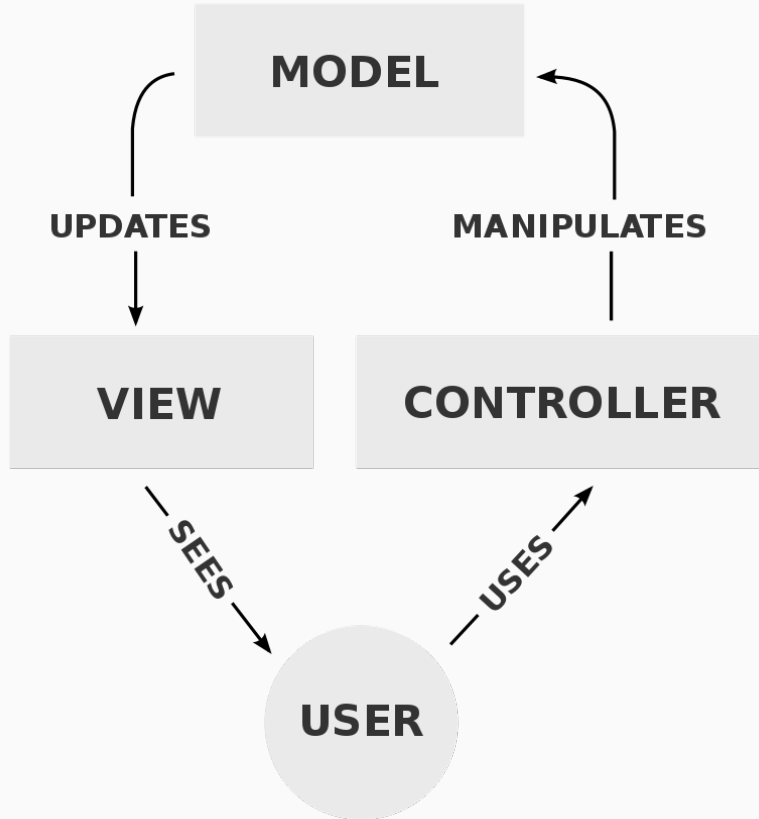


# Patrones de Arquitectura

75.10 Técnicas de Diseño FIUBA



# MVC



Tres roles distinguidos

- Modelo
- Vista
- Controlador

## 3 Roles

### Modelo

Objeto invisible que representa información sobre el dominio modelado, y se comporta según las reglas intrínsecas del negocio

### Vista

Representación del Modelo.  
Una forma de presentar su estado de manera que sea perceptible.

Sólo **muestra** información.

### Controlador

Operador. Encargado de tomar los estímulos recibidos, provocar las perturbaciones correspondientes al modelo y disparar actualizaciones de la/las vistas

## Importancia de la separación

- Vista, Modelo y Controlador son incumbencias separadas
- Múltiples vistas posibles para percibir el modelo
- Múltiples controladores permiten diferentes interacciones con el modelo
- Objetos de modelo más fácilmente *testeables*

# Layers

- Construcción de niveles donde cada nivel recibe servicios del que tiene debajo
- La interacción esperable es entre capas adyacentes

## Beneficios

- Las capas son
  - más simples que el todo
  - independientes/desacopladas
  - intercambiables
  - estandarizables
  - reusables

## Contras

- No se puede encapsular todo
- Cambios en cascada
- La interacción entre capas implica overhead
- Se debe cuidar la cantidad
- No es trivial definir la división más correcta

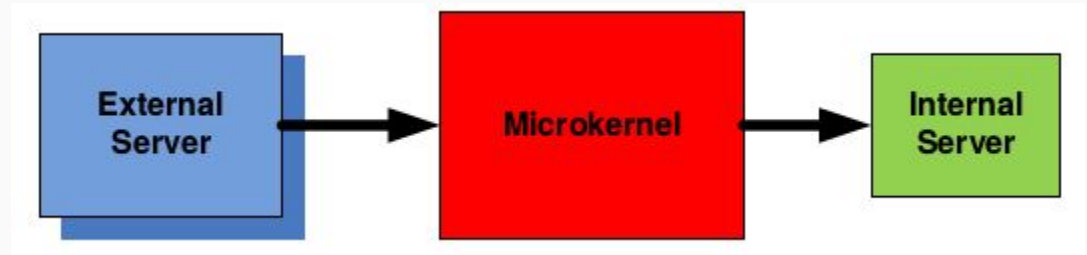
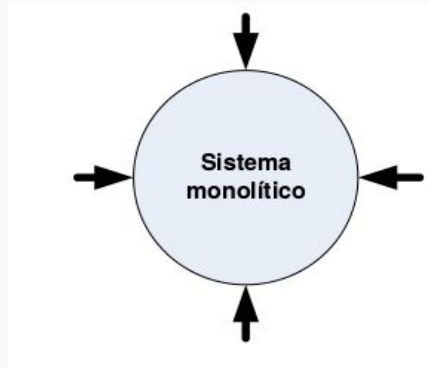
# Microkernel

## Cuando se usa?

- En sistemas de larga vida que deben evolucionar ante cambios de distinto tipo y con diferente frecuencia.

