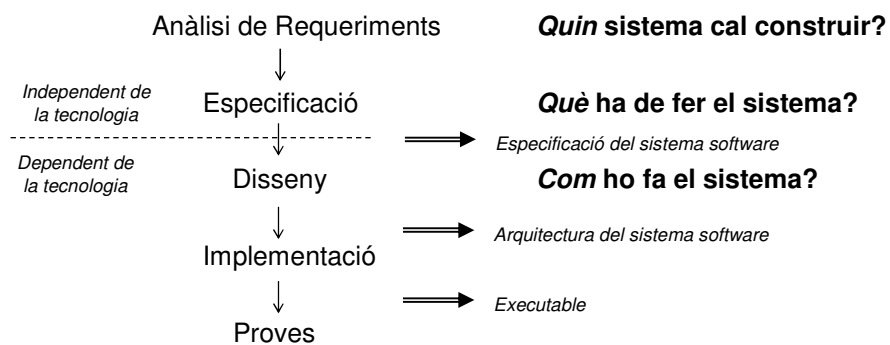


Introducció al disseny de software

- Etapes del desenvolupament de software
- Disseny de software
- Disseny amb patrons
 - Patró arquitectònic en capes
 - Patró arquitectònic orientat a objectes
 - Disseny en UML
- Bibliografia

Etapes del desenvolupament de software



Disseny de software

- **Disseny de software** és l'activitat d'aplicar diferents tècniques i principis amb el propòsit de definir un sistema amb el suficient detall per permetre la seva construcció física (implementació)
- Punt de partida:
 - Resultat de l'especificació: (Què ha de fer el sistema?)
 - Especificació de dades (Model de Dades)
 - Especificació de processos (Models Funcional i de Comportament)
 - Requeriments no funcionals del sistema
 - Tecnologia (Amb quins recursos?)
 - Recursos hardware i software disponibles
- Resultat del disseny: (Com ho fa el sistema?)
 - Arquitectura del Sistema Software: descripció dels subsistemes i components d'un sistema software, i les seves relacions
- Procés del disseny:
 - Metodologies de disseny
 - Adaptació de solucions genèriques a problemes de disseny (Patrons de disseny)

Arquitectures software: Vistes

Els subsistemes i els components s'especifiquen normalment en diferents vistes:

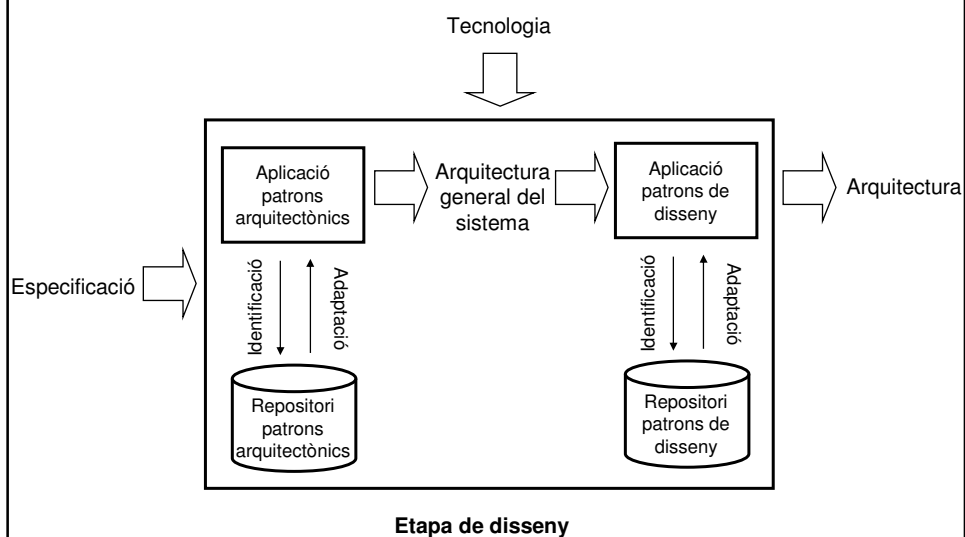
- Vista lògica: organització del sistema en classes
- Vista de processos: descripció de fluxs de control de dades
- Vista física (desplegament): assignació dels elements de disseny a elements hardware
- Vista de desenvolupament: encapsulament dels elements de disseny en mòduls
- Vista d'escenaris (casos d'ús): mostra els usos més significants del sistema i com afecten a les altres vistes
 - articula les altres 4 vistes (se'n diu 4+1)

Altres autors defineixen altres vistes: de dades, de seguretat, etc.

