

Arquitectura de Software

Contenido

- Definiciones
- Motivación
- Propuesta
- Aplicaciones monocapa
- Aplicaciones cliente-servidor
- Aplicaciones de 3 capas

Definiciones

- La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, tales como la *confiabilidad*, *escalabilidad*, *portabilidad*, y *disponibilidad*.
- La arquitectura de software define, de manera abstracta, los componentes que llevan a cabo alguna tarea de computación, sus interfaces y la comunicación entre ellos. Toda arquitectura debe ser implementable en una arquitectura física, que consiste simplemente en determinar qué computadora tendrá asignada cada tarea.

Motivación

- Facilitar la comunicación entre los diferentes participantes en el desarrollo.
- Resaltar las decisiones de diseño que pueden tener un gran impacto en todo el proceso de desarrollo posterior.
- Aportar una visión de cómo se estructura el sistema y cómo colaboran sus componentes.
- Facilitar el desarrollo de aplicaciones flexibles, escalables y reutilizables.

Propuesta

No hay que inventar una nueva arquitectura de software para cada sistema que se desee construir. Lo habitual es adoptar una arquitectura conocida en función de sus ventajas e inconvenientes para cada caso concreto.

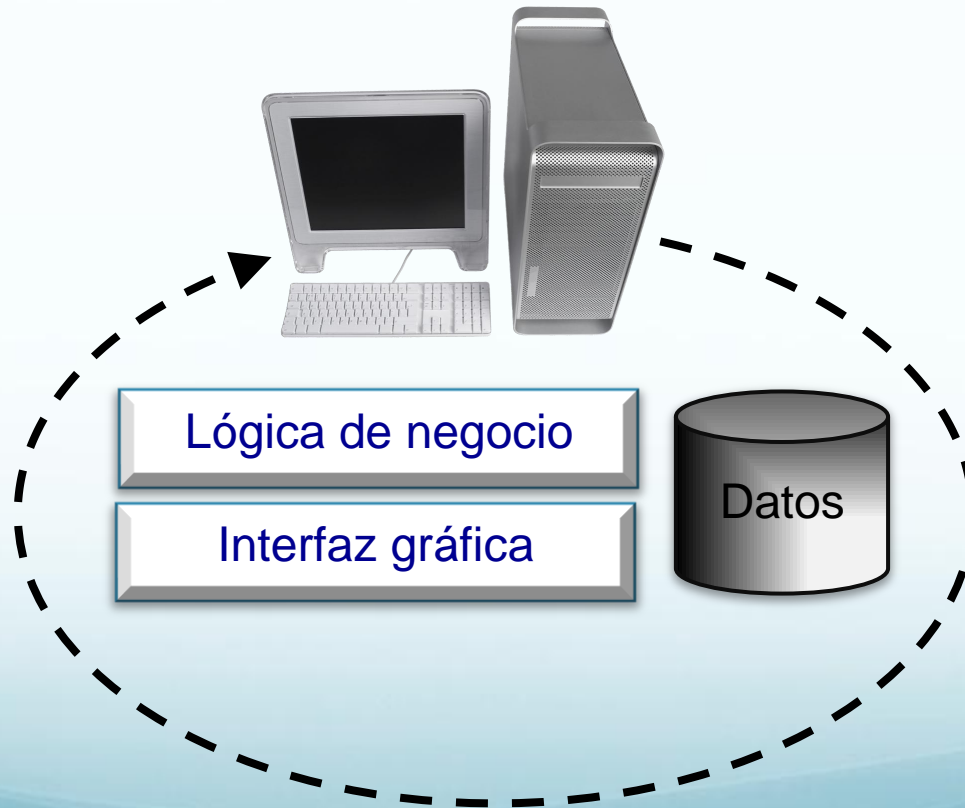
Las arquitecturas más universales son:

- *Monocapa*: el software se estructura en grupos funcionales muy acoplados.
- *Cliente-servidor*: el software reparte su carga en dos partes independientes.
- *Tres capas*: la carga se divide en tres partes (o capas).

La división de la aplicación en capas permite que los desarrolladores modifiquen partes de la aplicación sin necesidad de reescribirla en su totalidad.

Aplicaciones monocapa

- Tanto los datos de la aplicación como la interfaz gráfica y la lógica de negocio residen en una unidad.



