

Sistemas Distribuidos

- CI – 2205
- III - 2011
- Lunes y jueves, 5:00 pm a 9:00 pm
- Aula 205
- Profesor: Diego Villalba

Sistemas Distribuidos

- Sistemas distribuidos

Sistemas Distribuidos

- Sistemas distribuidos
- Procesos distribuidos

Sistemas Distribuidos

- Sistemas distribuidos
- Procesos distribuidos
- Computación distribuida

Sistemas Distribuidos

- Sistemas distribuidos
- Procesos distribuidos
- Computación distribuida
- ...

Sistemas Distribuidos

- Sistemas distribuidos
- Procesos distribuidos
- Computación distribuida
- **Middleware**

Algunas definiciones

- Un sistema distribuido es una colección de entidades independientes que cooperan para resolver un problema que no pueden resolver individualmente.
- Un sistema distribuido es un sistema en que componentes localizados en computadoras en red se comunican y coordinan sus acciones por medio del intercambio de mensajes (cdk5.net).
- “Uno se da cuenta que está usando un sistema distribuido cuando se cae un servidor del que nunca habías oído antes y no puedes hacer tu trabajo”.

Más definiciones

- Una colección de computadoras que no comparten una memoria común o un reloj físico, que se comunican pasándose mensajes por una red de comunicaciones y donde cada computadora tiene su propia memoria y su propio sistema operativo. Típicamente, las computadoras son semiautónomas y están débilmente acopladas, al mismo tiempo que cooperan para enfrentar un problema colectivamente.

Más definiciones

- Una colección que computadoras independientes que se presentan al usuario como un único sistema coherente.
- Este término describe una gama amplia de computadoras, desde sistemas débilmente acoplados como redes de área amplia, pasando por sistemas fuertemente acoplados como redes de área local, hasta sistemas muy fuertemente acoplados como sistemas multiprocesadores.

Nuestra definición

- Un sistema distribuido es un conjunto de componentes (hardware y software), localizados en computadoras independientes, que se presenta a sus usuarios como un solo sistema coherente [1, 2].

Aspectos fundamentales

- Dos aspectos fundamentales sobresalen de esta definición:
 - Un sistema distribuido consiste de componentes (i.e., computadoras o aplicaciones) que son autónomos.
 - Sus usuarios (sean éstos personas o programas) piensan que están interactuando con un único sistema, es decir, el aspecto distribuido del sistema es transparente para el usuario.

Algunos algoritmos fundamentales

- Exclusión mutua
- Detección de bloqueos
- Detección de fallas
- Recuperación de fallas
- Autenticación
- Seguridad
- Estado global y tiempo global
- Esta lista no es exhaustiva ni está ordenada por importancia

Principales motivaciones

- Compartir recursos
- Escalabilidad
- SaaS
- Recursos computacionales como un servicio básico
- ...

Principales desafíos

- Heterogeneidad
- Apertura (extensibilidad)
- Seguridad
- Escalabilidad
- Manejo de fallas
- Concurrencia
- Transparencia
- Calidad de servicio

Algunos dominios de aplicación

Finance and commerce

eCommerce e.g. Amazon and eBay, PayPal, online banking and trading

The information society

Web information and search engines, ebooks, Wikipedia; social networking: Facebook and MySpace.

Creative industries and entertainment

online gaming, music and film in the home, user-generated content, e.g. YouTube, Flickr

