

CHAPTER 2: DISTRIBUTED SYSTEM CONCEPTS AND ARCHITECTURES

Characteristics of distributed systems

- *multiplicity*
 - multiple users/clients
 - concurrent processes/servers
 - replication of resources
- *dispersion*
 - distributed resources
 - decentralized control
- *Non-negligible communication delay*
- *Lack of global information*
- *Failures*

Goals

- *Efficiency*
- *Flexibility*
- *Consistency*
- *Robustness*

Transparency

- *Access*
- *Location* a.k.a. *Name*
- *Migration*
- *Concurrency*
- *Replication*
- *Parallelism*
- *Failure*
- *Performance*
- *Size*
- *Revision*

Transparency

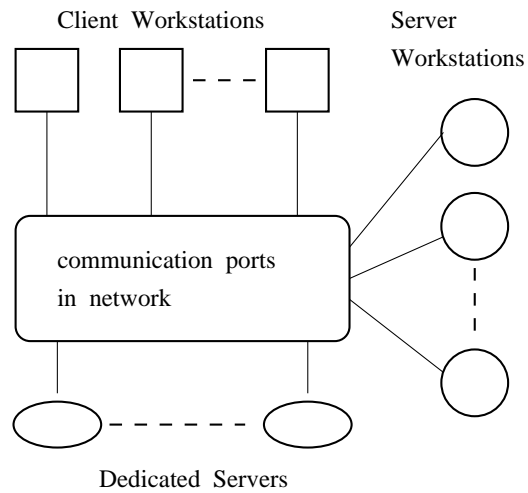
- *Access*
Uniform way to access local and remote resources
- *Location*
Logical names for resources independent of location
- *Migration*
Ability to move object without renaming it (*location independence*)
- *Concurrency*
Ability to share objects without interference
- *Replication*
Consistency across multiple instances/shares of objects
- *Parallelism*
Parallel activities not visible to users
- *Failure*
Fault tolerance to avoid damage, disruption (*graceful degradation*)
- *Performance*
Consistent and predictable performance level with changes in system structure, load distribution; reasonable delays, variations
- *Size*
Incremental growth without user awareness needed - size has no effect on user perceptions (need modularity, scalability)
- *Revision*
Vertical growth (version changes in software) not noticeable

Transparency Categorization Based on Goals

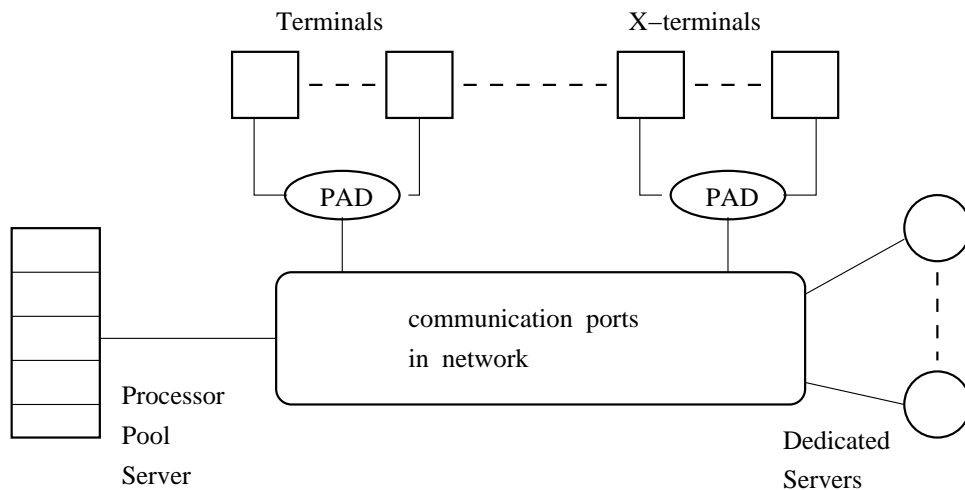
- Efficiency
 - *Concurrency*
 - *Parallelism*
 - *Performance*
- Flexibility
 - *Access*
 - *Location*
 - *Migration*
 - *Size*
 - *Revision*
- Consistency
 - *Access*
 - *Replication*
 - *Performance*
- Robustness
 - *Failure*
 - *Replication*
 - *Size*
 - *Revision*

