BITS Pilani
Pilani Campus

CS G623: Advanced Operating Systems

Lecture 1

Amit Dua

August 2, 2018

Topics to be discussed

- Advanced OS
- Types of AOS
- Motivation for Distributed systems
- Components of distributed system
- Goals
  - Access remote resources
    - Resource
    - Why collaboration
    - Price for collaboration
  - Transparency
    - Acc, loca, reloca, migra, repli, concurr, security
    - limitations
  - Open
    - Interface specification complete, neutral
    - Interoperability, portability, extensible
  - Scalable
    - Size, geography, administration
    - Central server, LAN, different organizations
    - Hiding latency, distribution, replication
    - Asynchronous, client side, DNS, Internet, caching, cost of concurrent updation
- Types of DS
  - Cluster and grid computing systems

TEXT BOOK

T1 M. Singhal & N. Shivaratri, “Advanced Concepts in Operating Systems: Distributed, Database and Multiprocessor Operating Systems”, Tata McGraw Hill, 2015.

EVALUATION SCHEME

S No.	Component & Nature	Duration	Weightage	Date and Time
1.	Assignment (Open Book)	*	10%	*
2.	Mid-semester Test (Closed Book)	90 mins	25%	9/10 11:00 - 12:30 PM
3.	Surprise Quizzes (Closed Book) – (Total 4)	20 mins each	15%	*
3.	Term paper presentation	15 mins (presentation) + 5 mins (Q/A)	10%	*
4.	Comprehensive Exam (20% OB and 20% CB)	3 hrs	40%	1/12 AN

Advanced OS

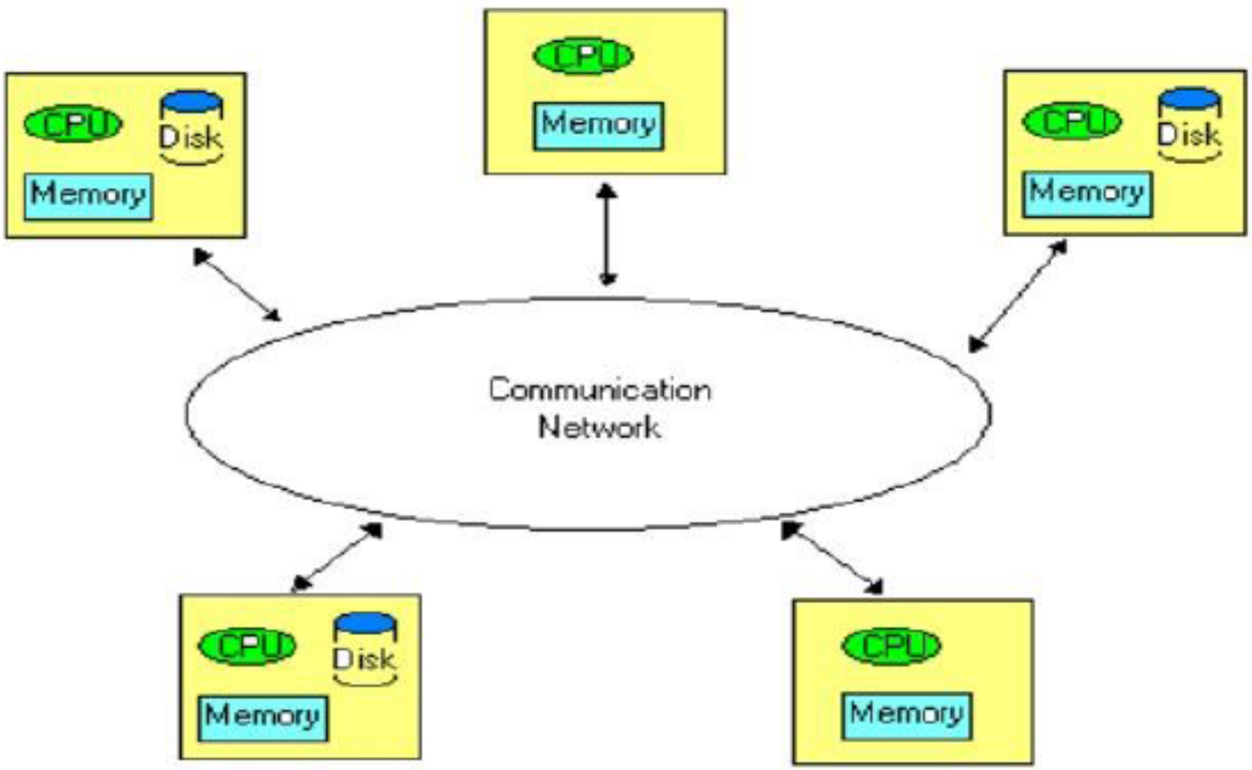
- Advanced OS possible because

1. Advances in architecture of multiprocessor systems and
2. Applications require high speed systems

Distributed OS:

- OS for network of autonomous computers connected by communication network
- Control and manage H/w and S/w resources
- User views as single powerful computer system.
- User not aware of where program executed or location of resources accessed.

Architecture of Distributed OS



Multiprocessor OS:

- A Tightly coupled system where
- Set of processors that share the main memory

Database OS:

- Systems support concurrency control and failure recovery
- Store, retrieve and manipulate large volumes of data

Real-time OS:

- Systems have job completion deadline.

Reason for distributed system's existence

✓ Development of powerful microprocessors

- From 1980's to today
- From 8 bit machines to 16 bit,32 bit, to 64 bit CPU became common
- From machine costing $10 million for one instruction per second to $1000 machine executing 1 billion instructions per second
- Gain of 10^{13}
- Totally unprecedented in other industries
- Imagine cost of mercedes/ rolls royce
- $1 dollar running billion miles per liter
- Telling 200 page manual how to open the door

Cont..

- ✓ High speed computer networks
- Information transferred in micro seconds
- 10 billion bits/ sec in LAN to
- 64 Kbps and gigabits per seconds for communication anywhere in globe

- Because of these two technologies it is easy to put computing resources connected with high speed networks.

Definition of distributed systems

innovate achieve lead

A distributed system is a collection of independent computers that appears to its users as a single coherent system

- Autonomous components/ computers
- Users think they are dealing with a single system

- These AS collaborate for some purpose.

- How they collaborate is the subject matter of this course.

DS

- No assumption on type of computer
- No assumption on way the computers are interconnected

- Difference between various computers and way they are interconnected is hidden from user

- Users and applications can interact with a distributed system in a consistent and uniform way,
- No matter where and when interaction takes place

DS cont..

