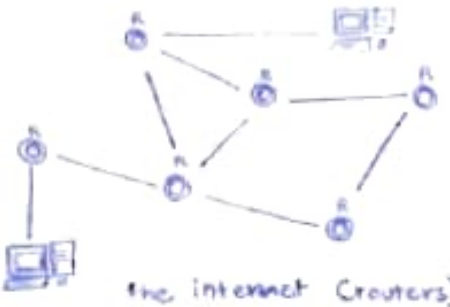
1 INTRODUCTION



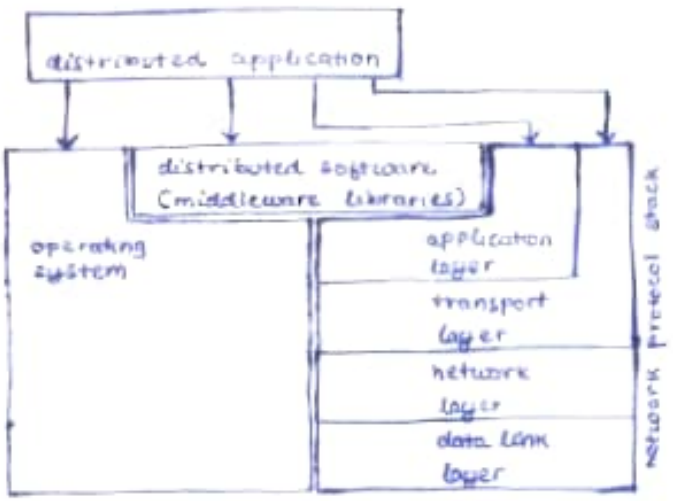
the internet (routers)

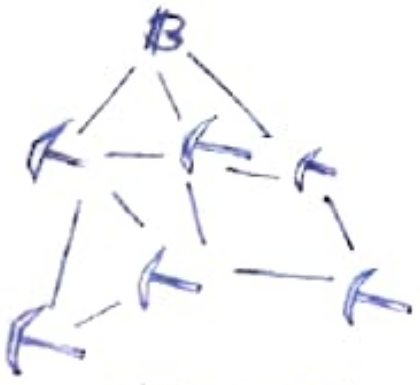
a distributed system = collection of 'independent' entities that 'cooperate' to solve a problem.

- no common physical clock
- no shared memory
- autonomy & heterogenity
- geographical separation

} "independent"

weakly coupled	strongly coupled	very strongly coupled
WAN	LAN	multiprocessor systems





bitcoin mining

(distributed execution)

OMG's → CORBA , RPC / RMI / DCOM

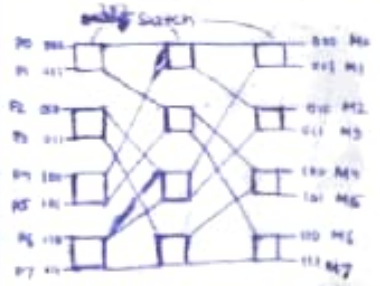
- "inherently distributed" computations (banking)
- resource "sharing" (world primer)
- access to geographically remote data & resources
- Enhanced "reliability" (internet)
- increased performance / cost ratio
- "scalability" (agriculture)
- "modularity" and incremental expandability (PL)

UMA
uniform memory access (time)

NUMA
non-uniform memory access

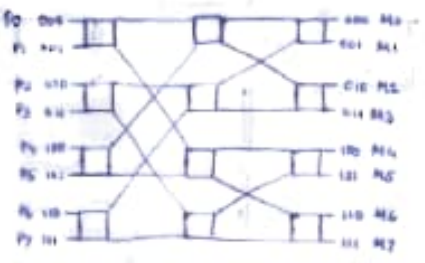
(GPU) multiprocessor system - multiple processors w/ shared memory

(supercomp) multicomputer system - multiple processors w/ shared memory



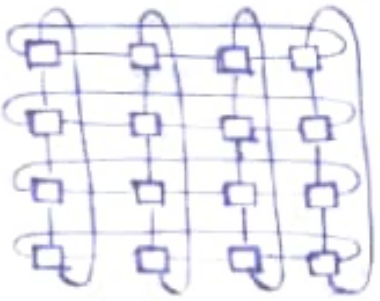
3-stage omega network (n=8, M=4)

abc → bca



3-stage butterfly network (n=8, M=4)

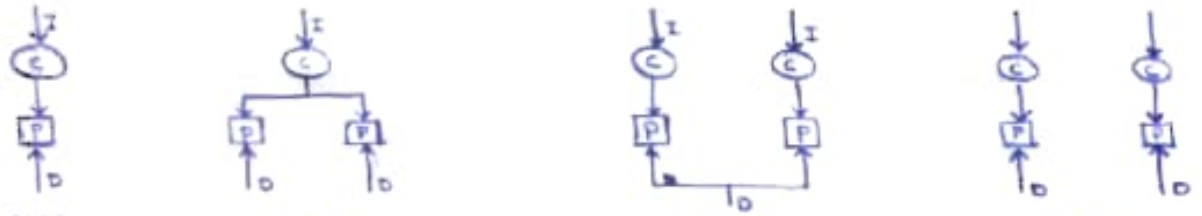
ebd aeb



wrap-around 2D-mesh (torus)

hypercube of dimension 4.

FLYNN's TAXONOMY



SISD
(CPU)

SIMD
(MMX arch.)

