

# Arquitectura Software Dirigida por Modelos.

*Departamento de  
tecnologías de la  
Información*



# TEMA 1: ARQUITECTURA SOFTWARE



- › 1.1. Introducción
- › 1.2. Patrones Arquitectónicos. Estructura
- › 1.3.- Patrones Arquitectónicos. Comunicaciones.



# 1.- Introducción.



# Introducción – Definición de Arquitectura Software

## Ingeniería - IEEE 610.12-1990.

Un componente de software es un elemento de un sistema software que ofrece un conjunto de servicios, o funcionalidades, a través de interfaces definidas

*“La Ingeniería de Software es la aplicación de una estrategia sistemática, disciplinada y cuantificable para el desarrollo de software.”*

Consiste en las colaboraciones entre los distintos componentes para llevar a cabo una función

## Arquitectura - IEEE

Consiste por una parte en el entorno de desarrollo y el entorno de desempeño de la aplicación, es decir, donde va a funcionar esa aplicación

*“La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el entorno y los principios que orientan su diseño y evolución.”*



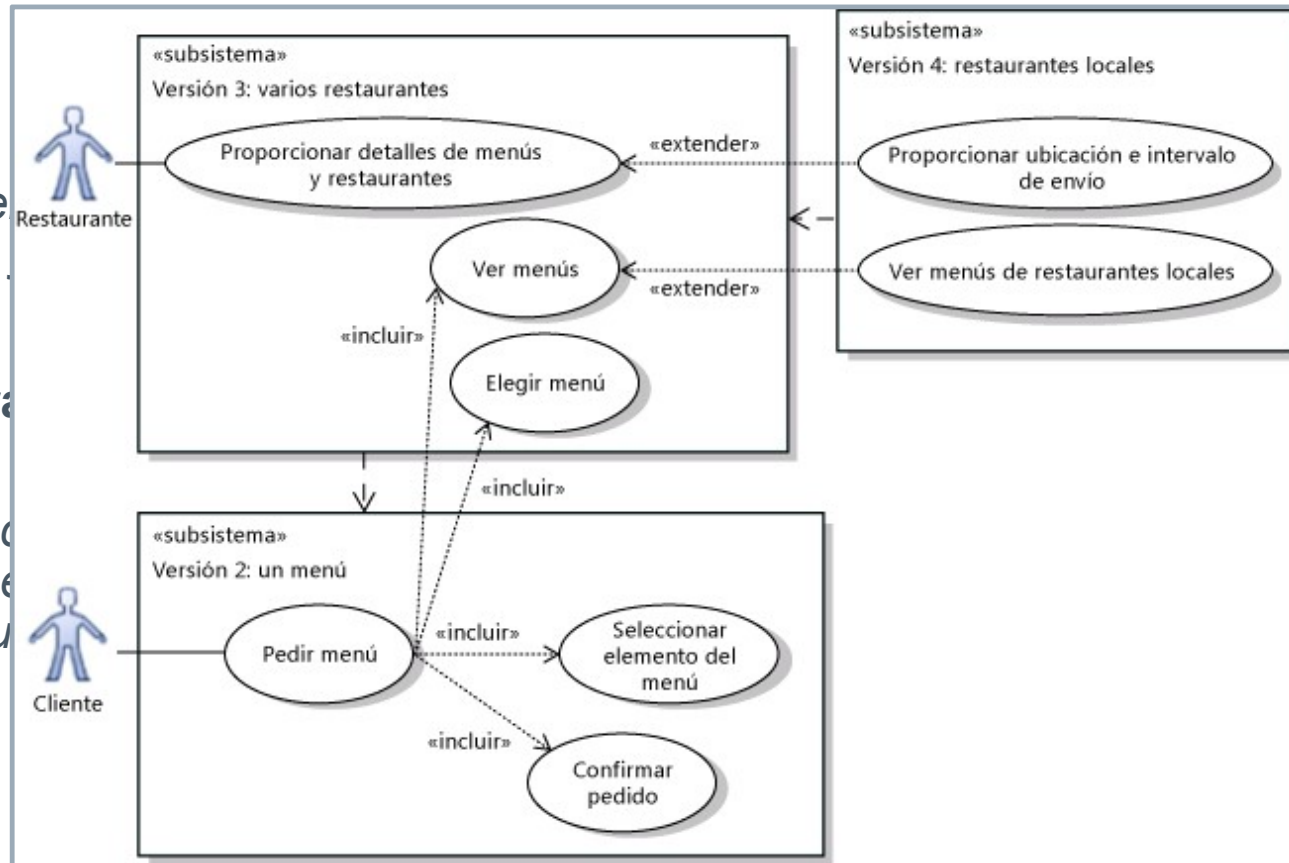
# Introducción – Definición de Arquitectura Software

Ingeniería -

“La Ingeniería es una disciplina”

Arquitectura

“La Arquitectura es la encarnación de los principios que rigen el sistema”



“La Ingeniería es una disciplina, el software.”