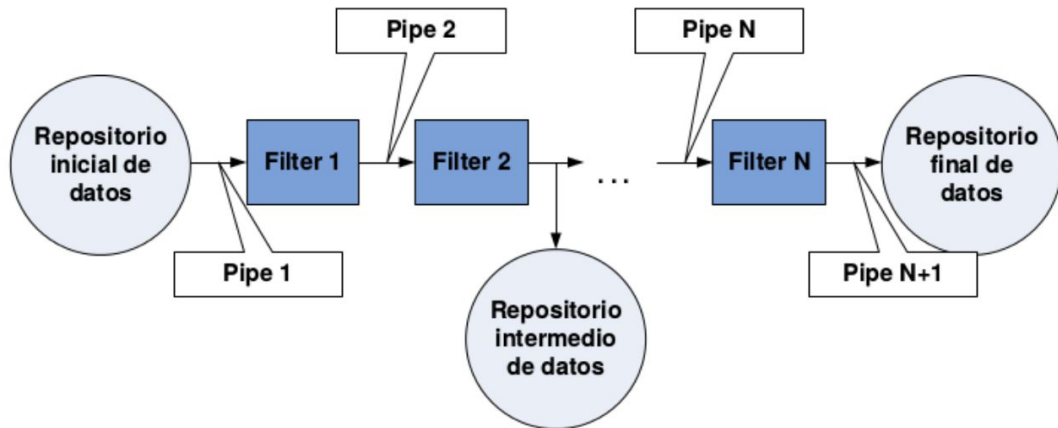
Ventajas / beneficios: soporta cambios con menor impacto.

Desventajas: diseño más elaborado



## Que es un filter? y un pipe?

- **Filter:** componente que recibe datos y devuelve el mismo conjunto de datos, en general, modificado.
- **Pipe:** componente que se utiliza para conectar dos filters.

