

## **Patró arquitectònic: Arquitectura en capes**

- Context
- Problema
- Solució
- Consideracions en la definició de l'arquitectura
- Beneficis i inconvenients
- Aplicació als Sistemes d'Informació
  - Arquitectura en tres capes
  - Vista de desenvolupament de l'arquitectura en tres capes
  - Vista física (de desplegament) de l'arquitectura en tres capes
  - Especificació dels serveis de cada capa
- Bibliografia

## **Context**

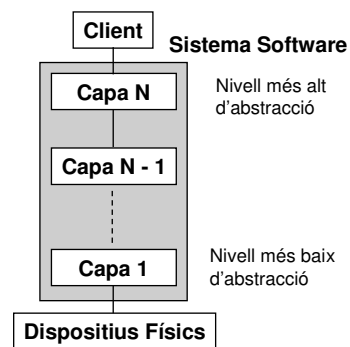
Un sistema gran que requereix ésser descomposat en grups de subtasques, tals que cada grup de subtasques està a un nivell determinat d'abstracció.

## Problema

- Cal dissenyar un sistema amb la característica dominant d'incloure aspectes d'alt i baix nivell (d'abstracció), on les tasques d'alt nivell es basen en les de baix nivell.
- Les tasques d'alt nivell no es poden implementar utilitzant directament els serveis de la plataforma, a causa de la seva complexitat. Calen serveis intermediaris.
- El sistema requereix també una estructuració horitzontal.
- Forces a equilibrar:
  - Canvis en el codi no haurien de propagar-se en tot el sistema (mantenibilitat)
  - Els components s'haurien de poder reutilitzar i reemplaçar per implementacions alternatives (reusabilitat)
  - Responsabilitats semblants s'haurien d'agrupar per afavorir la comprensibilitat i mantenibilitat
  - Es desitja portabilitat a d'altres plataformes
  - La definició de serveis intermedis pot exigir més crides a procediments (eficiència)

## Solució

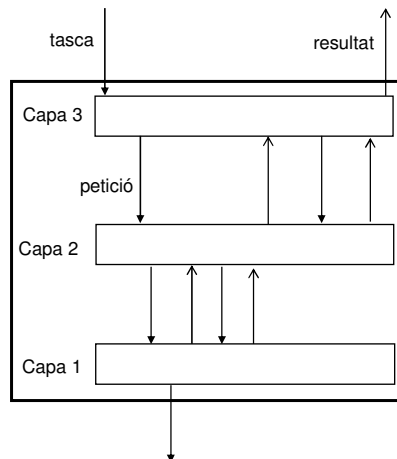
- Estructurar el sistema en un nombre apropiat de capes.
- Col·locar les capes verticalment.
- Tots els components d'una mateixa capa han de treballar al mateix nivell d'abstracció.
- Els serveis que proporciona la capa **j** utilitzen serveis proporcionats per la capa **j - 1**. Alhora, els serveis de la capa **j** poden dependre d'altres serveis en la mateixa capa.



## Solució: Comportament

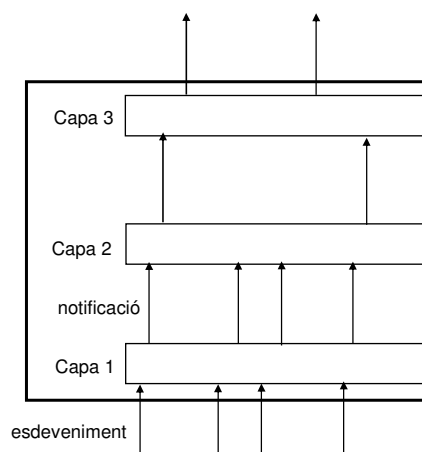
### Comunicació de dalt cap a baix

Un usuari realitza una petició d'una tasca a la capa superior i rep un resultat



### Comunicació de baix cap a dalt

Un dispositiu físic detecta l'ocurrència d'un esdeveniment a la capa inferior i ho notifica a les superiors