“La Arquitectura es la encarnación de los principios que rigen el sistema”



# Introducción – Definición de Arquitectura Software

**Clements, and Kazman (2003)**

*“La Arquitectura Software de un programa o sistema informático es la estructura del sistema, que comprenden los elementos software, las propiedades visibles externamente de esos elementos y las relaciones entre ellos”.*

**David Garlan**

*“La Arquitectura Software, abarca todo lo relativo a la estructura de alto nivel de los sistemas: su organización en subsistemas y la relación entre ellos.”*

*“La Arquitectura de Software establece un puente entre el requerimiento y el código.”*



# Introducción – Definición de Arquitectura Software

Una noción clave de la arquitectura es la organización y está relacionada con:

- › Aspectos de rendimiento
- › Reutilización
- › Limitaciones económicas y tecnológicas



La arquitectura software se encuentra entre los requisitos y el diseño.

Una arquitectura sigue un patrón o conjunto de patrones.

**La arquitectura software es la estructura general del software, que en su forma más sencilla es la estructura de organización de los componentes de un programa, la forma en la que éstos interactúan y la estructura de datos que utilizan.**



# Introducción – Definición de Arquitectura Software

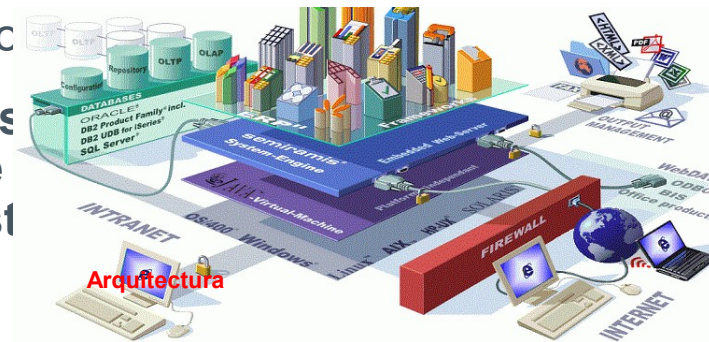
Una noción clave de la arquitectura es la organización de un sistema con:

- › Aspectos de rendimiento
- › Reutilización
- › Limitaciones económicas y tecnológicas

La arquitectura software se encuentra entre los requisitos y el diseño.

Una arquitectura sigue un patrón c

**La arquitectura software es la estructura de un programa, la forma en la que éstos utilizan.**



**La arquitectura software es la estructura de un programa, la forma en la que éstos utilizan.**





# Introducción – Definición de Arquitectura Software

Una noción clave de la arquitectura es la organización y está relacionada con:

- › Aspectos de rendimiento
- › Reutilización
- › Limitaciones económicas y tecnológicas

La arquitectura software se encuentra entre los re

Una arquitectura sigue un patrón o con

**La arquitectura software es la estruc**  
**más sencilla es la estructura de orga**  
**programa, la forma en la que éstos in**  
**utilizan.**



que en su forma  
tes de un  
e datos que



# Introducción – Definición de Arquitectura Software

Una noción clave de la arquitectura es la org

- › Aspectos de rendimiento
- › Reutilización
- › Limitaciones económicas y tecnológicas



elacionada con:

La arquitectura software se encuentra entre los requisitos y el diseño.

Una arquitectura sigue un patrón o conjunto de patrones.

**La arquitectura software es la estructura general del software, que en su forma más sencilla es la estructura de organización de los componentes de un programa, la forma en la que éstos interactúan y la estructura de datos que utilizan.**



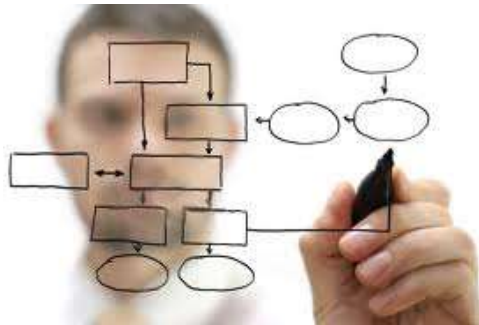
# Introducción – Definición de Arquitectura Software

- › Resumiendo:
  - Componentes software
  - Propiedades externamente visibles
  - Relaciones entre los componentes
  - El entorno



# Introducción – Definición de Arquitectura Software

- › ¿Diferencias entre los términos arquitectura software y diseño software?
- › Patrones arquitectónicos.



