Operating Systems

Operating System Support for
Continuous Media

Why Study Multimedia?

- Improvements:
  – Telecommunications
  – Environments
  – Communication
  – Fun
- Outgrowth from industry
  – telecommunications
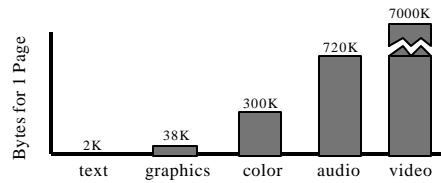 – consumer electronics
  – television

Continuous Media

- Subset of multimedia
- Includes timing relationship between server and client
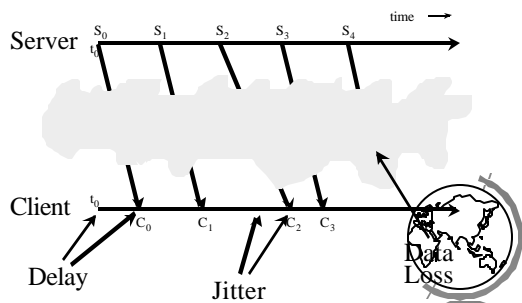- Stream:
  – video: mpeg, H.261, avi
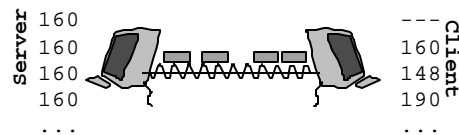 – audio: MP3, μ-law

---

Multimedia Resource Requirements

Bytes for 1 Page

text	graphics	color	audio	video
2K	38K	300K	720K	7000K

- Step up in media requires more bytes
- But not as much as some applications!
  – Graphics or transaction processing

Influences on Quality

time →

Server s_0 s_1 s_2 s_3 s_4
t_0

Client t_0
c_0 c_1 c_2 c_3

Delay

Jitter

Data Loss

An End-To-End Problem

Server 160
160
160
160
...

Client ---
160
148
190
...

- Server Application
- Operating System
- Network Protocol

Network Routers

- Client Application
- Operating System
- Network Protocol

1

Traditional OS Support

- Same:
  - arbitrate resource demands (efficient)
  - abstractions of low-level devices (convenient)
- Different:
  - no longer just protect memory of process
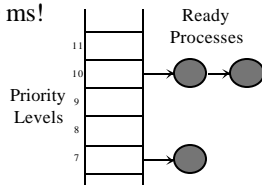 - negotiated slice of CPU time
  - I/O bandwidth
  - timing!

OS Problems in Supporting Multimedia

- Process Scheduling (now)
- Memory Management (later)
- Storage Scheduling (later, cs4513)
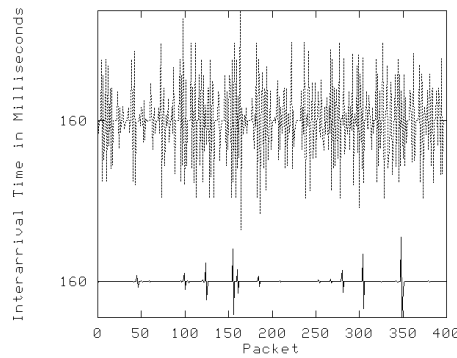- Network Interface (later, cs4514)
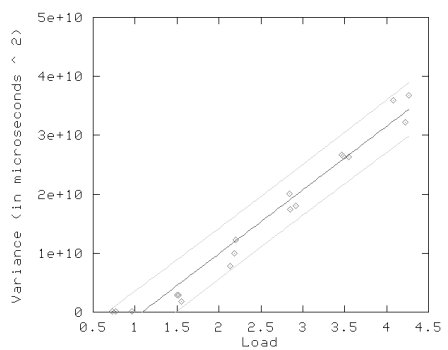
Process Scheduling Shortcomings

- Multi-level feedback queue
- Typical time slice 100 ms
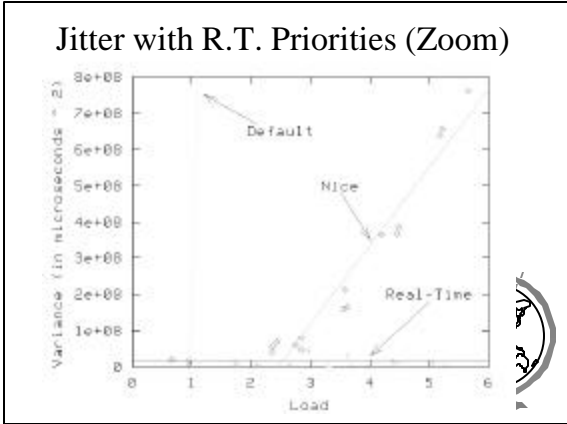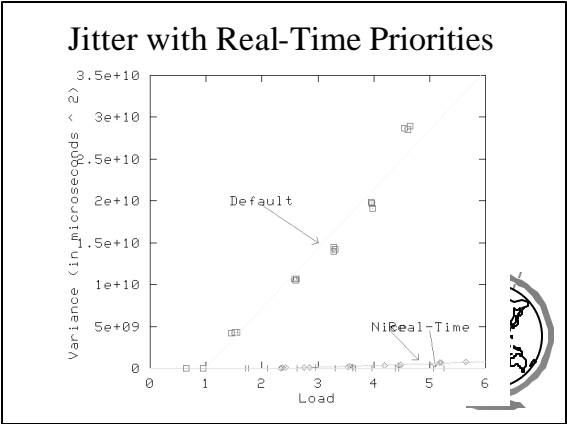- Dispatch latency 100 ms!
  - Varies (Jitter)

Ready Processes

Priority Levels

11
10
9
8
7

Jitter

Interarrival Time in Milliseconds

160

160

0   50   100 150 200 250 300 350 400
Packet

Jitter vs. Processor Load

Variance (in microseconds ^ 2)

5e+10
4e+10
3e+10
2e+10
1e+10
0

0.5   1   1.5   2   2.5   3   3.5   4   4.5
Load

Process Scheduling Fix?