## Consideracions a la definició de l'arquitectura

- Definir el criteri d'abstracció i el nombre capes
- Especificar els serveis de cada capa:
  - Cap component pot estar repartit en dues o més capes.
  - Capa superior conté serveis a nivell de sistema.
  - Pocs serveis en les capes inferiors.
- Especificar una interfície per cada capa:
  - Normalment, capa com a caixa negra per la capa superior
  - Alternatives: caixa blanca, caixa gris
- Estructurar les capes individualment
- Especificar la comunicació entre capes adjacents:
  - Model "empenta": la informació es comunica en la petició del servei
  - Model "estirada": el servei demanat estira la informació de la capa superior
- Dissenyar una estratègia de tractament d'errors.
  - Tractar errors en la capa on es detecten o a capes superiors

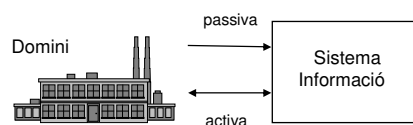
## Beneficis i Inconvenients

- Beneficis:
  - Canviable, Reusable, Portable, Provable
- Inconvenients:
  - Eficiència
  - Feina innecessària o redundant
  - Dificultat en establir la granularitat i nombre de capes



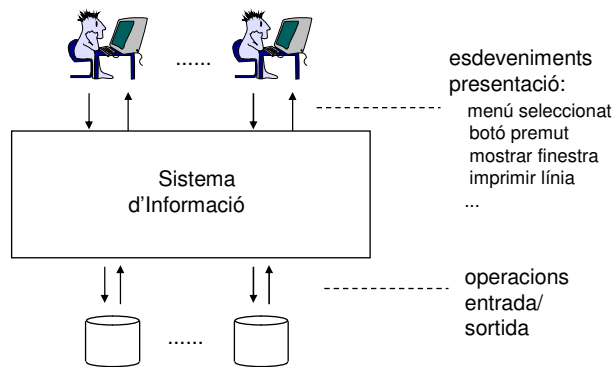
- **Arquitectura en capes relaxat**
  - Una capa pot usar els serveis de qualsevol capa inferior
    - . Tots o només part dels serveis de la capa (opacitat parcial)
  - Conseqüències:
    - . Possible guany en flexibilitat i eficiència
    - . Possible pèrdua en la canvialilitat, reusabilitat

## Aplicació als Sistemes d'Informació

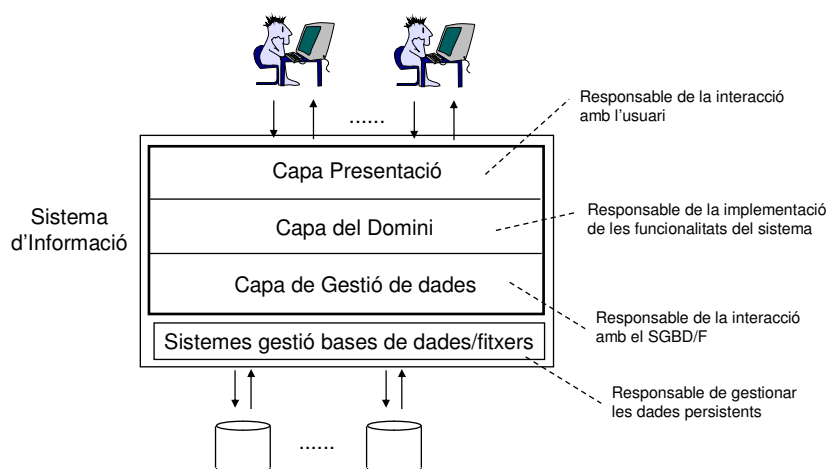


- Funcionalitat passiva del SI:
  - Mantenir una representació consistent de l'estat del domini:
    - . Capturar els esdeveniments que ocorren al domini
    - . Actualitzar l'estat del sistema d'informació com a conseqüència d'aquests esdeveniments
    - . Assegurar la consistència de la representació
- Funcionalitat activa del SI:
  - Respondre a consultes sobre l'estat del domini.
  - Produir reaccions quan es donen certes condicions predefinides.