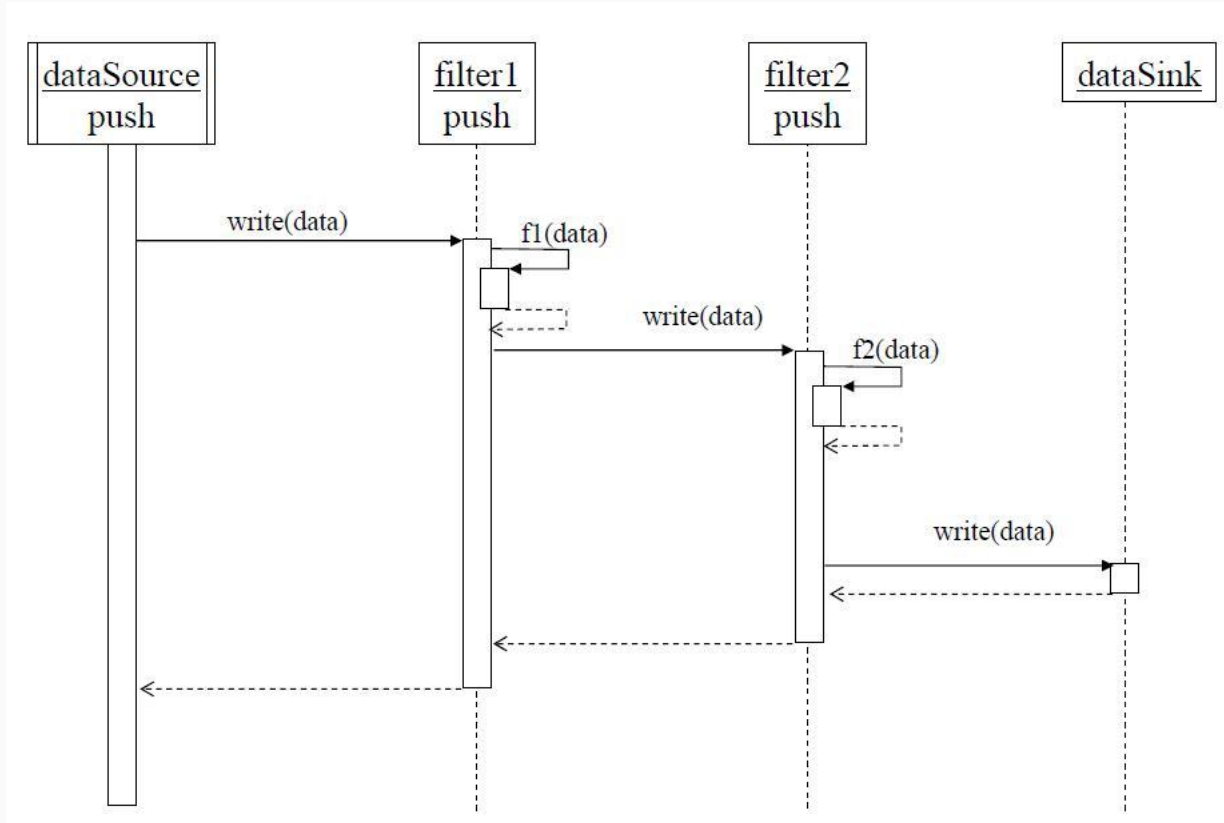


## Cuando se usa?

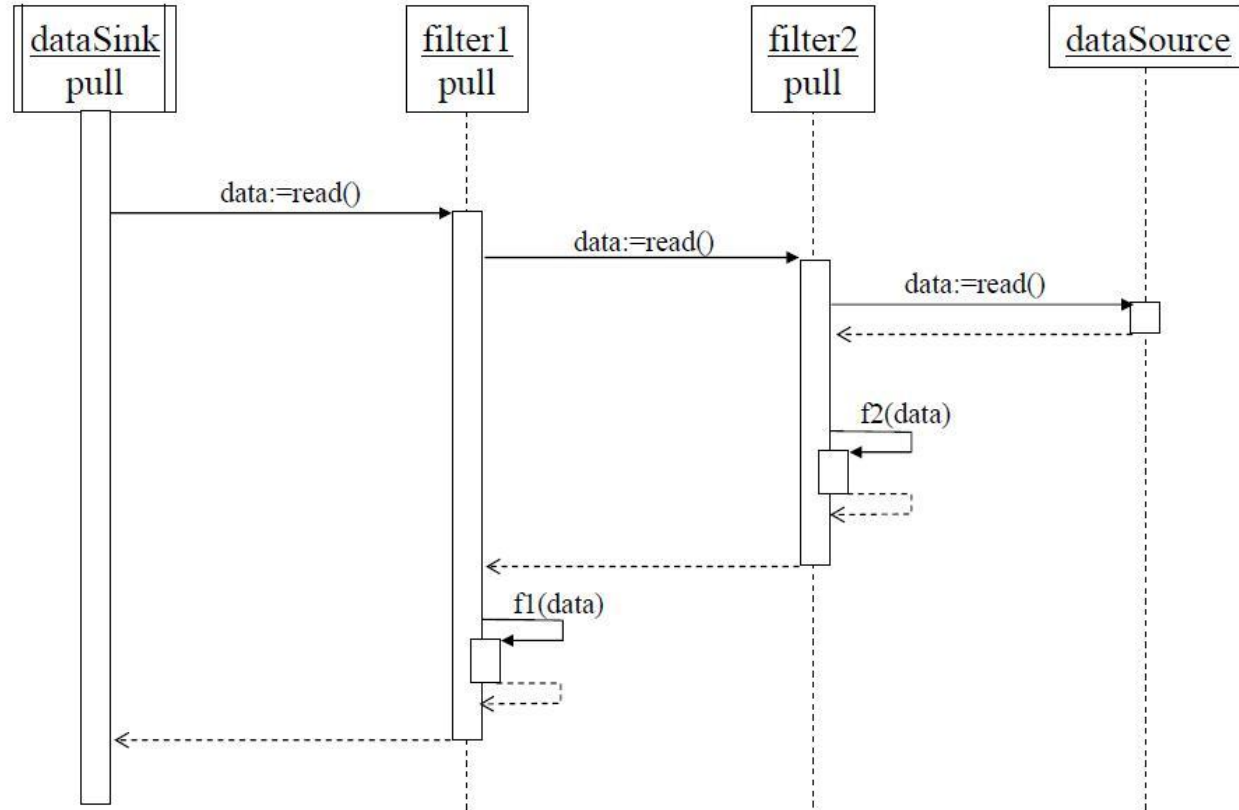
- Queremos encapsular distintas tareas, sobre un mismo conjunto de datos, en componentes individuales; cada uno encargado de realizar una tarea específica.
- Necesitamos poder interconectar dichos componentes.
- Queremos poder construir distintas "líneas" de procesamiento interconectando componentes reutilizables.



