Pattern di Progettazione

Primo Livello di una Linea di Prodotti

Architettura con Tre Strati Principali ...

- **Presentazione**. La logica di gestione dell'interfaccia esterna per rendere un servizio da parte dell'applicazione ad altri sistemi, umano o programma remoto. Il sistema utilizzatore potrebbe richiedere interfacce di diversi livelli di complessità. Lo scopo della logica realizzata in questo strato è articolato in:
  - interpretare le richieste di servizio dell'utilizzatore;
  - raccolta dei dati e
  - restituzione dei risultati.

  Può utilizzare interfaccia grafica, HTML...

... Architettura con Tre Strati Principali ...

- **Sorgenti di Dati.** La logica di gestione dell'interfaccia con sistemi che rendono servizi all'applicazione. Possono essere monitor di transazioni, sistemi di messaggi... .
- Per molte applicazioni la gran parte di questa logica è nel trattamento della base di dati che ha la responsabilità di raccogliere, organizzare e mettere a disposizione i dati persistenti.

… Architettura con Tre Strati Principali

⇨ **Dominio.** Logica di business, ovvero tutte le funzioni software necessarie per supportare il business destinatario dell'applicazione:

- calcoli basati su dati di input e dati archiviati,
- validazione dei dati, secondo la logica di business, che vengono dalla presentazione,
- utilizzo della logica della sorgente di dati per richiedere i dati necessari alla costruzione delle informazioni da dare ai sistemi che ne fanno richiesta…

I Package e le Varianti …

⇨ Una linea di prodotti spesso ha, per ogni strato, molti package, ognuno composto da componenti, che hanno uguale scopo ma lo realizzano in modo diverso:

- La presentazione può prevedere un'interfaccia grafica od un' interfaccia a comando o un'interfaccia Web;

… I Package e le Varianti

- La sorgente di dati può avere packages diversificati per diversi tipi di data base o diversi progetti di uno stesso data base;
- Il dominio può essere diviso in problemi differenti ed ogni problema può essere risolto con modalità differenti dipendenti dai destinatari dell'applicazione.

Architettura Fisica…

⇨ Ogni strato può essere realizzato per funzionare:

- Completamente su Server, per facilitare i cambiamenti del software e l'amministrazione del sistema :
  - I servizi possono essere fruiti su sistemi distribuiti via HTML; ogni piccola decisione dell'utilizzatore deve essere elaborata dal server ma i clients sono completamente indipendenti dai server
  - Possono essere costruiti degli applet scaricabili sul client la dipendenza dal server rimarrebbe inalterata ma il client deve essere più potente

…Architettura Fisica …

- Parte su Server e parte sul Client; il vantaggio di questa divisione è la reattività e il funzionamento del sistema anche con il server sconnesso

…Architettura Fisica

- Ibrida. Un servizio sul Server è realizzato come un "documento" composto da parti elaborabili dal server e parti elaborabili dal client. Ogni volta che viene richiesto il sevizio il Server trasmette il documento corrispondente con: i risultati dell'esecuzione delle parti elaborabili dal server e le parti che devono essere elaborate dal client (Esempio: ASP, PHP…). Il client deve avere configurazione dipendente dalle richieste delle parti che deve elaborare. Il client risulta indipendente dal server.

PATTERNS ARCHITETTURALI

Motivazioni

- ⇨ Il progetto di sistemi software basato su componenti è complesso e difficile
- ⇨ È quindi opportuno riusare l'esperienza maturata in progetti precedenti
- ⇨ Una proposta per il trasferimento dell'esperienza è quella che fa uso di pattern

Definizione

- Un pattern architetturale descrive e risolve un problema di progettazione mediante macrostrutture di componenti
- Il pattern architetturale esprime, quindi, strutture organizzative fondamentali per configurare sistemi software complessi

Descrizione di un Pattern

- Obiettivo: descrizione sintetica del contenuto del pattern
- Come funziona : architettura del pattern e scopo delle componenti
- Quando si usa :Motivazione o problema che il pattern vuole risolvere
- Soluzione: come si progettano e come interagiscono le componenti del pattern

Patterns di Dominio

Decisioni per la Struttura del Dominio

...