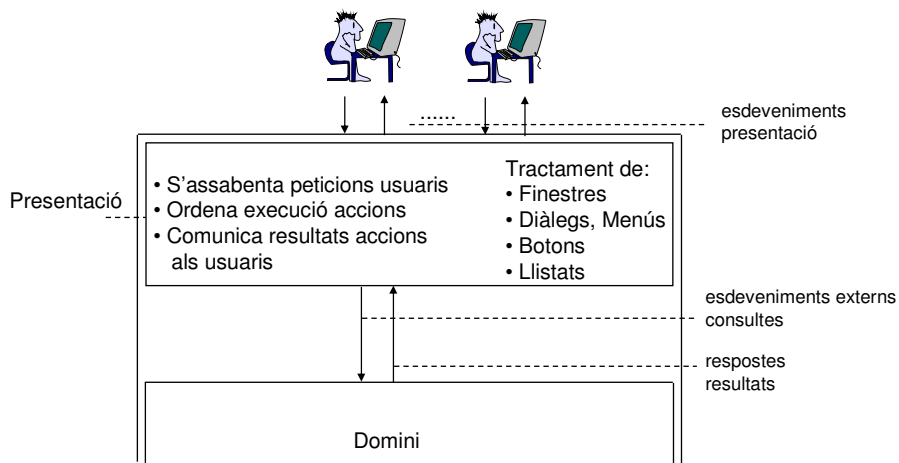
## Context dels Sistemes d'Informació



## Arquitectura en capes d'un Sistema d'Informació: Definició de criteris d'abstracció i nombre de capes

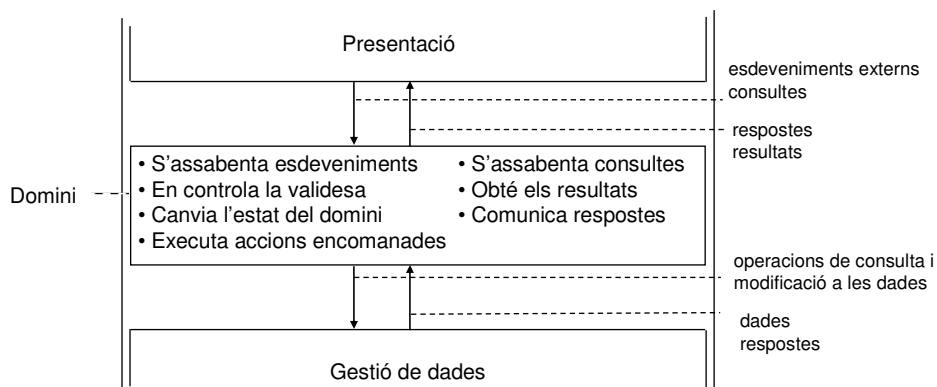


### Arquitectura en tres capes d'un Sistema d'Informació: Capa de Presentació



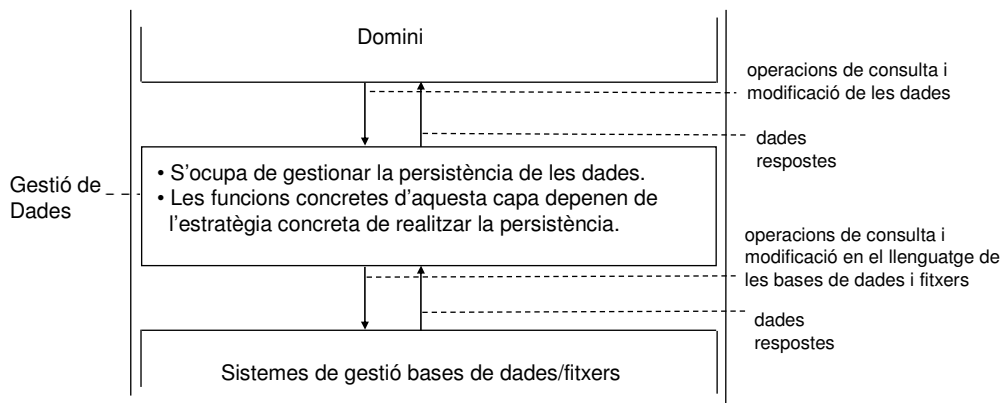
La Capa de Presentació coneix com presentar les dades a l'usuari, però ignora quines transformacions cal fer per donar resposta a les peticions de l'usuari