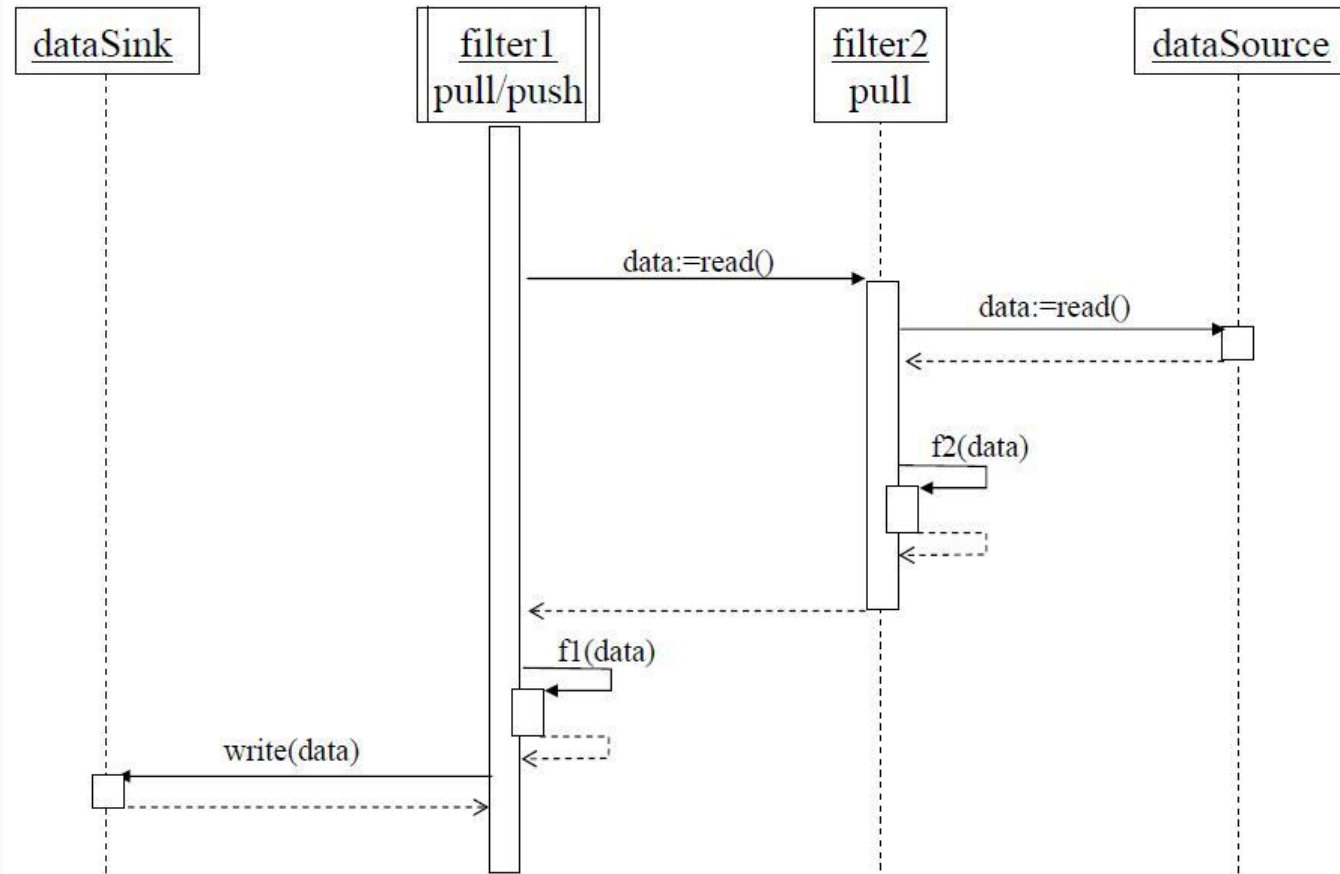
# Pipe & Filter



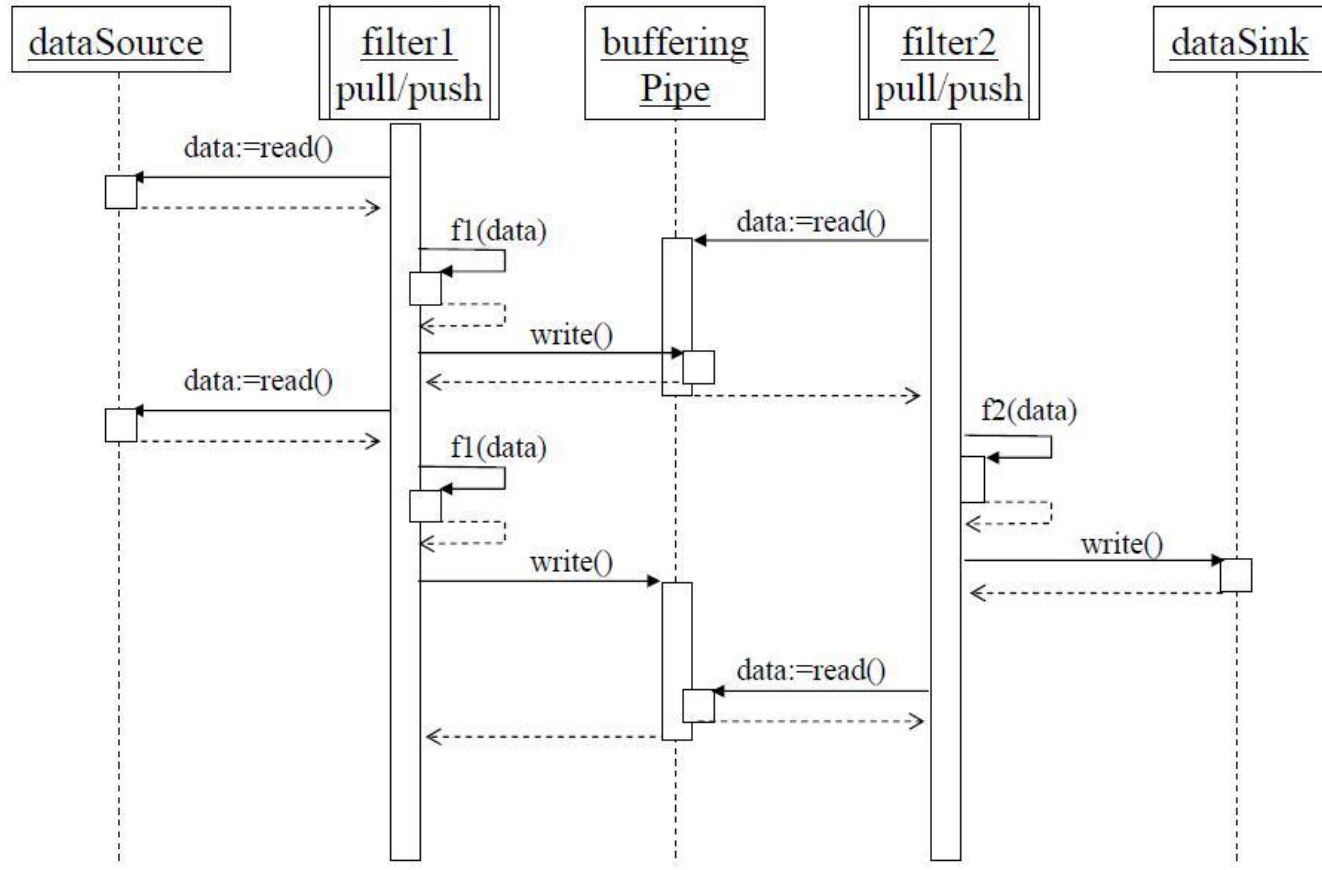
# Pipe & Filter



# Pipe & Filter



# Pipe & Filter



## Usos Conocidos

- HTTP Filters
- UNIX commands
- Compilador

## Que es un broker?

- Broker = Intermediario (Definición)
