Distributed Systems

Daniel Andresen

CIS520 – Operating Systems

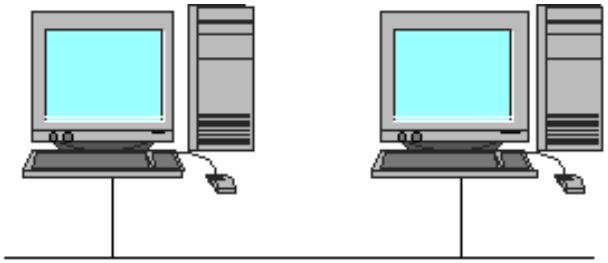
Distributed systems

- …are computing systems composed of large number of CPUs connected by a high-speed network.

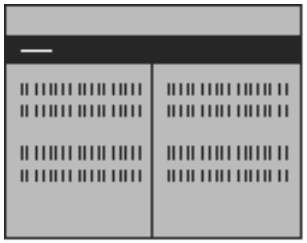
- A typical computer network connects many hosts, each capable of supplying computing services to network users.
- Topologies vary: ring, bus, switch, Internet

Main idea

Idea: use many distinct but connected computers as single large computational resource



Use this.



Instead of this.

Advantages:

- **Economics**: better price-performance ratio than mainframe.
- **Resource sharing**
- **Speed**: more total computing power than mainframe.
- **Reliability**: if one crashes, system survives. But: partial failures more likely
- **Scalability**: possible to add more nodes.

Disadvantages

- **Networking**: network can be saturated, messages can get lost.
- **Software**: radically different software needed, including operating systems.
- **Security**

Design Issues

Performance-

- Fine-grained parallelism- jobs that involve a large number of small computation may cause trouble in a distributed system w/ slow communication.
- Coarse-grained parallelism- jobs involving large computations usually are better for distributed systems.

Scalability

Avoid

1. centralized components
   - - a single mail server.
2. centralized tables
   - - a telephone book.
3. centralized algorithms
   - - doing routing based on complete information.

Keep ***computation/communication ratio*** high.

Hardware Concepts

SIMD = Single instruction, multiple data

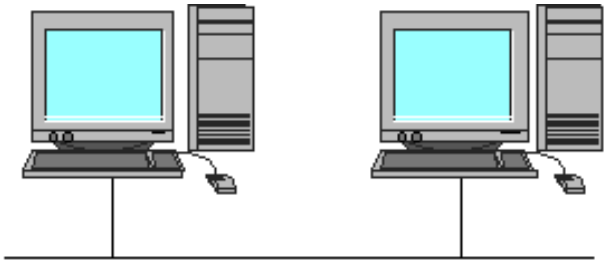
- vector processors, MMX

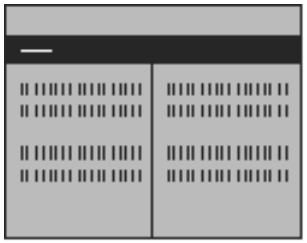
MIMD = Multiple Instructions, Multiple data.

- All distributed systems, SMP's.

multiprocessors => have shared memory.

multicomputers => do not have shared memory.





Types of Distributed Systems:

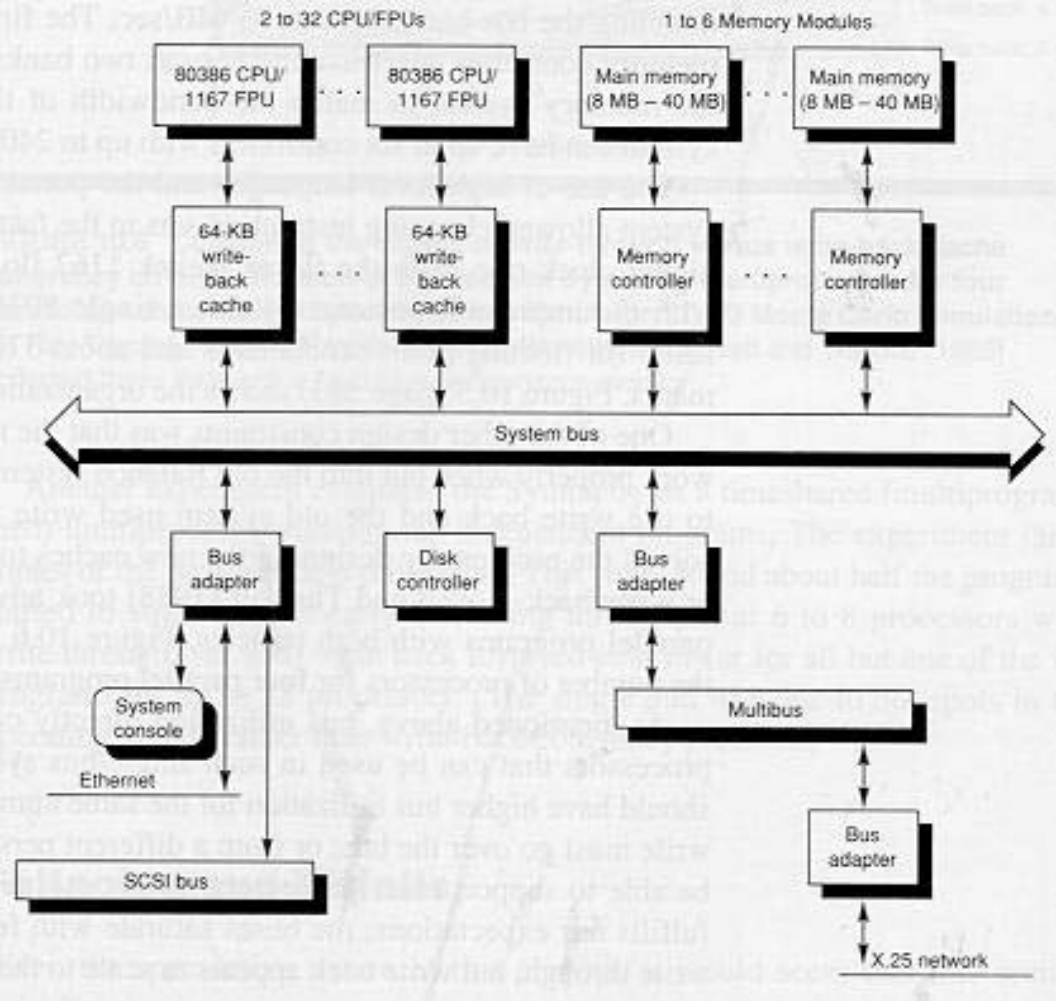
1. bus-based or switched
   - bus-based => single cable connecting all machines.
   - switched => individual wires from machine to machine
2. tightly-coupled (shared data) or loosely-coupled (no shared data)

Bus-based multiprocessors

Coherent memory - if X is just written by processor A, then a subsequent read by process B will return the value just written.

To maintain memory coherence:

- write-through cache.
- snoopy + cache - snoops on bus - value written by another CPU => update cache entry or invalidate.



Software

Types of Operating Systems

1. Loosely-coupled - allow users to be independent, but interact in limited ways.

    - Network Operation System and NFS

2. Tightly-coupled - allow single application to run on several CPU's in parallel.

    - Example: SGI Irix on O2K's