Router Design

Nick Feamster
CS 7260
January 24, 2007
Today’s Lecture

• The design of big, fast routers
• Partridge et al., A 50 Gb/s IP Router
• Design constraints
  – Speed
  – Size
  – Power consumption
• Components
• Algorithms
  – Lookups and packet processing (classification, etc.)
  – Packet queueing
  – Switch arbitration
What’s In A Router

• Interfaces
  – Input/output of packets

• Switching fabric
  – Moving packets from input to output

• Software
  – Routing
  – Packet processing
  – Scheduling
  – Etc.
What a Router Chassis Looks Like

**Cisco CRS-1**
- 19” Width
- 6ft Height
- Capacity: 1.2Tb/s
- Power: 10.4kW
- Weight: 0.5 Ton
- Cost: $500k

**Juniper M320**
- 17” Width
- 3ft Height
- Capacity: 320 Gb/s
- Power: 3.1kW
What a Router Line Card Looks Like

1-Port OC48 (2.5 Gb/s) (for Juniper M40)

4-Port 10 GigE (for Cisco CRS-1)

Power: about 150 Watts
Big, Fast Routers: Why Bother?

• Faster link bandwidths
• Increasing demands
• Larger network size (hosts, routers, users)
Summary of Routing Functionality

- Router gets packet
- Looks at packet header for destination
- Looks up routing table for output interface
- Modifies header (ttl, IP header checksum)
- Passes packet to output interface
Generic Router Architecture

Question: What is the difference between this architecture and that in today’s paper?
Innovation #1: Each Line Card Has the Routing Tables

• Prevents central table from becoming a bottleneck at high speeds

• **Complication:** Must update forwarding tables on the fly.
  – How does the BBN router update tables without slowing the forwarding engines?
Generic Router Architecture

Header Processing
- Lookup IP Address
- Update Header
- Address Table

Buffer Manager
- Buffer Memory

Data Hdr

Interconnection Fabric
First Generation Routers

Typically <0.5Gb/s aggregate capacity
Second Generation Routers

Typically <5Gb/s aggregate capacity
Third Generation Routers

“Crossbar”: Switched Backplane

Typically <50Gb/s aggregate capacity
Innovation #2: Switched Backplane

- Every input port has a connection to every output port
- During each timeslot, each input connected to zero or one outputs

- **Advantage:** Exploits parallelism
- **Disadvantage:** Need scheduling algorithm
Router Components and Functions

- Route processor
  - Routing
  - Installing forwarding tables
  - Management

- Line cards
  - Packet processing and classification
  - Packet forwarding

- Switched bus (“Crossbar”)
  - Scheduling
Crossbar Switching

• Conceptually: \( N \) inputs, \( N \) outputs
  – Actually, inputs are also outputs
• In each timeslot, one-to-one mapping between inputs and outputs.
• Goal: Maximal matching

\[
S^*(n) = \arg \max_{S(n)} (L^T(n) \cdot \underline{S}(n))
\]

Traffic Demands

Bipartite Match

Maximum Weight Match
Early Crossbar Scheduling Algorithm

• Wavefront algorithm

Problems: Fairness, speed, …
Alternatives to the Wavefront Scheduler

• **PIM: Parallel Iterative Matching**
  – **Request:** Each input sends requests to all outputs for which it has packets
  – **Grant:** Output selects an input at random and grants
  – **Accept:** Input selects from its received grants

• **Problem:** Matching may not be maximal
• **Solution:** Run several times

• **Problem:** Matching may not be “fair”
• **Solution:** Grant/accept in round robin instead of random
Scheduling and Fairness

• What is an appropriate definition of fairness?
  – One notion: Max-min fairness
  – Disadvantage: Compromises throughput

• Max-min fairness gives priority to low data rates/small values

• Is it guaranteed to exist?
• Is it unique?
Max-Min Fairness

- A flow rate $x$ is **max-min fair** if any rate $x$ cannot be increased without decreasing some $y$ which is smaller than or equal to $x$.

- How to share equally with different resource demands
  - small users will get all they want
  - large users will evenly split the rest

- More formally, perform this procedure:
  - resource allocated to customers in order of increasing demand
  - no customer receives more than requested
  - customers with unsatisfied demands split the remaining resource
Example

• Demands: 2, 2.6, 4, 5; capacity: 10
  – 10/4 = 2.5
  – **Problem:** 1st user needs only 2; excess of 0.5,

• Distribute among 3, so 0.5/3=0.167
  – now we have allocs of [2, 2.67, 2.67, 2.67],
  – leaving an excess of 0.07 for cust #2
  – divide that in two, gets [2, 2.6, 2.7, 2.7]

• Maximizes the minimum share to each customer whose demand is not fully serviced
How to Achieve Max-Min Fairness

• **Take 1:** Round-Robin  
  – Problem: Packets may have different sizes

• **Take 2:** Bit-by-Bit Round Robin  
  – Problem: Feasibility

• **Take 3:** Fair Queuing  
  – Service packets according to soonest “finishing time”

Adding QoS: Add weights to the queues…
Why QoS?