Healthcare

health informatics, on online patient records, monitoring patients

Education

e-learning, virtual learning environments; distance learning

Transport and logistics

GPS in route finding systems, map services: Google Maps, Google Earth

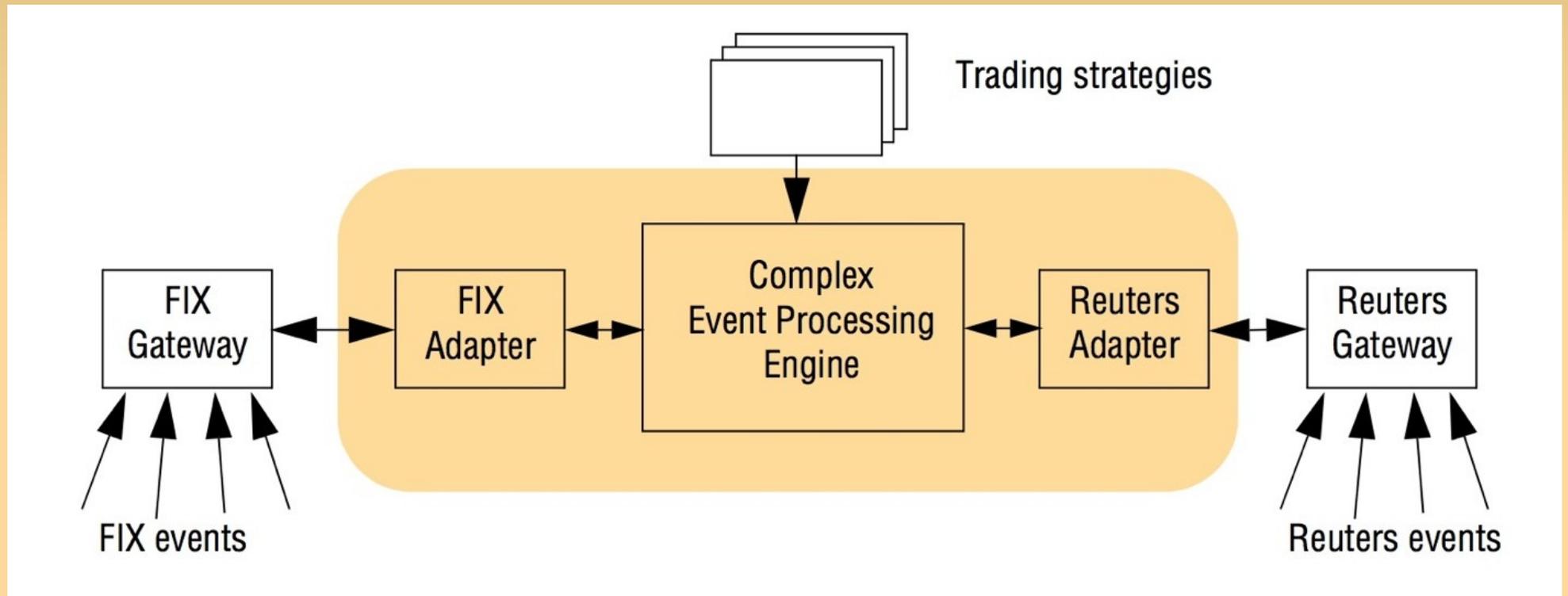
Science

The Grid as an enabling technology for collaboration between scientists

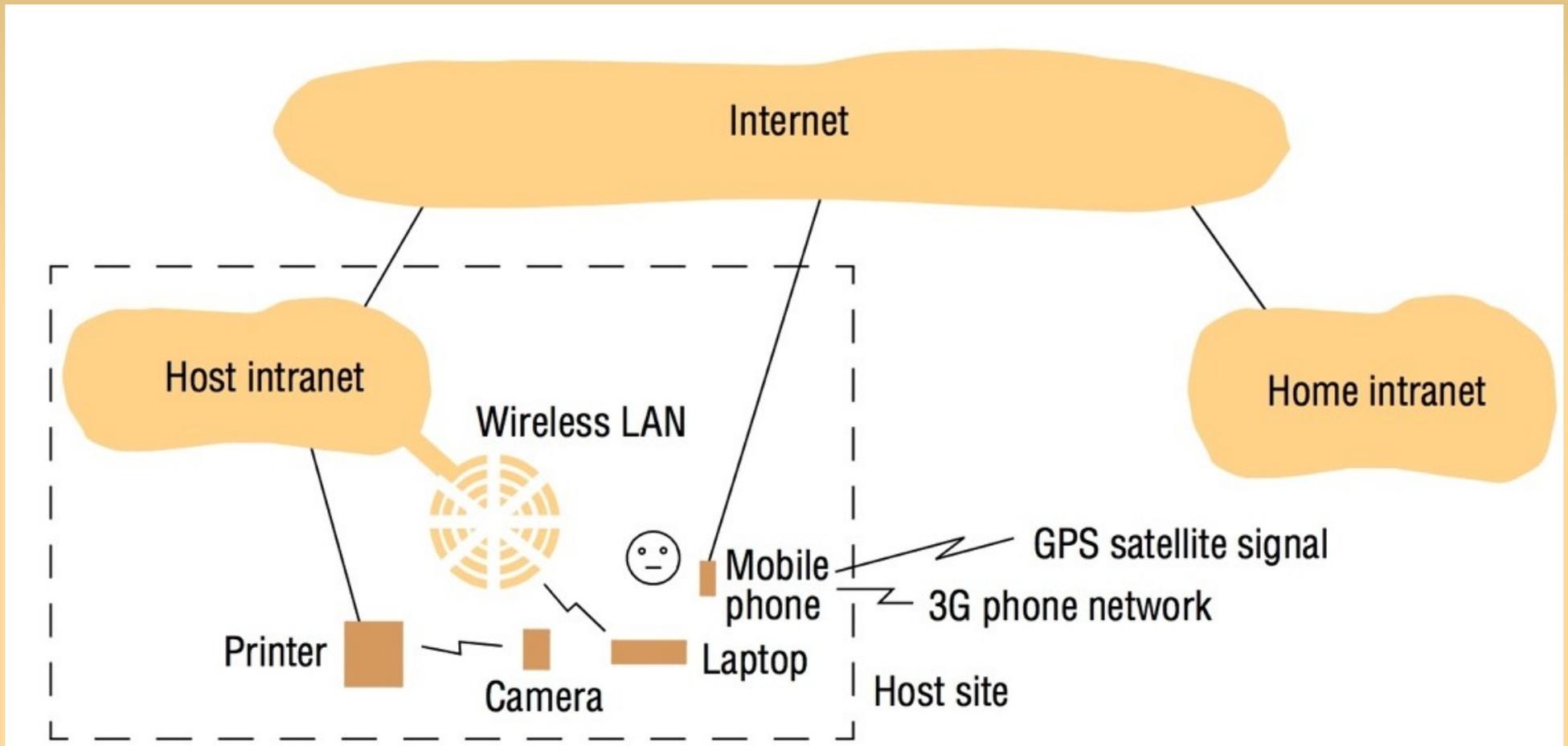
Environmental management

sensor technology to monitor earthquakes, floods or tsunamis

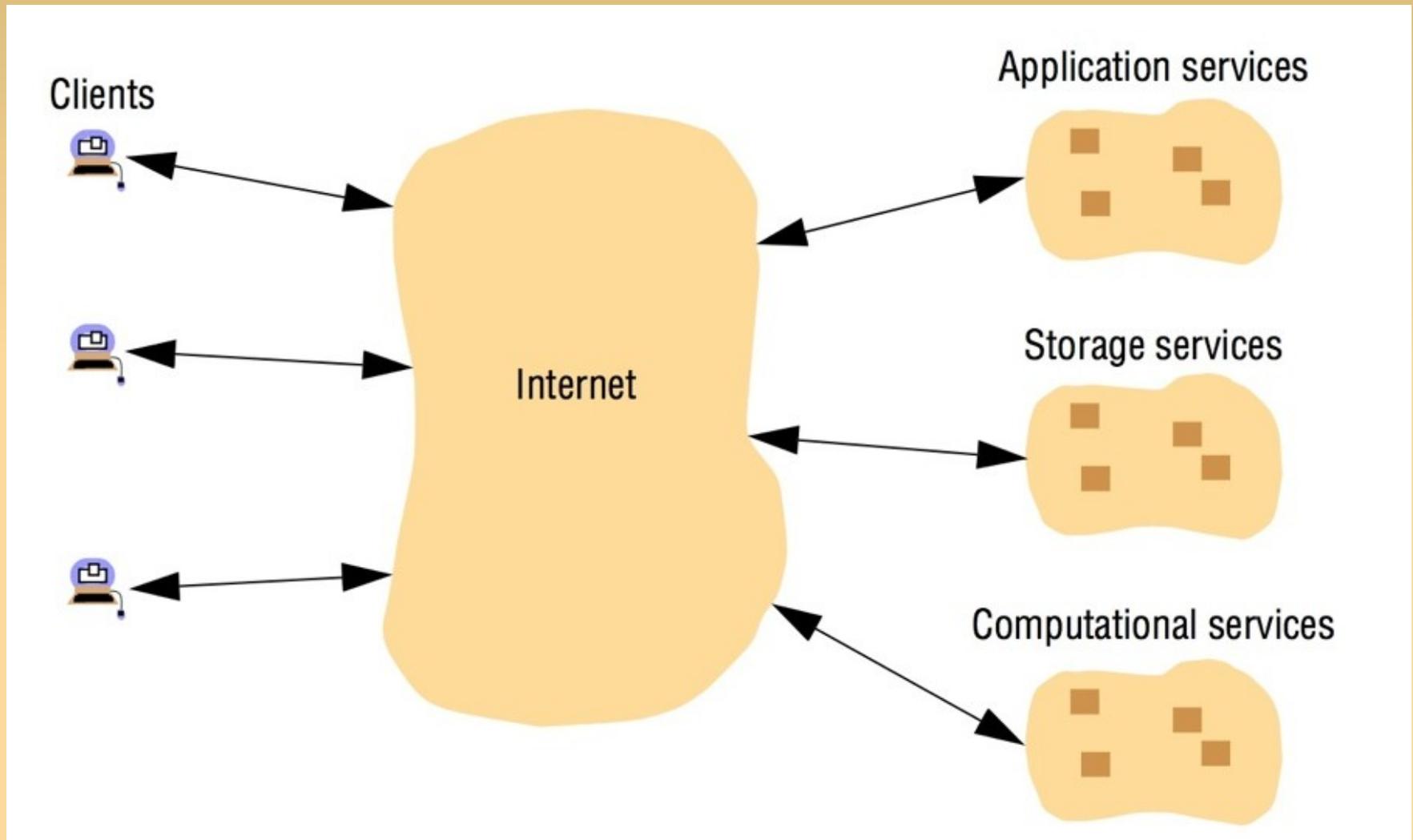
Ejemplo de sistema financiero



Aparatos portátiles en un SD



Cloud computing



Crecimiento de Internet

| <i>Date</i> | <i>Computers</i> | <i>Web servers</i> | <i>Percentage</i> |
|-------------|------------------|--------------------|-------------------|
| 1993, July | 1,776,000 | 130 | 0.008 |
| 1995, July | 6,642,000 | 23,500 | 0.4 |
| 1997, July | 19,540,000 | 1,203,096 | 6 |
| 1999, July | 56,218,000 | 6,598,697 | 12 |
| 2001, July | 125,888,197 | 31,299,592 | 25 |
| 2003, July | ~200,000,000 | 42,298,371 | 21 |
| 2005, July | 353,284,187 | 67,571,581 | 19 |

