

Distributed Systems

CSE 4084

1. What is a Distributed System?

Definition: A *distributed system* is one in which **components** located at networked computers communicate and **coordinate** their actions only by passing **messages**. This definition leads to the following characteristics of distributed systems:

- Concurrency of components
- Lack of a global 'clock'
- Independent failures of components

1.1 Centralized System Characteristics

- One component with non-autonomous parts
- Component shared by users all the time
- All resources accessible
- Software runs in a single process
- Single point of control
- Single point of failure

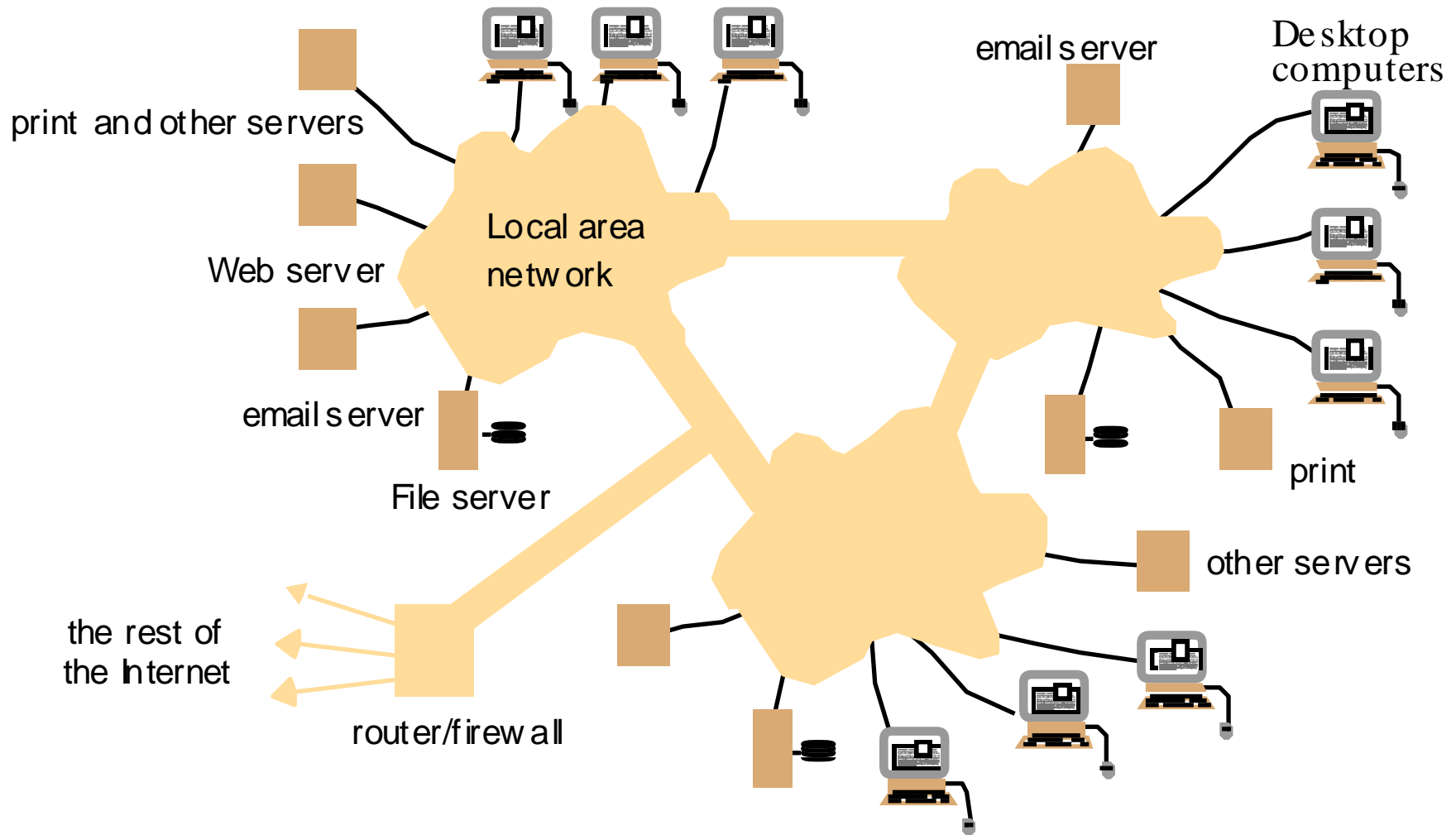
1.2 Distributed System Characteristics

- Multiple autonomous components
- Components are not shared by all users
- Resources may not be accessible
- Software runs in concurrent processes on different processors
- Multiple points of control
- Multiple points of failure

