draft to group - next week
pseudo-code - two weeks

7.0 Distributed Documents

The following documents and papers were distributed at the meeting. As indicated, a number of them are drafts. For copies or additional information, please contact the authors or the SRI Network Information Center.

The Profile Naming Service
(Larry Peterson, University of Arizona)

A Descriptive Naming Service for the
DARPA/NSF Internet (Larry Peterson, University of Arizona)

Kerberos Authentication and Authorization System
Project Athena Technical Plan (S.P. Miller et al)