Transparencias

- Transparencia de acceso (local o remoto)
- Transparencia de ubicación (eg IPs)
- Transparencia de concurrencia: varios procesos operan concurrentemente utilizando recursos compartidos sin interferir entre ellos
- Transparencia de replicación: muchas instancias de recursos pueden ser utilizadas para incrementar confiabilidad o rendimiento sin que los usuarios sepan (eg cachés)

Transparencias

- Transparencia de fallas: permite esconder fallas, permitiendo a los usuarios completar tareas
- Transparencia de movilidad: permite mover recursos o clientes sin afectar operación
- Transparencia de rendimiento: permite reconfigurar el sistema para mejorar rendimiento cuando la carga de trabajo varía
- Transparencia de escalabilidad: permite expanderse sin cambiar estructura o algoritmos

Generaciones

| <i>Distributed systems:</i> | <i>Early</i> | <i>Internet-scale</i> | <i>Contemporary</i> |
|-----------------------------|--|---|--|
| <i>Scale</i> | Small | Large | Ultra-large |
| <i>Heterogeneity</i> | Limited (typically relatively homogenous configurations) | Significant in terms of platforms, languages and middleware | Added dimensions introduced including radically different styles of architecture |
| <i>Openness</i> | Not a priority | Significant priority with range of standards introduced | Major research challenge with existing standards not yet able to embrace complex systems |
| <i>Quality of service</i> | In its infancy | Significant priority with range of services introduced | Major research challenge with existing services not yet able to embrace complex systems |

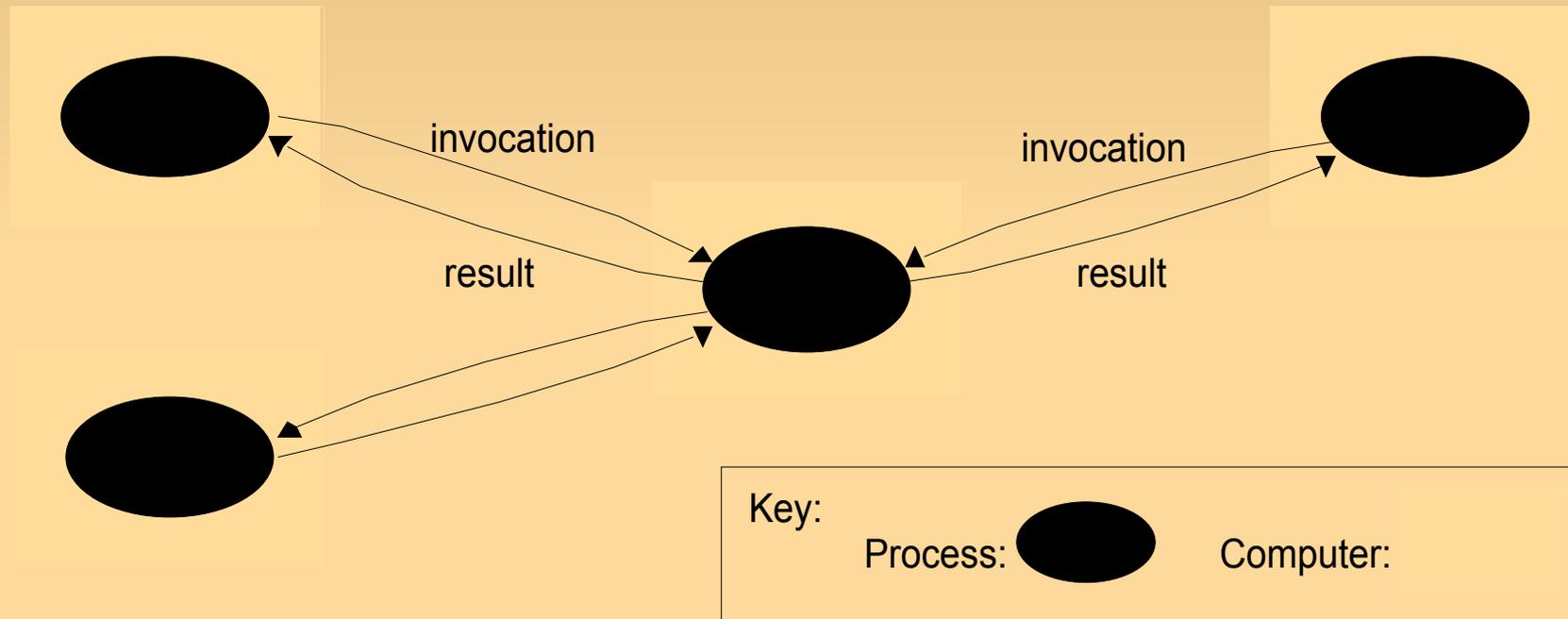
Elementos arquitectónicos

- Cuatro preguntas fundamentales:
 - ¿Qué entidades se están comunicando?
 - ¿Cómo se comunican? ¿Qué paradigma de comunicación utilizan?
 - ¿Qué roles y responsabilidades tienen en la arquitectura global?
 - ¿Cómo son mapeados a la infraestructura física distribuida?