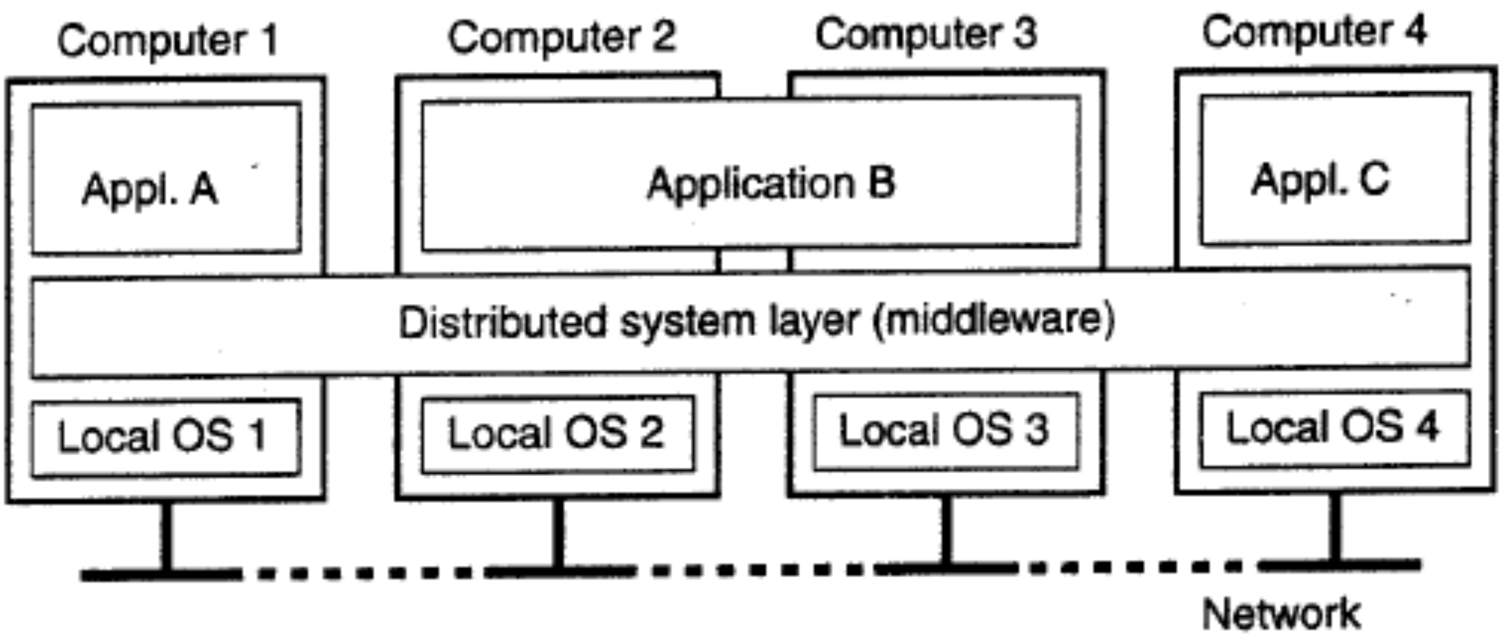
- Easy to expand
- Hiding how these computers actually take part in system as a whole

- Users and applications should not notice that parts are being replaced, fixed or new parts are added.

- **How this system can be made??**

DS

A layer of software between local OS and application is provided.



Ques

- Just because we can build distributed system does not mean it’s a good idea.

- We can have 4 floppy drives in a computer but who uses floppy now?

- What are the goals that should be met to make building a DS worth an effort?

Goals

DS should

1. Make resources easily accessible
2. Hide the fact that resources are distributed across n/w
3. Should be open
4. scalable

Make resources easily accessible

- distributed system is to make it easy for the users (and applications)
- to access remote resources, and
- to share them in a controlled and efficient way.

- **<u>What are resources??</u>**

Ques

- printers, computers, storage facilities, data, files, Web pages, and networks,

to name just a few.

- Why share resources??

Resources cont..

- ✓ Economics

it is cheaper to let a printer be shared by several users in a small office than having to buy and maintain a separate printer for each user.

- ✓ Connecting users and resources also makes it easier to collaborate and exchange information

Internet with its simple protocols for exchanging files, mail. documents, audio, and video

groupware, that is, software for collaborative editing, teleconferencing, and so on.

Is there a cost to sharing ??

Distribution Transparency

A distributed system that is able to present itself to users and applications as if it were only a single computer system is said to be transparent.

What do DSs hide or can hide?

Distribution transparency

Transparency	Description
Access	Hide differences in data representation and how a resource is accessed
Location	Hide where a resource is located
Migration	Hide that a resource may move to another location
Relocation	Hide that a resource may be moved to another location while in use
Replication	Hide that a resource is replicated
Concurrency	Hide that a resource may be shared by several competitive users
Failure	Hide the failure and recovery of a resource

Degree of Transparency

there are situations in which attempting to completely hide all distribution aspects from users is not a good idea

electronic newspaper to appear in your mailbox before 7 A.M. local time

connects a process in San Francisco to a process in Amsterdam

Signal transmission is not only limited by the speed of light. but also by limited processing capacities of the intermediate switches.

cont

many Internet applications repeatedly try to contact a server before finally giving up.

we need to guarantee that several replicas, located on different continents, need to be consistent all the time

single update operation may now even take seconds to complete, something that cannot be hidden from users

where the very notion of location and context awareness is becoming increasingly important, it may be best to actually *expose distribution* rather than trying to hide it.

Cont

distribution transparency should be considered together with other issues such as performance and comprehensibility.

The price for not being able to achieve full transparency may be surprisingly high.

Openness

✓ Making Resources Accessible
✓ Distribution Transparency
Openness

Openness

a system that offers services according to standard rules that describe the syntax and semantics of those services.

Difference between Syntax and semantics??

in computer networks, standard rules govern the format, contents, and meaning of messages sent and received.

Services are generally specified through interfaces

Cont..

Syntax defines the names of the functions that are available together with types of the parameters, return values, possible exceptions that can be raised, and so on.

the semantics of interfaces tell what those services do

Cont..

Proper specifications allows two independent parties to build completely different implementations of those interfaces, leading to two separate distributed systems that operate in exactly the same way.

