Top ten things network engineers wish grid programmers knew

Jon Crowcroft

Jon.Crowcroft@cl.cam.ac.uk

http://www.cl.cam.ac.uk/homes/jac22

Tuesday, 10.30, July 23, 2002

GGF-5, Edinburgh, Scotland

Abstract

- This is a draft contribution for a document for the GHPN (see `http://www.csm.ornl.gov/ghpn` amongst other places)

- List topics that the network community is working on and is sometimes asked alarming questions about by folks who make intensive (and quite well educated) use of networks, e.g. GGF¿

- Currently list of topics and references. May expand list, and it certainly needs lots of explanatory text.

- TBD: *Top ten things grid programmers wish network engineers knew*

1.Congestion Control, contrariwise: see QoS

Slow Start is this always necessary? no, but beware ISPs who mandate it, and if you think you can use less than recent history rather than recent measurements, look at Congestion Manager and TCP PCB state sharing work first!

Congestion Control This is not optional in a non QoS network (which is just about any network) - adaption is mandatory

AIMD and Equation Based AIMD is not the only solution to a fair, convergent control rule for congestion avoidance and control. Alternate: Rate based, using loss, or ECN feedback, can be TCP fair, but not generate characteristic Saw Tooth.

Assumptions and errors *Most* connections are not like Padhye equation; most bytes are shipped on a small number of connections, and do - c.f. Mice and Elephants.

Jury still out on whether there are non greedy TCP flows (ones who do not have infinite sources of data at any moment) (see HPDC talk!

RMT and Unicast Reliable Multicast Transport protocols (PGM, ALC) use a variety of techniques to mimic TCP mainly.

Mobile and Congestion Control Mobile nodes experience temporary indications of loss AND congestion during a hand-off. People have proposed mechanisms for indicating whether these are "true" or chimera.

Economics, Fairness etc Congestion control gives approximately fair distribution of bottleneck bandwidth - not great if you paid more to get fat pipe to net. But you are probably nearer core and have every right to ask the ISP to upgrade bottlenecks anyhow; people that paid less should be bottlenecked at THEIR access links in that case.

`http://www.psc.edu/networking/tcp_friendly.html`

2.Routing

Fast Forwarding Packet classification and switched routers have come a long way recently - we are unlikely in s/w to beat h/w in core routers, but can compete nicely in access devices - certainly, no reason why small cluster couldn't make good 10Gbps router - but there's every reason why a PCI bus machine maxes out at 1Gbps!

Faster Convergence Routers and links fail. the job of OSPF/ISIS and BGP is to find the alternate paths quickly - in reality they take a whole to converge - IGPs take a while (despite being mainly link state nowadays) because link failure detection is NOT obvious - sometimes you have to count missed HELLO packets (since some links don't generate an explicit clock). BGP convergence is a joke. But there are smart people on the case.

theory and practice Most the problems with implementing routing protocols are those of classic distributed (p2p/autonomous) algorithms: dealing with bugs in other peoples implementations - it takes a good programmer about 3 months to do a full OSPF. It then takes around *3 years* to put in all the defences.

Better (multi-path, multi-metric) routing Equal Cost Multipath OSPF and QOSPF have been dreamt up - are they used a lot? multipath in limited cases appears to work quite well. Multimetric relies on understanding traffic engineering and economics; hasn't seen the light of day. Note also tier one networks' end-to-end delays approaching transmission delays, so asking for delay (jitter) bound getting fairly pointless - asking for a throughput guarantee good idea, but doesn't need clever routing!

Does MPLS Help? Yes and No. For example, esxperience shows that for level 2 protection, and for provisioning of diffserve SLAs, MPLS can help. Other experience shows that some function (e.g. multicast) are not well supported on an MPLS sub-strate.

Policies are hard BGP allows one to express unilateral policies to the planet. this is cute (the same idea could be used for policy management of other resources like CPUs in the GRID) however, it results in difficulties in computing global choices (esp Multihoming) - there are fixes.

`http://www.potaroo.net/`,
`http://www.telstra.net/gih`, NANOG

3.Packet Sizes

- Go faster LANs always pushed the MTU up - since ATM LANs (remember the fore asx100) we tried 9280 byte packets, and enjoyed things. But the GRID is global, so the MTU is that of the weakest link. Most stuff is on 100BaseT somewhere on the path so not likely to see more than the occasional special case non 1500 byte path. However, with path MTU discovery, get that auto-magically

- Multicast MSS is a real problem:)

- Sub-IP packet size is a consideration - some systems (ATM) break packets into tiny little pieces, then apply various level2 schemes to these pieces (e.g. rate/congestion control) - most these are anathema to good performance.