### Arquitectura en tres capes d'un Sistema d'Informació: Capa del Domini



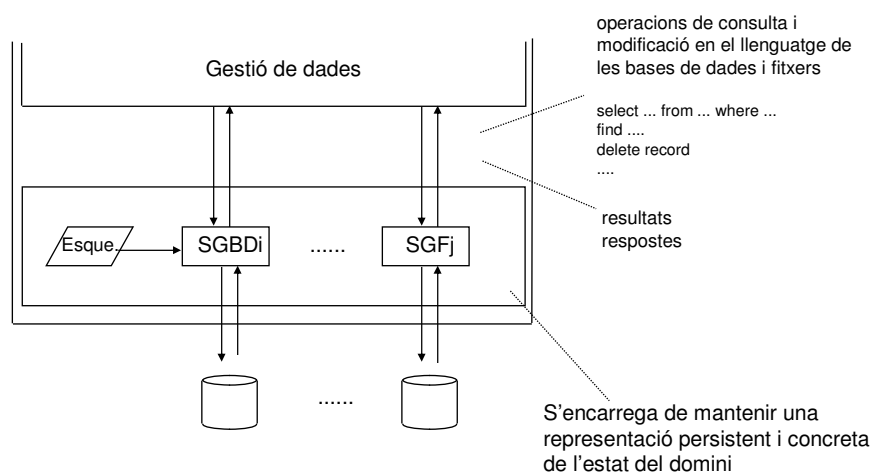
La Capa de Domini coneix com satisfer les peticions de l'usuari, però ignora on es guarden les dades i com es presenten a l'usuari

### Arquitectura en tres capes d'un Sistema d'Informació: Capa Gestió de Dades

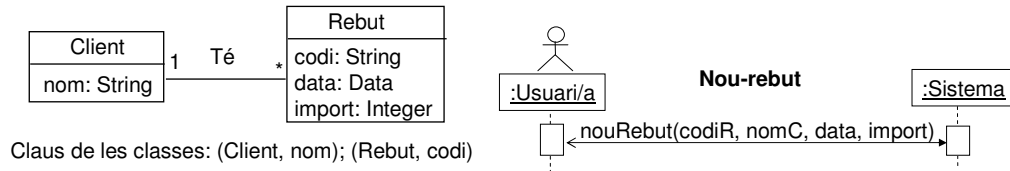


La capa de Gestió de Dades coneix on i com estan emmagatzemades les dades, però desconeix com tractar-les