Proper specifications are complete and neutral

Complete means that everything that is necessary to make an implementation has indeed been specified.

specifications do **not** prescribe what an implementation should look like: they should be neutral

Completeness and neutrality are important for interoperability and portability

Interoperability characterizes the extent by which two implementations of systems or components

from different manufacturers can co-exist and work together by merely relying on each other's services as specified by a common standard.

Portability characterizes to what extent an application developed for a distributed system *A*

can be executed. without modification, on a different distributed system *B that* implements the same interfaces as *A.*

it should be easy to add new components or replace existing ones without affecting those components that stay in place.

What is the forth goal?

Scalability

Scalability of a system can be measured along at least three different dimensions

with respect to its size, meaning that we can easily add more users and resources to the system

geographically scalable system is one in which the users and resources may lie far apart.

system can be administratively scalable, that it can still be easy to manage even if it spans many independent administrative organizations.

ques

What are disadvantages of a single server?

Single server problems

the server can become a bottleneck as the number of users and applications grows.

virtually unlimited processing and storage capacity, communication with that server will eventually prohibit further growth.

DNS maintains information on millions of computers worldwide and forms an essential service for locating Web servers.

cont..

In a large distributed system, an enormous number of messages have to be routed over many lines.

Centralized algorithm bad idea. Why?

Concept	Example
Centralized services	A single server for all users
Centralized data	A single on-line telephone book
Centralized algorithms	Doing routing based on complete information

Ques

Why it is currently hard to scale existing distributed systems that were designed for local-area networks?

Problems with scaling existing LAN

- they are based on synchronous communication
- a party requesting service, client, blocks until a reply is sent back.
- Consider WAN

- Communication in wide-area networks is inherently unreliable
- virtually always point-to-point.

- what would happen if we tried to locate a service this way in the Internet
- a single mail server is used for an entire country

Ques

Can different organizations collaborate?

Problems with collaboration in different organizations

- independent administrative domains
- conflicting policies with respect to resource usage (and payment),
- management, and
- Security

- system administration may have tested and certified applications
- trust does not expand naturally across domain boundaries.

Measures to collaborate

1. protect itself against malicious attacks from the new domain
2. the new domain has to protect itself against malicious attacks from the distributed system

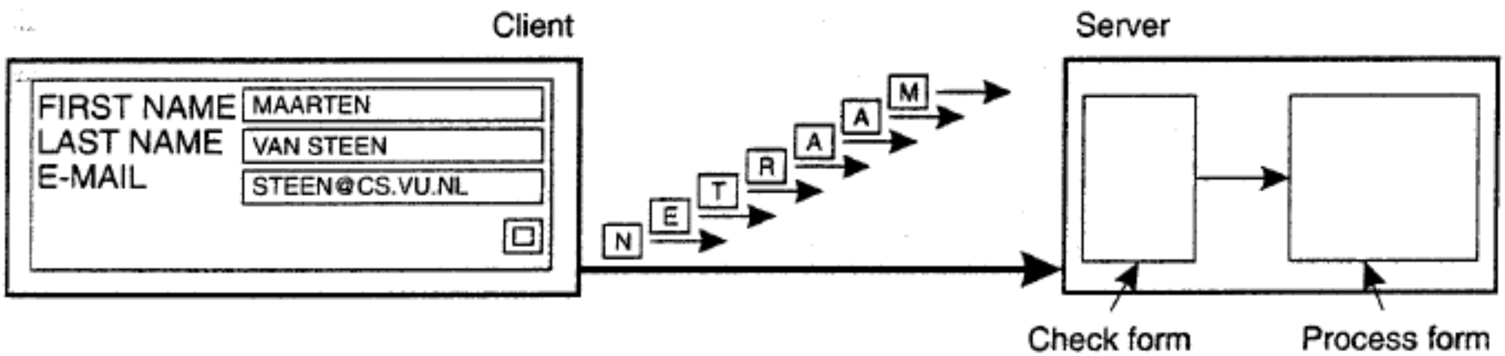
- downloading programs such as applets in Web browsers.
- new domain does not know behavior what to expect from such foreign code

How can we scale??

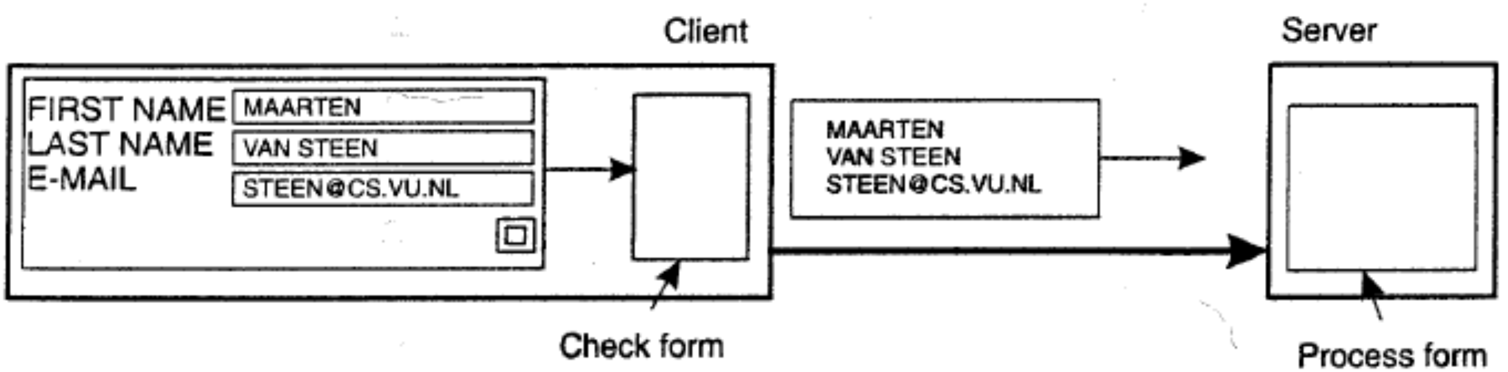
Ans

- hiding communication latencies,
  - asynchronous communication
  - Shifting to customer side
- distribution,
  - Dns
  - Internet
- Replication
  - Caching
  - Difference
  - Cost ??