`http://www.nlanr.net/NA/Learn/packetsizes.html`

`http://www.faqs.org/rfcs/rfc1191.html`

4. Overlays

- Overlays and P2p (e.g. Pastry, CAN, Tapastry) becoming commonplace - routing overlay du jour is RON from MIT - these (at best) are an auto-magic way of configuring a set of Tunnels (IPinIP, GRE etc). I.e. they build VPNs

- P2P: are slightly different - they do content sharing and have index/search/replication strategies varying from mind-numbingly stupid (napster, gnutella) to very cute (CAN, Pastry). Problems with Locality and Metrics so *not* the tool for the job for low latency file access....in trying to mitigate this, they (and overlay routing substrates) use ping and pathchar to try to find proximal nodes:

- Note limitations of Ping/Pathchar and Convergence when not native (errors/confidence

(similar to route link outage discovery problems)

Peer-to-Peer Harnessing the Power of Disruptive Technologies Edited by Andy Oram, March 2001, 0-596-00110-X

5. QoS (contrariwise: see Congestion Control)

QoS would be a nice thing

Parameters typically include Throughout, Delay, Availability. Some people add security/integrity; Some people also mention loss...

Threats Theft and Denial of Service

Offers Protection is really what people want - If I send x bps to site S, what y bps will be received, how much d later?

To guarantee

$$y = x$$

, and d is minimised, you need *Admission Control* (so we are not sharing as

we would if we adapted under congestion control) and *Scheduling* (so we do not experience arbitrary queueing delays)

Re-routing may also need to be controlled and pre-empted alternate routes (also known, unfortunately as protection paths) may be needed if we want QoS to include availability as well as throughput guarantees and delay bounds.

Network Structure *edge*, *core*, etc is a myth; in global net the average traffic path includes 7 ASs - most inter-domain traffic traverses heavily used Internet Exchange points (e.g. London) where capacity only just about matches demand; core networks are often *over-provisioned* (UK academic net now runs at

$$< 5\%$$

utilisation). Wont be true when we all install Gig and 10Gig Ethers at our cluster/server farm sites!

Aggregation technique to scale traffic management for QoS - by only managing classes of aggregates of flows, we get to reduce the state and signaling/management overhead for it. VPNs/tunnels of course are aggregation techniques, as are things that treat packet differently on subfields like DSCP, port etc etc

Network Structure *edge*, *core*, etc is a myth; in global

SLAs are around already despite non widespread QoS - however, SLAs are only intra-ISP to my knowledge (some Internet Exchanges offer SLAs but end 2 end SLAs are as scarce as dragons).

Economics - are important here again as you can imagine!

Reference: An Engineering Approach to Computer Networking Keshav, 1997, Addison-Wesley Pub Co; ISBN: 0201634422, Internet QoS: Architectures and Mechanisms for Quality of Service by Zheng Wang, 2001, Morgan Kaufmann Publishers; ISBN: 1558606084

6. Multicast

- Tier 1 routing works. Most ISPs run core native multicast

- Interdomain only just limps (its getting better... MSDP Problems, App Relay Solutions

- RMT - we have some candidate protocols for reliable multicast - nothing as solid as 1988 TCP quite yet tho.

- Address Allocation and Directories are not great yet, hence beacons and so on.

- Access Network are in bad shape...e.g. DSLAMs dont do IGMP snooping; Cable dont do IGMP snooping;Dialup cant hack it at all

- Does IPv6 Help (don't laugh!) - yes it might!

Developing IP Multicast Networks: The Definitive Guide to Designing and Deploying CISCO IP Multicast Networks by Beau Williamson, 2000, Cisco Press; ISBN: 157870077

Interdomain multicast solutions guide, Cisco Press, ISBN 1-58705-083-8

Multicast Communication: Protocols, Programming, and Applications by Ralph Wittmann, Martina Zitterbart Morgan Kaufmann Publishers; ISBN: 1558606459

7.Operating Systems

- Linux, Solaris etc...there's a lot we could say here - lots of things can and should be configured - see www.psc.edu for a LOT Of help

- zero copy stack - we'd all like this - zero copy receive is hard; RDMA is not obviously the answer

- Interrupts (self selecting NICs) we should minimises these if we want TCP to go to 10Gbps on a reasonable processor - there are nice techniques

- socket buffer considerations -there are lots!

- protection and scheduling domains - if we could get away from OSs that confused these , life would be easier!