- Il flusso di controllo nell'applicazione è complesso?
  - Application Controller

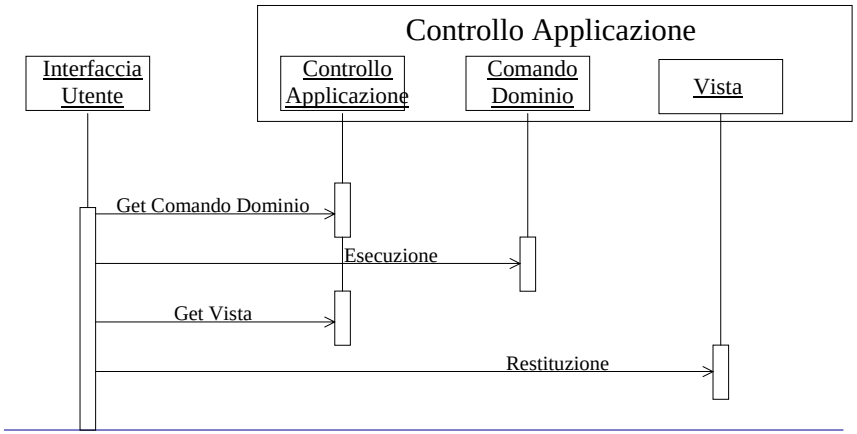
... Decisioni per la Struttura del Dominio

- Com'è opportuno strutturare la logica di Dominio?
  - Se è orientata alle transazioni ed è semplice
    - Transaction Script
  - Se è semplice ed è orientata alla elaborazione di records di dati
    - Table Module
  - Se è complessa
    - Domain Model

Application Controller: Obiettivo

- Isolare lo strato di dominio da quello di presentazione
- Avere un unico punto centrale in cui è gestito il flusso di controllo dell'intera applicazione
- Ha due scopi:
  - Decidere quale parte della logica di dominio deve essere eseguita
  - Come devono essere restituiti i dati risultati dalla elaborazione

Application Controller : Come funziona ...



...Application Controller : Come funziona ...

- Una Interfaccia Utente (WEB, Grafica, A Comando...) rileva la richiesta dell'utente e la comunica all'Application Controller
- L'application Controller restituisce il Comando di Dominio da attivare e la Vista

...Application Controller : Come funziona

- Il Comando di Dominio è utilizzato per attivare lo strato di dominio
- La Vista è utilizzata per attivare lo strato di Sorgenti di dati o di Presentazione, dipendentemente da come devono essere restituiti i dati

Application Controller : Quando si usa

- Quando l'ordine delle pagine di presentazione dell'applicazione è complesso, presenta viste di dati differenti dipendenti dallo stato degli oggetti
- Un sintomo che richiede l'uso di questo strato di software è:
  - La diffusione del flusso di controllo nell'applicazione; si rivela quando il cambio del flusso di controllo richiede molte modifiche nell'applicazione

Application Controller : Soluzione...

- 1. Formalizzare la logica dell'applicazione distinguendo la sequenza delle videate dal loro controllo: Il controllo lo si assegna all'Application Controller;
  - Spesso è necessario decidere se la logica di un comportamento deve essere assegnata all'Application Controller od al Dominio (es: I dati di recapito di uno studente devono essere rilevati se egli deve fare stage fuori Università; la sequenza di videate è condizionata dallo stato dello Studente; se questa condizione è molto diffusa nel dominio conviene metterla nell'Application Controller)

...Application Controller : Soluzione...

- 2. Se l'applicazione è molto articolata si può dividere per aree la U.I. e, di conseguenza, l'Application Controller
- 3. Se una variante è il tipo di interfaccia (WEB, Grafica, a comando...) allora:
  - Se la sequenza delle pagine è indipendente dal tipo di interfaccia si può utilizzare un unico Application Controller,
  - Se la sequenza deve essere differente per rendere più efficace il tipo di interfaccia allora è necessario utilizzare un Application Controller per ogni tipo di interfaccia