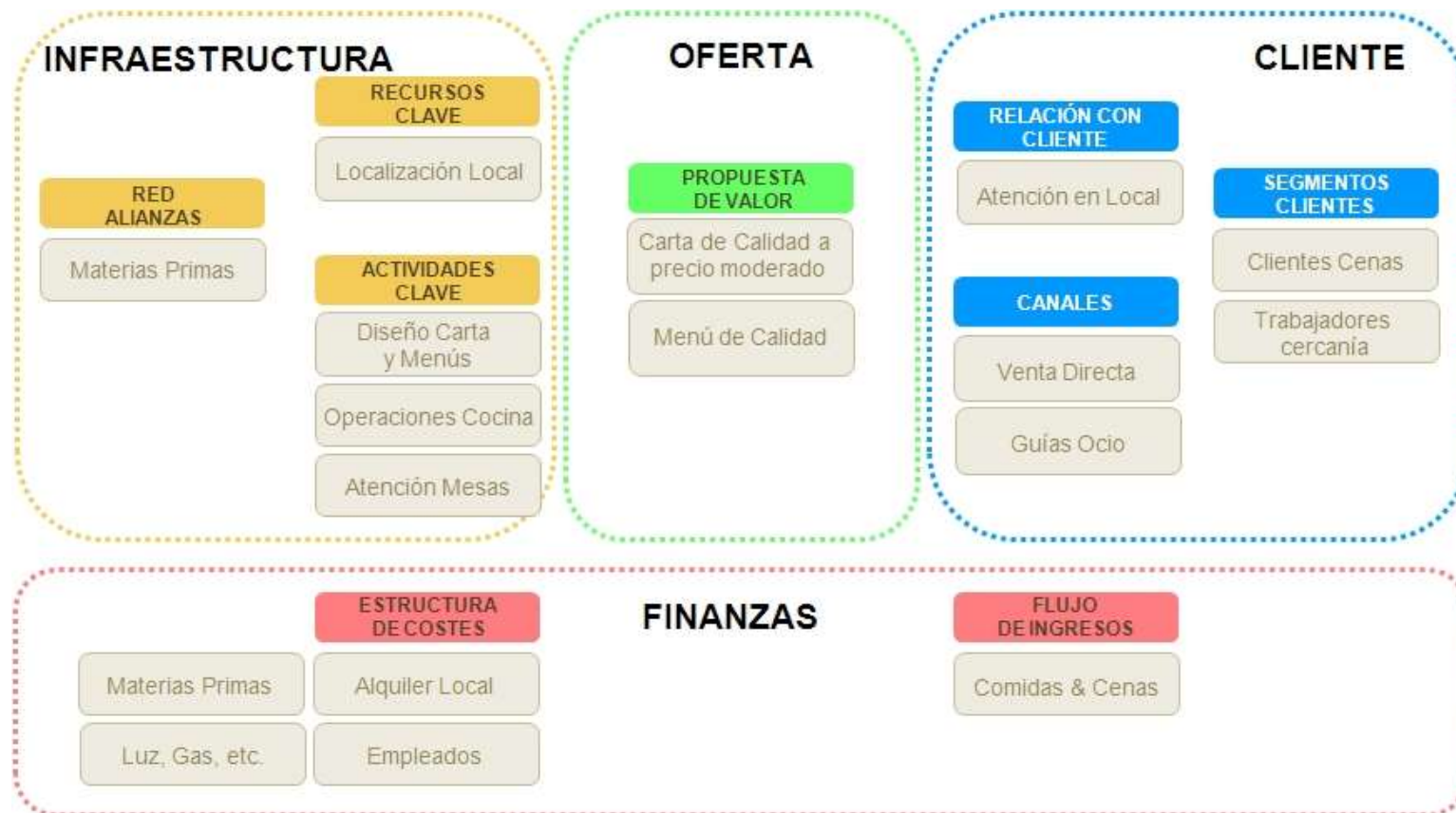


# Introducción – Definición de Arquitectura Software

- › El modelo arquitectónico proviene de tres fuentes:
  - 1. Información sobre el dominio de la aplicación.
  - 2. Elementos específicos del modelo de requerimientos.
  - 3. Disponibilidad de estilos arquitectónicos y sus patrones.