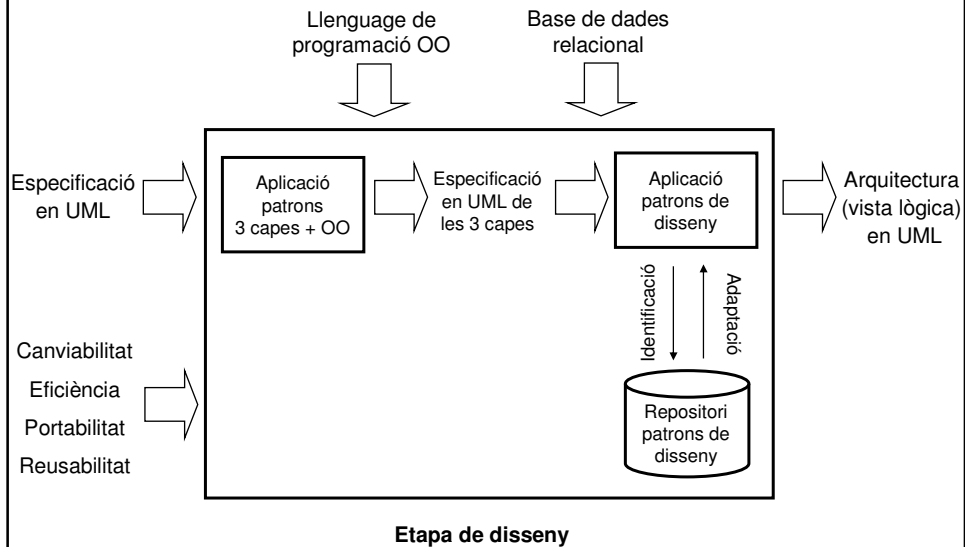
Arquitectures software: Factors de qualitat

- Externs: observables per l'usuari
- Interns: observables pel professional informàtic
 - Els factors interns són un mitjà per assolir els externs
- (Un) Catàleg de factors de qualitat externs (Meyer):
 - Fiabilitat (correctesa + robustesa)
 - Modularitat (extensibilitat + reusabilitat)
 - Compatibilitat
 - Eficiència
 - Portabilitat
 - Facilitat d'ús
 - Funcionalitat
 - Puntualitat
 - Verificabilitat
 - Integritat
 - Reparabilitat
 - Economia
- Existeixen múltiples interrelacions entre ells
 - Sinèrgies positives: un factor afavoreix un altre
 - Conflictes: un factor perjudica un altre
- En un procés de disseny, ens interessa obtenir architectures tals que els seus factors de qualitat es corresponguin amb els requisits no funcionals establerts