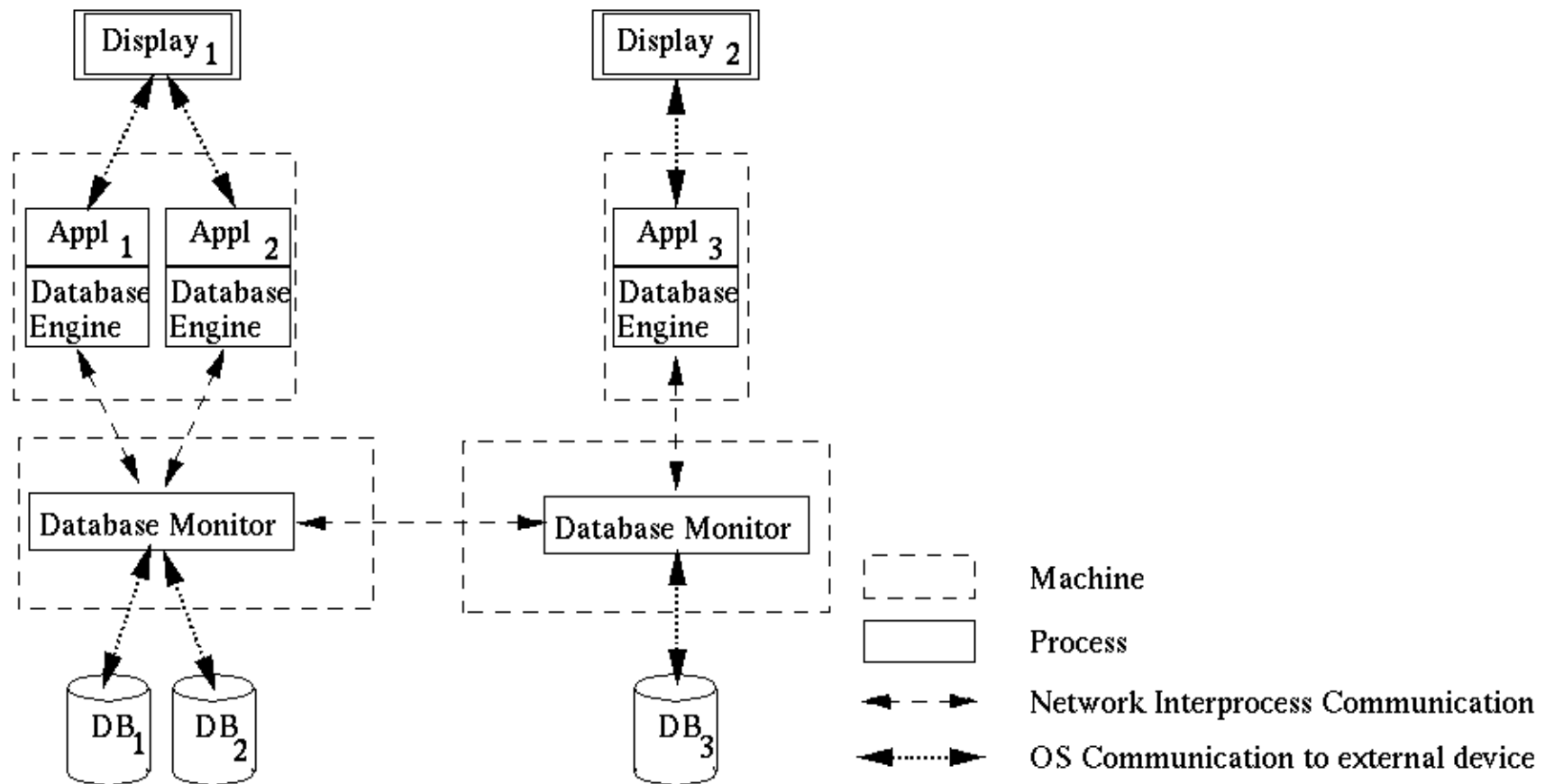
2. Examples of Distributed Systems

- Local Area Network and Intranet
- Database Management System
- Automatic Teller Machine Network
- Internet/World-Wide Web
- Mobile and Ubiquitous Computing

