Demands for Highspeed Network Processing Capabilities for Gigabit Ethernet Link Speed and Beyond

Kernel and network interface cards adaptations to fullfill upcoming demands

Hagen Paul Pfeifer‡
hagen@jauu.net

‡Hochschule Furtwangen,
Computer Networking – Fakultät Informatik
Furtwangen, Germany

November 2007

Agenda

1. Why high speed networks
2. What are the major and minor challenges (we will pick representative hot-spots)
3. Hard- and software adoptions
    - What are the current bottlenecks
    - Ways to “fix” (or bypass) them
    - What can we learn from the history and other techniques/protocols
4. Network analysis at high speed links
5. At the end we will realize that old innovations demonstrate us how we solve upcoming challenges

6. . . . but now lets stop this prose and dig into the technical details!

Why High Speed Network

- More comprehensive term: broadband network
- New upcoming technologies require more bandwidth (e.g. IPTV)
- New innovations shift traditional (non-wired) technologies and equip with INET access (e.g. embedded hardware)

Background Knowledge

- Since the advent of Myrinet, Gigabit Ethernet and Infiniband the bottleneck shifted from interconnects to end-hosts (RX/TX paths)
- $\{1{,}10{,}100{,}1000{,}10000{,}\ldots\}$MBit/s $\rightarrow$ Moor's law helped 30 years
- Rule of thumb: 1MBit/s $\leftrightarrow$ 1Mhz (rough rule)
- One challenge: one big producer - one NIC, there is no "I/O virtualization" (no real possibility for I/O virtualization, see literature list at the end)
- Overheads in the protocol stack (fragmentation, checksum, data copy, DMA overhead)
- A story of parallelism: Mainframe, Workstation, IBM-PC; UNIX tool chain; PIO/DMA; Cluster
- For the network to scale – all involved components must scale
- Demands have changed: years back memory consumption was the biggest issue, now access time is the big challenge
- Industry Debate: network interface design

Throughput and Latency

- Throughput
  - Amount of data per time
  - The term “throughput” without further specification is senseless!
    - Received amount of data at physical layer? What about CRC errors?
    - Received amount of data at transport layer? Application layer?
    - This sounds of minor interest, but it isn’t!

- Maximum throughput: capacity
  - How can we determine the capacity of a certain link? Normally you <u>can't</u>!
  - You can ask the carrier provider or you can use *packet dispersion techniques*:
    - ⋆ There are some fundamental limitations with these techniques
    - ⋆ pchar, pathchar, bing, pathrate, clink, pipechar
- *Goodput* is the application level throughput (without protocol overhead)
- **`iperf(1)`**, **`netperf(1)`**, **`netsend(1)`** and similar tools measure the current throughput of the link (and sometimes not even that)
- Nomenclature: decimal prefixes vs. binary prefixes

- Latency
  - "It's the Latency, Stupid" (see the reference section at the end)
  - Network latency (sum of intermediate host processing time and L1 characteristic $\rightarrow 0.7 * c$)
  - Why latency matters: VoIP, data centers (think about time-critical, automated trading systems)
- Interplay between *throughput* and *latency* (see congestion control, especially BDP)

10 Gbit/s Processing Requirements

- Organization
  - Well defined path through kernel and userspace
  - **One** connection
    - One CPU queue
    - CPU affinity
    - One lockless journey through the kernel (is the destination!)
- Closely interaction with memory/CPU subsystem
  - Reduce latency
  - Direct connection between frame multiplexing and CPU
- Effective notification scheme
  - Interrupt driven (TX path) or completion queue (Infiniband)