Disseny amb patrons

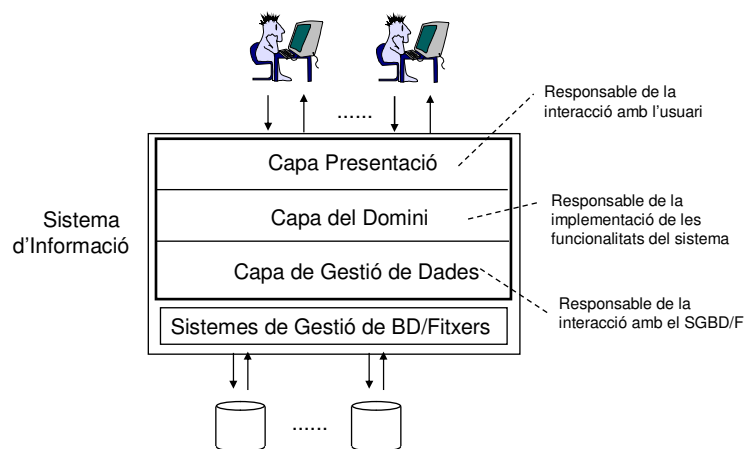


Context de l'assignatura ES2



Patró arquitectònic en capes

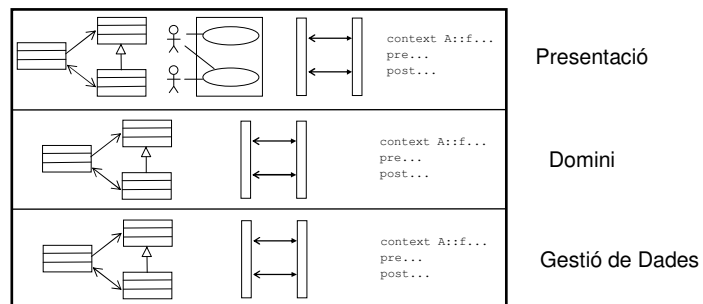
Aplicació del patró "Arquitectura en 3 Capes" a un Sistema d'Informació



Patró arquitectònic d'orientació a objectes

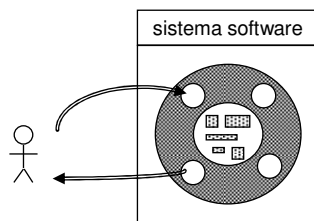
Aplicació del patró "Orientació a Objectes" a cada capa, que es dissenya independentment

Useu UML com a llenguatge de disseny: utilitzem els mateixos models al disseny que a l'especificació

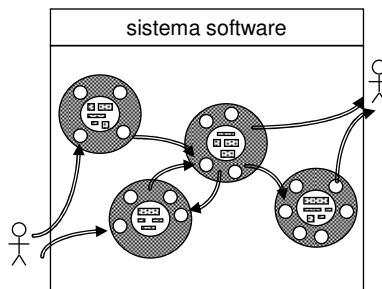


Disseny orientat a objectes Visió d'un sistema software

Especificació



Disseny



Especificació: el sistema software es veu com una sola classe d'objectes que engloba tota la informació i totes les operacions.

Disseny: cada classe d'objectes té les seves pròpies operacions de manipulació d'informació. Els objectes interactuen per satisfer les operacions del sistema.

Disseny orientat a objectes en UML

Punt de partida: especificació en UML

- **Especificació:** *QUÈ* fa el sistema software?
- **Resultat de l'especificació en UML:**
 - Model de Casos d'Ús:
 - Quina interacció hi ha entre els actors i el sistema software?
 - Esquema Conceptual:
 - Quins són els conceptes rellevants del món real de referència ?
 - Diagrames de seqüència del sistema:
 - Quina resposta dona el sistema als esdeveniments externs? (quines operacions ha de tenir el sistema?)
 - Contractes de les operacions:
 - Què fan les operacions del sistema?
 - Diagrama d'estats:
 - Per quins estats evolucionen els objectes del sistema?

Disseny en UML

Resultat a assolir: disseny en UML

- **Disseny:** *COM* estructurarem el sistema perquè faci el que ha de fer?
⇒ *el disseny és una activitat iterativa i és difícil seqüencialitzar tot el que s'hi fa.*
- **Resultat del disseny en UML:**
 - Model de casos d'ús:
 - Defineix la interacció real, amb una interfície concreta.
 - Diagrama de classes de disseny:
 - Descriu les classes del software i les seves interfícies (operacions).
 - Diagrames de seqüència:
 - Defineixen la interacció entre les classes d'objectes per respondre un esdeveniment extern.
 - Contractes de les operacions:
 - Defineixen què fan les operacions de les classes d'objectes.
- **Altres artefactes complementaris:**
 - Esquema de la base de dades:
 - Defineix les taules que configuren la base de dades del sistema d'informació.
 - Disseny extern de la interfície:
 - Defineix els elements de disseny de la interfície, referits des dels casos d'ús reals.

Bibliografia

- *Software Engineering. A Practitioner's Approach*
R.G. Pressman
McGraw Hill, 2005 (Sisena edició), caps. 9 i 10
- *Enginyeria del software: Especificació*
D. Costal, X. Franch, M.R. Sancho, E. Teniente
Edicions UPC, 2004
- *Object-Oriented Software Construction*
B. Meyer
Prentice Hall, 1997, cap. 1
- *Applying UML and Patterns*
C. Larman
Prentice Hall, 2005 (Tercera edició), caps. 33, 34 i 39
- The 4+1 View Model of Architecture
P. Krutchen
IEEE Software, 12(6), Nov. 1995, pp. 42-50