moving part of the computation to the client process

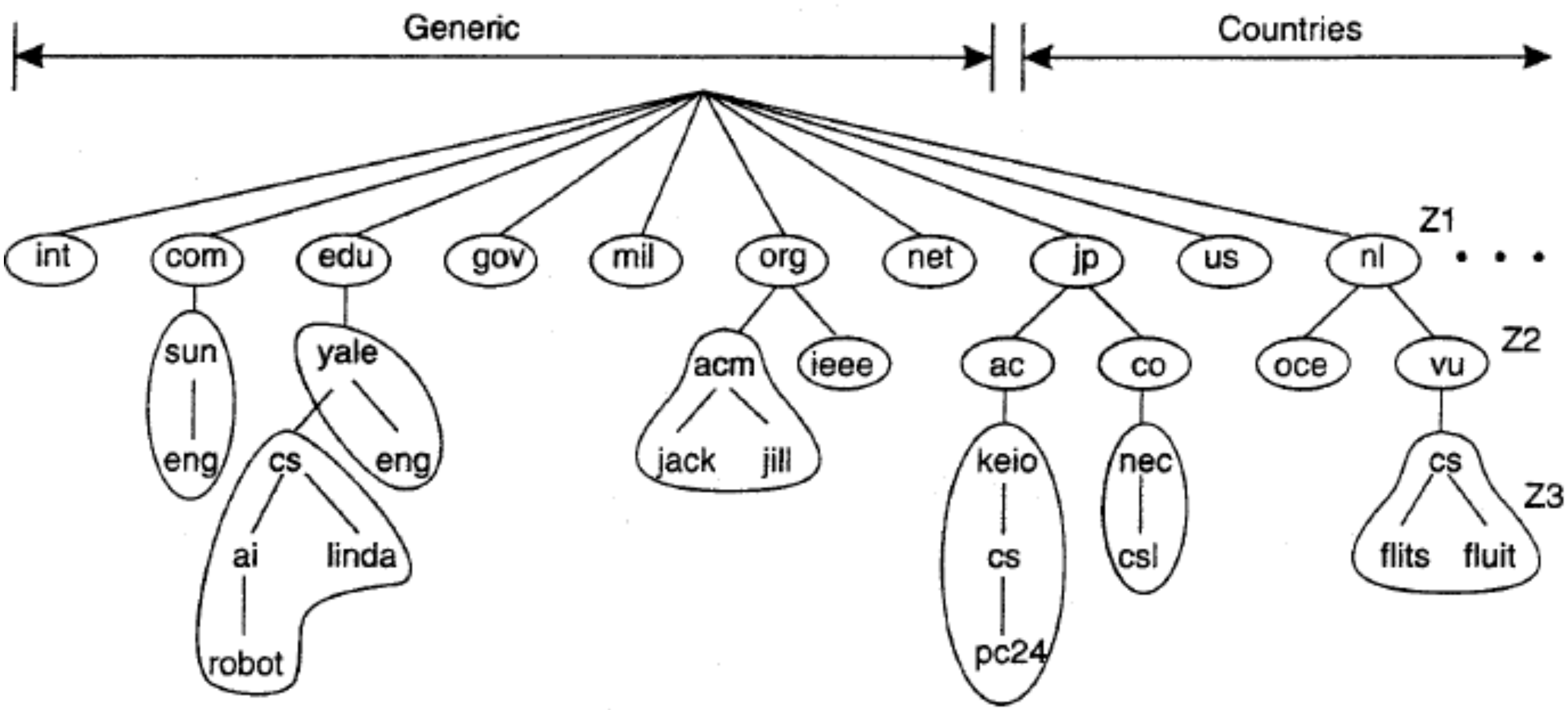


(a)



(b)

Distribution in DNS



False assumptions: be careful

1. The network is reliable.
2. The network is secure.
3. The network is homogeneous.
4. The topology does not change.
5. Latency is zero.
6. Bandwidth is infinite.
7. Transport cost is zero.
8. There is one administrator

TYPES OF DISTRIBUTED SYSTEMS

1. Distributed computing systems,
    1. Cluster Computing Systems
    2. Grid Computing Systems
2. Distributed information systems,
    1. Transaction Processing Systems and
    2. Enterprise Application Integration
3. Distributed embedded systems/Distributed Pervasive Systems
    1. Home Systems
    2. Electronic Health Care Systems
    3. Sensor Networks.

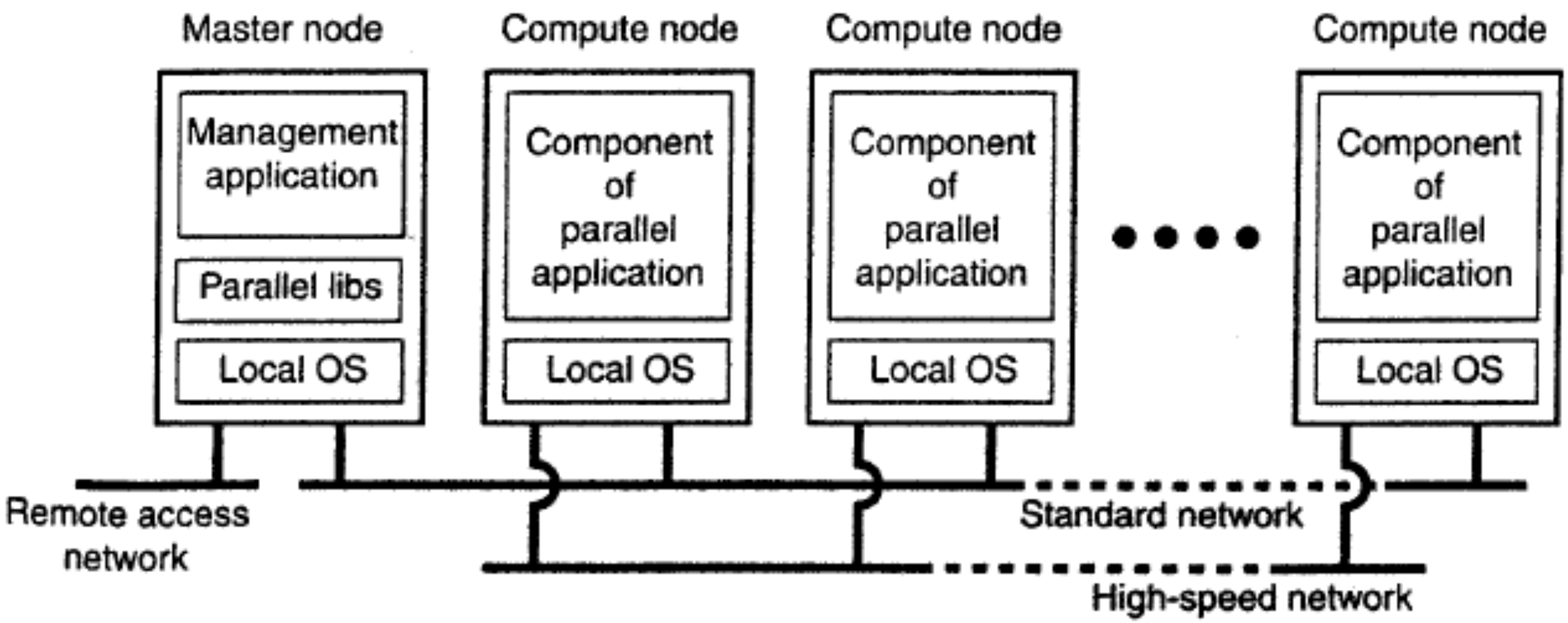
Distributed computing systems

- used for high-performance computing tasks.