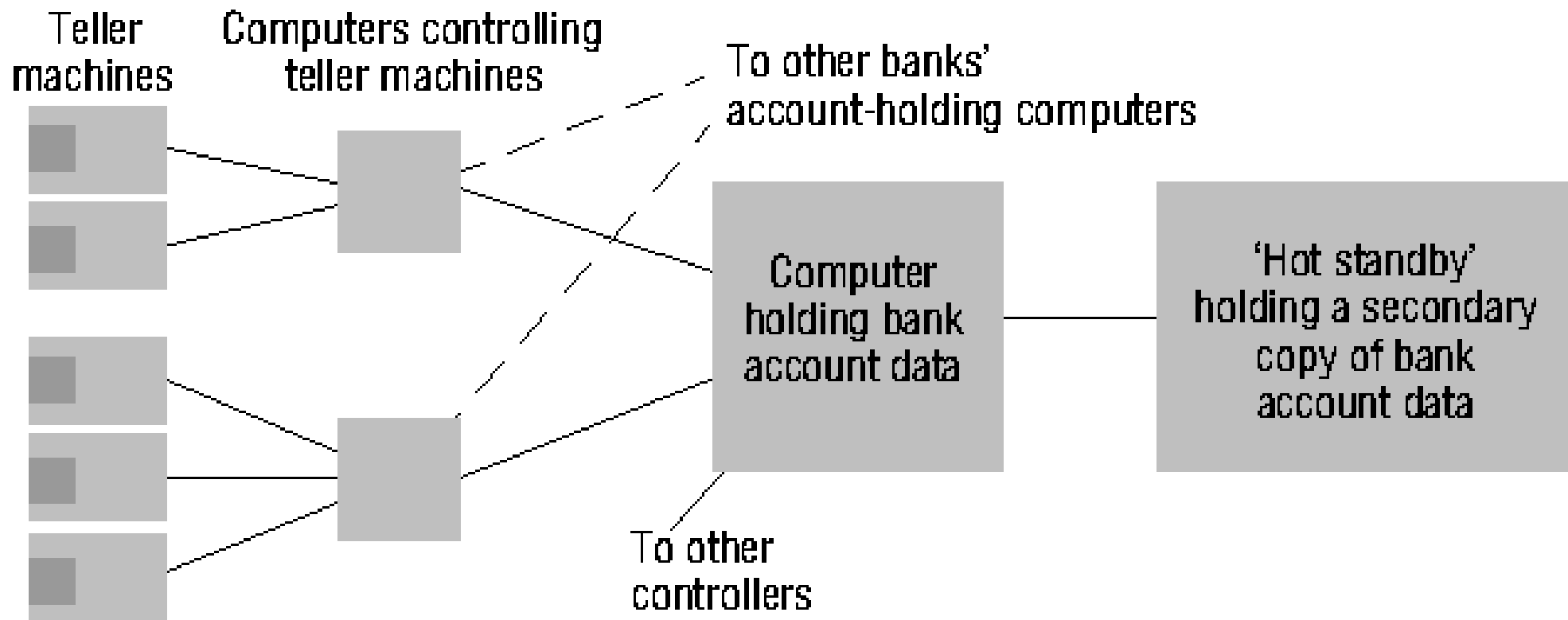
2.1 Local Area Network



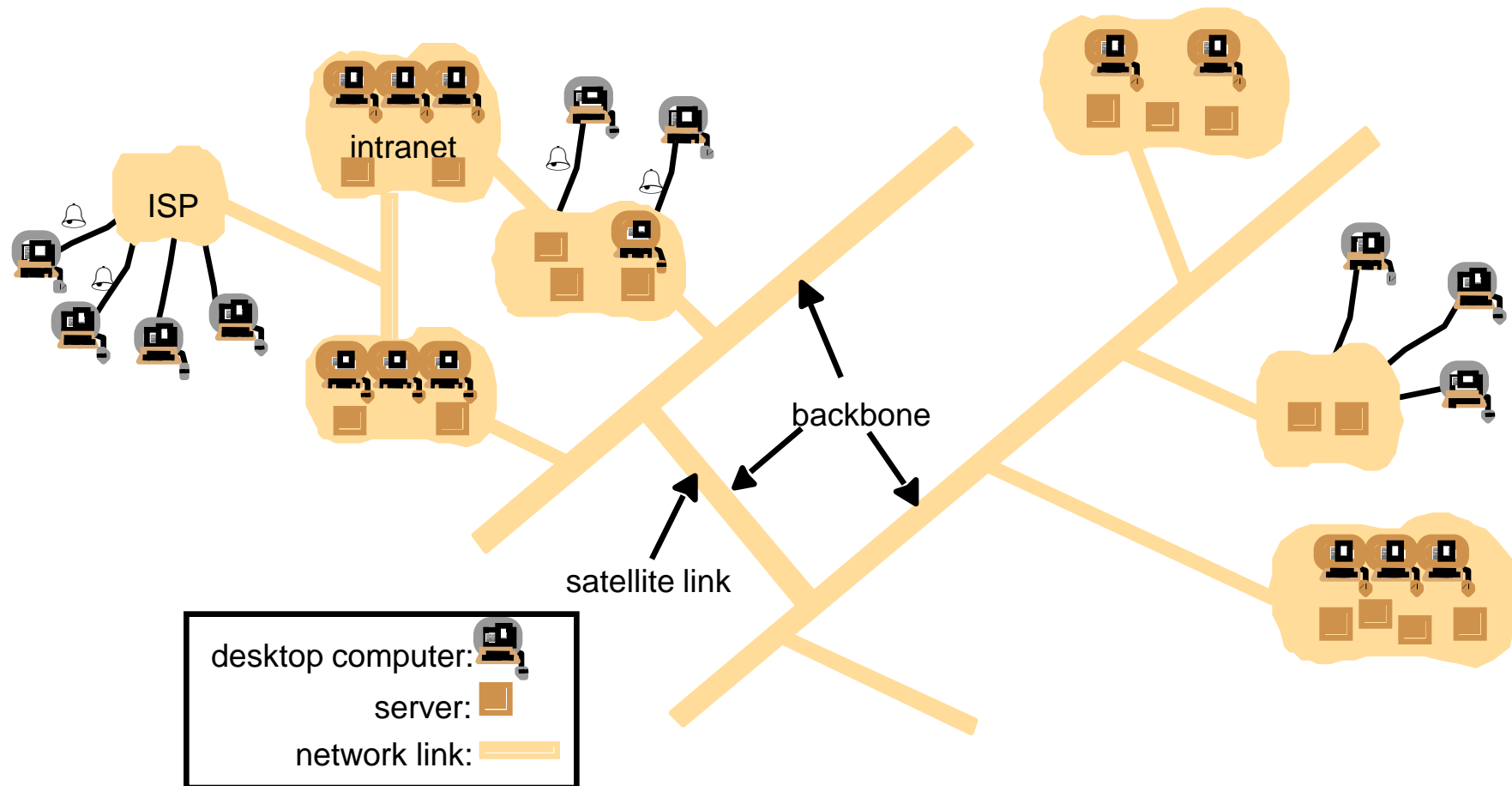
2.2 Database Management System



2.3 Automatic Teller Machine Network



2.4 Internet



2.4.1 World-Wide-Web

Uppsala universitet - Windows Internet Explorer

http://www.uu.se/

File Edit View Favorites Tools Help

Google Search + Bookmarks Check Translate

Windows Live Bing What's New Profile Mail Photos Calendar MSN Share

Uppsala universitet

UPPSALA UNIVERSITET

In English Karta Personal Enhet Lyssna Fritext RSS Sök

Startsida Utbildning Forskning Samverkan Om UU Bibliotek Kontakt

Välkommen till Uppsala universitet!



Rector Anders Hallberg:
"Vid Uppsala universitet arbetar vi målmedvetet och långsiktigt för att alltid kunna erbjuda de bästa förutsättningarna för vår utbildning och forskning." **Välkommen till Uppsala!**

Nyheter

Ännu fler miljoner i anslag till Hans Ellegren
 FN:s Karen AbuZayd årets Dag
 Hammarskjöldföreläsare

Genvägar

- » Alumn
- » Doktorand
- » Fakulteter och enheter
- » Internt för anställda
- » Lediga anställningar
- » Pressinformation
- » Student
- » Stöd Uppsala universitet