# Introducción – Definición de Arquitectura Software

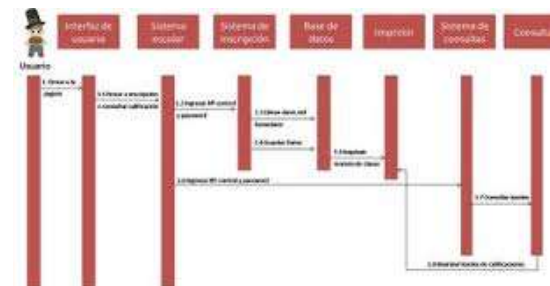
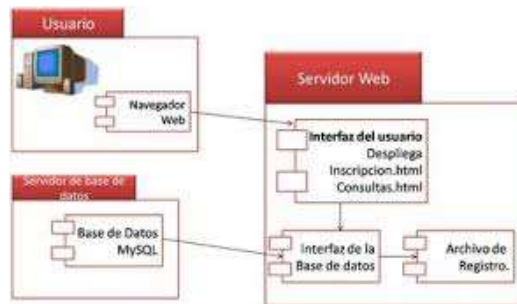
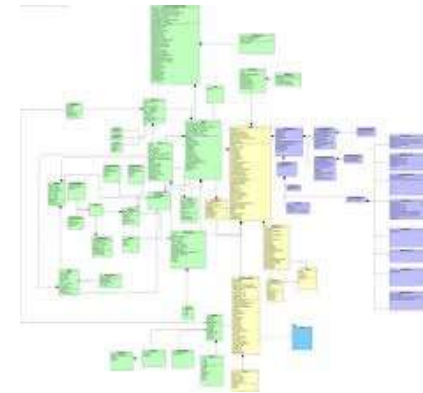




# Introducción – Definición de Arquitectura Software

La Arquitectura de Software debe representar los distintos aspectos del software.

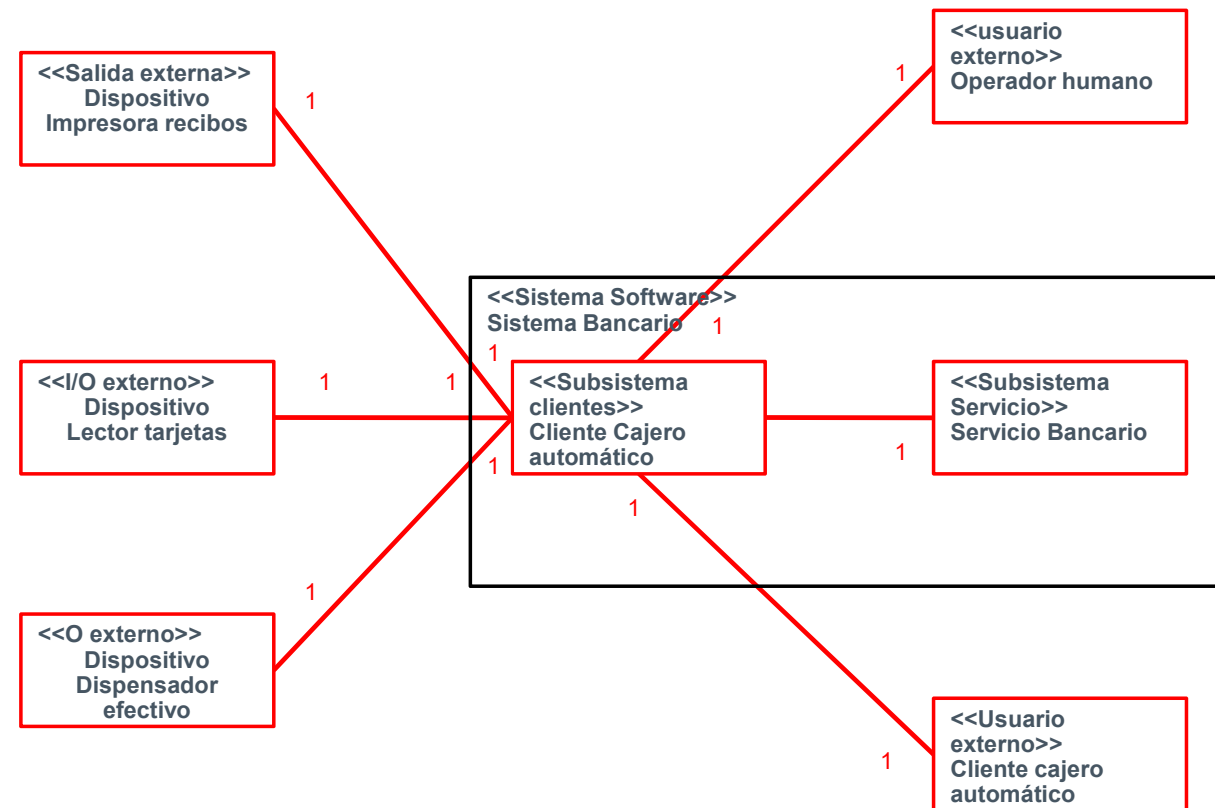
- Visión estática o vista estructural
- Visión dinámica
- Visión funcional o de implementación