Aplicaciones cliente-servidor

- El software reparte su carga de cómputo en dos partes independientes pero sin reparto claro de funciones.
- Se distinguen tres posibilidades, en función de la localización de la lógica de negocio...

Aplicaciones cliente-servidor

- Posibilidad 1: la lógica de negocio está en el Servidor



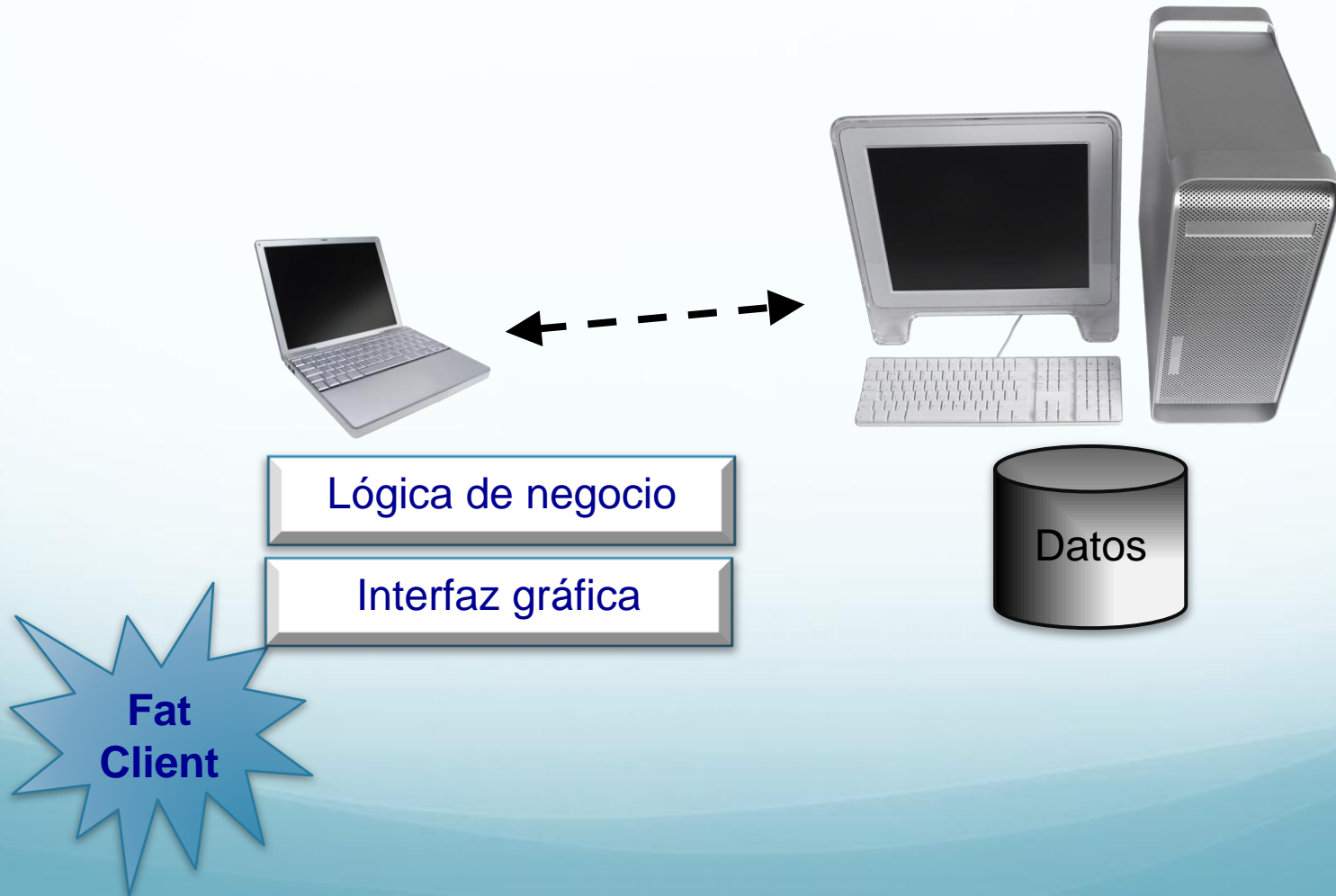
Aplicaciones cliente-servidor

- Posibilidad 2: se reparte la lógica de negocio entre el cliente y el servidor.

