UML2

Package di Analisi

Andrea Polini

Laboratorio di Ingegneria del Software
Corso di Laurea in Informatica – L-31
Università di Camerino

Package

Entità di raggruppamento del linguaggio che fornisce un meccanismo generalizzato per organizzare e raggruppare nomi.

Uso

- fornire spazio di nomi in cui le etichette saranno univoche
- raggruppare elementi semanticamente correlati
- creare confini semantici interni al modello
- definire unità di lavoro che possano procedere in parallelo

Package

Entità di raggruppamento del linguaggio che fornisce un meccanismo generalizzato per organizzare e raggruppare nomi.

Uso

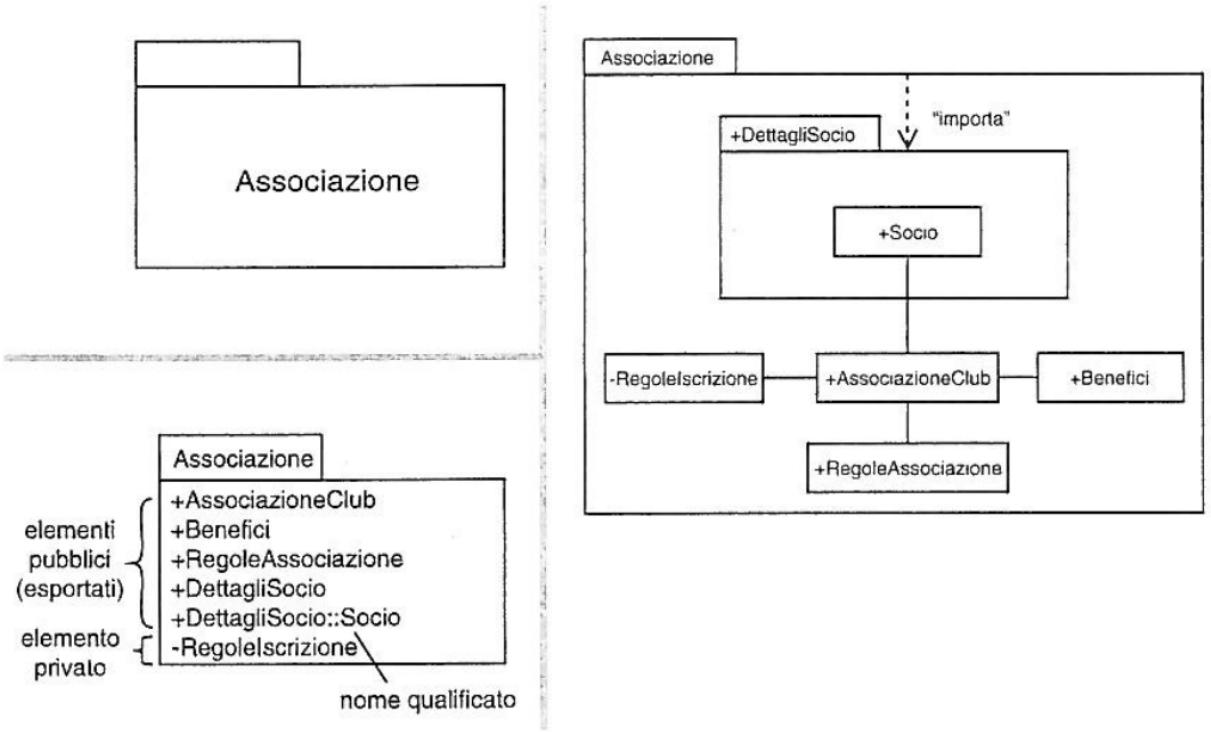
- fornire spazio di nomi in cui le etichette saranno univoche
- raggruppare elementi semanticamente correlati
- creare confini semantici interni al modello
- definire unità di lavoro che possano procedere in parallelo

Package di analisi

Possono contenere:

- casi d'uso
- classi di analisi
- realizzazione dei casi d'uso

Il simbolo utilizzato per indicare un package richiama una cartella. Un package può contenere elementi sia accessibili dall'esterno che privati.



Package: dipendenze e stereotipi

Quando?

In genere la scomposizione in package avviene quando sono state identificate un buon numero di classi di analisi e di casi d'uso.

È importante ridurre le dipendenze tra package ovvero è meglio avere pochi elementi pubblici o protetti

UML fornisce due stereotipi per il package:

- `<<framework>>`: architettura riutilizzabile
- `<<modelloLibreria>>`: elementi da riusare in altri package

Package: dipendenze e stereotipi

Quando?

In genere la scomposizione in package avviene quando sono state identificate un buon numero di classi di analisi e di casi d'uso.

È importante ridurre le dipendenze tra package ovvero è meglio avere pochi elementi pubblici o protetti

UML fornisce due stereotipi per il package:

- `<<framework>>`: architettura riutilizzabile
- `<<modelloLibreria>>`: elementi da riusare in altri package

Annidamento di package

Il linguaggio permette di annidare i package. Possono utilizzarsi due possibili notazioni per rappresentare la relazione:

- notazione a “matrioska”
- notazione con relazione di contenimento

Il concetto di annidamento porta con se specifiche regole di accesso agli elementi tra package contenuti/contenenti

Dipendenze tra package

UML definisce 5 diversi tipi di dipendenza tra i package:

- «usa»: relazione tra gli elementi non tra package
- «importa»: fusione publica dello spazio dei nomi
- «accede»: fusione privata dello spazio dei nomi
- «traccia»: relazione tra modelli e non elementi nello stesso modello
- «incorpora»: solo per “meta-modeling”

Transitività

Attenzione la dipendenza importa risulta essere transitiva mentre questo non vale per la dipendenza accede

Generalizzazione tra package

La generalizzazione tra package permette di definire package ereditando da package già definiti (simile a quanto avviene per le classi). Un package eredita tutti i membri protetti e pubblici del genitore.

Un package che erediti può dunque:

- aggiungere
- ridefinire

Relazione più forte delle altre che si esplica nel principio di sostituibilità come per le classi.

Package di analisi

Importante minimizzare le dipendenze tra gli elementi di un sistema

- minimizzare le dipendenze tra package
- minimizzare il numero degli elementi pubblici o protetti
- massimizzare il numero di elementi privati

Individuare package di analisi

Dal modello statico si cerca di identificare aggregati coesi nei diagrammi delle classi:

- gerarchia: dipendenze, aggregazione, composizione, ereditarietà
- in particolare si considerino relazioni di ereditarietà

Successivamente si tenta di minimizzare (tenendo conto che in genere un numero superiore alla decina di classi per package può essere un indicazione ragionevole e che è meglio ridurre annidamento. Inversamente package con meno di 4 classi suggeriscono accorpamento).

Strategie per la gestione delle dipendenze

Come si può agire per ridurre le dipendenze:

- spostando classi tra i package
- aggiungendo package
- eliminando package

Attenzione al caso delle dipendenze circolari tra i package

Come è possibile rimuovere dipendenze di creazione?

Strategie per la gestione delle dipendenze

Come si può agire per ridurre le dipendenze:

- spostando classi tra i package
- aggiungendo package
- eliminando package

Attenzione al caso delle dipendenze circolari tra i package

Come è possibile rimuovere dipendenze di creazione?