MISD
(visualization)

MIMD
(Server)

- coupling - interdependency / binding among modules
- parallelism - % time processors are busy
- concurrency - multiple ops., not necessarily at once
- granularity - computation - communication ratio

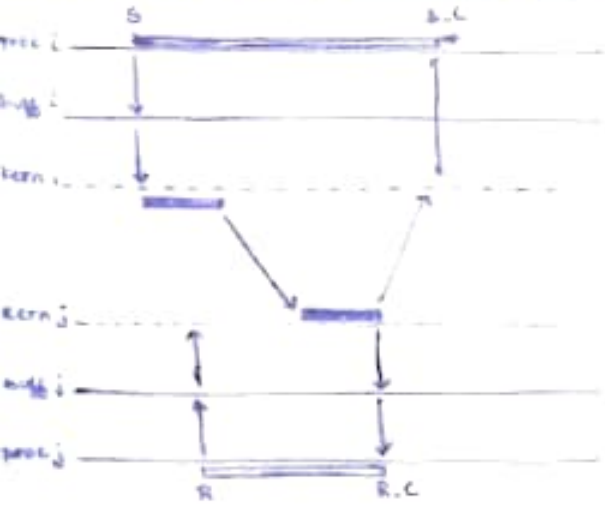
SHARED MEMORY SYSTEM

- shared address space
- use semaphores and monitors
- conceptually easier
- can emulate message passing.

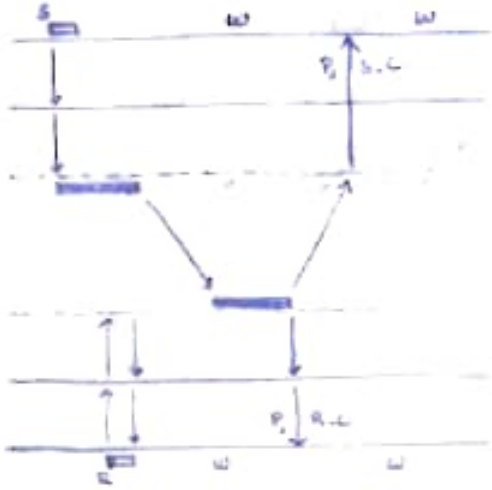
MESSAGE PASSING SYSTEM

- separated address space
- can emulate (distributed) shared memory

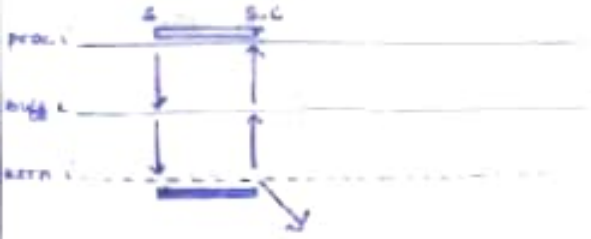
BLOCKING SYNC SEND, BLOCKING RECEIVE



NONBLOCKING SYNC SEND, NONBLOCKING RECEIVE

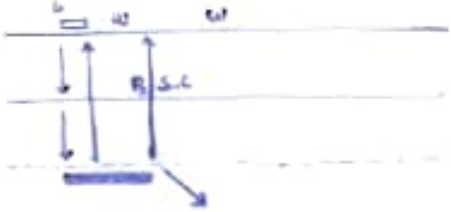


BLOCKING ASYNC SEND



- send (destination, buffer)
  - buffered or unbuffered

NONBLOCKING ASYNC SEND



- receive (source, buffer)
  - normally buffered

processor synchrony - processors execute in synchronized steps (GPU shader)

ASYNCHRONOUS EXECUTION

- no processor synchrony
- no bound on drift rate of proc. clocks
- message delays are unbounded
- execution time is unbounded

SYNCHRONOUS EXECUTION

- processors are synchronized
- drift rate of proc. clocks is bounded
- message delays occur within one step
- execution time is bounded
- algorithms easier to design & verify
- difficult to build (often emulated)
  synchronizer

SYSTEMS CHALLENGES

- communication
- synchronization
- fault tolerance
- scalability & modularity
- processes
- data storage & access
- security
- naming
- consistency & replication
- API & transparency

ALGORITHMIC CHALLENGES

- designing useful models & frameworks
- dynamic distributed graph algorithms & distributed routing algorithms
- time & global state in a distributed system
- synchronization / coordination mechanisms
  - physical clock synchronization
  - mutual exclusion
  - termination detection
  - leader election
  - deadlock detection and resolution
  - garbage collection
- group communication, multicast, and ordered message delivery
- monitoring distributed events & predicates
- distributed program design & verification tools
- debugging distributed programs
- data replication, consistency models & caching
- World Wide Web design - caching, searching, scheduling
- distributed shared memory abstraction
  - wait-free algorithms
  - register constructions
  - mutual exclusion
  - consistency models
- reliable and fault-tolerant distributed systems
  - consensus algorithms
  - voting & quorum systems
  - replication & replica management
  - distributed DBs & distr. commit

- self-stabilizing systems
- checkpointing & recovery algos
- failure detectors

- load balancing
  - data migration
  - computation migration
  - distributed scheduling
- real-time scheduling
- performance
  - metrics
  - measurement methods / tools

APPLICATIONS

- mobile systems
- sensor networks
- ubiquitous / pervasive computing
- peer-to-peer computing
- publish-subscribe, content-distribution & multimedia
- distributed agents
- distributed data mining
- "grid computing" ← PROJECT
- security in distributed systems