

Patrons GRASP

Toni Sellarès
Universitat de Girona

Reponsabilitats i Mètodes

Un cop s'han identificat els **requeriments** del Sistema a desenvolupar i s'ha creat un **model del domini**, cal afegir els **mètodes a les classes** del domini i definir els **missatges** que els objectes s'hauran d'intercanviar per tal d'assolir els requeriments.

L'etapa d'**afegir mètodes** a les classes és una de les més crítiques del Disseny del Software.

Reponsabilitat no és el mateix que **mètode**, pero els dos conceptes estan relacionats.

Les responsabilitats s'implementen amb mètodes.

Frequentment els mètodes han de **col·laborar** amb altres mètodes i objectes.

Qu  cal fer ?

1. Seleccionar un Diagrama de Sequ ncia Simplificat (del Sistema) corresponent a un escenari d'un Cas d' s.
2. Seleccionar el primer missatge que indica una Responsabilitat del Sistema.
3. Identificar les Classe apropiades Software pera a implementar la Responsabilitat.
4. Repetir el proc s amb el missatge seg ent.
5. Revisar les assignacions. A l'assignar una nova Responsabilitat poden haver canbiat les regles fetes per a altres assignaments.

PATRONS DE DISSENY

Els patrons de disseny s n descripcions (amb nom) de problemes espec fics que van acompanyades de les seves solucions.

Els patrons de disseny capten l'ess ncia de principis b sics que han estat aplicats amb  xit.

L'estructura d'un patr   s:

- Nom del patr  :
- Problema que resol:
- Soluci  :

Patrons GRASP

(General Responsibility Assignment Patterns)

Els patrons GRASP descriuen els principis a utilitzar en el procés de disseny de classes i d'assignació de responsabilitats a aquestes classes.

S'apliquen durant la creació de diagrames d'interacció.

Els cinc patrons GRASP més àmpliament utilitzats són: Expert, Creador, Controlador, Baix Acoblament i Alta Cohesió.

Altres patrons GRASP: Fabricació Pura, ...

Patró Expert

Problema: quin és el principi general a l'hora d'assignar responsabilitats als objectes ?

Solució: Assignar responsabilitat a l'expert en la informació, és a dir a la classe que posseeix la informació necessària per a complir amb una responsabilitat

El primer pas en l'aplicació d'aquest patró és buscar les classes expertes.

El compliment d'una responsabilitat sovint requereix informació que està distribuïda entre diverses classes d'objectes. En conseqüència hi haurà diversos experts parcials que col·laboraran en la tasca.

Patró Expert

L'utilització del patró expert permet mantenir l'encapsulació de la informació ja que els objectes fan servir la seva pròpia informació per a assolir les tasques. Com a conseqüència, normalment hi ha baix acoblament.

El comportament es distribueix entre les classes que tenen la informació requerida. Això permet una definició cohesionada de les classes que generalment és fàcil d'entendre i mantenir.

Expert ha de ser el primer patró considerat en qualsevol disseny a no ser que el disseny impliqui un problema de control o creació.

Patró Creador

Problema: quina classe ha de ser la responsable de crear noves instàncies d'altres classes.

Solució: assignar la responsabilitat de crear instàncies de la classe A a la classe B si es compleixen una més de les següents condicions:

- la classe B "agrega" objectes de tipus A ;
- B conté objectes de tipus A;
- B "enregistra" instàncies de tipus A;
- B fa servir "íntimament" objectes de tipus A;
- B té la informació d'inicialització d'un objecte de tipus A quan aquest és creat.

Amb la correcte utilització d'aquest patró s'obté baix acoblament, per tant es facilita el manteniment i es creen oportunitats de reutilització.

Patró Controlador

Problema: qui ha de tenir la responsabilitat de gestionar un esdeveniment del sistema.

Solució: assignar la responsabilitat de gestionar els esdeveniments del sistema a una classe que representi:

- tot el "sistema" (facade controller);
- un gestor artificial de tots els esdeveniments d'un cas d'ús.

És desitjable fer servir el mateix classe controlador per a tots els esdeveniments del sistema relacionats amb un cas d'ús.

Una classe controlador delega als altres objectes l'execució del treball a fer i ella s'ocupa de coordinar o controlar l'activitat.

La utilització del patró controlador potencia la creació d'interfícies reutilitzables i connectables.

Permet separar la interfície de presentació, la lògica del diàleg i la lògica del negoci.

Patró Baix Acoplament

Problema: com aconseguir abaixar la dependència, disminuir l'impacte dels canvis i incrementar la reutilització.

Solució: assignar responsabilitats de manera que l'acoblament es mantingui baix.

L'acoblament mesura en quin grau un element està connectat o té coneixement d'altres elements.

Una classe amb baix acoblament no queda afectada per canvis en altres components, i per tant és fàcil d'entendre el seu comportament i de ser reutilitzada.

Una subclasse està molt fortament lligada a les seves superclasses. Cal meditar molt bé la decisió de derivar una classe d'una superclasse, ja que comporta un alt acoblament.

A la pràctica el nivell d'acoblament no es pot considerar aïllat i cal tenir en compte altres patrons, com Expert i Alta Cohesió.

Patró Alta Cohesió

Problema: com aconseguir que la complexitat sigui gestionable ?

Solució: assignar les responsabilitats de manera que la cohesió es mantingui alta.

La cohesió mesura en quin grau les responsabilitats estan relacionades i focalitzades amb un element.

Una classe amb alta cohesió té relativament pocs mètodes amb funcionalitats molt relacionades i no executa massa treball. Col·labora amb altres classes per a compartir l'esforç a l'hora de fer una tasca.

Una classe amb baixa cohesió fa moltes coses no relacionades o fa massa treball. Aquests tipus de classes present els següents problemes: són difícils d'entendre, són difícils de reutilitzar i estan afectades constantment pels canvis.

A la pràctica el nivell de cohesió no es pot considerar aïllat i cal tenir en compte altres patrons, com Expert i Baix Acoblament.

És recomanable dissenyar les classes d'un sistema de forma que presentin alta cohesió i baix acoblament. Generalment l'alta cohesió i el baix acoblament van lligats.

Patró Fabricació Pura

Problema: Què cal fer quan les solucions de disseny trobades comprometen la cohesió i l'acoplament?

Solució: Assignar un conjunt cohesionat de responsabilitats a una classe **artificial** (no representa cap concepte del domini del problema i està dissenyada exclusivament per a aquesta finalitat)

Avantages

- Millora la cohesió eliminant responsabilitats a altres classes.
- Permet bon acoplament perquè fa servir un conjunt cohesionat de responsabilitats

Contraindicacions

- Cal evitar-ne l'ús abusiu.