Gigabit Flush

- The truth throughput is often less then netto Gb/s (expecially SoHo sphere)
- Often: $\leq$ 100 Mb/s
- PCI bus: 32bit 33MHz, require 64-bit 66MHz
- CPUs are also disburdening: often the CPU is the limiting factor (OS limitations)
- Packet processing overhead (small packet problem)
- 30 Megabyte transfer
  - 802.11g $\longrightarrow$ 148m
  - 100BASE-T $\longrightarrow$ 40m
  - 1000BASE-T $\longrightarrow$ 4m
  - 10GBASE-T $\longrightarrow$ 24s

Technology Responses

- Software based optimizations
  - Kernelspace
    - NAPI (interrupt mitigation, packet throttling)
    - LRO (large receive offload)
    - Driver lines
    - Automatic buffer size management (TCP)
  - Userspace
    - `splice()`, `tee()`, `mmap()`
    - `TCP_CORK`
    - `SO_RCVBUF`, `SO_SNDBUF` (not that clean – the user shouldn't touch this)

- NIC based optimizations (bypass OS)
  - TOE – TCP Offload Engine (many patents, M$ chimney: but drivers are unusable)
  - Hardware fragmentation
  - Checksums

Triumphantly Principals - Key To Success

- Cache data
- $O(1)$ data structures where possible (and avoid $O(n)$ and worser)
- Fine grained multiplexing (early demultiplexing)
- Only essential fragmentation (Jumbo frames, VM, ...)
- Avoid unnecessary operations (zero-copy)
- Optimize the common path (fast path, pre-computer header)
- Invest in appropriate hardware (sounds like design weakness, but it isn't)

Integrated NIC versus Offload Engine

- Two concurrent developments
- The former attempt to shift network processing tight to the CPU, the later tempt to shift a major part to a dedicated unit
- TOE's are less flexible, especially the OS integration is terrible
- New protocols must support by the vendor, security holes are now "hard-coded"
- Integrated NIC
  - CPU integrated FIFO's (RX/TX)
  - Dedicated PHY interface (exchangeable)
  - Checksum functionality
- Another approach: no CPU integration but in one memory domain

Router Demultiplexing

- Demultiplexing based on: Address, Multicast, QoS, Security, ...
- Demultiplexing happend before forwarding → to back-up line speed
- Space Shift: integrated, optimized circuits process routing (realize the arising technology chains?)
- 10000000 (OC-192/1000MBit/s) lookups per second
- Patricia Trie (longest prefix)
- Many providers deploy switched infrastructure, because of limited router performance – lookup algorithms as bottleneck