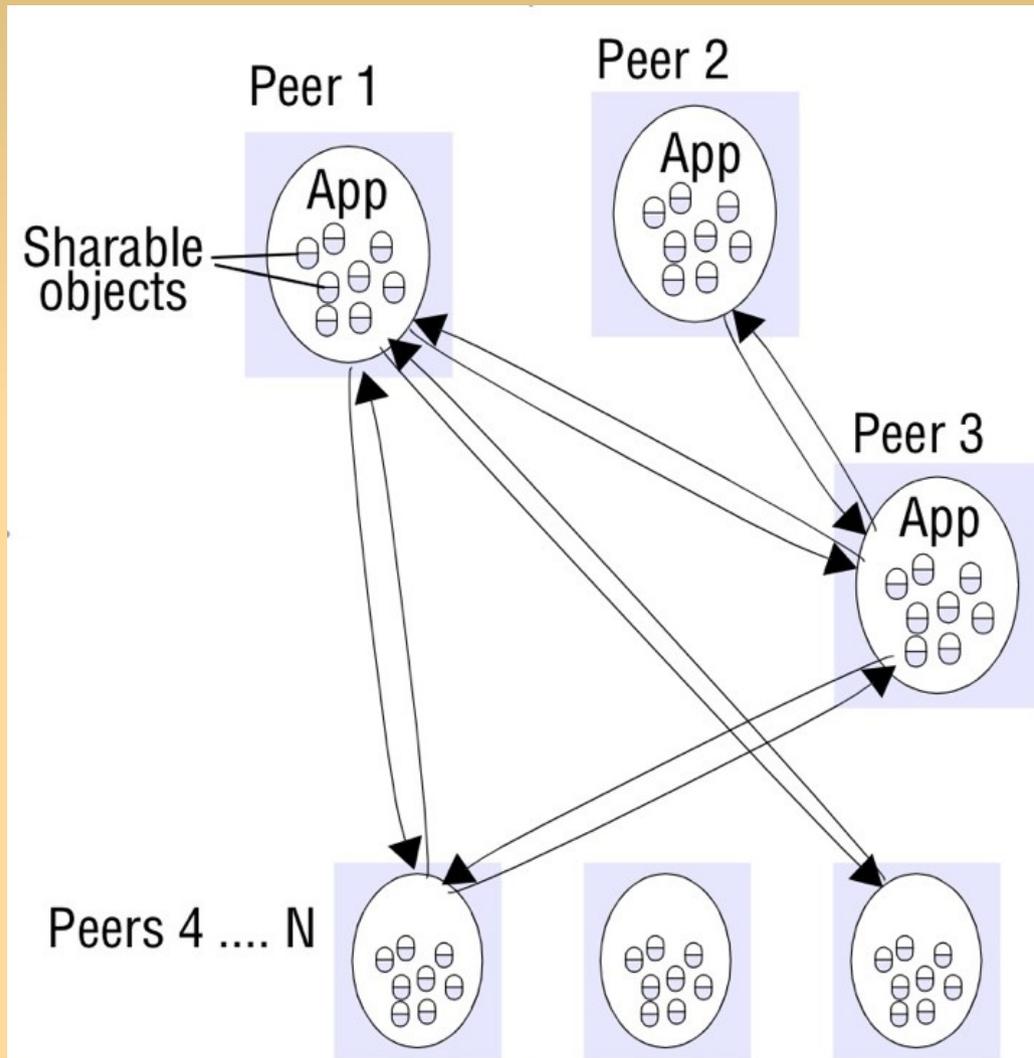
Elementos arquitectónicos

| <i>Communicating entities (what is communicating)</i> | | <i>Communication paradigms (how they communicate)</i> | | |
|---|----------------------------------|---|--------------------------|-------------------------------|
| <i>System-oriented entities</i> | <i>Problem-oriented entities</i> | <i>Interprocess communication</i> | <i>Remote invocation</i> | <i>Indirect communication</i> |
| Nodes | Objects | Message passing | Request-reply | Group communication |
| Processes | Components | Sockets | RPC | Publish-subscribe |
| | Web services | Multicast | RMI | Message queues |
| | | | | Tuple spaces |
| | | | | DSM |