Blåsenhus



Blåsenhus - Uppsala universitets nyaste campusområde för nya

Aktuellt

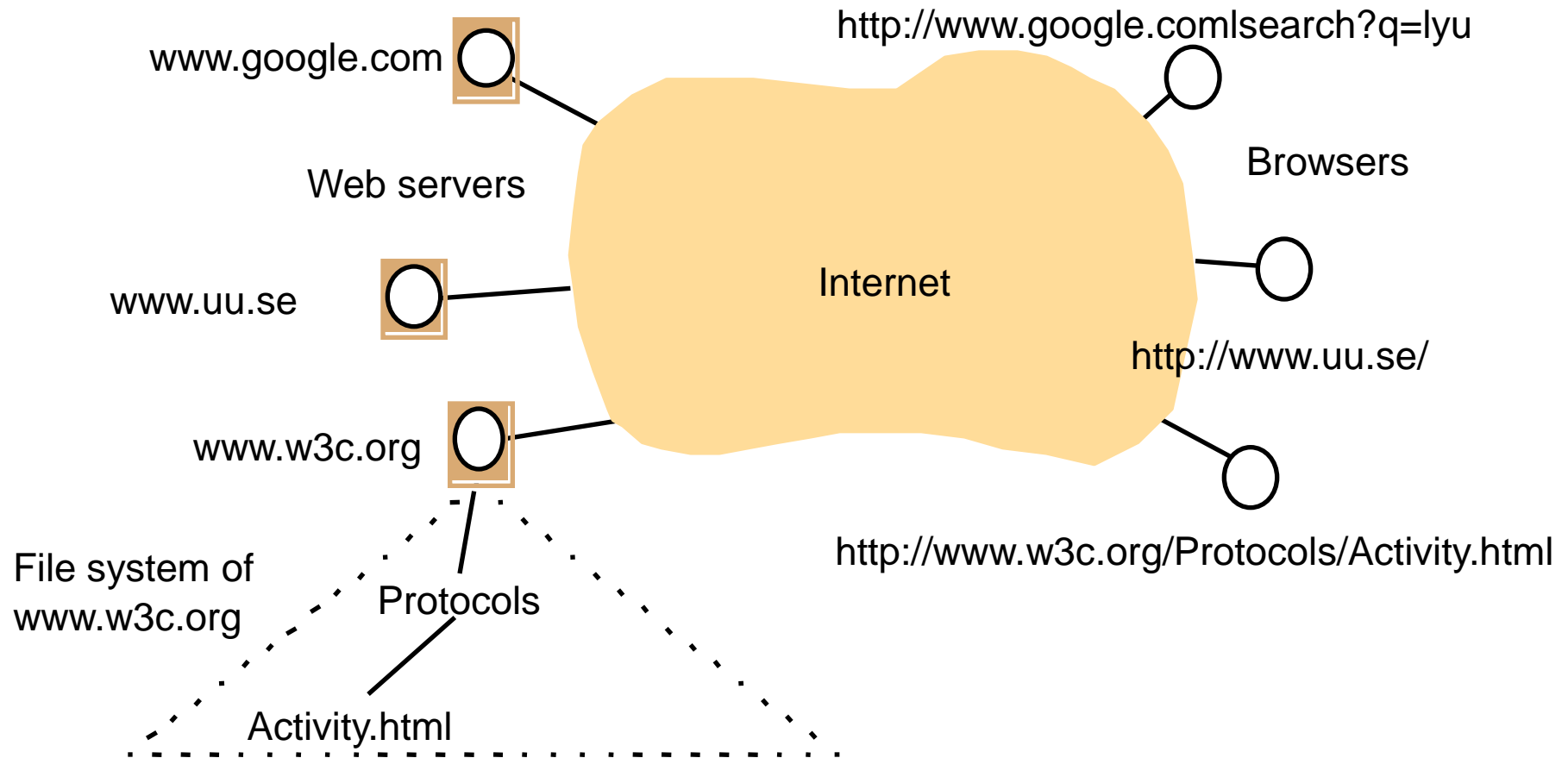
Om Influensa A (H1N1)
 Här kan du vaccinera dig

Expedition Antarktis

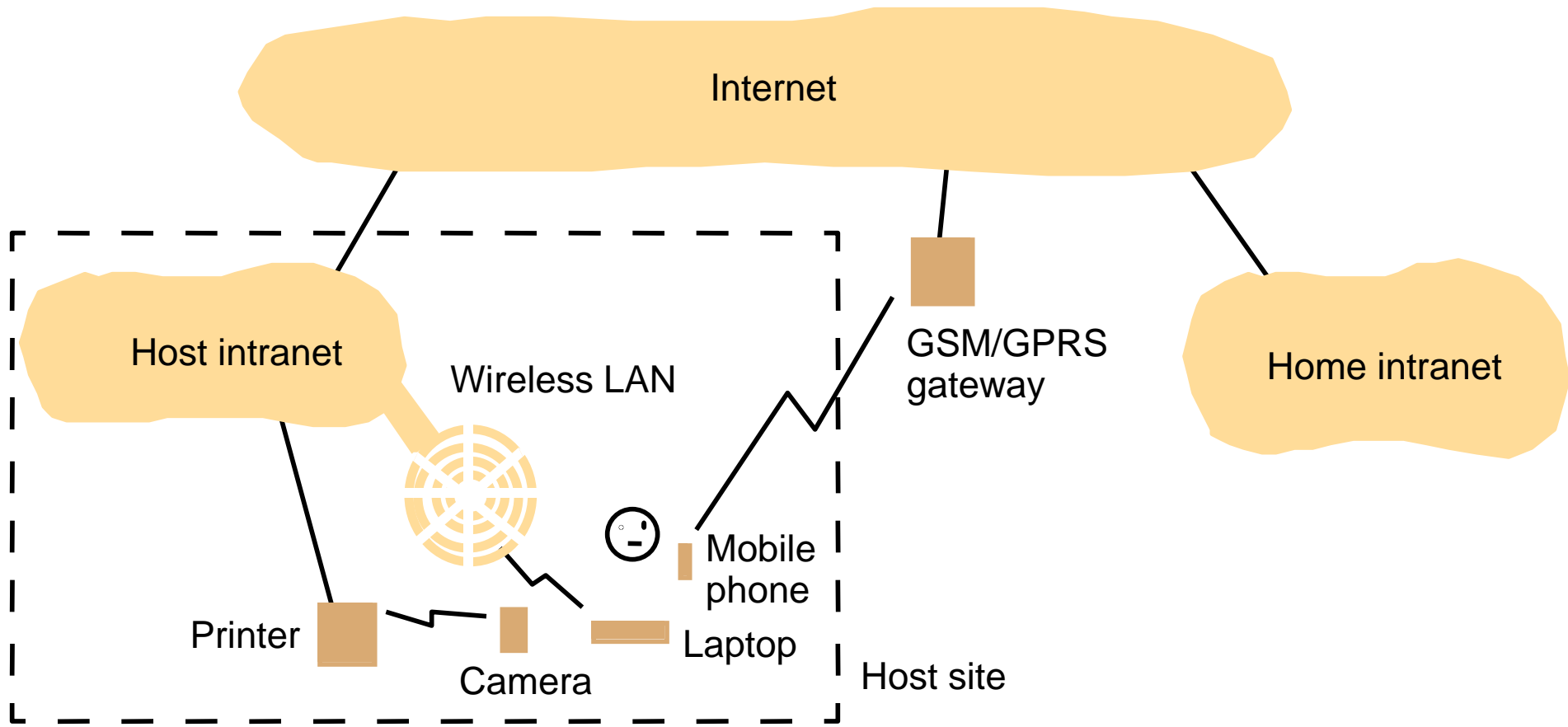


Forskningsexpedition till Antarktis om influensavirus och antibiotikaresistens. Följ forskningsresan på professor Björn Olsens blogg.