In cluster computing

1. the underlying hardware consists of a collection of similar workstations or PCs,
2. closely connected by means of a high speed local-area network.
3. In addition, each node runs the same operating system

cluster computing system.



cluster consists of a collection of compute nodes that are controlled and accessed by means of a single master node.

The master handles the

1. allocation of nodes to a particular parallel program,
2. maintains a batch queue of submitted jobs, and
3. provides an interface for the users of the system.
4. Actually runs the middleware

Cluster summary

- no assumptions are made concerning

1. hardware,
2. operating systems,
3. networks,
4. administrative domains,
5. security
6. policies, etc

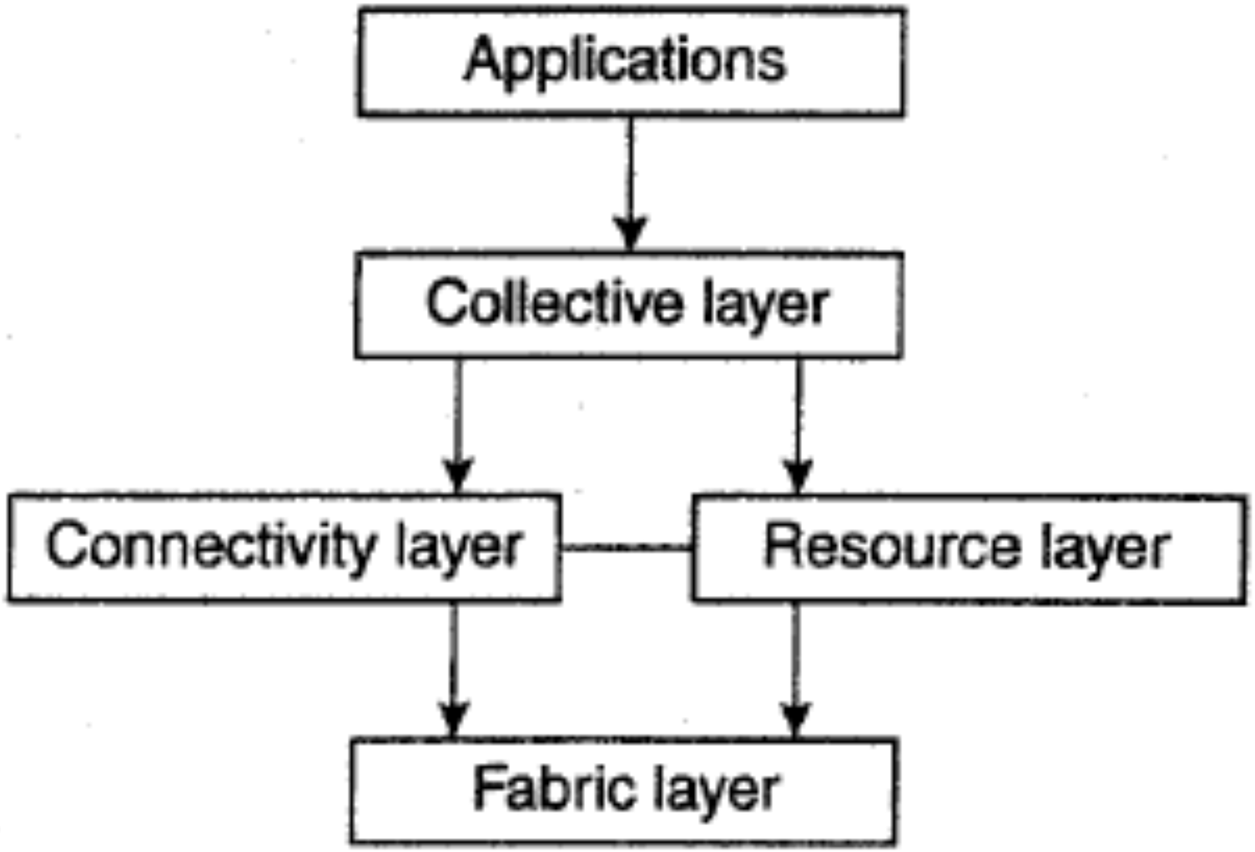
Distributed Computing Systems : Grid Computing

- resources from different organizations are brought together to allow the collaboration of a group of people or institutions.

resources consist of

- compute servers (including supercomputers, possibly implemented as cluster computers),
- storage facilities, and
- databases.

layered architecture for grid computing systems



Components in detail

- lowest *fabric layer provides interfaces* to local resources at a specific site.

- The *connectivity layer consists of communication protocols for supporting* grid transactions that span the usage of multiple resources.
- E.g transfer data between resources
- access a resource from a remote location.
- contain security protocols to authenticate users and resources

Resource layer

- The *resource layer is responsible for managing a single resource*
- responsible for access control
- rely on the authentication and functions provided by the connectivity layer
- to perform specific operations such as creating a process or reading data