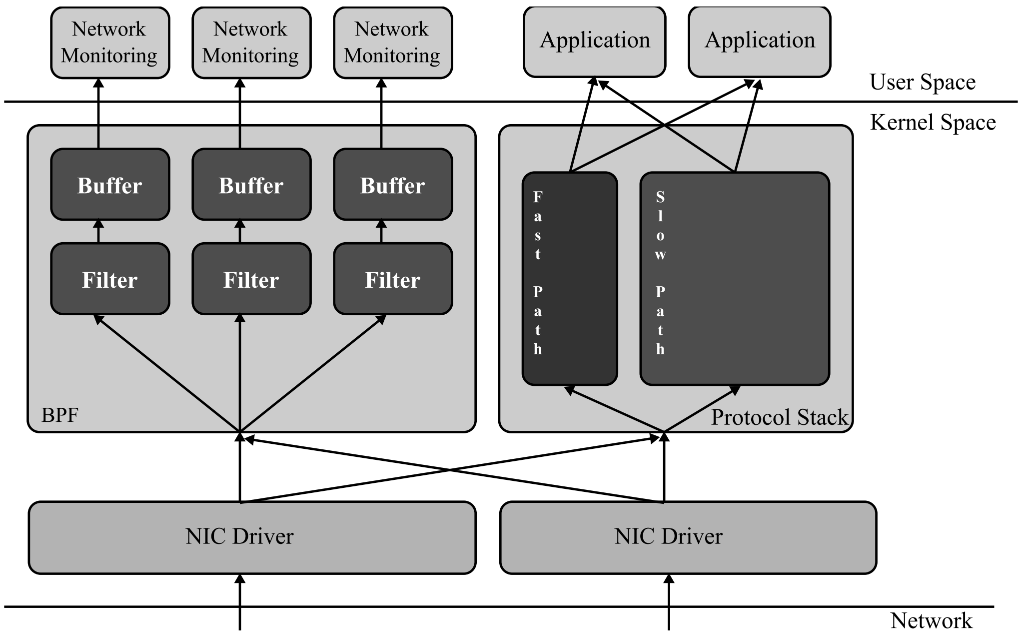
Why Network Analysis

- The drivers for network measurements at high rates (research and business drivers):
  - QoS Measurements (Determine product or service quality assurance (QA) at prefedined network conditions)
  - Traffic Engineering
  - Accounting (simple port based analysis aren't sufficient)
  - Failure Management
  - Traffic Control
  - IDS
- Measurements Metrics
  - Packet Loss, Round Trip Delay, Delay Variation, Throughput, Packet Loss Patterns, Link capacity (Used bandwidth, available bandwidth) packet reordering, ...

- Research Community
  - Internet traffic modeling and simulation
  - Test network protocol behavior
    - Simulator: ns2
    - Emulators: Dummynet, Netem, NISTnet
    - Hardware Emulators: Simena Network Emulator Appliance
  - Traffic generation *can* be realistic if derived from real measurements
    - Many premises in the measurement process must be covered
    - At the end the traffic is an approximation

BSD Packet Filter - BPF

- Steven McCanne and Van Jacobson
- Stanford do not perform well on modern RISC architectures
- BPF: register based filter evaluator (up to 20 times faster)
- New buffering strategy (avoid packet triggered copy mechanism)



BPF and PCAP Interplay

- `PF_PACKET` based data gathering
- `libpcap-0.8.1/pcap-bpf.c`
- `libpcap-0.8.1/bpf/net/bpf_filter.c` is the userspace filter pendant (sometimes the filter can't be applied to the kernel → filtering in userspace (e.g. no socket filters support))
- Enable filter via `setsockopt(..., SO_ATTACH_FILTER)` (`/pcap-linux.c`)
- Processing Chain:
  1. `scanner.l` parse human filter rule,
  2. `gencode.c` compile human filter to intermediate format (`pcap_compile()`)
- Now the fun begins:
  - Principally: intern representation as a graph ("flowgraph intermediate representation")
  - `bpf_optimize()` (`opt_loop()` → `opt_root()` → ...)

- `icode_to_fcode()` – Convert flowgraph intermediate representation to BPF array representation

- BPF and Linux:
  - `net/core/filter.c:sk_run_filter()`

Analysis With Consumer Hardware

- Gigabit and 10-gigabit NIC's are incredible fast
- Real-Time analysis: unthinkable – capturing: feasible
- Consumer Hardware: FSB and Disk aren't fast enough, but ...
- Hardware suggestions:
  - Fast CPU (Opterons and Xeons)
  - Much DRAM (2GByte and beyond)
  - RAID Array
  - OS: try Linux and FreeBSD
- 10 Gigabit: split traffic on multiple 1GB links (e.g. Cisco Switch functionality)

Fin

- Thank you very much!
- Questions?

Additional Information

- **The BSD Packet Filter**, *A New Architecture for User-level Packet Capture*, Steven McCanne, Van Jacobson,
- **PCAP - Packet Capture library**, *http://www.tcpdump.org/*
- **UDP & TCP Throughput measurements using the Myricom 10 Gigabit Ethernet NIC**, *http://www.hep.man.ac.uk/u/rich/net/NIC_tests_10GE_Myricom/Myricom_10GE_NIC.htm*
- **The Performance Potential of an Integrated Network Interface**, *http://www.eecs.umich.edu/ stever/pubs/asplos06-nic.pdf*
- **Performance Analysis of System Overheads in TCP/IP Workloads**, *http://www.eecs.umich.edu/ stever/pubs/pact05.pdf*
- **Analyzing NIC Overheads in Network-Intensive Workloads**, *http://www.eecs.umich.edu/techreports/cse/2004/CSE-TR-505-04.pdf*
- **Optimizing 10-Gigabit Ethernet for Networks of Workstations, Clusters, and Grids: A Case Study**, *http://www.sc-conference.org/sc2003/paperpdfs/pap293.pdf*