Distributed system architectures

The workstation-server model



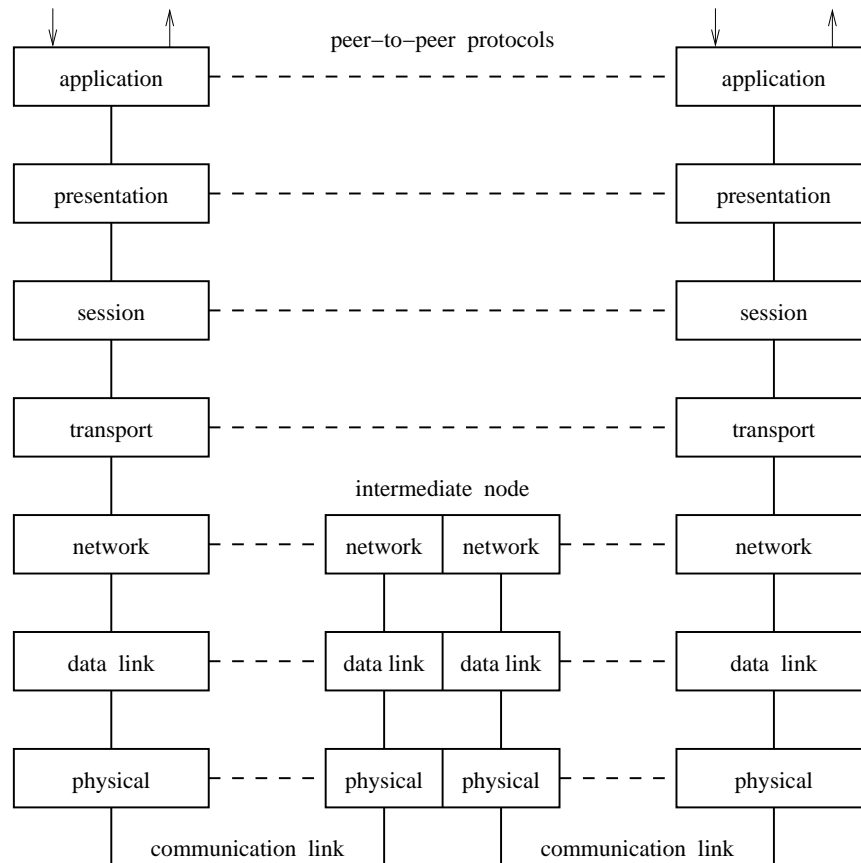
The processor-pool model



Communication network architectures

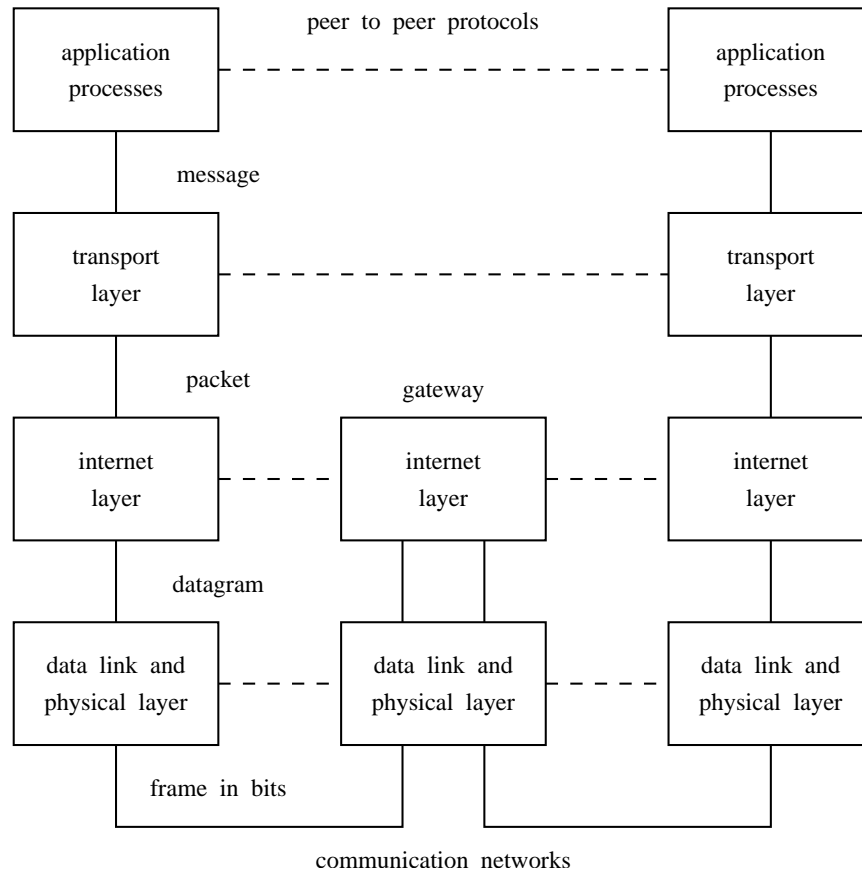
OSI protocol suite

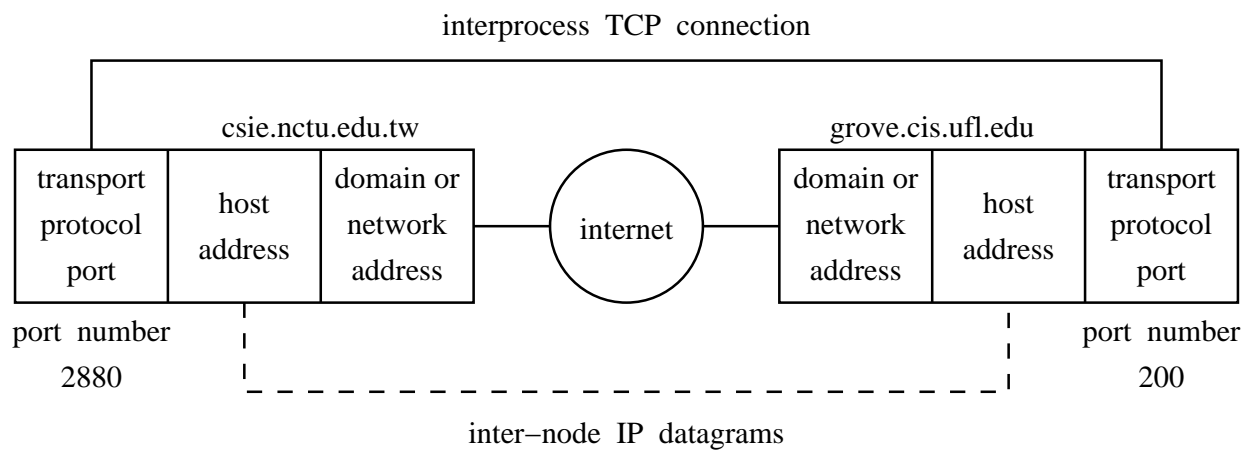
Connecting hosts



TCP/IP protocol suite

Connecting networks





Major design issues

- *Object models and naming scheme*: object, client/server, name server
- *Distributed coordination*: synchronization and distributed algorithms
- *Interprocess communication*: socket and RPC
- *Distributed resources*: sharing and replication
- *Fault-tolerance and security*: redundancy, recovery, protection

DCE example

