Overview

- Router basics
- Interconnection architecture
  - Input Queuing
  - Output Queuing
  - Virtual output Queuing
  - Scheduling
- Future bottlenecks
- Case Studies
What’s in a router?

- **Physical components**
  - One or more *input interfaces* that receive packets
  - One or more *output interfaces* that transmit packets
  - A chassis (box + power) to hold it all

- **Functions**
  - **Forward** packets
  - **Drop** packets (congestion, security, QoS)
  - **Delay** packets (QoS)
  - **Transform** packets? (Encapsulation, Tunneling)
What a router does: the normal case

- Receive incoming packet from link input interface
- Lookup packet destination in forwarding table
  - (destination, output port(s))
- Validate checksum, decrement ttl, update checksum
- Buffer packet in input queue
- Send packet to output interface (interfaces? Mcast)
- Buffer packet in output queue
- Send packet to output interface link
What a router looks like?

Cisco 2500

Capacity: <10Mbps

Linksys DEFSR81

Capacity: <10Mbps
What a router looks like (2)

Cisco GSR 12416
- Capacity: 160Gb/s
- Power: 4.2kW
- Dimensions: 6ft x 2ft x 2ft

Juniper M160
- Capacity: 80Gb/s
- Power: 2.6kW
- Dimensions: 3ft x 2.5ft x 2ft
What a router looks like (3)

Pluris Teraplex 20 w/7 Racks

Capacity: >1Tb/s
Power: 45kW (~250 homes?)

1 room
High-performance routers

- Geared to core and distribution service needs
  - Requirements: high speed & high density

- Why do we care?
  - Moore’s Law slower than link speed growth (and BW demand)
    - OC48c (2.5Gbps), now, 128ns/packet
    - OC192c (10Gbps), in deployment, 33ns/packet
    - OC768c (40Gbps), 2002-3, 8ns/packet
  - Need high density/low power to manage POP complexity
    - $20-100k & 2-400W per port, 50% ports frequently for internal connectivity
    - DWDM can help with the former, but requires more interfaces
Functional architecture

Control Plane
- Complex
- Per-control action
- May be slow

Data plane
- Simple
- Per-packet
- Must be fast
Packet classification

- **Forwarding**
  - Longest prefix match of destination against forwarding table
  - Returns (output port, Next-hop MAC header) tuple
  - Key issue: forwarding table growth
  - George will talk about this next time

- **QoS tagging**
  - Certain traffic tagged with higher priority
  - Per flow (src ip, src port, dst ip, dst port), pre source or dest prefix, per protocol (Napster, etc…)

- **Firewall rules**
  - Block access to TCP packets with dst port != 80
Interconnect architecture

- Input & output connected via switch fabric

- Kinds of switch fabric
  - Bus
  - Crossbar
  - Shared Memory

- How to deal with transient contention?
  - Input queuing
  - Output queuing
  - Combination
First Generation Routers

- Single CPU and shared memory;
- All classification by main CPU
Second Generation Routers

CPU

Route Table

Shared Bus(s)

Direct DMA on cache hit

Line Card

Buffers

Forwarding Cache

MAC

Cache of recent routes
Third Generation Routers

- Shared interconnect (usually crossbar)
- Centralized scheduler
- Full forwarding table in line card
- Fixed cells (why?)
Output queuing

- **Output interfaces buffer packets**

- **Pro**
  - Simple algorithms
  - Single congestion point

- **Con**
  - N inputs may send to the same output
  - Requires speedup of N
Input queuing

- Input interfaces buffer packets

- Pro
  - Single congestion point
  - Simple to design algorithms

- Con
  - Must implement flow control
  - Low utilization due to Head-of-Line (HoL) Blocking
    - Utili limited to \( 2 - 2^{0.5} = 58\% \)
Head-of-Line Blocking
Virtual Output Queues
(courtesy Nick McKeown)
IQ + Virtual Output Queuing

- Input interfaces buffer packets in per-output virtual queues

- Pro
  - Solves blocking problem

- Con
  - More resources per port
  - Complex arbiter
  - Still limited by input/output contention (scheduler).
  - RR: \( 1 = 1/e = 63\% \)
Switch scheduling

- **Problem**
  - Match inputs and outputs
  - Resolve contentions
  - No output packet drops
  - Maximize throughput
  - Do it in constant time…