# Introducción – Definición de Arquitectura Software

– Visión estática o vista estructural

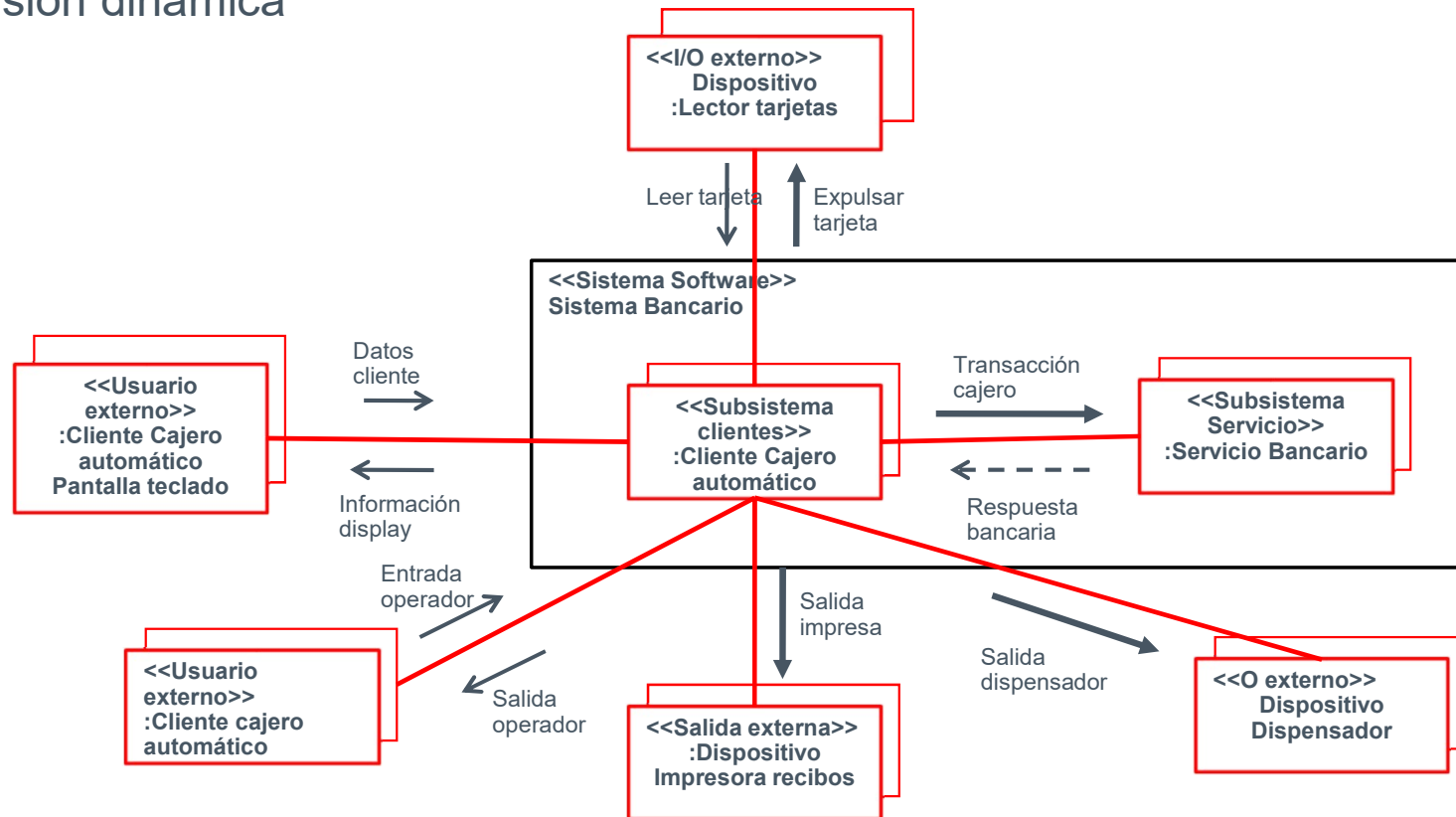






# Introducción – Definición de Arquitectura Software

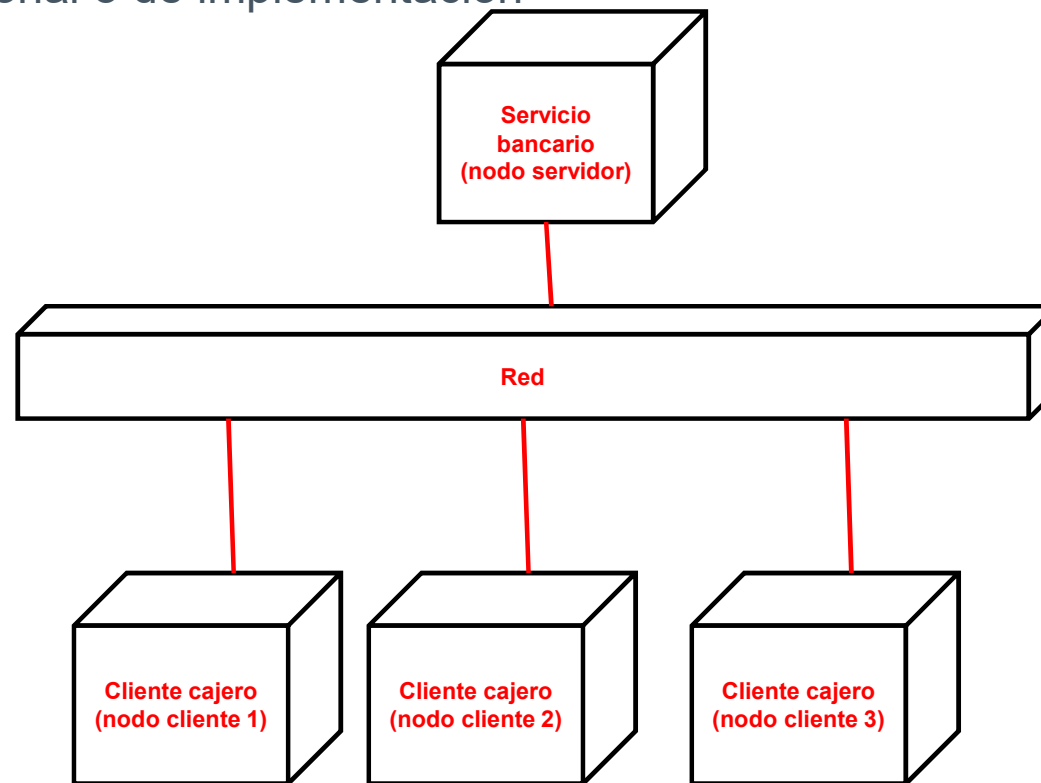
– Visión dinámica





# Introducción – Definición de Arquitectura Software

– Visión funcional o de implementación





# Introducción – Definición de Arquitectura Software

Según Shaw y Garlan existen