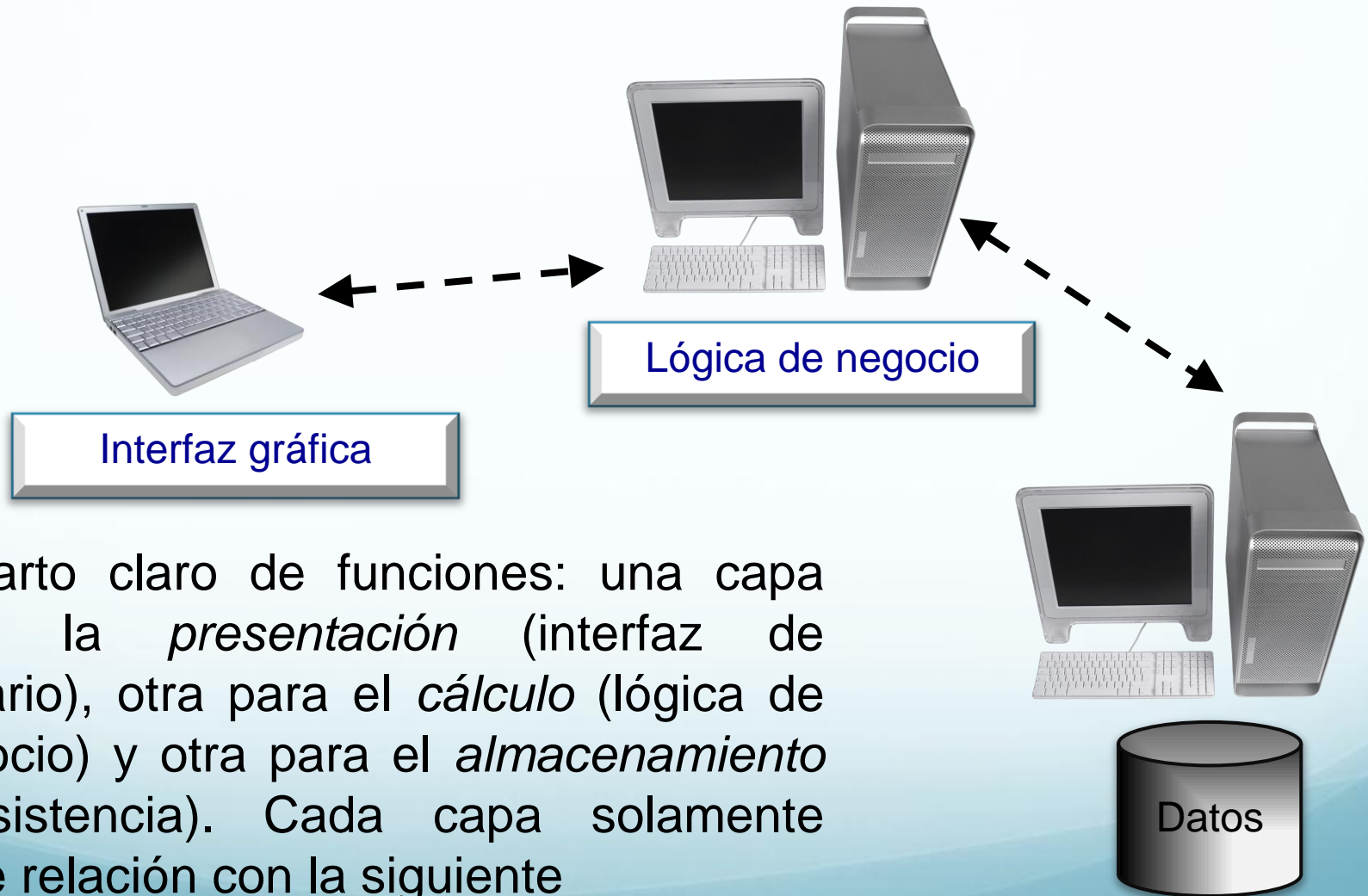


Aplicaciones cliente-servidor

- Posibilidad 3: la lógica de negocio está en el cliente.