F Priority to multimedia processes
F nice

Jitter with Real-Time Priorities

Default

NiReal-Time

Jitter with R.T. Priorities (Zoom)

Default

Nice

Real-Time

Memory Management

F Paging:
  – page faults cause jitter
  – allocation causes jitter
    ‣ global vs. local
  – solution: lock in pages
F Memory allocation generally not tied to scheduling priority

Network Interface

- TCP
  – guarantees delivery
  – stream semantics
  – fixed flow control
  – unicast
  – … big bleah!

F UDP
  – multicast add-on
  – checksum cannot be turned off
  – no notion of priority
  – no flow control
  – … little bleah!

- RTP
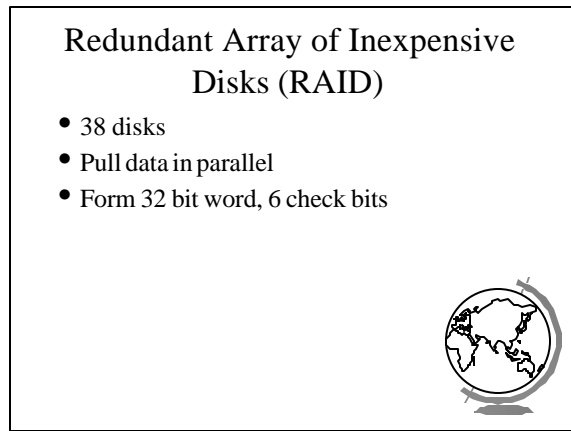 – multicast add-on
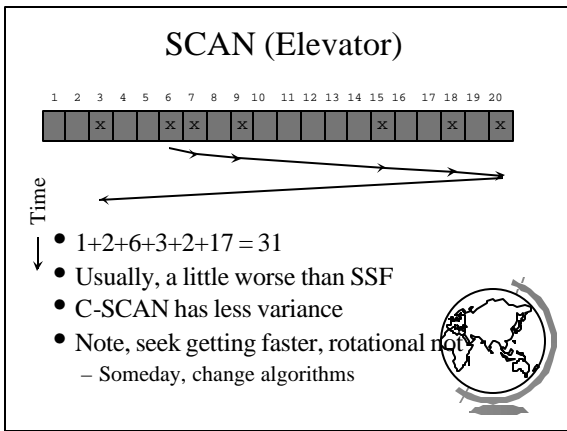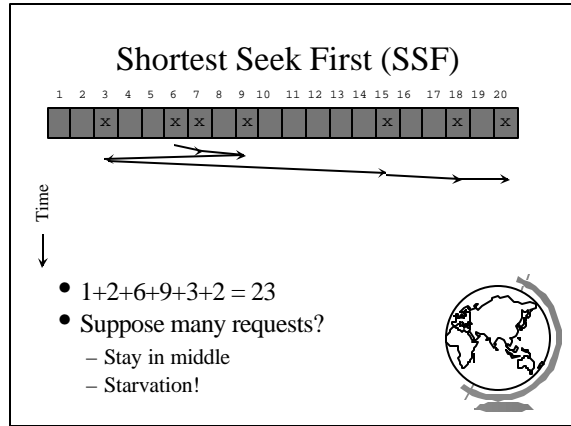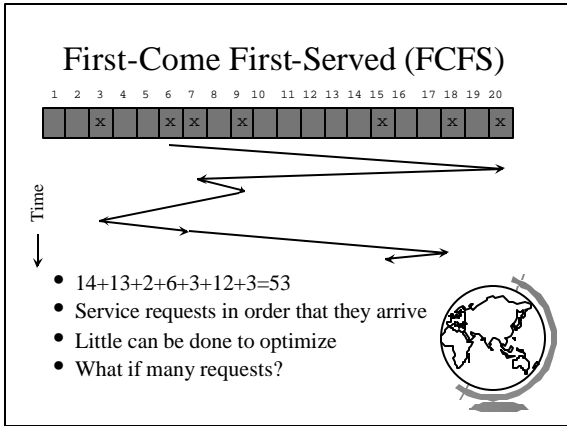 – packet sequence
  – flow control

Storage Scheduling

- Disk scheduling and layout
- DBMS

Disk Arm Scheduling

- Read time:
  – seek time (arm to cylinder)
  – rotational delay (time for sector under head)
  – transfer time (takes bits off disk)
- Seek time dominates
- How does disk arm scheduling affect seek?

First-Come First-Served (FCFS)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Time

- 14+13+2+6+3+12+3=53
- Service requests in order that they arrive
- Little can be done to optimize
- What if many requests?

Shortest Seek First (SSF)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Time

- 1+2+6+9+3+2 = 23
- Suppose many requests?
  – Stay in middle
  – Starvation!

SCAN (Elevator)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Time

- 1+2+6+3+2+17 = 31
- Usually, a little worse than SSF
- C-SCAN has less variance
- Note, seek getting faster, rotational not
  – Someday, change algorithms

Redundant Array of Inexpensive Disks (RAID)

- 38 disks
- Pull data in parallel
- Form 32 bit word, 6 check bits

Conclusion

ϝ Much work to be done
  – scheduling
  – memory management
  – network
  – disk

ϝ MQP anyone?
  – One piece in OS support puzzle