

Test 07 – Object Design

Gli UML Deployment diagram sono utili per mostrare il progetto del sistema dopo che le seguenti decisioni sono state prese:

- Decomposizione in sottosistemi
- Consistenza dei nodi
- Mapping Hardware/Software

Cosa è l'interfaccia di un sottosistema?

- Un insieme di operazioni con signature completamente specificata
- Un gruppo di operazioni che condividono uno scopo comune
- Un insieme di associazioni, eventi e vincoli legati tra di loro

Quali delle seguenti attività non rientrano in quelle da fare per la specifica delle interfacce, durante l'object design:

- Identificare attributi e operazioni mancanti
- Modificare le signature e la visibilità di ogni operazione
- Controllare le precondizioni

Se il mio obiettivo è quello di realizzare una architettura chiusa

- Uso un bridge pattern
- Uso un adapter pattern
- Uso un façade pattern

Quale di queste non è una attività di ottimizzazione effettuata durante la fase di object design e implementazione del sistema?

- Trasformazione di oggetti in attributi
- Ristrutturazione della gerarchia delle classi
- Memorizzazione di attributi derivati

Aggiungere associazioni ridondanti ad un diagramma delle classi in fase di object design serve a:

- Memorizzare attributi derivati
- Ottimizzare cammini di accesso
- Ritardare calcoli costosi

Se devo effettuare una trasformazione prima sul modello a oggetti e poi ottengo il codice, faccio

- Prima operazioni di forward engineering e poi di refactoring
- Prima operazioni di object model transformation e poi di forward engineering
- Prima operazioni di refactoring e poi di reverse engineering

[Mappare le relazioni di ereditarietà quando usiamo database relazionali-Mapping Verticale]

Quale delle seguenti affermazioni sono corrette utilizzando una tabella separata:

- E' più complicato aggiungere un attributo alla superclasse aggiungendo una colonna alla tabella superclasse;
- Ricercare tutti gli attributi di un oggetto richiede una operazione di Join.
- Aggiungere una sottoclasse significa aggiungere una tabella per la sottoclasse con un attributo per ogni colonna della sottoclasse;

Come vanno implementate le associazioni “multi-a-molti” tra oggetti persistenti in un DB relazionale ?

- Con una chiave esterna inserita nelle due tabelle corrispondenti alle classi dell’associazione
- Con una chiave esterna inserita in una delle due tabelle corrispondenti alle classi dell’associazione
- Con una ulteriore tabella oltre a quelle corrispondenti alle classi dell’associazione

Quando introduciamo una dipendenza in un component diagram, quali delle seguenti affermazioni sono false:

- La dipendenza non può essere transitiva
- I cicli non sono ammessi
- Quando una componente è modificata, altre componenti che dipendono da questa potrebbero richiedere modifiche

In che modo possono essere specificati i contratti in UML?

- In Javadoc
- In Object Constraint Language
- In Object Contract Language

Se devo effettuare una trasformazione prima sul codice sorgente e poi sul modello a oggetti faccio

- Prima operazioni di forward engineering e poi di refactoring
- Prima operazioni di object model transformation e poi di forward engineering
- Prima operazioni di refactoring e poi di reverse engineering

Se il mio obiettivo è fornire implementazioni diverse per uno stesso sottosistema

- Uso un bridge pattern
- Uso un adapter pattern
- Uso un façade pattern

Elencare e descrivere i tipi di trasformazione che possono essere impiegate nello sviluppo del software

Descrivere i tre meccanismi principali per il (Global) Software Control (tra le attività del System Design)

Elencare le attività da fare per la specifica delle interfacce, durante l'object design

Elencare e descrivere le euristiche per effettuare le trasformazioni nell'object design

Elencare le attività di ottimizzazione dell'object design

Descrivere brevemente i tre tipi di collezioni in OCL.

Descrivere il Proxy pattern.

Descrivere il design pattern Observer.

Descrivere l'Adapter pattern.

Descrivere il Bridge pattern.

Descrivere il Composite Pattern