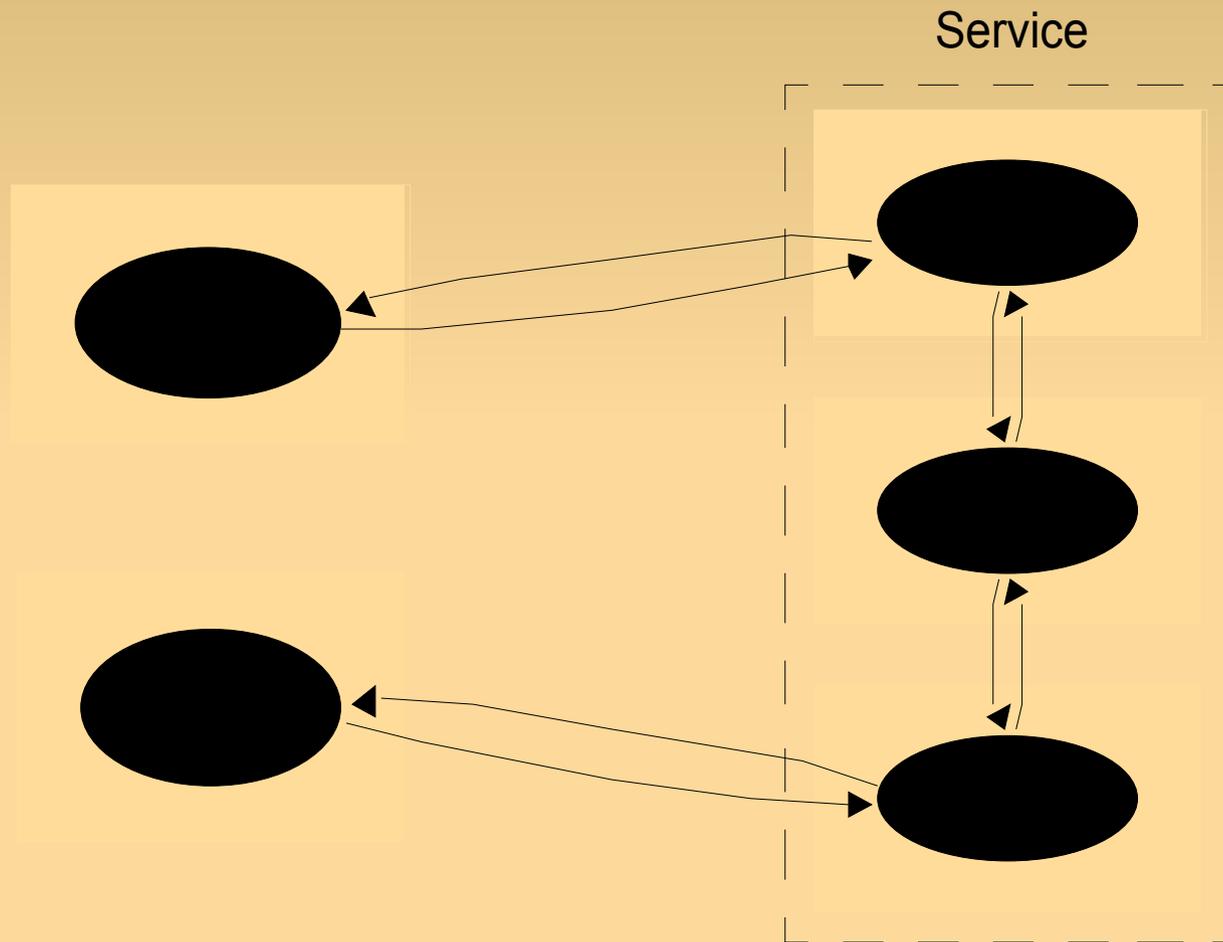
Cientes invocan servidores



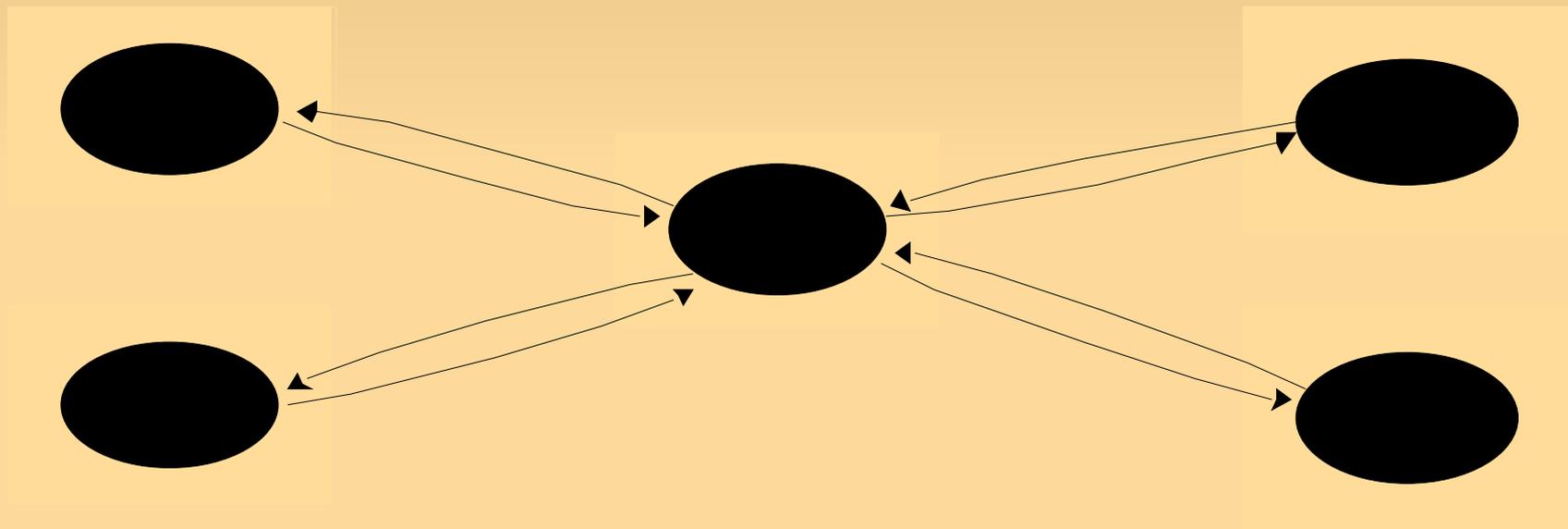
Arquitecturas Peer to Peer



Servicio provisto por n servidores

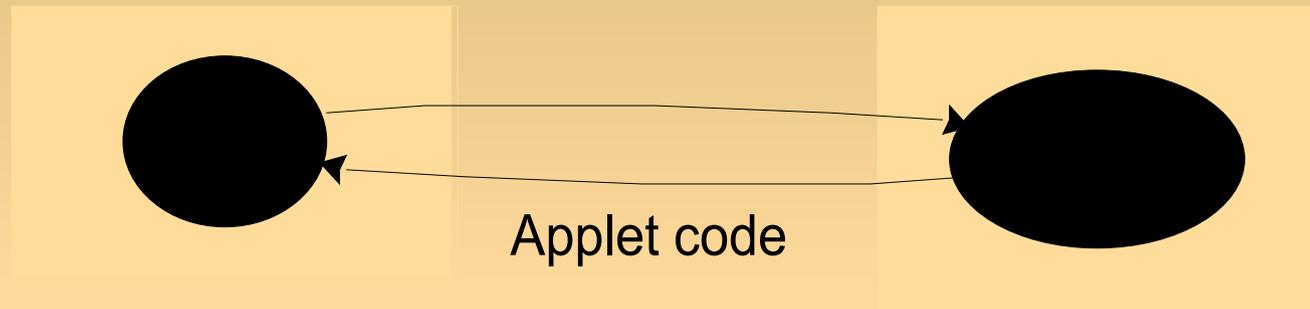


Un servidor Proxy



Applets

a) client request results in the downloading of applet code



b) client interacts with the applet



Capas de HW y SW

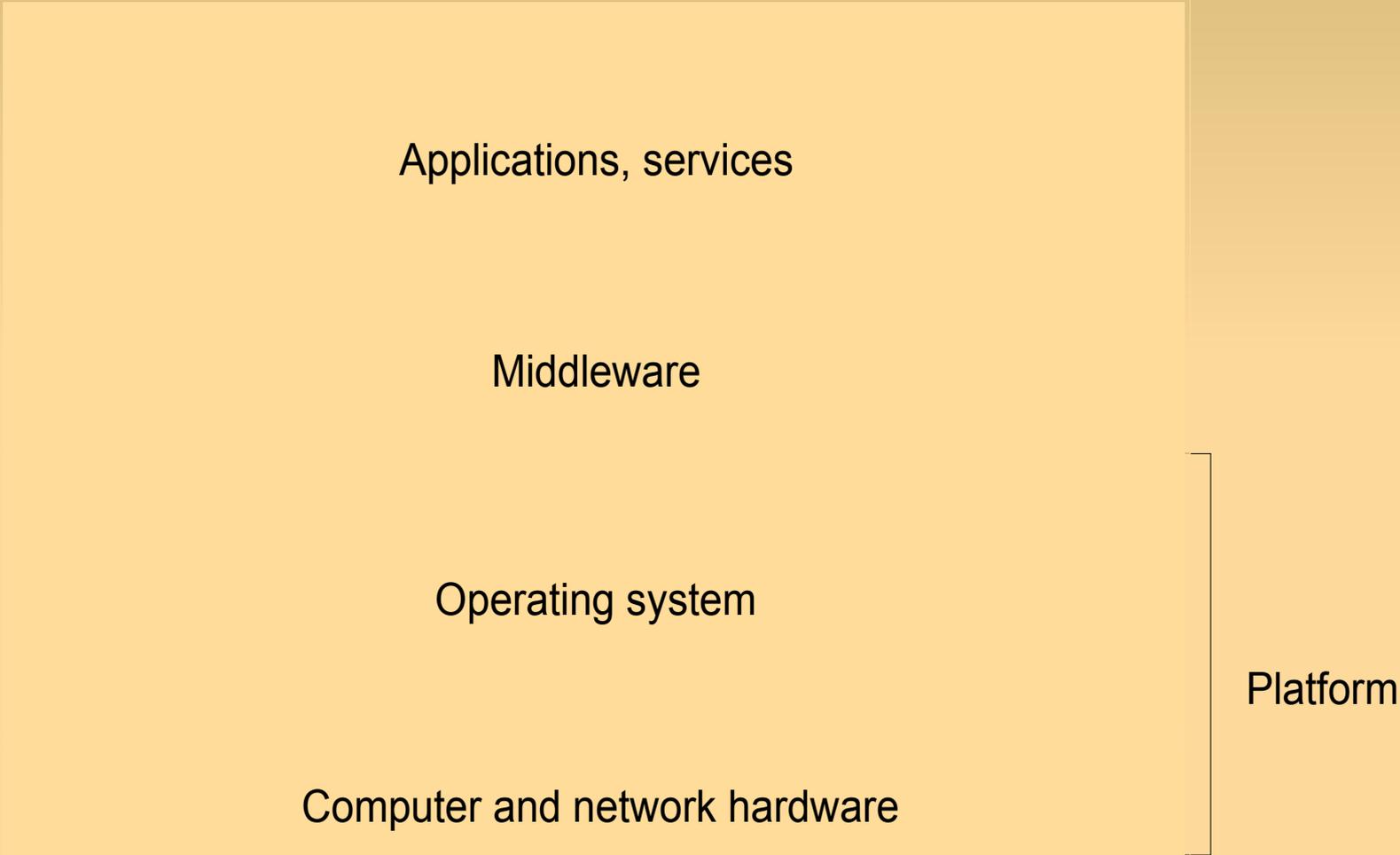
Applications, services

Middleware

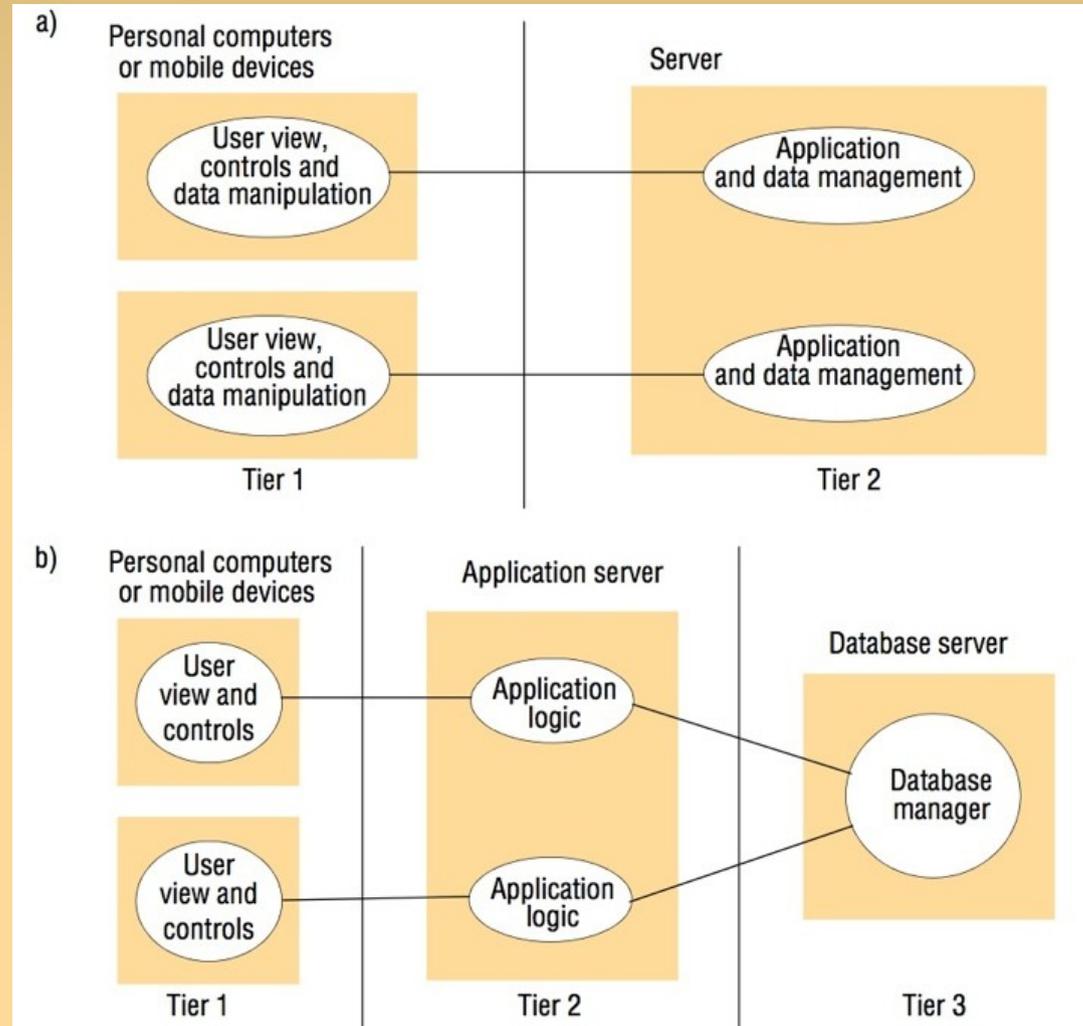
Operating system

Computer and network hardware

Platform



Arquitecturas de 2 y 3 capas



Ajax

```
new Ajax.Request('scores.php?
                 game=Arsenal:Liverpool',
                 {onSuccess: updateScore});
```

```
function updateScore(request) {
```

```
.....
```

(request contains the state of the Ajax request including the returned result.

The result is parsed to obtain some text giving the score, which is used to update the relevant portion of the current page.)

```
.....
```

```
}
```