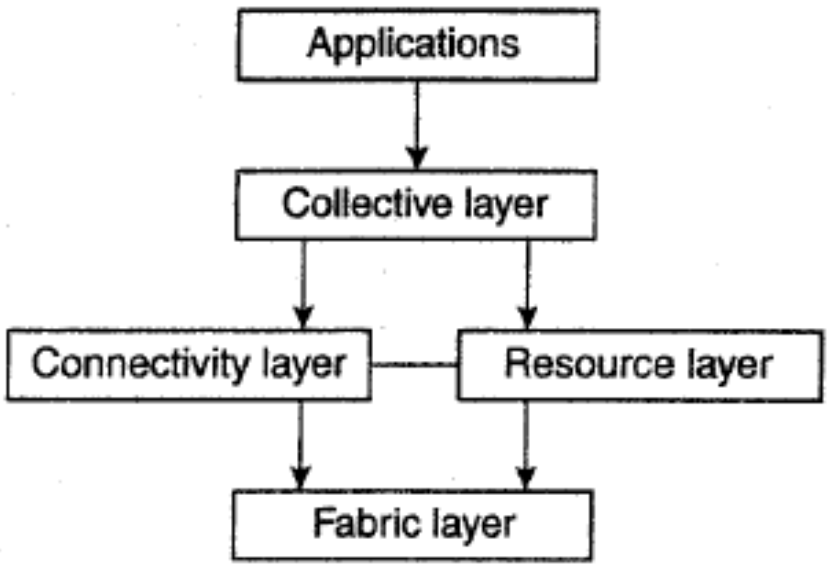
collective layer

- It deals with handling access to multiple resources and
- typically consists of services for
- Resource discovery, allocation and scheduling of tasks onto multiple resources,
- data replication

- many different protocols for many different purposes, reflecting the broad spectrum of services it may offer to a virtual organization.

application layer

- the *application layer consists of the applications that operate within a* virtual organization and
- which make use of the grid computing environment.



Distributed Information Systems

- Interoperation among applications

1. Transaction Processing Systems

- Clients wrap a number of requests, possibly for different servers, into a single larger request and have it executed as a distributed transaction.
- all, or none of the requests would be executed

2. enterprise application integration (EAI)

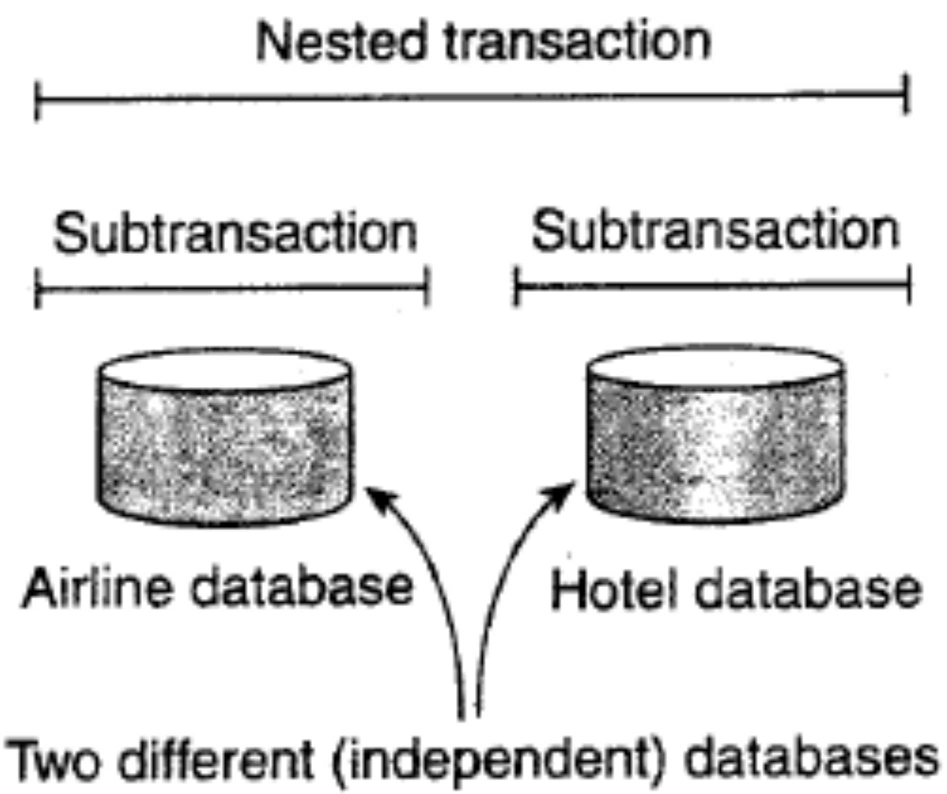
- applications communicate directly with each other

