Switching Hardware

- General purpose I/O bus is not enough
  - Bandwidth of 1Gbps can only support three 155Mbps links. Why?

- Design goals
  - Throughput
    - Packets/sec
  - Scalability
    - Number of input/output ports
  - Cost
    - $/port
Throughput

- Can be defined in terms of
  - Total bandwidth: ATM networks
  - Forwarding rate in packets/sec: Switched Ethernet

- Throughput is a function of traffic
  - Need to model traffic accurately

- Traffic modeling parameters
  - Packet arrival time
  - Destination distribution
  - Packet size distribution
Scalability

- Cost is a function of number of I/O ports (n)

- Scalability can be thought of as the rise in cost with increasing n.

- Switch designs also have hardware limitations when the number of ports becomes very large.
Switching Fabrics
Switching Fabrics

- Ports communicate with the outside world
  - Optoelectronics

- Ports also support the routing model.
  - E.g. for VC model, ports manage connection setup/routing/teardown

- Fabrics are fast bit pushers. They take data from the input port and move it to an output port.

- Complexity of determining the output port for a packet is handled by the input ports.

- Such fabrics are called self-routing
Switching Fabrics

- **Performance bottlenecks**
  - Input port: header analysis, routing table lookup

- **Buffering can occur at the**
  - Input port
  - Fabric
  - Output port

- **Design and implementation of buffers have the greatest impact on performance**

- **Simple scheme:**
  - Input buffering with FIFO queue
  - Drawback: head of line blocking. Max. throughput 59%

- **Buffering discipline determines QoS**
Crossbar Switches

- Every input is connected to every output.
- Can handle data arriving at all n inputs simultaneously.
Crossbar Switches

- Speed of the output buffer has to be proportional to $n$

- Complexity of the switch fabric grows as $n^2$
Knockout Switch

Assumption:
- Not all input ports need to get to the same output port simultaneously

The design supports no more than $L$ inputs out of $n$ to proceed to the output

Note:
- Choice of $L$ is hard. Servers on an output port can ruin the above assumption
Knockout Switch

Knockout has 3 components

- A set of packet filters that determine the inputs destined for the current output

- A set of knockout concentrators. Each concentrator has n inputs and L outputs. Given n inputs destined for the same output, it selects L out of them fairly. The remaining n – L inputs are discarded
  - Fairness implies that no input gets dropped more or less than any other

- A per output buffer that can accept L inputs at a time.
Knockout Switch Concentrator

- Uses a tennis tournament style knockout system.
- You play division I first, if you lose, you go to division II, if you lose again you go to division III and so on till L
- If you lose in division L, you’re kicked out.
Knockout switch output buffer

- **Simple option:**
  - Speed of output buffer is $L \times$ link speed.

- **Problem:** requires very fast buffers

- **Knockout** uses a clever round robin buffering scheme with $L$ parallel buffers and a shifter.

- The fabric writes data to $L$ buffers simultaneously. The output port reads from one
Knockout switch complexity

- Number of packet filters = n
- Size of output buffer/port = L
- Complexity of concentrator/output port: n \* L. This is proportional to n
  - Total cost of concentrators = n^2

- Total complexity is proportional to n^2
■ Used a high speed bus to transfer data between input port and shared memory.

■ Output port reads data from shared memory.
Shared memory switches

- Offer better resource utilization through statistical multiplexing

- Disadvantage:
  - The speed of the buses between the MUX and memory and DEMUX and memory has to scale linearly with number of inputs. This restricts the use of this design

- Shared memory switches are commercially very common.