Thin clients

Network computer or PC

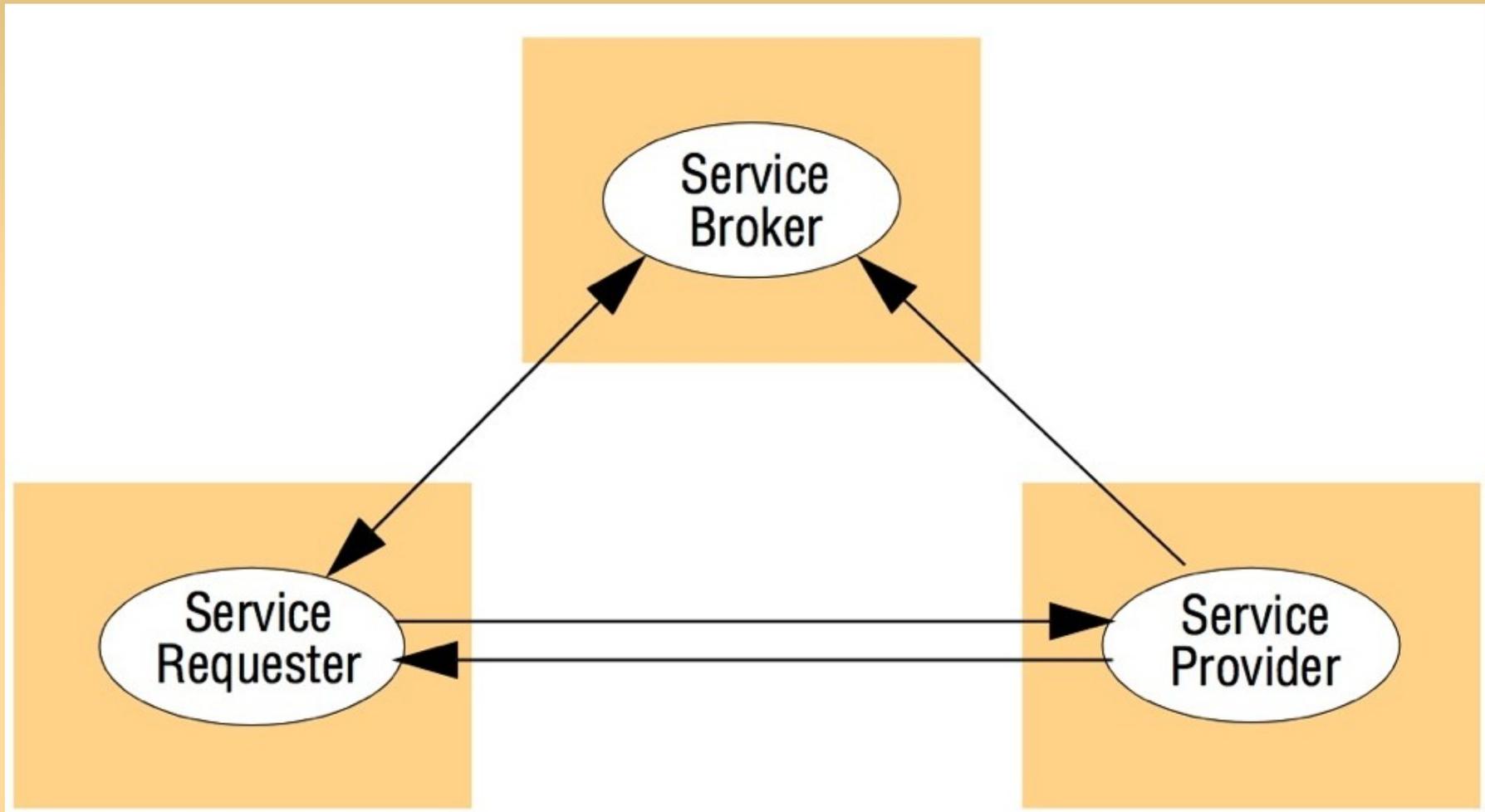


network

Compute server



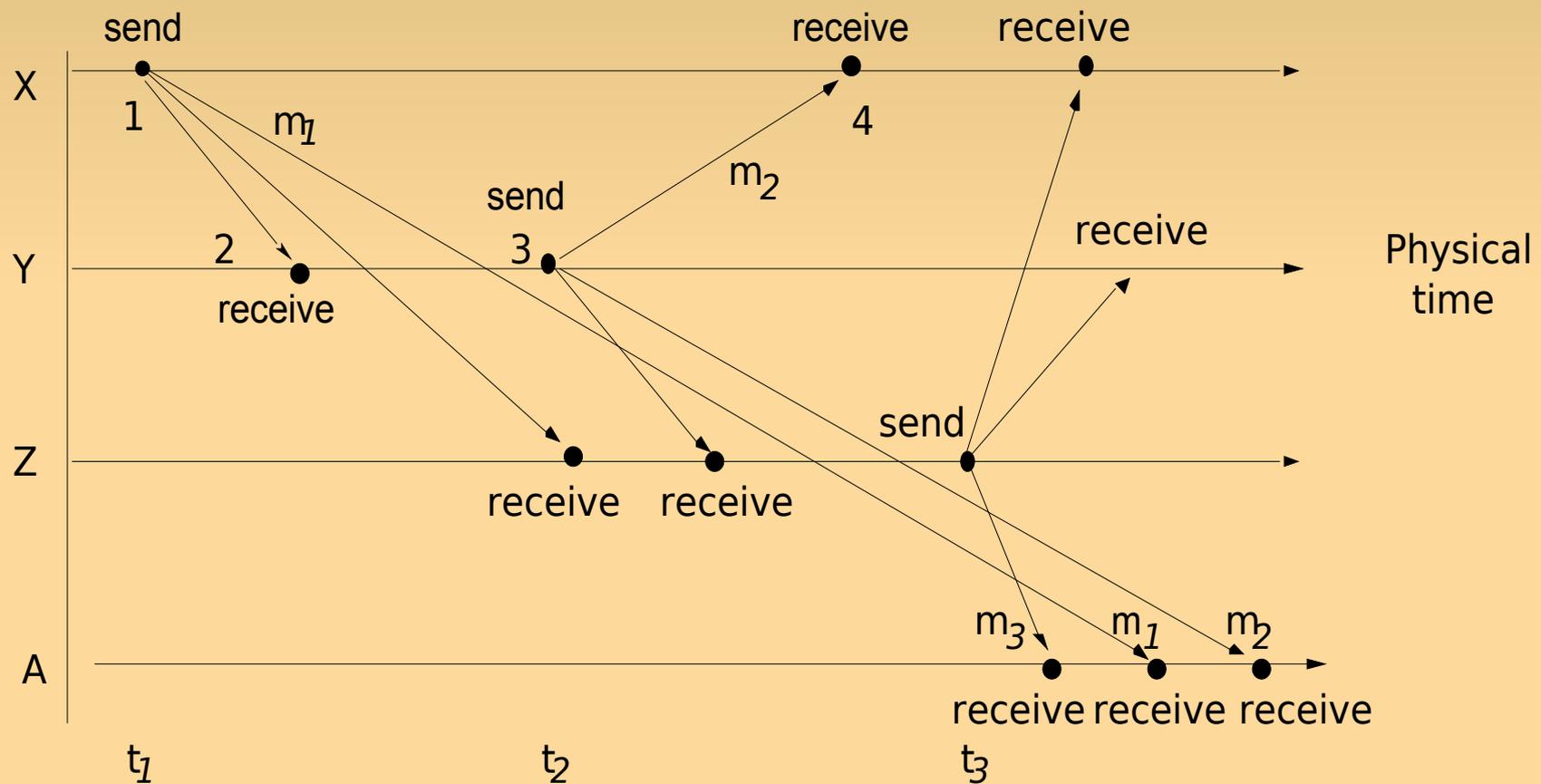
Web services



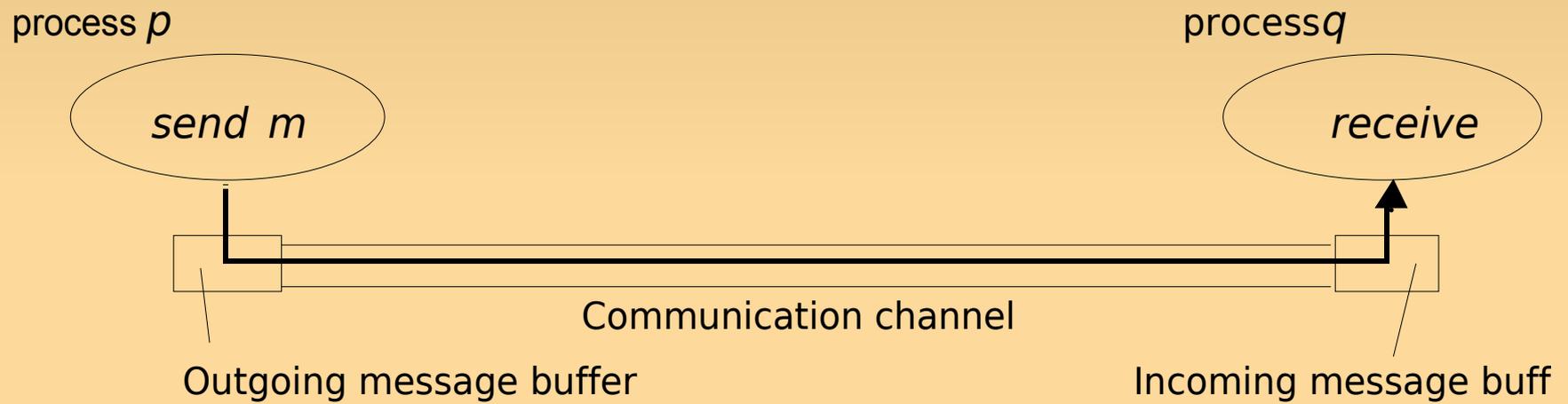
Middleware

| <i>Major categories:</i> | <i>Subcategory</i> | <i>Example systems</i> |
|--|------------------------|------------------------|
| <i>Distributed objects (Chapters 5, 8)</i> | Standard | RM-ODP |
| | Platform | CORBA |
| | Platform | Java RMI |
| <i>Distributed components (Chapter 8)</i> | Lightweight components | Fractal |
| | Lightweight components | OpenCOM |
| | Application servers | SUN EJB |
| | Application servers | CORBA Component Model |
| | Application servers | JBoss |