Transaction Processing Systems

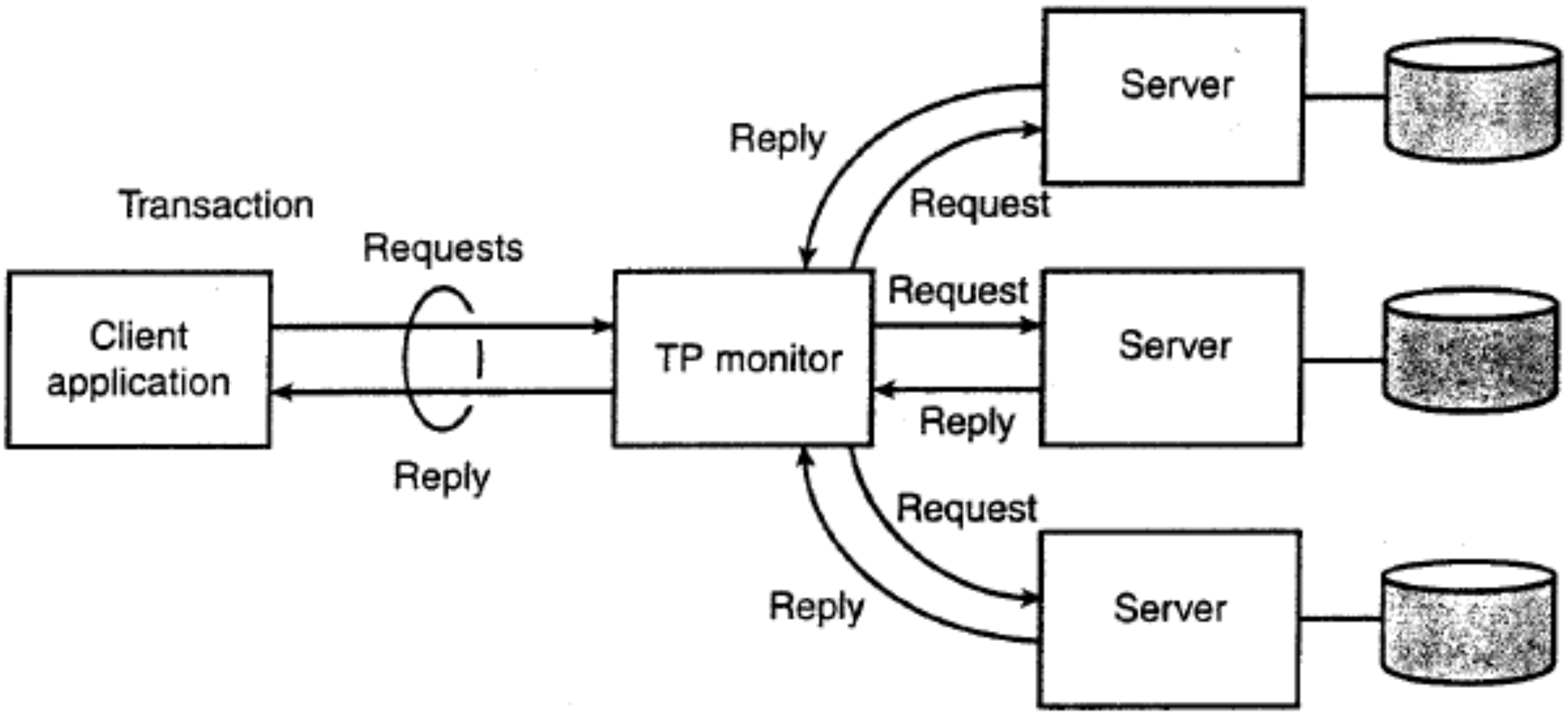
Follows ACID property

1. Atomic: To the outside world, the transaction happens indivisibly.
2. Consistent: The transaction does not violate system invariants.
3. Isolated: Concurrent transactions do not interfere with each other.
4. Durable: Once a transaction commits, the changes are permanent.

Nested transactions



Transaction processing monitor

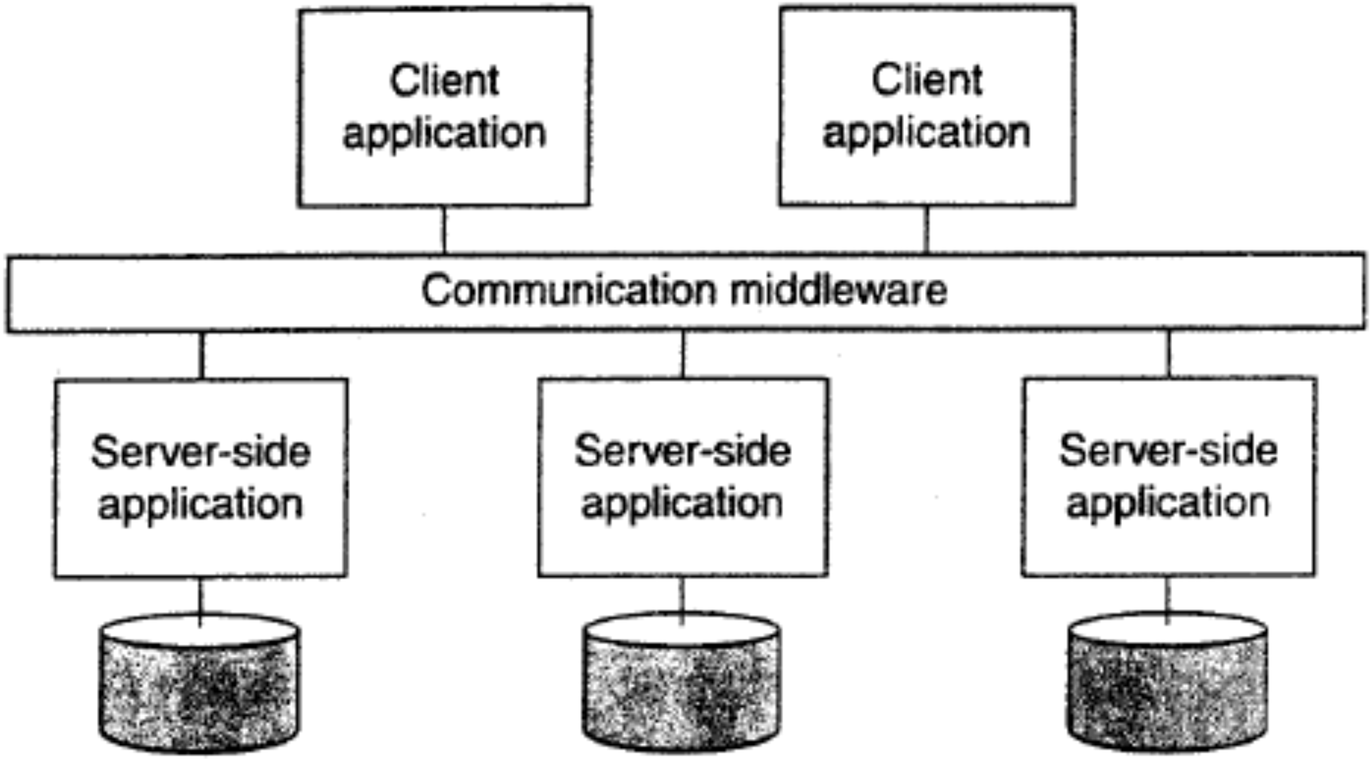


Enterprise Application Integration

- With remote procedure calls (RPC),
- an application component can effectively send a request to another application component
- by doing a local procedure call
- Request being packaged as a message and sent to the callee.

- Remote Method Invocations (RMI) operates on objects instead of applications

Distributed Information Systems :Enterprise Application Integration



Characteristics common

In Distributed Information Systems and Distributed Computing Systems

1. DS are stabile because of transparency
2. nodes are fixed and have a more or less permanent and
3. high-quality connection to a network.

Distributed Pervasive Systems

- instability is the default behavior

Devices are

1. small,
2. battery-powered,
3. mobile, and
4. having only a wireless connection,