- **Many algorithms for uniform traffic assumption**
  - E.g. TDM, Maximum size bipartite match
  - Approximate answers (e.g. iSLIP, submaximal match)

- **Recent result (Dai et al, 2000)**
  - Maximal size matching + speedup of two guarantees
  - 100% utilization for most traffic assumptions
Modern router

- IQ + VoQ + OQ
  - Speedup of 2
  - Central scheduler
  - Fixed-sized internal cells

Pro
- Can achieve utilization of 1
- Can scale to multiple Tb/s

Con
- Multiple congestion points
- Complexity
Typical function breakdown

- **Input interface**
  - Forwarding
  - Virtual output queuing

- **Switch**
  - Scheduling input interface requests to output interfaces
  - Multicast scheduling

- **Output interface**
  - Queue packets for transmission
  - Classification
  - Buffer management (which pkt to drop)
  - Scheduling (which pkt to send from buffer)
Next bottlenecks

- **Buffering at high speed**
  - SRAM density too low for BW*D of 40Gbps link
  - DRAM too slow
  - SRAM memory management as cache for DRAM

- **Scheduler and arbiter overhead**
  - Limits size of switch and link BW
  - Two-state switch (Chang et al, 2000); no scheduler

- **High density (100’s-1000’s of line cards)**
  - Physical distance to support density; electrical links degrade
  - Optical links; optical cross connect (MEMs, tunable lasers)

- **Time to market, Power/Heat**
Case study Partridge et al 98

- IQ + VOQ + OQ

- 15 line-cards each with up to 16 interfaces
  - Forwarding engines separate from lc (why?)

- Point-to-Point switch
  - 50Gbps capacity, 25% lost to overhead
  - 32Mpps
Router Architecture

Packet header

Switch

Input Packets

Output Packets

Packet Processor

FSU

TSU

Packets

Headers

Line Card

Forwarding Engine

Route Memory

Processor

Reply Fifo

TSU

Req Fifo

FSU

Packet Processor

FSU

TSU

Packets

Headers

Line Card

Switch

Packet

header

Input Packets

Output Packets
Implementation

- **Line cards**
  - 53Mhz FPGA, I/O processing, queue management

- **Forwarding engine**
  - 430Mhz 21164 Alpha using L2 as route cache, L3 holds all
    - 12,000 routes.. claim 95% hit rate… reasonable?
  - Lots of pain with uP I/O being not quite right
  - Doesn’t check IP checksum. Is this ok?

- **Switch**
  - Scheduler FPGA

- **Control plane**
  - NetBSD running GateD on 233-Mhz 21064
  - Pushes routing tables to FEs and handles exceptional packets
Summary: Innovations

1. Each FE has a complete set of the routing tables
2. A switched fabric is used instead of the traditional shared bus
3. FEs are on boards distinct from the line cards
4. Use of an abstract link layer header
5. Include QoS processing in the router

Questions
- Would you use a microprocessor for forwarding today?
- How important is programmability?
Case study: McKeown et al.98

- Tiny Terra: an inspiration for Cisco GSR (nee BFR)
- IQ + VOQ + OQ

- 32 interfaces at 10Gbps each
- 32x32 crossbar (320Gbps aggregate)
- 64bit cells
- Line cards ASIC-based

- iSLIP scheduling algorithm
  - Greedy, submaximal bipartite matching
  - Easy to implement (2N arbiters)
Conclusion

- It is feasible to build high-speed routers
- 40 Gbps link speeds
- 40 Tbps aggregate capacity
- 10-20 Tbps aggregate capacity
- High complexity, slow time to market
- Limited programmability
- "But..."

Next gen (OC3072 160Gbps LC) may be close to cross-over
- Starting to require significant on-chip SRAM
- Juniper I2 ASIC 2.5M gates
- Typical OC192 LC ~30M gates

Point for CMOS (luckily, not clear there is demand anyway)
Open issues

- **Network processors**
  - Quick TTM + high performance
  - Application-specific programmability (for high-touch features)

- **Software design**
  - Neglected, but critical

- **Optical switching**
  - Looks like circuit switching, very low power
  - Ok to lose packet switch efficiency if BW is cheap and reliable
  - Currently very expensive to light a lambda (amps, lasers)
For next time...

- George Varghese will be presenting on IP Lookups

- There is a packet lookup survey paper to read that will be posted tonight.
Points of Presence (POPs)