un conjunto de propiedades que se deben especificar como parte del diseño arquitectónico:

- › Propiedades estructurales.
- › Propiedades extra-funcionales.
- › Familias de sistemas relacionados.



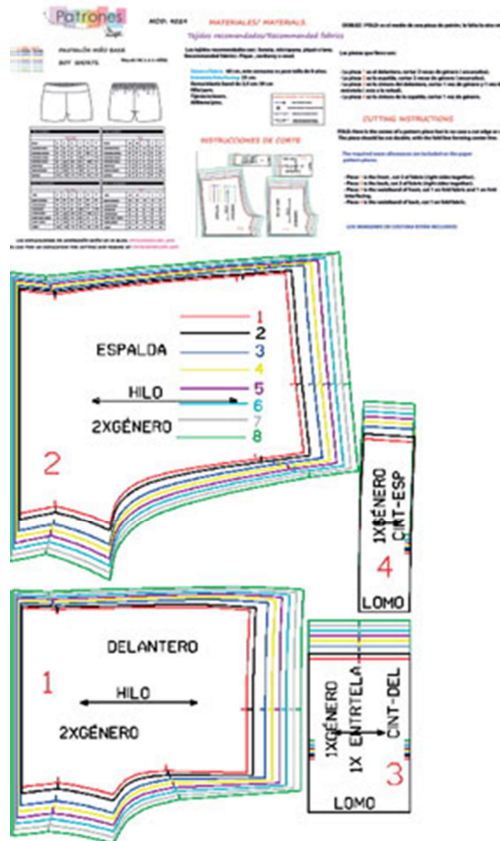
# Introducción – Definición de Patrón Arquitectónico

La arquitectura podría ser única para cada sistema software, pero existen muchos sistemas software cuya arquitectura son similares.