• Internet currently provides one single class of “best-effort” service
  – No assurances about delivery

• Existing applications are **elastic**
  – Tolerate delays and losses
  – Can adapt to congestion

• Future “real-time” applications may be **inelastic**
Other Goal: Utilization

• “100% Throughput”: no packets experience head-of-line blocking
• Does the previous scheme achieve 100% throughput?
• What if the crossbar could have a “speedup”?

Key result: Given a crossbar with 2x speedup, any maximal matching can achieve 100% throughput.
Head-of-Line Blocking

Problem: The packet at the front of the queue experiences contention for the output queue, blocking all packets behind it.

Maximum throughput in such a switch: $2 - \sqrt{2}$
Combined Input-Output Queueing

- Advantages
  - Easy to build
    - 100% can be achieved with limited speedup
- Disadvantages
  - Harder to design algorithms
    - Two congestion points
    - Flow control at destination
Solution: Virtual Output Queues

- Maintain N virtual queues at each input
  - one per output
Processing: Fast Path vs. Slow Path

- **Optimize for common case**
  - BBN router: 85 instructions for fast-path code
  - Fits entirely in L1 cache

- **Non-common cases handled on slow path**
  - Route cache misses
  - Errors (e.g., ICMP time exceeded)
  - IP options
  - Fragmented packets
  - Multicast packets
Recent Trends: Programmability

• NetFPGA: 4-port interface card, plugs into PCI bus (Stanford)
  – Customizable forwarding
  – Appearance of many virtual interfaces (with VLAN tags)

• Programmability with Network processors (Washington U.)
IP Address Lookup

Challenges:

1. Longest-prefix match (not exact).
2. Tables are large and growing.
3. Lookups must be fast.
IP Lookups find Longest Prefixes

Routing lookup: Find the longest matching prefix (aka the most specific route) among all prefixes that match the destination address.
IP Address Lookup

Challenges:

1. Longest-prefix match (not exact).
2. Tables are large and growing.
3. Lookups must be fast.
Address Tables are Large
IP Address Lookup

Challenges:

1. Longest-prefix match (not exact).
2. Tables are large and growing.
3. Lookups must be fast.
Lookups Must be Fast

<table>
<thead>
<tr>
<th>Year</th>
<th>Line</th>
<th>40B packets (Mpkt/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1997</td>
<td>622Mb/s</td>
<td>1.94</td>
</tr>
<tr>
<td>1999</td>
<td>2.5Gb/s</td>
<td>7.81</td>
</tr>
<tr>
<td>2001</td>
<td>10Gb/s</td>
<td>31.25</td>
</tr>
<tr>
<td>2003</td>
<td>40Gb/s</td>
<td>125</td>
</tr>
</tbody>
</table>

Cisco CRS-1 1-Port OC-768C (Line rate: 42.1 Gb/s)

Still pretty rare outside of research networks
IP Address Lookup: Binary Tries

Example Prefixes:

- a) 00001
- b) 00010
- c) 00011
- d) 001
- e) 0101
- f) 011
- g) 100
- h) 1010
- i) 1100
- j) 11110000
IP Address Lookup: Patricia Trie

Example Prefixes

a) 00001
b) 00010
c) 00011
d) 001
e) 0101
f) 011
g) 100
h) 1010
i) 1100
j) 11110000

Problem: Lots of (slow) memory lookups
Address Lookup: Direct Trie

- When pipelined, one lookup per memory access
- Inefficient use of memory
Faster LPM: Alternatives

• Content addressable memory (CAM)
  – **Hardware-based** route lookup
  – Input = tag, output = value

  – Requires exact match with tag
    • Multiple cycles (1 per prefix) with single CAM
    • Multiple CAMs (1 per prefix) searched in parallel
  – Ternary CAM
    • (0,1,don’t care) values in tag match
    • Priority (i.e., longest prefix) by order of entries

Historically, this approach has not been very economical.
Faster Lookup: Alternatives

• Caching
  – Packet trains exhibit temporal locality
  – Many packets to same destination
• Cisco Express Forwarding
IP Address Lookup: Summary

• Lookup limited by memory bandwidth.
• Lookup uses high-degree trie.
• State of the art: 10Gb/s line rate.
• Scales to: 40Gb/s line rate.
Fourth-Generation: Collapse the POP

High Reliability and Scalability enable “vertical” POP simplification

Reduces CapEx, Operational cost
Increases network stability
Fourth-Generation Routers

Limit today ~2.5Tb/s
- Electronics
- Scheduler scales <2x every 18 months
- Opto-electronic conversion
Multi-rack routers
Future: 100Tb/s Optical Router

(100Tb/s = 625 * 160Gb/s)

McKeown et al., Scaling Internet Routers Using Optics, ACM SIGCOMM 2003
Challenges with Optical Switching

- Missequenced packets
- Pathological traffic patterns
- Rapidly configuring switch fabric
- Failing components