References: W Richard Stevens, TCP/IP Illustrated, All Volumes, Understanding the Linux Kernel, D.P. Bovet and M. Cesati, O'Reilly, 2001, ISBN 0-596-00002-2

8.Layer 2 Considerations

- layer 2 NBMA nets - lots - a pain
- layer 2 shared media nets - was decreasing due to switched ether, now increasing due to wireless.
- switching and routing re-cursed - layer 2 switching and routing usually makes life HARDER for the IP engineer.
- flow and congestion control re-cursed - layer 2 reliability and flow control almost ALWAYS make life worse for the IP and TCP engineer.
- signaling (implicit, explicit) is just painful.

http://www.apple.com/ibook/wireless.html

http://www.ietf.org/html.charters/pilc-charter.html

http://www.cl.cam.ac.uk/Research/SRG/netos/coms/index.html

http://www.cis.ohio-state.edu/ jain/refs/opt_refs.htm

9.Light v. Heavyweight Protocols

Header prediction Packet templates make Code complexity a lot lower in the common case even for a big protocol like TCP or SCTP.

User space **v. kernel myths** - in this authors experience it is really worth getting people to put transports into the kernel - reasons include independent failure of application and protocol as well as good control of end system resources. It ain't that hard and user space will just almost never be as fast. Research OSs (nemesys, xenoservers, Scout) for which this is not true, but they are not in mainstream COTS yet.)

Computer Networks, A Systems Approach Peterson and Davie, Morgan Kaufmann, 1996, ISBN 1-55860-368-9 2nd ed

10. Macroscopic Traffic and System Considerations

Self similarity traffic is self similar (i.e. arrivals are not i.i.d) - this doesn't actually matter much (there is a horizon effect)

traffic phase effects p2p (IP router, multiparty applications etc) have a tendency (like clocks on a wooden door, or fireflies in the mekong delta) to synchronise :- this is a *bad* thing

flash crowds e.g. genome publication of new result followed by simultaneous dbase search with similar queries from lots of different places...

Asymmetry Many things in the net are asymmetric - see ADSL lines, see client-server, master-slave, see most NAT boxes. See

BGP paths. beware - assumptions about symmetry (e.g. deriving 1 way delay from RTT) are often wildly wrong. Asymmetry also breaks all kinds of middle box snooping behaviour.

The Art of Computer Systems Performance Analysis Raj Jain, 1991, Wiley, ISBN 0-471-50336-3

Web Protocols and Practice B. Krishnamurthy & J. Rexford, Addison Wesley, 2001, ISBN 0-201-710885

Security Engineering, Ross Anderson, 2001 Wiley & Sons; ISBN: 0471389226

Global Reference

- ACM CCR 25th Anniversary Edition, ACM SIGCOMM CCR, Volume 25, No.1 January 1995, ISSN #: 0146-4833 http://www.acm.org/

- J. Sterbenz, J. Touch, "High-Speed Networking: A Systematic Approach to High-Bandwidth Low-Latency Communication," John Wiley & Sons, April 2001, ISBN: 0471330361. *http://catalog.wiley.com/remtitle*

- "Computational Grids: The Future of High-Performance Distributed Computing," eds. I. Foster and C. Kesselman, Morgan Kaufmann, ISBN 1-55860-475-8, July 1998. http://www.mkp.com/grids/

More, you want MORE?

- Just as for networks, we can do the same for classical distributed systems;

- I just read the special issue of Kluwer's Cluster Computing Journal on Grid Computing (see `http://www.kluweronline.com/issn/1386-7857` for details)

- many papers contain amazing lacunae about tangable results; amount of work done in the 80s and 90s in distributed systems and networks communities that is simply ignored (unreferenced).

- Berners-Lee created www so we could find scientific papers- in all cases a quick search on google or citeseer found relevant work highly cited elsewhere - work with performance results and code (and even work on optimisation)... (n.b. this is 5 mins reading and searching)...

A.Online Prediction of the Running Time of Tasks

Peter A. Dinda

- actually this paper is ok, except it really ought to discuss how you avoid the Halting Problem!

B.Condor-G: A Computation Management Agent for Multi-Institutional Grids

James Frey, Todd Tannenbaum, Miron Livny, Ian Foster, Steven Tuecke

- there is large body of policy work (c.f. policy workshops run by Morris Sloman et al) including the successful deployment of policies In the internet - looking at that as a nice simple example (avoiding all the messy stuff about roles and so on)

- Inter-domain routing with BGP includes several very simple but powerful ideas - the ideas that policies are expressed locally about ingress, egress and transit, but are then carried globally (subject to filtering by BGP path attributes is very

cool - it allows massive scale systems because it permits inconsistency, but doesnt breat

- there are problems with BGP but they are due tro the path vector algorithm and updating, not due to the policy computations)- the equilavlent is completey missing from most thinking i've seen about multi-adminsitratic computational economies...