### Arquitectura en tres capes d'un Sistema d'Informació: Els sistemes de gestió de bases de dades/fitxers

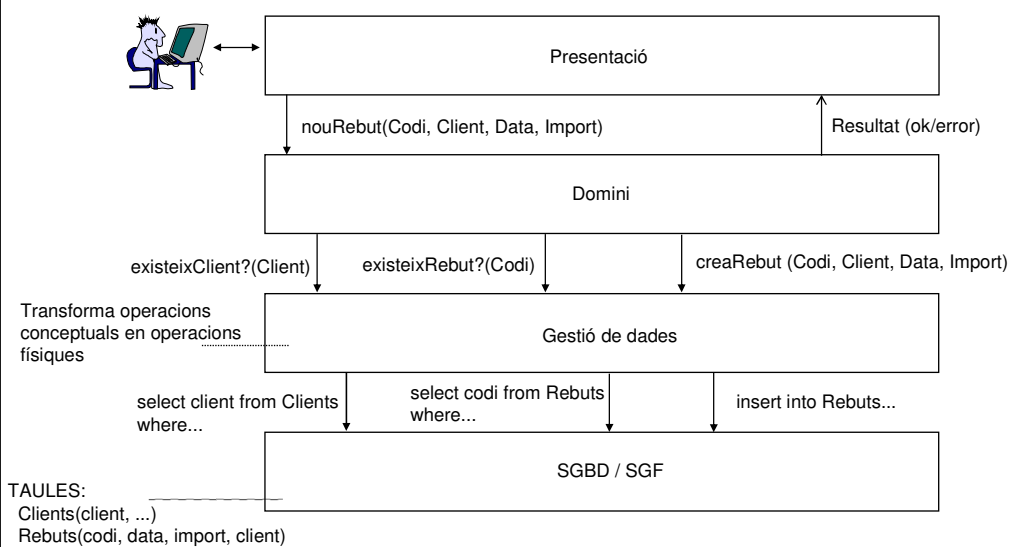


### Exemple: models de partida



**context** nouRebut (codiR: String, nomC: String, data: Data, import: Integer)  
 -- enregistrar un nou rebut per un client  
**pre:** 1.1 existeix el client amb nomC  
**post:** 2.1 alta d'una instància de rebut R amb R.codi=codiR, R.data=data i R.import=import  
 2.2 alta d'una instància de l'associació 'Té' que associa el rebut R i el client amb nom=nomC.

### Exemple: comunicació entre capes

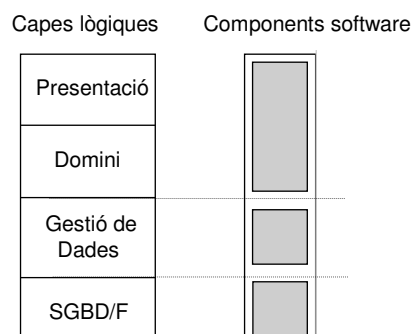




### Vista de desenvolupament de l'arquitectura en tres capes

- Configuració de l'arquitectura en components software
  - Com es distribueixen els elements de disseny en components software?
- De vegades poden fusionar-se capes adjacents:
  - Millora d'eficiència
  - Es perjudiquen la canviabilitat, la reusabilitat i la portabilitat
  - Possibilitats: a nivell de paquet, de mòdul o fins i tot de mètode
- Opcions:
  - Fusionar capa de domini i de gestió de dades
  - Fusionar capa de presentació i de domini
  - Fusionar les tres capes

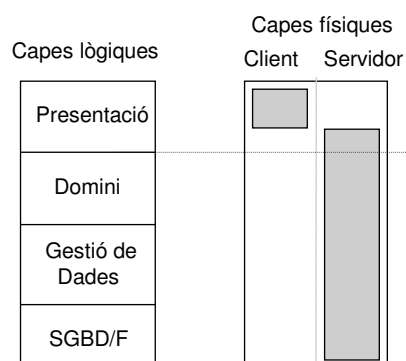
### Exemple de vista de desenvolupament



### Vista física (desplegament) de l'arquitectura en tres capes

- Desplegament de l'arquitectura (assignació d'elements de disseny a elements hardware)
  - Com es distribueixen físicament les capes?
- De vegades poden fusionar-se capes adjacents i/o dividir capes per tal de:
  - Assolir els requeriments no funcionals
- Opcions:
  - Totes les capes estan a la mateixa localització física (usualment, sistema centralitzat)
  - Les capes estan en diferents localitzacions físiques (sistema distribuït)
    - . Cada capa en una localització diferent
    - . Dues capes en una mateixa localització i l'altre en una localització diferent
    - . Repartició d'una o més capes en localitzacions diferents

### Exemple de vista física



Exemple: Sistema web

### **Arquitectura en capes d'un Sistema d'Informació: Especificació de les capes**

- Punt de partida
  - Especificació funcional
  - Tecnologia (llenguatge de programació orientat a objectes, bases de dades relacional)
  - Requisits no funcionals (canviabilitat, eficiència, portabilitat, reusabilitat)
- Cal assignar les responsabilitats que es deriven dels contractes a les diferents capes i agrupar-les en operacions
- Dues grans estratègies segons la gestió de la persistència
  - Generació automàtica → la gestió de la persistència és transparent
  - Disseny directe → aprofitament de la potència de la base de dades
  - El disseny de la capa de presentació no es veu pràcticament afectat
- A l'assignatura estudiarem les dues estratègies

### **Bibliografia**

- *Pattern-oriented Software Architecture. A System of patterns*  
F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal  
John Wiley & Sons, 1996. Pàgines 31-51
- *Analysis Patterns*  
Martin Fowler  
Addison-Wesley, 1997, cap. 12.