| <i>Publish-subscribe systems (Chapter 6)</i> | - | CORBA Event Service |
| | - | Scribe |
| | - | JMS |
| <i>Message queues (Chapter 6)</i> | - | Websphere MQ |
| | - | JMS |
| <i>Web services (Chapter 9)</i> | Web services | Apache Axis |
| | Grid services | The Globus Toolkit |
| <i>Peer-to-peer (Chapter 10)</i> | Routing overlays | Pastry |
| | Routing overlays | Tapestry |
| | Application-specific | Squirrel |
| | Application-specific | OceanStore |
| | Application-specific | Ivy |
| | Application-specific | Gnutella |

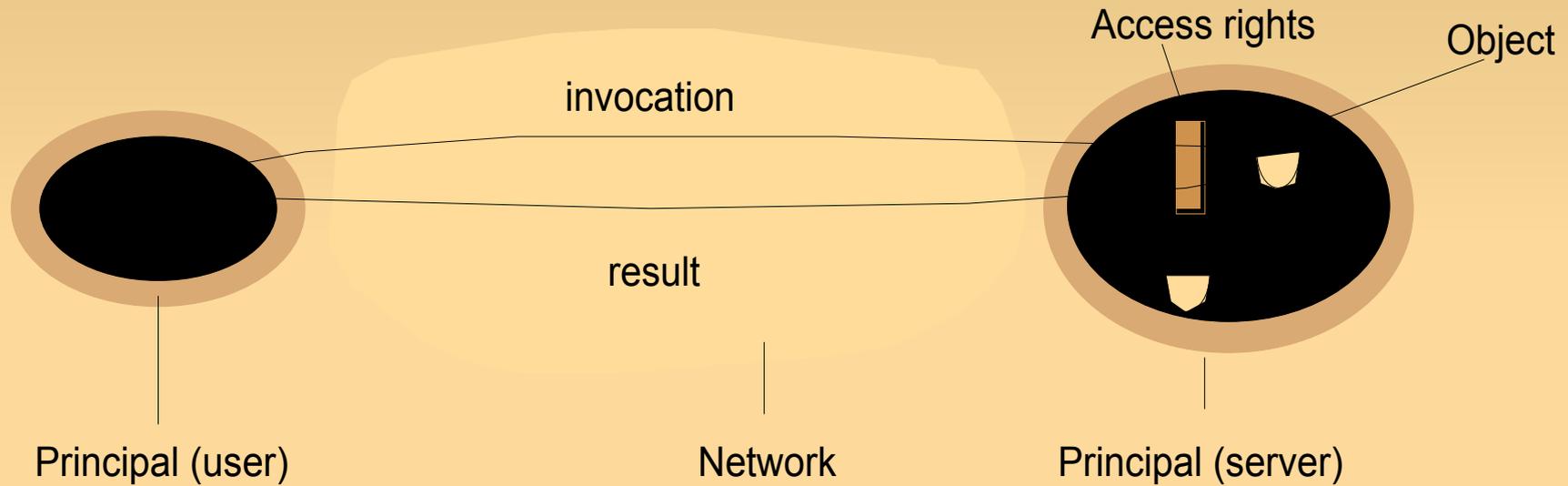
Ordenamiento de eventos en TR



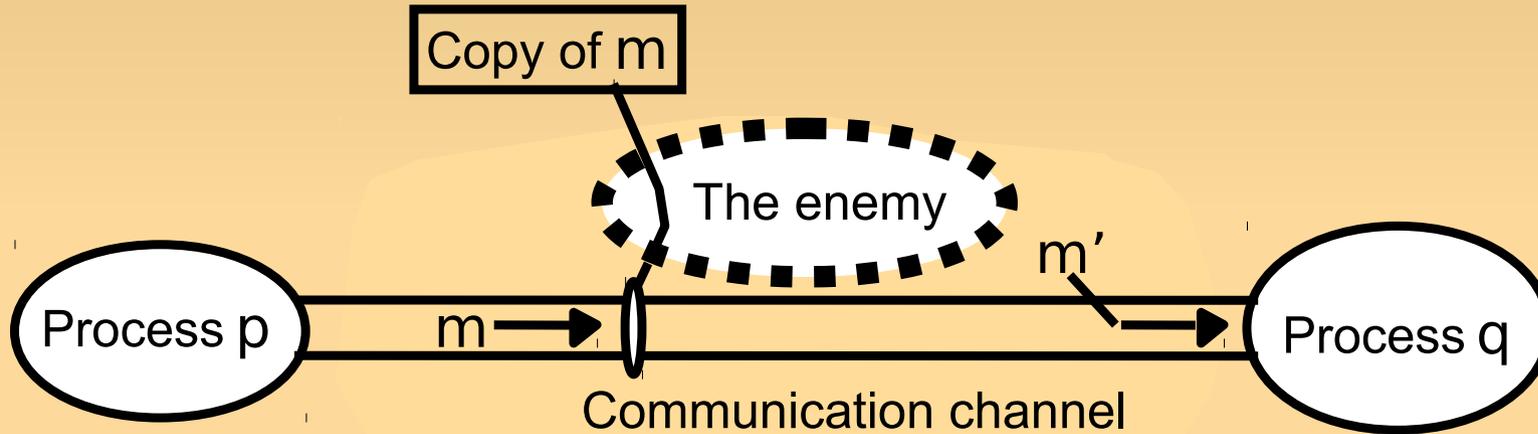
Procesos y canales



Objetos y principales



El enemigo



Canales seguros