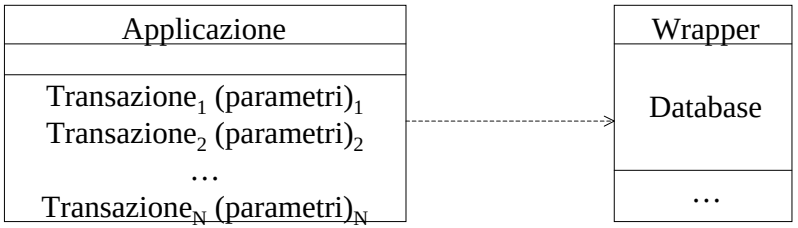
... Application Controller : Soluzione

- 4. Se la UI è progettata come una macchina a stati dove ogni evento richiede risposte differenti dipendenti dallo stato di alcuni oggetti chiave dell'applicazione allora l'Application Controller potrebbe utilizzare efficacemente i Metadati per rappresentare gli stati.
- 5. Collegare direttamente l'Application Controller alla UI, facendolo essere uno strato di collegamento tra Presentazione e Dominio

Transaction Scripts: Obiettivo

- Organizzare la logica di business per procedure;
- Ogni procedura risponde ad una singola richiesta dalla presentazione;
- Ad ogni procedura corrisponde una transazione

Transaction Scripts : Come funziona



- Ogni transazione comporta la interazione tra un client ed un server che può comportare:
  - Visualizzazione di dati in un data base
  - Un insieme di passi procedurali di validazione e di calcolo dati
- Una transazione è una procedura che chiama direttamente un database o wrapper leggeri di database

Transaction Scripts : Quando si usa

- La logica di business è semplice e può essere formalizzata con transazioni che hanno tra loro relazioni che possono essere espresse attraverso dati persistenti
- Quando la interazione tra le transazioni avviene anche attraverso la logica implementata in esse, è difficile fare un progetto semplice e comprensibile; inoltre c'è il rischio di duplicare codice comune a più transazioni
- Un esempio di applicazione semplice è la Gestione delle Prenotazioni in un Hotel: controllare la disponibilità delle camere richieste; determina il costo delle camere; aggiorna il database...

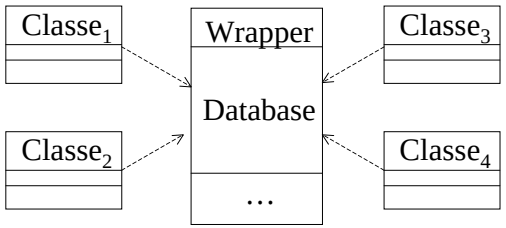
Transaction Scripts : Soluzione

- 1. Progettazione della base di dati normalizzata, partendo dal suo modello concettuale (E- R / Digramma delle Classi)
- 2. Progettazione della struttura dell'applicazione per transazioni
- 3. Progettazione di ogni transazione per moduli ottimizzando il riuso del codice (è opportuno intensificare il riuso di componenti del patrimonio di base e, quindi, anche di COTS)

Table Module : Obiettivo

- La logica di dominio è organizzata associando ad ogni tavola o vista del data base una classe
- La classe contiene tutti i metodi necessari al trattamento dei dati presenti nella tabella associata alla classe.

Table Module : Come funziona ...



- L'applicazione consiste in un insieme di oggetti che interagiscono tra loro attraverso le tavole;
- Il Wrapper nasconde: i metodi resi disponibili dal DBMS utilizzato per accedere fisicamente alle tavole od alle viste

... Table Module : Come funziona

- Se si deve trattare un Classe (Automobile) con molte proprietà (Targa, Modello, Prezzo, Costruttore, Parti Meccaniche); il Dominio Applicativo tratta un'unica classe (Automobile) il Table Module tratta più classi (Vettura:Targa,Modello,Prezzo…;
  Costruttore:Denominazione, presidente, N. Dipendenti, Indirizzo,…;
  Parti meccaniche: Tipi di freni; Tipo di Motore,…;)

Table Module : Quando si usa

- Se le regole del dominio applicativo non sono molto complesse, tanto da richiedere le caratteristiche previste nell'orientamento agli oggetti (ereditarietà e polimorfismo) e l'applicazione è centrata sui dati;
- Un esempio di regole poco complessa è un Sistema per la Gestione delle Auto Usate: rilevare le componenti dell'auto; aggiungere/cancellare/modificare le caratteristiche del costruttore; aggiungere/cancellare, nel magazzino, una nuova vettura usata