- ¿Entre quién? Entre dos cosas que queremos desacoplar.
- En software, por lo general, se trata de un cliente que solicita un servicio y el servidor que presta dicho servicio.

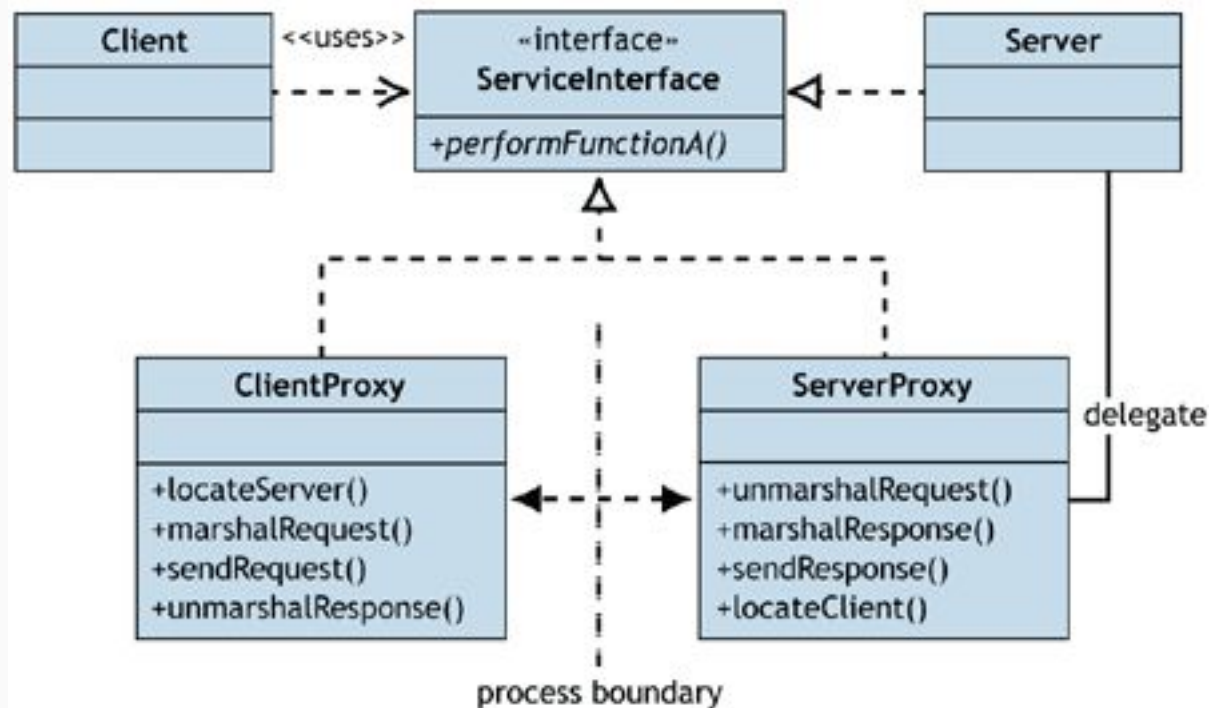
## Cuando se usa?