2.4.2 Web Servers and Web Browsers



2.5 Mobile and Ubiquitous Computing



3.1 Heterogeneity

- Variety and differences in
 - Networks
 - Computer hardware
 - Operating systems
 - Programming languages
 - Implementations by different developers
- *Middleware* as software layers to provide a programming abstraction as well as masking the heterogeneity of the underlying networks, hardware, OS, and programming languages (e.g., CORBA).
- *Mobile Code* to refer to code that can be sent from one computer to another and run at the destination (e.g., Java applets and Java *virtual machine*).

3.2 Openness

- Openness is concerned with extensions and improvements of distributed systems.
- Detailed interfaces of components need to be published.
- New components have to be integrated with existing components.
- Differences in data representation of interface types on different processors (of different vendors) have to be resolved.

3.3 Security

- In a distributed system, clients send requests to access data managed by servers, resources in the networks:
 - Doctors requesting records from hospitals
 - Users purchase products through electronic commerce
- Security is required for:
 - Concealing the contents of messages: security and privacy
 - Identifying a remote user or other agent correctly (authentication)
- New challenges:
 - Denial of service attack
 - Security of mobile code

3.5 Failure Handling (Fault Tolerance)

- Hardware, software and networks fail!
- Distributed systems must maintain *availability* even at low levels of hardware/software/network *reliability*.
- Fault tolerance is achieved by
 - recovery
 - redundancy

3.6 Concurrency

- Components in distributed systems are executed in concurrent processes.
- Components access and update shared resources (e.g. variables, databases, device drivers).
- Integrity of the system may be violated if concurrent updates are not coordinated.
 - Lost updates
 - Inconsistent analysis

4. Basic Design Issues

- General software engineering principles include rigor and formality, separation of concerns, modularity, abstraction, anticipation of change, ...
- Specific issues for distributed systems:
 - Communication
 - Software structure
 - System architecture
 - Workload allocation
 - Consistency maintenance

4.2 Communication

- Separated components communicate with sending processes and receiving processes for *data transfer* and *synchronization*.
- Message passing: *send* and *receive* primitives
 - synchronous or blocking
 - asynchronous or non-blocking
 - Abstractions defined: channels, sockets, ports.
- Communication patterns: client-server communication (e.g., RPC, function shipping) and group multicast

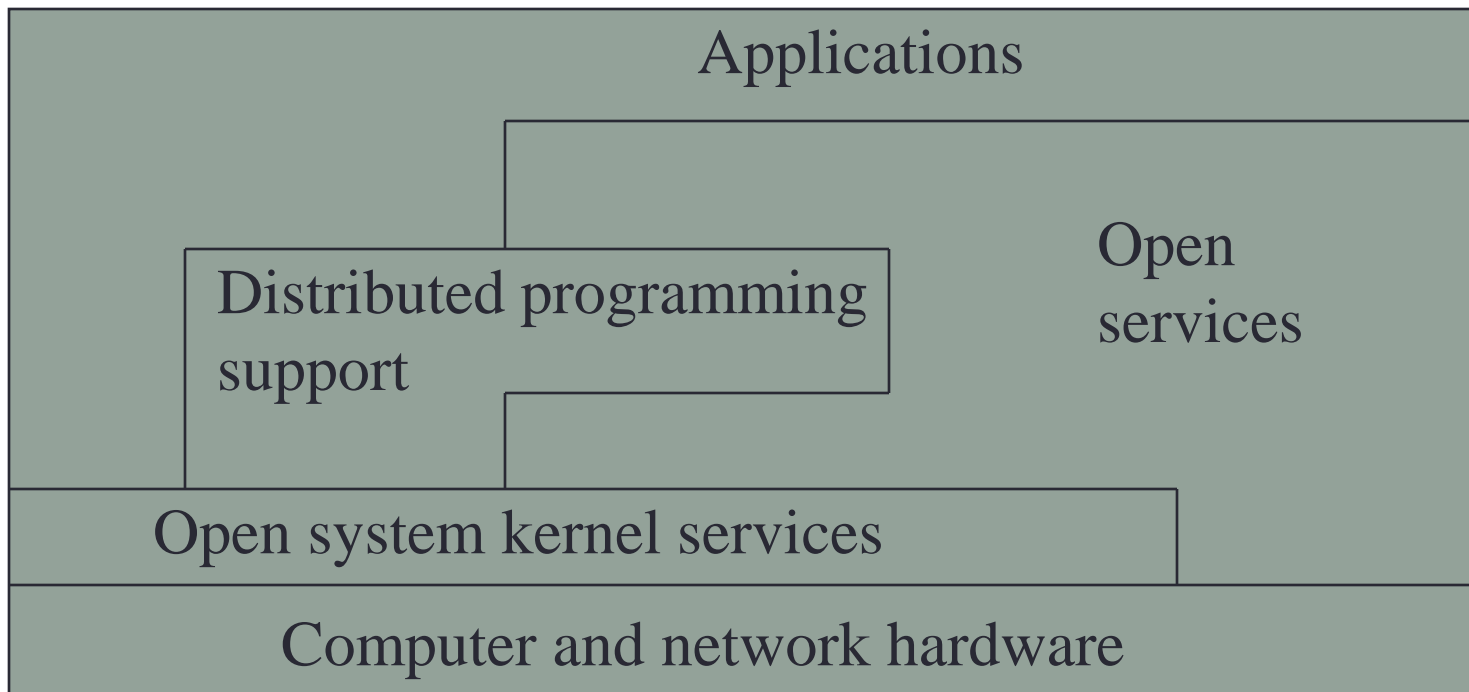
4.3 Software Structure

- Layers in centralized computer systems:

Applications
Middleware
Operating system
Computer and Network Hardware

4.3 Software Structure

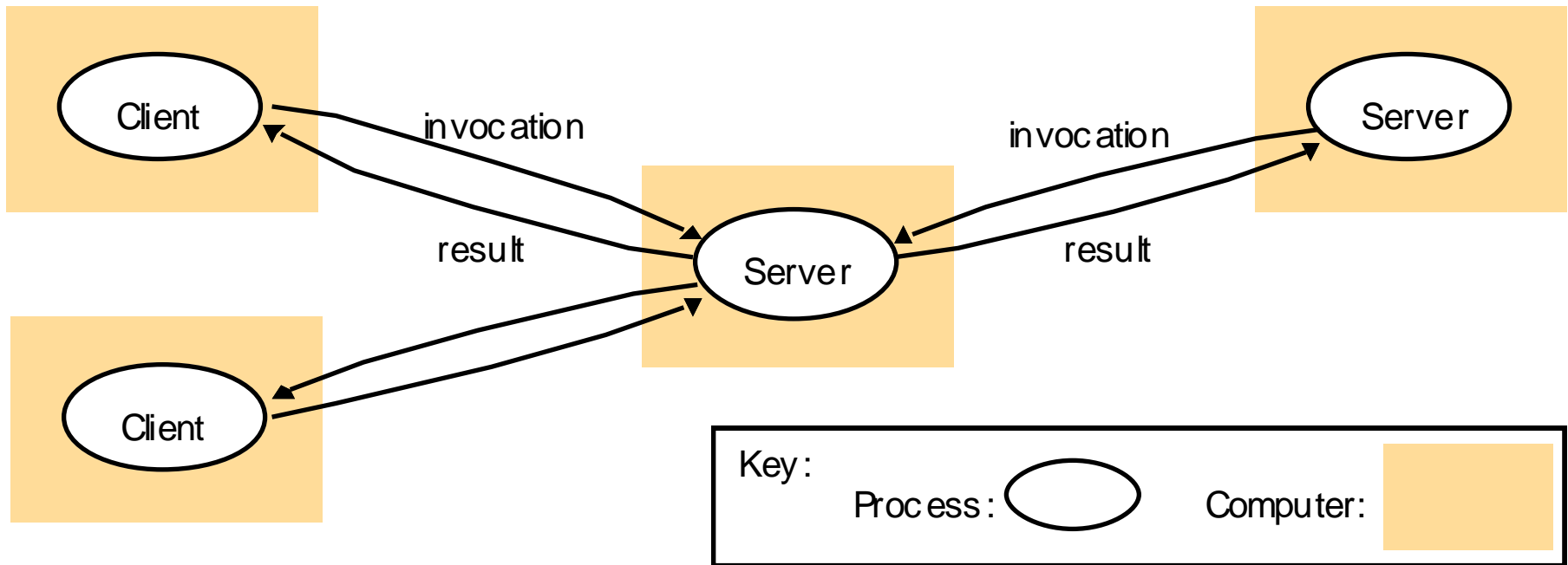
- Layers and dependencies in distributed systems:



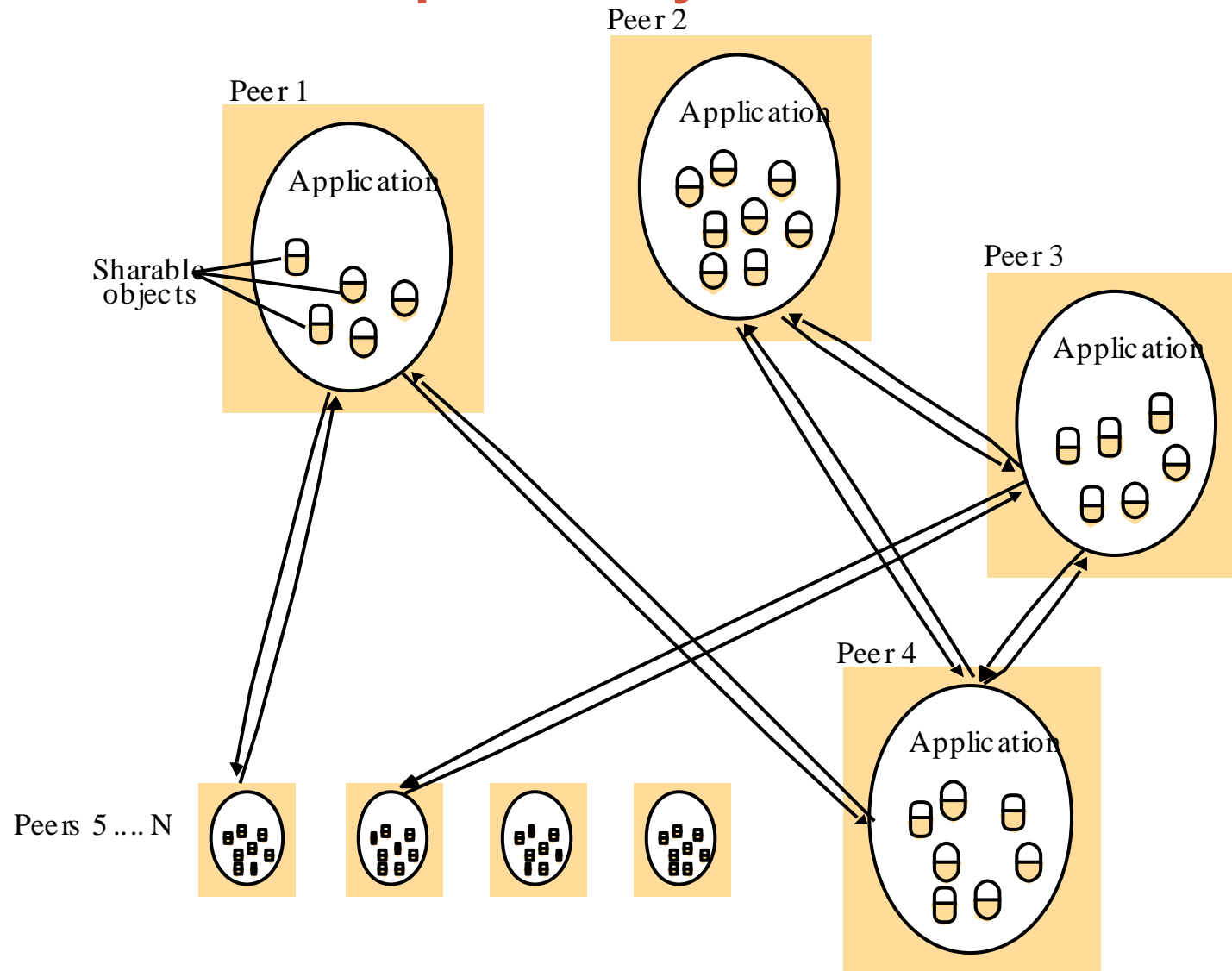
4.4 SYSTEM ARCHITECTURES

- Client-Server
- Peer-to-Peer
- Services provided by multiple servers
- Proxy servers and caches
- Mobile code and mobile agents
- Network computers
- Thin clients and mobile devices