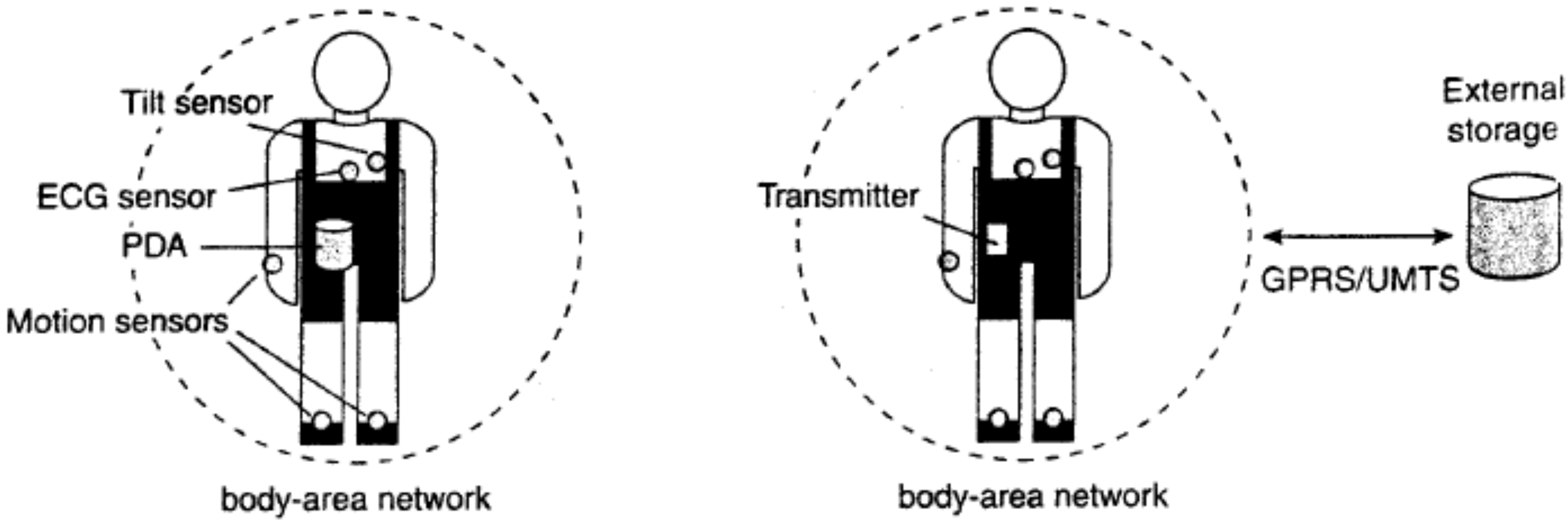
- Characteristics

1. Embrace contextual changes: aware of env. change
2. Encourage ad hoc composition: plug and play
3. Recognize sharing as the default: where info storage change

Home Systems

- *personal space*
- Universal Plug and Play (UPnP) standards
- recommenders

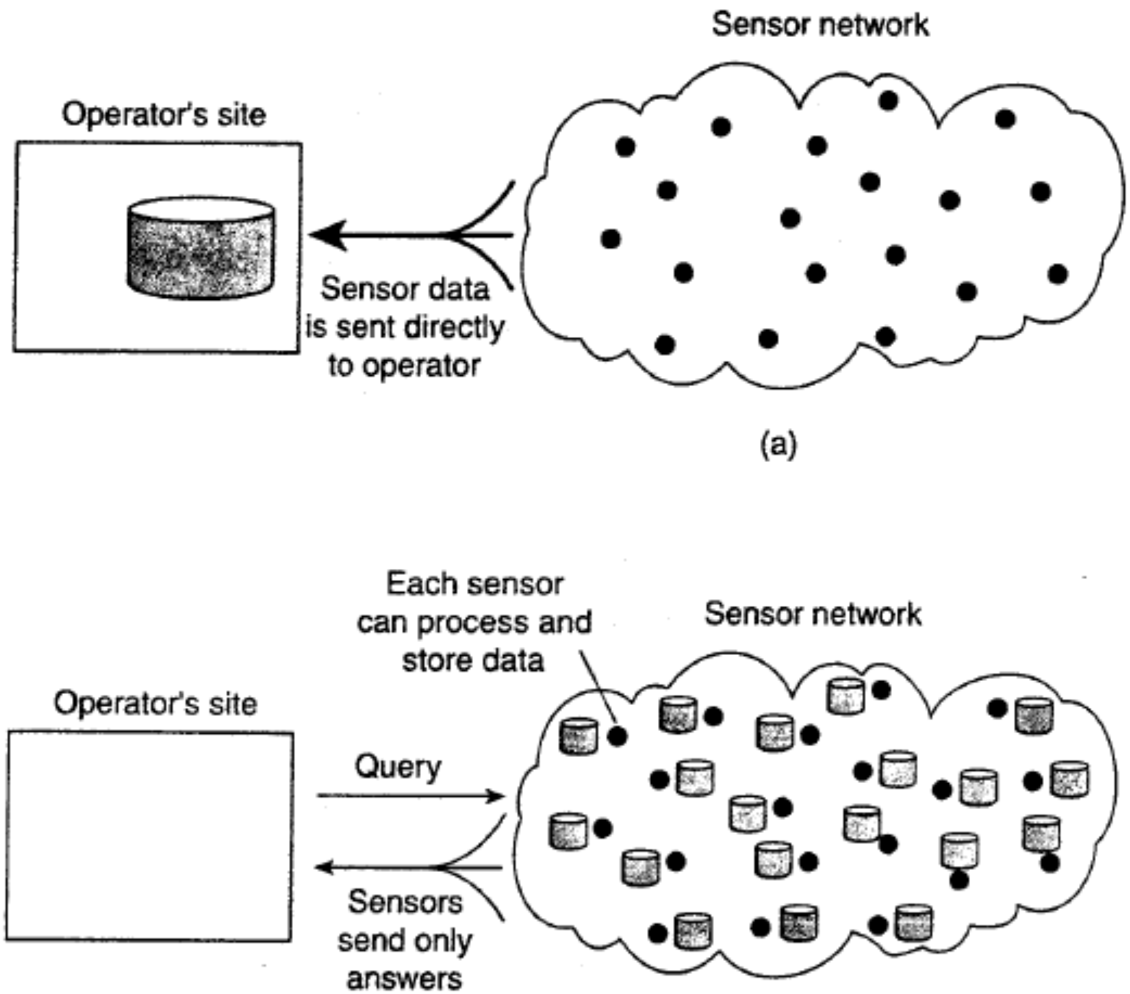
Electronic Health Care Systems



Electronic Health Care Systems

1. Where and how should monitored data be stored?
2. How can we prevent loss of crucial data?
3. What infrastructure is needed to generate and propagate alerts?
4. How can physicians provide online feedback?
5. How can extreme robustness of the monitoring system be realized?
6. What are the security issues and how can the proper policies be enforced?

Sensor Networks



Summary

Advancement in comt Tech, and CN

DS have come with goals of remote access, transparent access, open and scalable system

There are pitfalls

DS can be Distributed computing- cluster(systems, os same, high speed LAN)

Grid- administrative different to share resources

Fabric layer-interface to local resources

Connectivity-communication and security protocols, access

Resource- access control

Collective- protocols and multiple resources

application

Distributed information system: transaction and application integration- RPC, local procedure calls, message, Rmi, objects

Pervasive system, mobile, low power, wireless, small