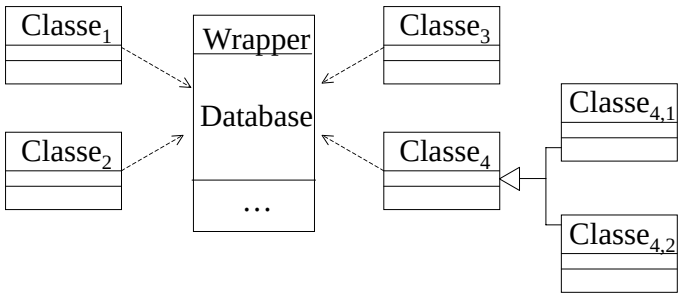
Table Module : Soluzione

- 1. Progettazione della base di dati normalizzata, partendo dal suo modello concettuale (E- R / Digramma delle Classi); ogni tavola corrisponde ad una vista dei dati nel dominio applicativo
- 2. Progettazione dei metodi, corrispondenti alle procedure richieste dal dominio per il trattamento dei dati contenuti in ogni classe, per moduli ottimizzando il riuso (è opportuno intensificare il riuso di componenti del patrimonio di base e, quindi, anche di COTS)

Domain Model: Obiettivo

- Strutturare il dominio in oggetti che contengono sia i dati sia il comportamento;
- Ogni oggetto rappresenta un significativo concetto del dominio; il significato incorporato nell'oggetto è tanto più profondo quanto più complesso è il dominio

Domain Model : Come funziona



- L'applicazione consiste in una rete di oggetti interconnessi tra loro; le classi non sono in corrispondenza 1:1 con tavole o viste
- Il Wrapper nasconde: la relazione tra i dati persistenti di ogni classe e le tavole di cui si compone il DB; i metodi resi disponibili dal DBMS utilizzato per accedere fisicamente alle tavole od alle viste

Domain Model : Quando si usa

- Se le regole del Dominio Applicativo sono molto complesse e possono cambiare frequentemente è opportuno modellarlo con gli oggetti e con le caratteristiche previste nell'orientamento agli oggetti;
- Un esempio di regole complesse è un Sistema di Configurazione di un'Offerta: rilevare le caratteristiche dell'utente, dipendentemente da quello che vuole ordinare; configurare l'offerta in relazione alle sue caratteristiche…

Domain Model : Soluzione…

- 1. Un modello di dominio semplice può essere progettato con la stessa logica di un database dove ogni tavola corrisponde ad un oggetto (il comportamento è incapsulato nei metodi)
- 2. Quando il modello di dominio risulta essere complesso si usano le caratteristiche tipiche dell'orientamento agli oggetti: ereditarietà ed il polimorfismo (ottimizzando l'I.H. attraverso gli oggetti specifici)

… Domain Model : Soluzione

- 3. Quando un metodo in un oggetto serve solo per un caso d'uso è opportuno prevedere oggetti specifici, per evitare l'aumento di complessità e la diminuzione della manutenibilità degli oggetti
- 4. Potrebbe insorgere il problema di eccessiva numerosità degli oggetti; solo in questo caso si riconsidera il punto 3.

Patterns di Sorgenti di Dati

Decisioni Per la Struttura delle Sorgenti di Dati…

- Come interagire con il Data Base?
  - Se si sta usando il Transaction Script
    - Row Data Gateway
  - Se si sta usando il Transaction Script e la piattaforma disponibile è orientata ai records
    - Table Data Gateway
  - Se si sta usando un Domain Model che ha oggetti assimilabili alle tavole del data base
    - Active Records
  - Se si sta usando un Domain Model molto articolato
    - Data Mapper

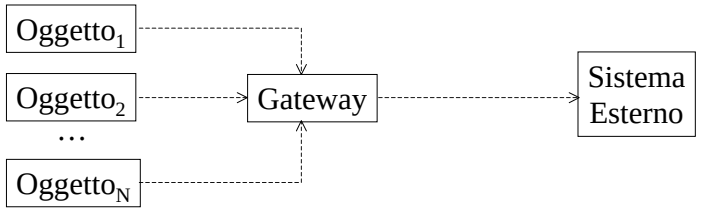
… Decisioni Per la Struttura delle Sorgenti di Dati