- En sistemas distribuidos, donde se requiere desacoplar cliente y servidor. Es decir, cuando ambos están físicamente separados. ¿Tiene sentido si están juntos?
- Necesitamos un intermediario que coordine comunicación (requests, responses and exceptions)
- Queremos abstraer el cómo el cliente y el servidor se comunican.
- Queremos que para el cliente sea transparente el hecho de que el servidor está físicamente en otra parte.

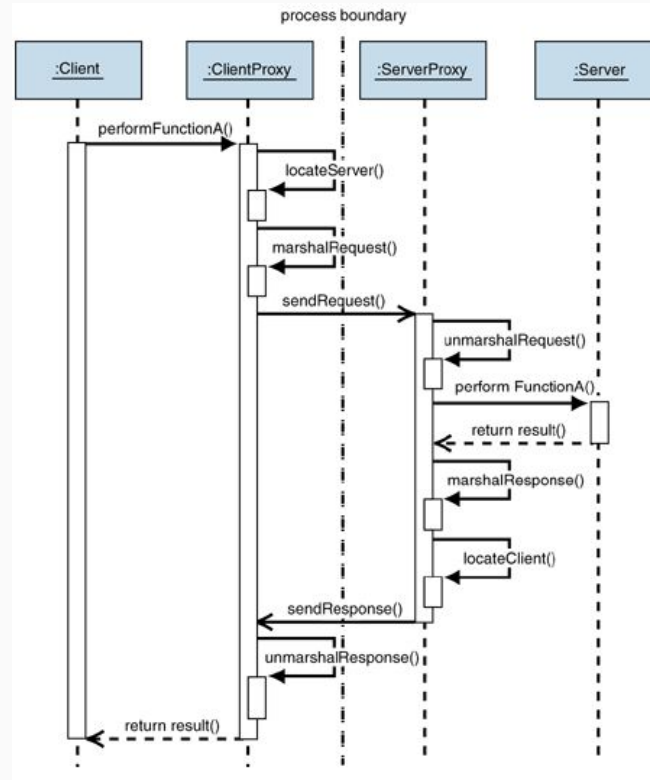




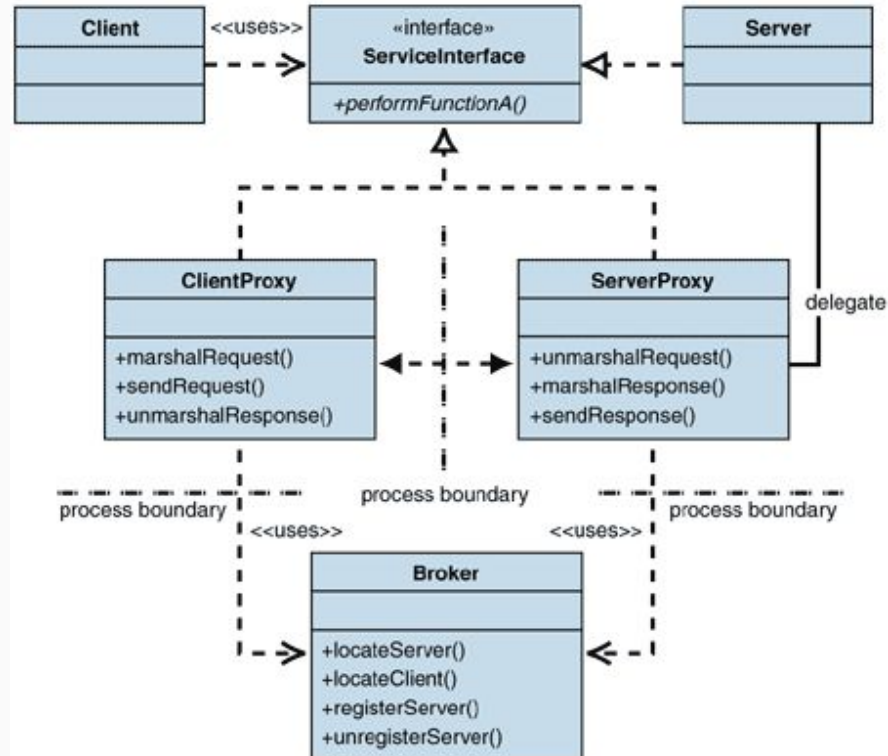
# Broker



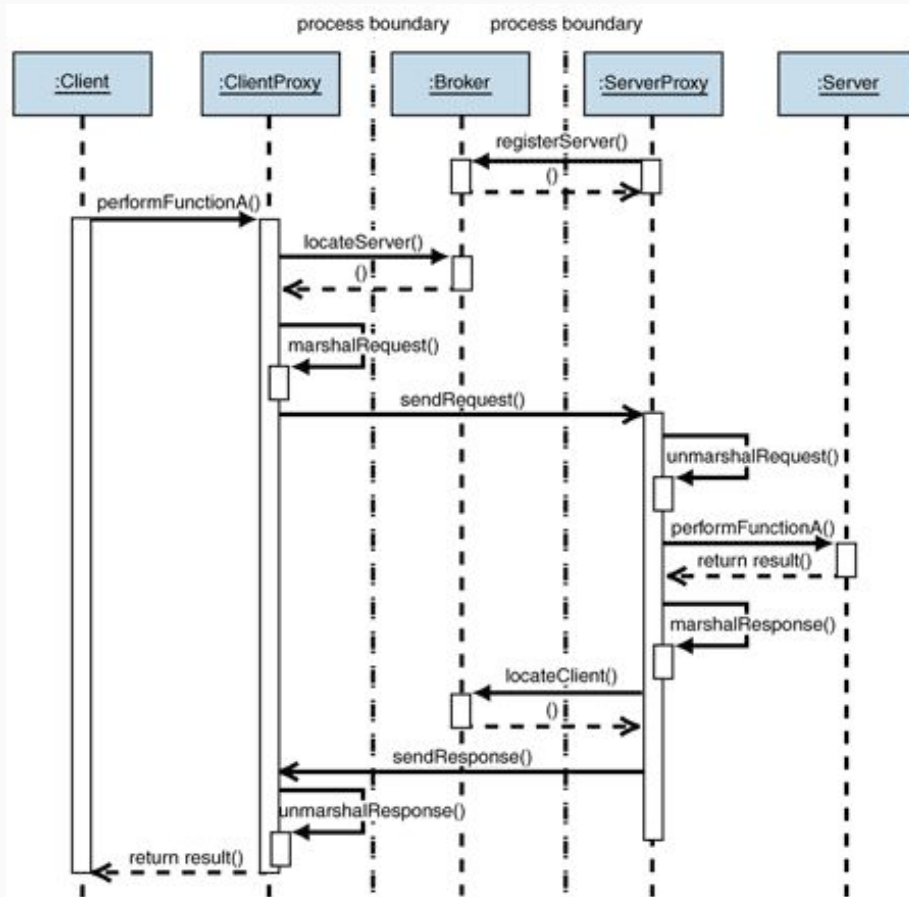