C. NAS Grid Benchmarks: A Tool for Grid Space Exploration

Michael Frumkin, Rob F. Van der Wijngaart

misses the *main* parameter of interest: latency

D. Security Implications of Typical Grid Computing Usage Scenarios

Marty Humphrey, Mary R. Thompson

well they ought to read Schneier, but more importantly Ross Anderson's book on Secutrity Engineering

- their comment about firewalls is unbelievably naive - in a trivial single grid instances, a VPN will work, but the BIG uissue is when we have a global grid of grids -

- then we have a massive DDoS engine - the securing of the world from this, given evidence (see anderson's book) that in a non-cooperative, or even cooperative, but large, endeavour, 90% of attacs come from internal miscreants, is a BIG problem -

- the practical implications of this are already hitting the UK E-science program, when we have a single national network administration - in anything bigger, just this simple example will be come a nightmare. scalable key distribution and webs of trust and accounting, and privacy and provenance etc etc etc.....

- so a "security challenges for the GRID" is a rather larger problem....("there are mor things in heaven and earth and the GG than most men have dreamed of")

E. Location Selection for Active Services

Roger Karrer, Thomas R. Gross

what about the really big US Projects on location services such as:

- HOPS (ACIRI and ISI)
- IDMaps (UMich, UMass) (see google)? or even
- RocketFuel (WashU)

F. CORBA-Based Distributed Component Environment for Wrapping and Coupling Legacy Scientific Codes

Gregory Follen, Chan Kim, Isaac Lopez, Janche Sang

- wrappers abound - there's lots of work in the CORBA community done in this sure -

- but what about the *performance* and *behaviour* descriptions - its not just a matter of syntactic mapping and wireline formats - its semantics -

- semantic brokers are cool - lots of the SE community work on componenent technologies uses XML or other metadata annotations to do this (as do the reflective programming community) - you need it for large systems - even something simple like rpmfind has some clue...

G. A Web-Based Data Architecture for Problem-Solving Environments: Application of Distributed Authoring and Versioning to the Extensible Computational Chemistry Environment

K.L. Schuchardt, J.D. Myers, E.G. Stephan

- so there's a LOT of work on collaborative authoring in the CSCW community - entire books have been written - the WEBDAV community have mentioned some of this but this paper doesn't much...

- one thing i like is the ability to mix synch and asynch in authoring tools - one cute example of this (includes CVS support etc) is Messie

- see `http://www.cse.ucsc.edu/ ejw/collab/` for lots of tools...

H. The Astrophysics Simulation Collaboratory: A Science Portal Enabling Community Software Development

Michael Russell, Gabrielle Allen, Greg Daues, Ian Foster, Edward Seidel, Jason Novotny, John Shalf, Gregor von Laszewski

can anyone say *vnc*?

at&t research general purpose portal/teleportal tool... (cross platform, efficient, and so on)

I. File and Object Replication in Data Grids

Heinz Stockinger, Asad Samar, Koen Holtman, Bill Allcock, Ian Foster, Brian Tierney

comments: well, no mention of SANs or s/w RAID! amazing...

J. Open Metadata Formats: Efficient XML-Based Communication for High Performance Computing

Patrick Widener, Greg Eisenhauer, Karsten Schwan, Fabin E. Bustamante

- Craig partridge did a paper yonks ago comparing ASN1 and XDR - there was a spate of such work in the network management days when we argued about what wire encoding and language level syntax to use in stubs

- automatic space/time tradeoff optimisation papers in stub compilers are many (see also xerox parc's universal stub compiler work about 10 years back)

- java object serialisation also spring to mind as areas where a lot has been done on

this - re-applying to SOAP and XML, one would hope thatp eople actually wrote some code (yacc/bison etc aren't hard to use):-)

K. Programming the Grid: Distributed Software Components, P2P and Grid Web Services for Scientific Applications

Dennis Gannon, Randall Bramley, Geoffrey Fox, Shava Smallen Al Rossi, Rachana Ananthakrishnan, Felipe Bertrand, Ken Chiu, Matt Farrellee, Madhu Govindaraju, Sriram Krishnan, Lavanya Ramakrishnan, Yogesh Simmhan, Alek Slominski, Yu Ma, Caroline Olariu, Nicolas Rey-Cenvaz

(physicists do so like huge numbers of authors!)‘

so the mention of p2p here is a bit lite given it shows up in the title....see `http://www.cl.cam.ac.uk/ jac22/p2p` for pointers to lots more..

Meta-Comment

A *lot* of the papers use *superlatives* a lot. really, a huge amount, like vast, massive, humungous, etc....so? i have a lapop with 100G of disk and 1G of memory - it is small. very small,. tiny in fact. computer science is more interesting when you express things in terms of complexity and computability - given moores law and equavilable expontial increases in memory, disk and network speed big numbers look silly 5 years later - k's turn into m's turn into g's, t's and p's, but the releative performance stays the same... surely physics people understand the differencebetween equations and constants:-)