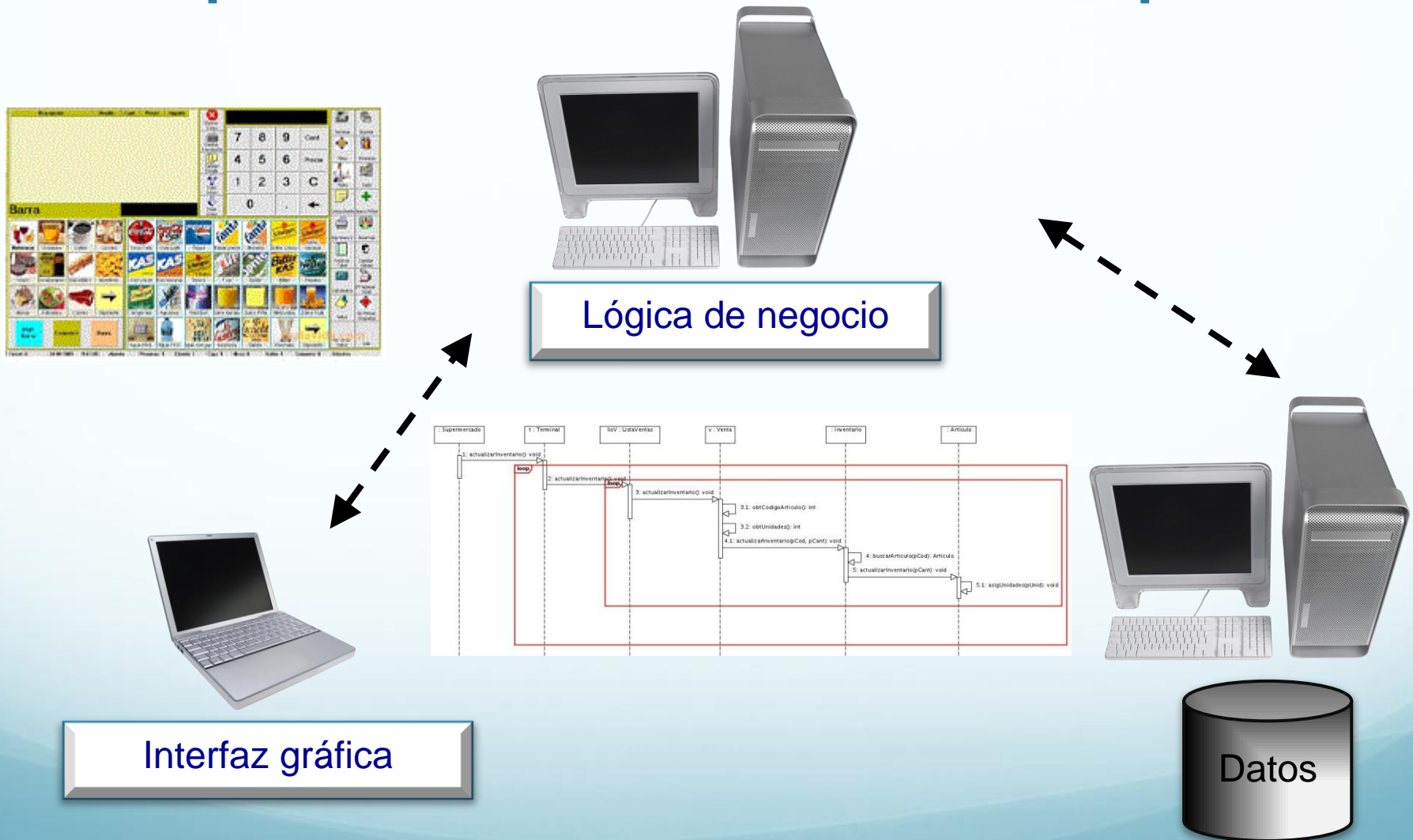


Aplicaciones de tres capas



Reparto claro de funciones: una capa para la *presentación* (interfaz de usuario), otra para el *cálculo* (lógica de negocio) y otra para el *almacenamiento* (persistencia). Cada capa solamente tiene relación con la siguiente

Aplicaciones de tres capas



Aplicaciones de tres capas

○ *Ventajas*

- Clientes livianos
- Capacidad de migración de SGBD (o persistencia datos)
- Escalabilidad
- Independencia de la plataforma
- Facilidad de actualización