- Come associare i dati di dominio ai campi del database?
  - Metadata Mapping

Gateway : Obiettivo

- Nascondere in un oggetto tutti i segreti della comunicazione con un sistema esterno che non siano orientati agli oggetti
- I sistemi esterni con cui comunicare potrebbero essere per esempio: database relazionali, sistema CICS per la comunicazione dei dati, sistemi basati su XML…

Gateway : Come funziona …



- E' un wrapper delle modalità con cui si accede al sistema esterno.

... Gateway : Come funziona

- Adatta l'interfaccia con cui comunicano gli oggetti all'interfaccia con cui comunica il sistema esterno
- Quando l'Applicazione (Oggetto i_{mo}) ha bisogno di qualche servizio dal Sistema Esterno lo richiede al Gateway che lo adegua al Sistema Esterno.

Gateway : Quando si usa

- Quando esiste una noiosa interfaccia verso il sistema esterno che deve essere utilizzata in più punti dell'applicazione
- Se cambia l'interfaccia del sistema esterno è necessario solo modificare la Gateway
- Il test del sistema diventa più facile perché non richiede necessariamente la contemporanea presenza dell'Applicazione e del Sistema Esterno

Gateway : Soluzione...

- 1. Progettazione della trasformazione tra l'interfaccia degli oggetti dell'applicazione e quella dei sistemi esterni
- 2. Costruire la Gateway come due oggetti: il back end per gestire la trasformazione tra interfaccia dell'Applicazione e la interfaccia generale verso il Sistema Esterno; il front end per gestire le parti invarianti tra le diverse interfacce che deve incapsulare e le specializzazioni verso la specifica interfaccia di cui ha bisogno per soddisfare la richiesta ricevuta

... Gateway : Soluzione

- 3. Se le interfacce specifiche sono molte si dividono in classi e si progetto un front end per ogni classe

Row Data Gateway : Obiettivo

- Rendere disponibile un oggetto che corrisponde ad un singolo record estratto dalla sorgente di dati ma può essere trattato con lo stesso linguaggio dell'applicazione, qualunque sia quello della sorgente di dati

Row Data Gateway : Come funziona ...

- L'applicazione ha bisogno di un oggetto e richiede di estrarlo alla gateway, che conosce l'esatta struttura della base di dati

... Row Data Gateway : Come funziona

- Il Gateway trasformerà i tipi di dati nella base di dati con i tipi di dati richiesti, in memoria dall'applicazione
- Una row data gateway potrebbe sovrastare ad una tavola reale oppure ad una vista

Row Data Gateway : Quando si usa

- Quando si vuole nascondere all'Applicazione la organizzazione della base di dati
- E' opportuno utilizzarlo quando la logica del dominio è realizzata in modalità orientata ai records; ad esempio Transaction Script. Diventerebbe molto complesso quando il dominio è organizzato in modo più complesso: ad esempio Domain Model

Row Data Gateway : Soluzione ...

- 1. Progettazione della base di dati e delle viste dell'Applicazione
- 2. Progetto e realizzazione dei Finder per ogni oggetto previsto dall'applicazione che ha alla base le tavole e/o le viste progettate al precedente punto

... Row Data Gateway : Soluzione

- 3. Per realizzare il polimorfismo è possibile avere dei Gateway che abbiano una parte invariante insieme con una specializzazione per ogni interfaccia da incapsulare; oppure una gateway per ogni interfaccia
- 4. Se il numero di Gateway da scrivere sono molte è efficace la generazione automatica basata su Metadata Mapping

Table Data Gateway : Obiettivo

- Rendere disponibile un oggetto che corrisponde ad una tavola o una vista estratta dalla sorgente di dati che può essere trattata con lo stesso linguaggio dell'applicazione, qualunque sia quello della sorgente di dati

Table Data Gateway : Come funziona...

Gateway

- L'applicazione chiede al gateway di accedere alla base di dati e di restituirgli una tavola
- Nel Gateway sono incapsulate tutte le informazioni per navigare nella base di dati e per accedere ad essa.