- **Server Switching: Yesterday and Tomorrow**, *http://www.cs.duke.edu/ari/publications/switch.pdf*
- **End-System Optimizations for High-Speed TCP**, *http://www.cs.duke.edu/ari/publications/end-system.pdf*
- **Balancing DMA Latency and Bandwidth in a High-Speed Network Adapter**, *http://www.cs.duke.edu/ari/publications/balancing.ps*
- **Experiences with a High-Speed Network Adaptor: A Software Perspective**, *http://citeseer.ist.psu.edu/cache/papers/.../druschel94experience.pdf*
- **Achieving Reliable High Performance in LFNs**, *http://citeseer.ist.psu.edu/ubik03achieving.html*
- **Wikipedia - List of device bandwidths**, *http://en.wikipedia.org/wiki/List_of_device_bandwidths*
- **IEEE P802.3ae 10Gb/s Ethernet Task Force**, *http://grouper.ieee.org/groups/802/3/ae/*
- **Linux Kernel – Large receive offload**, *http://lwn.net/Articles/243949/*
- **TOE and Linux**, *http://www.linux-foundation.org/en/Net:TOE*
- **INTEL – Virtualization Technology for Directed I/O**, *http://www.intel.com/technology/.../5-platform-hardware-support.htm*

- **Challenges for Scalable Networking in a Virtualized Server**, *http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4317816*
- **High performance and scalable I/O virtualization via self-virtualized devices**, *http://portal.acm.org/citation.cfm?id=1272390*
- **An efficient programmable 10 gigabit Ethernet network interface card**, *http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=1385932*
- **Impact of protocol overheads on network throughput over high-speed interconnects: measurement, analysis, and improvement**, *http://portal.acm.org/citation.cfm?id=1265197*
- **It's the Latency, Stupid**, *http://www.stuartcheshire.org/rants/Latency.html*
- **Reducing Web Latency Using Reference Point Caching**, *http://www.cs.ucsd.edu/ varghese/PAPERS/webinfocom.pdf*

Contact

- Hagen Paul Pfeifer
- EMail: hagen@jauu.net
  - Key-ID: `0x98350C22`
  - Fingerprint: `490F 557B 6C48 6D7E 5706 2EA2 4A22 8D45 9835 0C22`

Document-ID: da39a3ee5e6b4b0d3255bfef95601890afd80709

Simena Network Emulator Appliance

- Models: NE 2000, NE 3000
- Operates on Ethernet layer
- Fully tested products and services
- Gigabit wire speed
- Capture and Replay functionality.
- RFC 2544 network performance measurements

Myrinet

- High-speed LAN technology (mostly used at clusters)
- Minor protocol overhead (better throughput, reduced latency, ...)
- Fibre Optic technology
- Up to 10Gbit/s

IEEE 802.3an and 802.3ae

- Cabling: copper (IEEE 802.3an) and fiber optic (802.3ae)
- 10GBASE-EX → 40km (Wavelength: 1550nm)
- 825 Mbaud

Neptun NIC

- SUN Niagara II
- 2 x 10Gbit/s
- Ability to multiplex 10Gbit/s and distribute them among several CPU's
- "Virtualization" based on MAC, IP address or port

DAG Cards

- Endace – “world leader in network traffic monitoring technology”
- Passive measurement cards (ok - newer version include “lawfull interception” features)
    - Capture in real-time
    - Timestamping from GPS data