Switch fabrics 1
“Centralized” vs distributed switches

Switches can be centralized (e.g. implemented in a single box) or distributed (consisting of multiple interconnected boxes)

Potential advantages of centralized switches:
• Regular internal structure, c.f. heterogeneous distributed systems
• Feasible to have central control, c.f. propagation delays in distributed systems

We’ll consider centralized switches first, and later consider distributed switching (using bridges)

Regular “centralized” designs can conceivably be applied to distributed switching, provided component consistency is possible.
We’ll consider some of these structures (e.g. crossbar, Banyan, time-division) shortly...

Figures from H. Peyravi

Copyright © 2003, Tim Moors
Outline: Time-division switches

• Multiplexing and demultiplexing
  • add-drop multiplexers
  • inverse multiplexing
• Shared transmission medium
• Shared storage medium
  • Time-slot-interchange switch
  • other shared memory switches
Multiplexing and demultiplexing

**Multiplexing** (muxing): “The combining of two or more information channels onto a common transmission medium.” [FS-1037]

**Time-division multiplexing:** Different inputs are sent at different times. Alternatives: Frequency & Wavelength division muxing (FDM, WDM)

**Synchronous multiplexing:** Multiplexer alternates between different inputs in round-robin order (c.f. Asynchronous TDM)
Add-drop multiplexers (ADMs)

*Drop* one (or some) multiplexed signal(s) from trunk
*Add* one or more replacement signals

Drop circuit 4
Add circuit A

Popular in optical networks (OADM), where adding/dropping a wavelength is one of the few functions possible with photonic technology.
Inverse muxing (splitting)

After demuxing, outputs usually diverge

**Inverse multiplexing**: multiple outputs follow same path & later rejoin

**Application**: Construct high-speed link from multiple lower-speed links, e.g.:

- $1\times100\text{Mb/s}$ port from $10\times10\text{Mb/s}$ switch ports
- Interconnect routers through PSTN, with rate of connection varying according to demand.

Need to deal with potential mis-sequencing.
Shared transmission media switches

Most commonly produced form of switch
(c.f. academic emphasis on space-division switching)

A single transmission medium shared by all input and output ports
Often a bus, but can use unidirectional transmission to reduce
fanout: ring, dual ring/bus, folded bus

Dispersion is (relatively) manageable in a centralized switch
⇒ Wide buses to achieve high bandwidth, e.g. 424b (53B=ATM cell)
   (utilisation may be low for small packets)
Bandwidth often equals aggregate of input ports

Access from input ports is usually fixed round-robin TDM; but
other schemes are possible (e.g. Distributed Queuing MAC)
Output ports pick off packets destined to them.
Downside: high-speed filtering
Implementing time-division switches on a PC

How:
- Use general-purpose NICs for line interfaces
- PC’s bus & memory for switching fabric & buffering
  - Packet is read in from a port into memory
  - Processor decides where to forward the packet
  - Processor sends packet to appropriate outgoing NIC
Disadvantages of time-division switching on a PC

• General-purpose NICs don’t make the frame available immediately (only after integrity check) ⇒ limits forwarding modes.
• Each packet is sent across the bus twice (unless DMA and non-standard NICs); Doesn’t scale well with increasing numbers of ports
• General-purpose computer:
  • Has features that are wasted (unnecessary cost): video, disk, etc
  • Isn’t optimised for communications processing (e.g. VRAM in wrong place, CAMs for address matching); interrupt processing overheads
• CPU is a central bottleneck ⇒ switches with port processors
“Slots” carry constant-length segments of payload. Slots don’t contain headers with addressing information
- numbers/letters/’&” in this figure are for illustrative purposes only
  - rank of numbers indicates order of arrival
  - rank of letters indicates order of departure

Slots propagate left to right.
Differing numbers of slots on TSI input and output lines reflect different propagation delays.
In practice, these lines could have 0 propagation delay,
  e.g. include mux&demux in switch.
• Incoming slots are buffered
• Switch is preprogrammed (by order of linked list) to know how to switch slots, e.g.
  • every 4th slot (1c, 1c’, 1c’’) is destined to port c
  • the slot after that (2d, 2d’, 2d’’) is destined to port d
• Phase: red dashed lines point to the next slot (or port for accessing that slot) to leave each device.
Notes about TSI circuit switch

Used in PBXs & after a space-division switch to resequence traffic

Scalability:
• Can multiplex multiple physical inputs to TSI input, but memory bandwidth increases in proportion to aggregate input bandwidth.
• Memory size is proportional to number of inputs (if allowing switching between arbitrary sets of ports)

Note that this circuit switch does have buffering (albeit small, since arrivals are deterministic) i.e. it is not just packet switches that have buffers.
Shared memory switches

Memory organized into linked lists, one for each output port
  (Linked lists are useful in other switches to implement scheduling disciplines, e.g. fair queuing)
• Incoming traffic is appended to the end of the list for the required output port
• Single memory:
  • Allows statistical gain: busy port can use memory not being used by an idle port
  • Helps multicasting: Stored once in memory for multiple outputs (multiple links to payload)
Example of a shared memory switch
D-link DES-1008D 8-Port Switch

- AUD$138 ea
- unmanaged
- 8 port, 10/100Mb/s Ethernet (NWay auto-negotiated). Full or half duplex. RJ-45 connectors.
- LEDs to indicate port activity (power, speed, collision)
RTL8308 switch + glue:
RTL8208 transceiver
FC-638L transformers
74VHC164 8b SIPO
RTL8308

2MB DRAM
32K x 64b @ 50MHz
256B pages

Store-and-forward + cut-through operation

Address matching:
1. addresses are hashed to index the 8K lookup table.
2. Hash bash handled by storing colliding addresses in CAM
Video RAM

CPU can randomly access DRAM
Access to DRAM is shared between CPU and cache(s)

Cache(s):
- access DRAM one complete row at a time
- can be accessed independently of DRAM, e.g. to feed video raster

Application to network switches:
- ports access caches
- achieve switching by using DRAM to transfer between caches & for buffering
Example of a VRAM switch: IDT 77V400 chip
Locating buffering

On the topic of memory, but looking forward:

Buffers can be located at:
- **Input ports**: Reliance on simple FIFO input buffers leads to head-of-line blocking:

![Diagram showing input ports with FIFO buffers]

- **Output ports**: Reliance on output requires fast switch and buffers
- **Within the switch fabric**
Multicasting in shared media switches

Readily supported since all output ports are naturally exposed to the traffic.

- For shared memory, multicasting requires multiple reads from memory, reducing throughput in proportion to number of members of multicast group. However, multicasting does not affect the size of the buffer required.

- For shared transmission, traffic will likely propagate past all output ports in any case, so multicast can be switched as fast as singlecast.
Time vs space-division switching

Time-division advantage: **Fabric cost**
    e.g. adding one extra port may require that a space-division switch double its port capacity

Space-division advantage: **Internal port cost**
    Port costs can dominate costs, but ideally it is the external port costs, which are inevitable, that dominate.