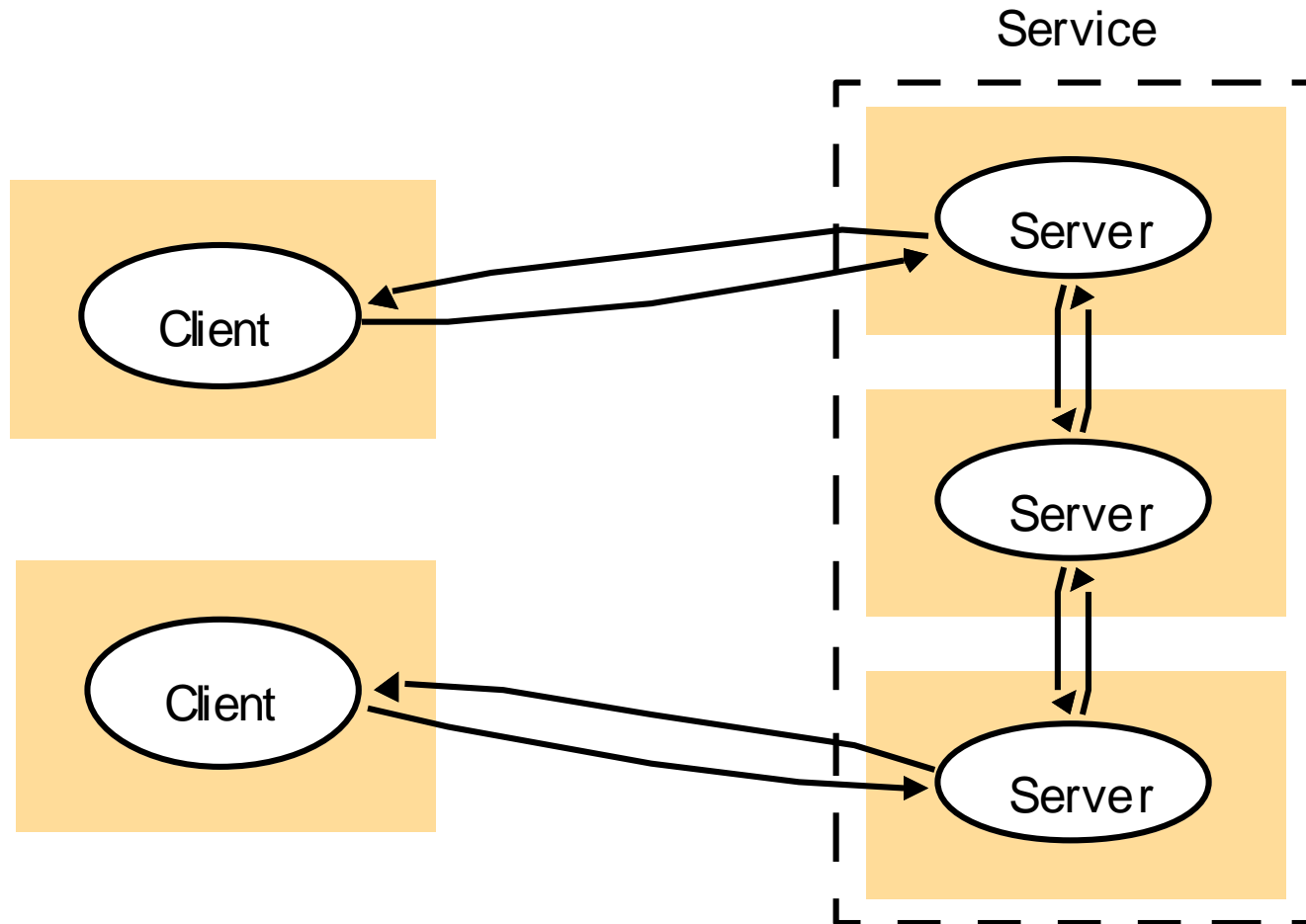
4.4.1 Clients Invoke Individual Servers



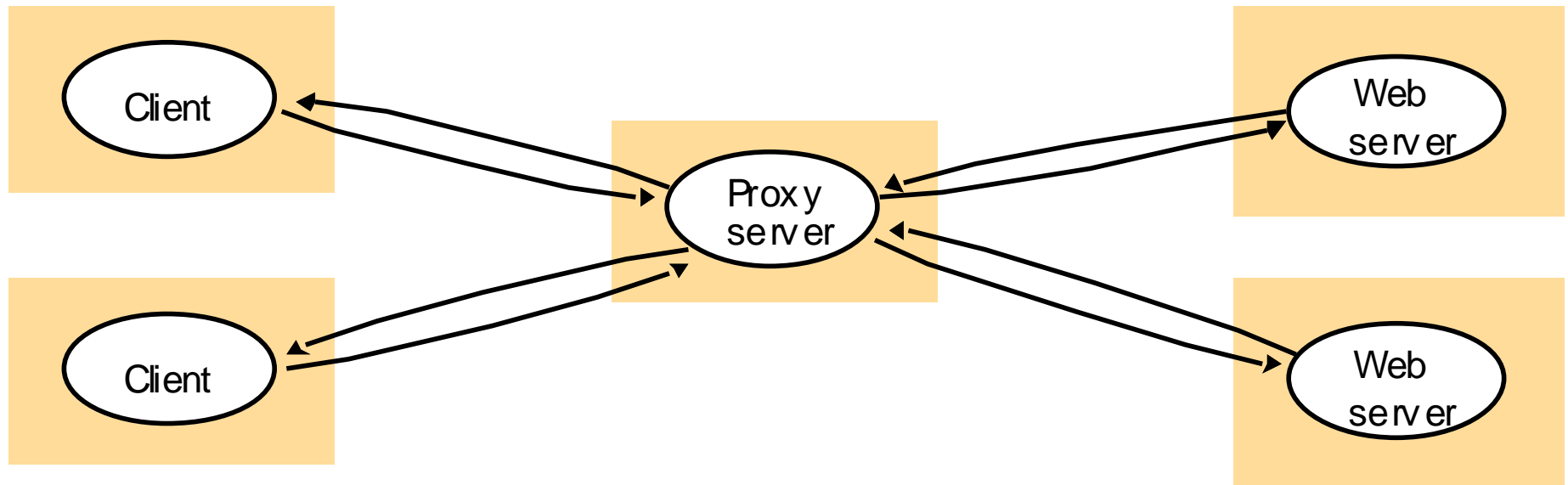
4.4.2 Peer-to-peer Systems



4.4.3 A Service by Multiple Servers

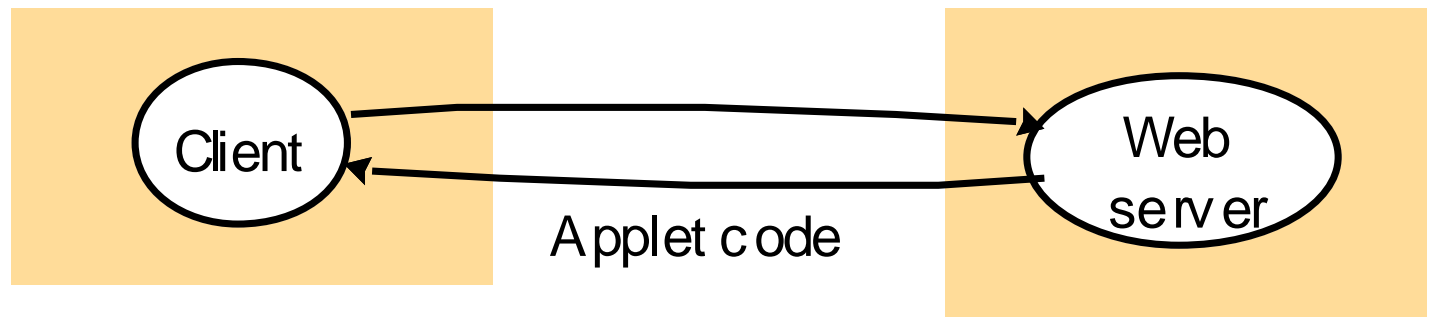


4.4.4 Web Proxy Server



4.4.5 Web Applets

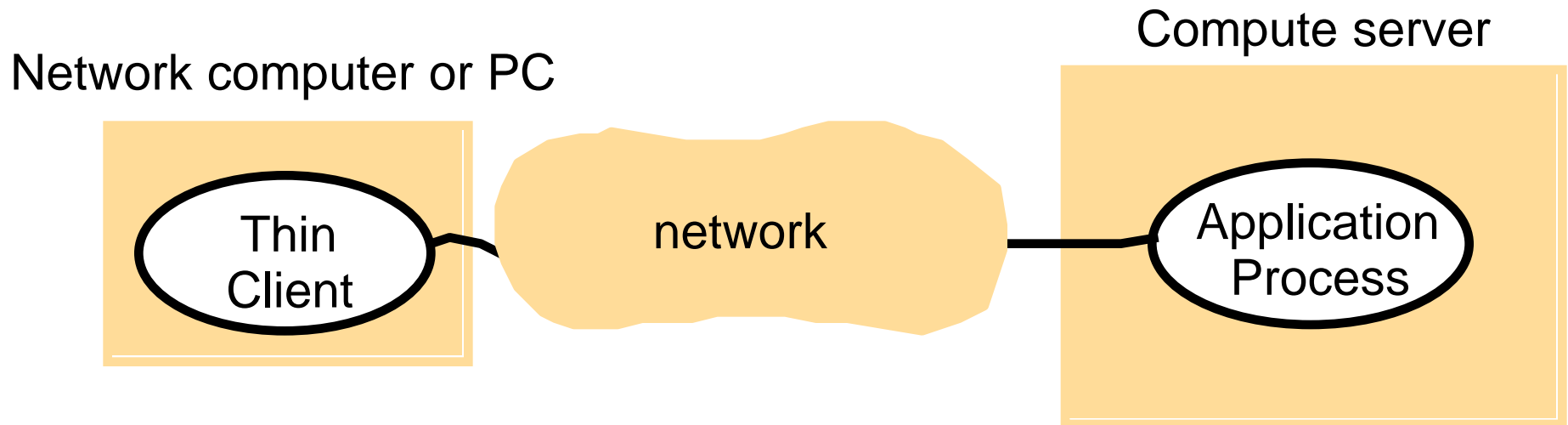
a) client request results in the downloading of applet code



b) client interacts with the applet



4.4.6 Thin Clients and Compute Servers



5. Summary

- Definitions of distributed systems and comparisons to centralized systems.
- The characteristics of distributed systems.
- The eight forms of transparency.
- The basic design issues.
- Read Chapter 1 and Chapter 2 of the textbook.