# Broker



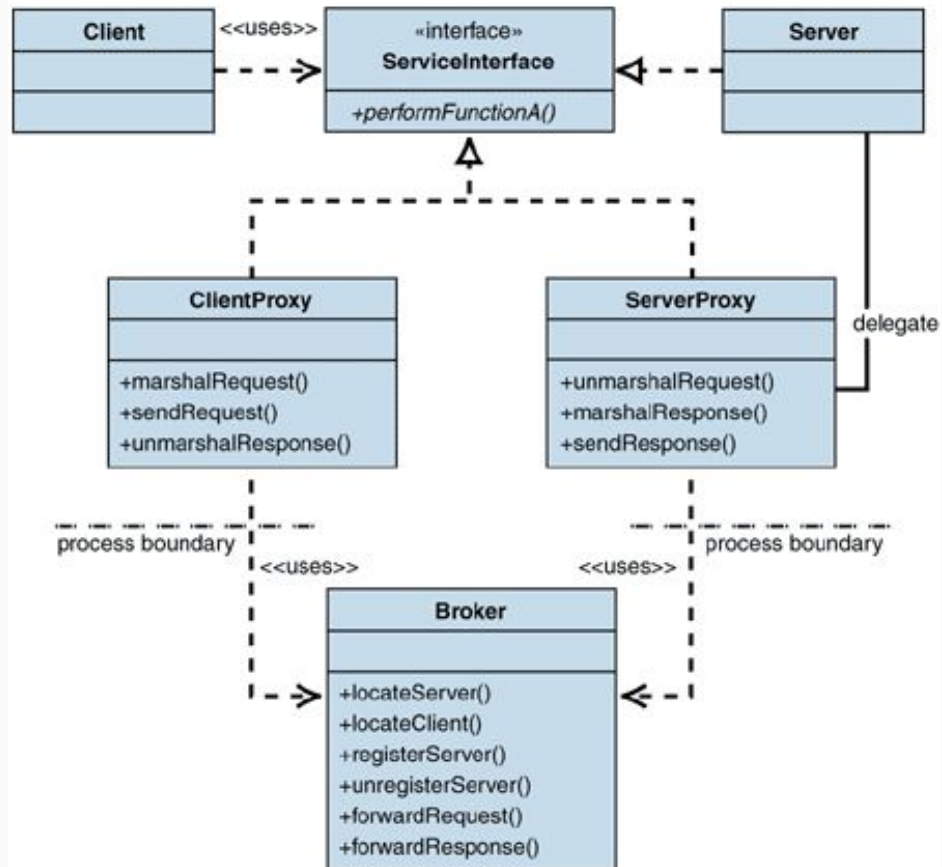
# Broker



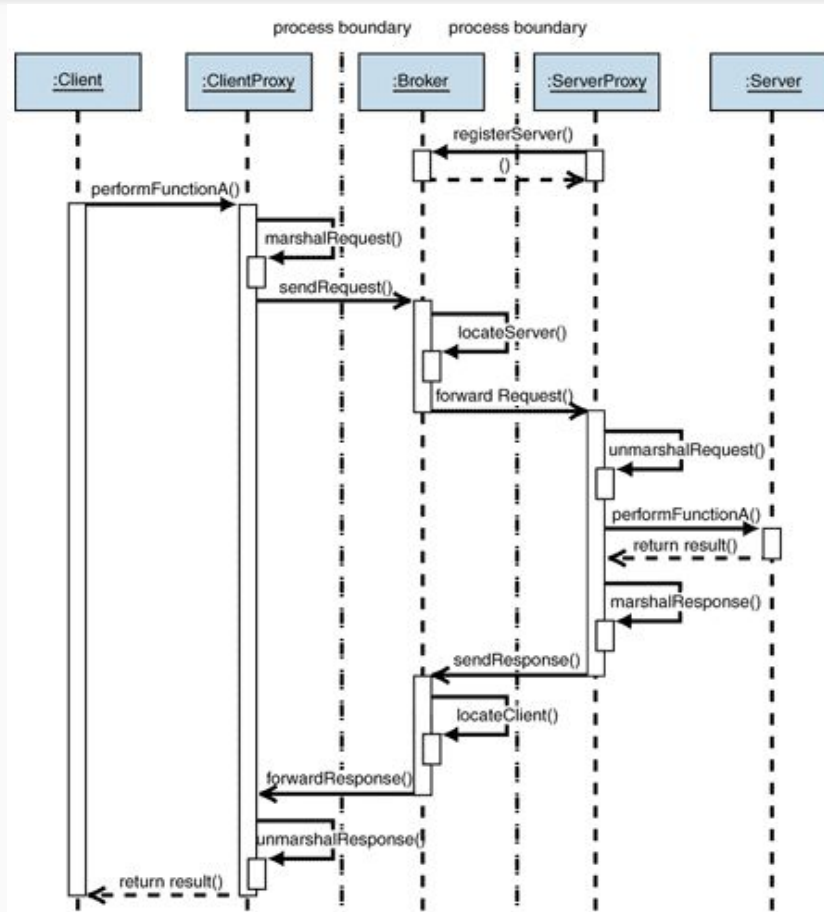
# Broker



# Broker



# Broker



## Brokers Conocidos

- CORBA (Common Object Request Broker Architecture)
- RMI
- .NET Remoting
- Active MQ

Una aplicación enterprise es un sistema fuertemente orientado a un negocio determinado, que debe cumplir ciertos requerimientos no funcionales.

## **Responde a ciertas necesidades**

- Codificar las reglas de negocio
- Representar las entidades de negocio
- Persistir su estado
- Garantizar la coherencia de datos
- Manejar la distribución de la aplicación

## **Características habituales**

- Lógica de negocio
- Persistencia de datos en forma masiva
- Multiusuarios
- Múltiples interfaces de usuario
- Disonancia conceptual



# Enterprise Architecture + Layers

## Modelo de 3 capas

- Presentación
- Dominio
- Persistencia

## Modelo de 4+ capas

- Presentación
- Servicios (+)
- Dominio
- Persistencia

Generalmente, además de estar aislada lógicamente, cada capa está aislada físicamente

A veces llamado N-tiers