# Introducción – Definición de Patrón Arquitectónico





### 2.- Patrones Arquitectónicos Estructura.

#### 3.1.- **Arquitectura Software en Subsistemas**

3.2.- Arquitectura Software Orientada a Objetos

3.3.- Arquitectura Software Cliente/Servidor

3.4.- Arquitectura Software Orientada a Servicios

3.5.- Arquitectura Software Orientada a Componentes distribuidos

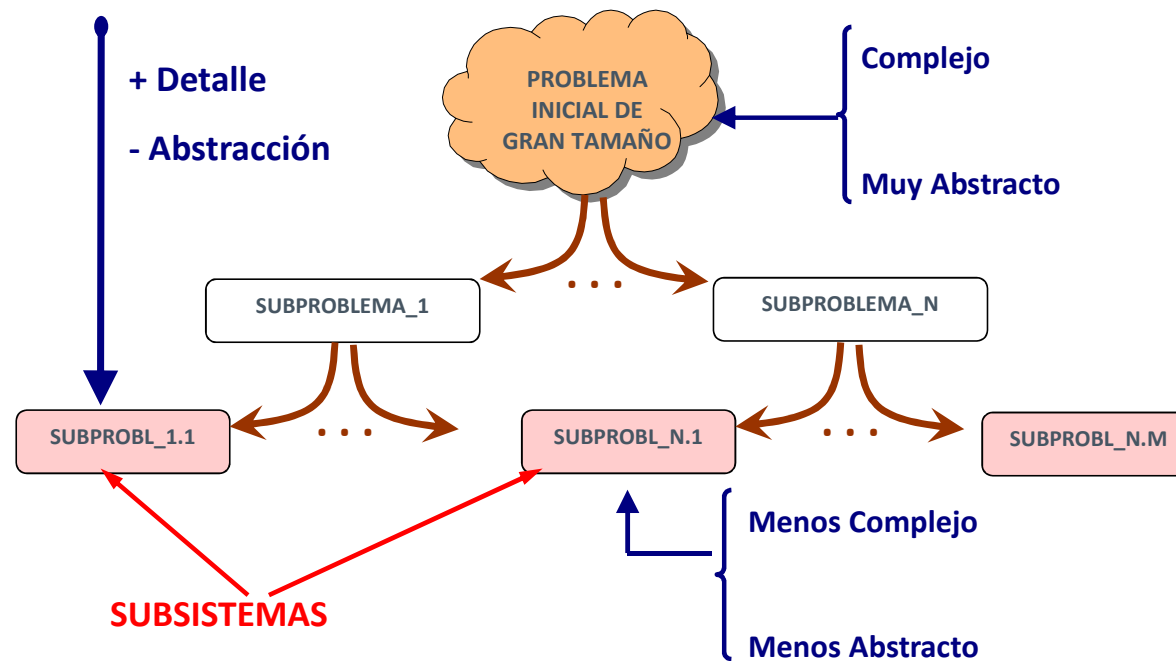


# Diseño de Arquitectura Software en Subsistemas

- › Aplicable a sistemas de gran tamaño
- › Objetivos:
  - Realizar una función concreta
  - Independiente



# Diseño de Arquitectura Software en Subsistemas

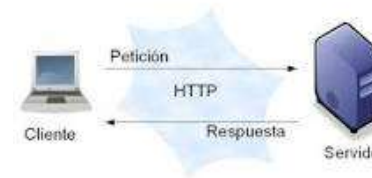






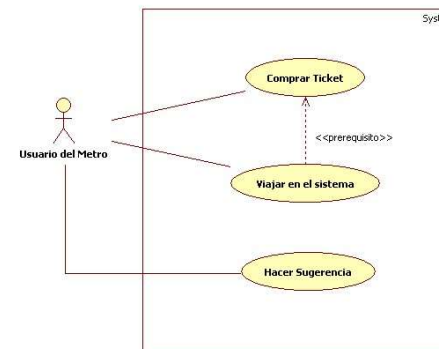
# Diseño de Arquitectura Software en Subsistemas

› Por distribución



› Casos de uso

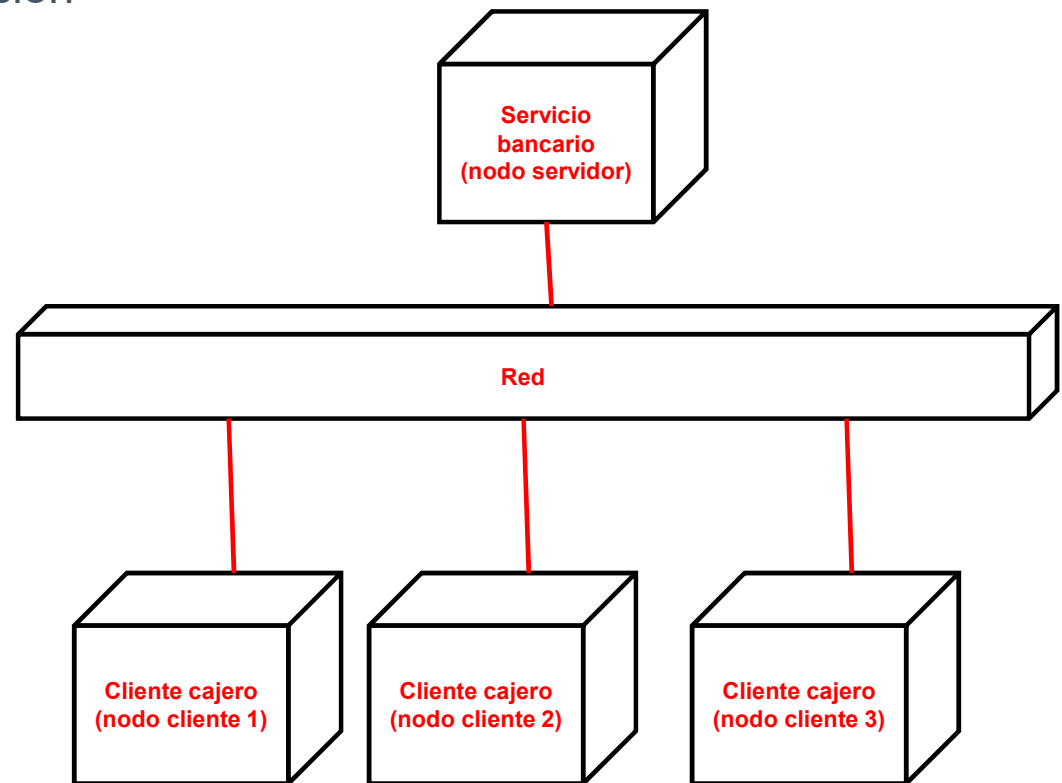
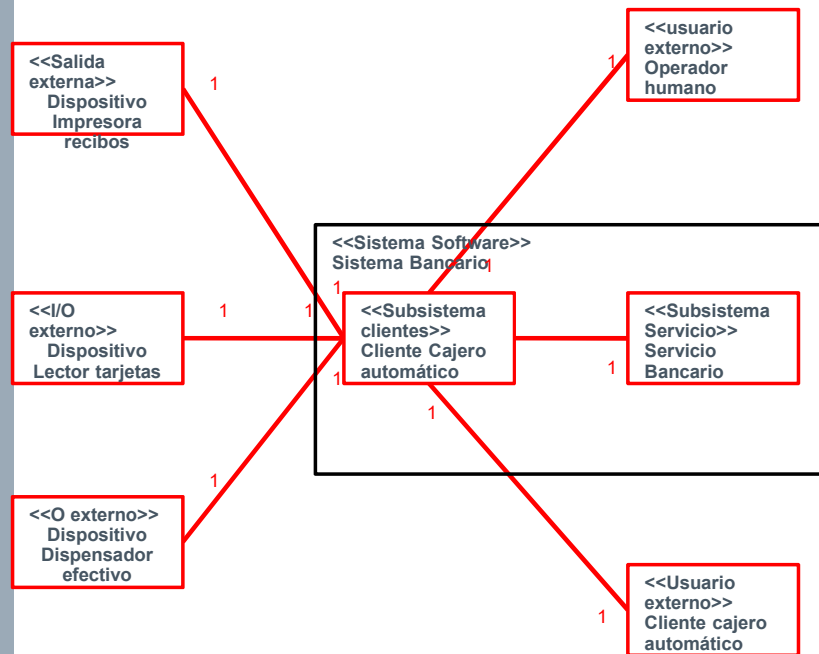
› Acoplamiento





# Introducción – Definición de Arquitectura Software

– Visión funcional o de implementación





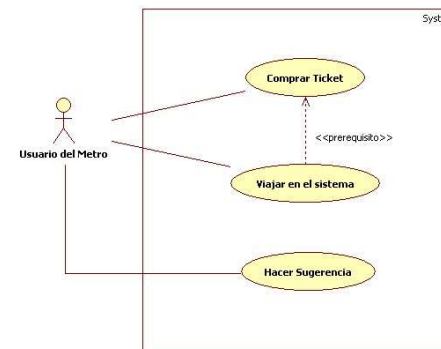
# Diseño de Arquitectura Software en Subsistemas

› Por distribución



› Casos de uso

› Acoplamiento





# Diseño de Arquitectura Software en Subsistemas

## › Asuntos o problemas en el diseño de subsistemas

1. Objetos compuestos
2. Localización geográfica
3. Clientes y servicios
4. Interacción de usuario
5. Interface a objetos externos
6. Ámbito de control.



# Diseño de Arquitectura Software en Subsistemas

## › Resumiendo:

1. Los objetos que son parte del mismo objeto compuesto deben estar en el mismo subsistema. Se crean juntos, viven juntos, y mueren juntos.
2. Si dos objetos están físicamente separados en diferentes ubicaciones geográficas, deben estar en diferentes subsistemas.
3. Los clientes y los servicios deben estar en subsistemas separados.
4. Los objetos de interacción de usuario deben estar en subsistemas separados del resto.
5. Un objeto del mundo real externo debe interconectar a un solo subsistema.
6. Un objeto de control y todas las entidades y los objetos de E/S que controla deben formar parte de un subsistema y no dividirse entre subsistemas.



# Diseño de Arquitectura Software en Subsistemas

## › Tipos de subsistemas:

```
<<subsystem>>  
<<component>>  
<<service>>
```

1. Subsistema cliente.
2. Subsistema de Interacción de usuario.
3. Subsistema de servicio.
4. Subsistema de control.
5. Subsistema de E/S.