...Table Data Gateway : Come funziona

- E' utile per i domini implementati con tecniche orientate alle Tavole, Table Module. Non è ottima per tecniche Domain Model perché una modifica della base di dati impatterebbe sull'implementazione del dominio e viceversa

Table Data Gateway : Quando si usa

- Quando i due sottosistemi che devono comunicare hanno percorsi e tempi di evoluzione diversi
- Quando i due sottosistemi hanno approcci o standard o gestori differenti tra loro
- E' usato frequentemente come Data Mapper

Table Data Gateway : Soluzione ...

1. Progettazione della base di dati e delle viste dell'Applicazione
2. Progetto e realizzazione dei Finder per ogni tavola o vista prevista dall'applicazione che ha alla base le tavole e/o le viste progettate al precedente punto

... Table Data Gateway : Soluzione

3. Per realizzare il polimorfismo è possibile avere dei Gateway che abbiano parti invarianti insieme con una specializzazione per ogni interfaccia da incapsulare; oppure una gateway per ogni interfaccia

Active Record : Obiettivo

- Rendere disponibile un oggetto che corrisponde ad un record di una tavola o di una vista estratta dalla sorgente di dati che può essere trattata con lo stesso linguaggio dell'applicazione, qualunque sia quello della sorgente di dati e la logica di dominio su quei dati

Active Record : Come funziona

Gateway con
Metodi del Dominio
applicabili ai dati che estrae

- E' simile al Row Data Gateway con l'aggiunta di metodi che rappresentano la logica di dominio applicata ai dati del record messo a disposizione in memoria

Active Record : Quando si usa

- E' particolarmente adatto a domini applicativi la cui logica è modulabile per tavole
- Questa scelta è più semplice rispetto al Data Mapper, a condizione che la logica di dominio sia adeguata ad essa. Infatti, poiché esso crea accoppiamento tra gli oggetti di dominio e la struttura del data base se i primi non sono compatibili con la struttura della base di dati si creerebbe una interdipendenza che renderebbe difficile le modifiche

Active Record : Soluzione ...

- 1. Progettazione della base di dati e delle viste dell'Applicazione
- 2. Progetto e realizzazione degli gateway che comprendono i Finder per ogni oggetto previsto dall'applicazione, corrispondenti alle tavole e/o le viste progettate al precedente punto, ed i relativi metodi

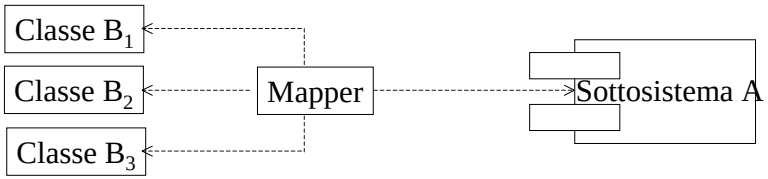
... Active Record : Soluzione

- 3. Per realizzare il polimorfismo è possibile avere dei Gateway che abbiano una parte invariante insieme con una specializzazione per ogni interfaccia da incapsulare; oppure una gateway per ogni interfaccia, i metodi sono nella parte invariante;
- 4. Se il numero di Gateway da scrivere sono molte è efficace la generazione automatica basata su Metadata Mapping

Mapper: Obiettivo

- Fare comunicare due sottosistemi facendo in modo che ognuno sappia il meno possibile dell'altro

Mapper : Come funziona ...



- Quando l'Applicazione ha bisogno di comunicare dati da uno o più oggetti di dominio ad un altro sottosistema: chiede al /agli oggetti di dominio i dati da comunicare , li adegua al sottosistema destinatario e li comunica a quest'ultimo

... Mapper : Come funziona

- Viceversa se la comunicazione ha il verso da un sottosistema ad uno o più oggetti di dominio: chiede al sottosistema i dati da comunicare, li adegua al od agli oggetti di dominio e li comunica a questi ultimi

Mapper : Quando si usa

- Quando i due sottosistemi che devono comunicare hanno cicli di evoluzione diversi
- Quando i due sottosistemi hanno approcci o standard o gestori differenti tra loro
- E' usato frequentemente come Data Mapper

Mapper : Soluzione ...

- 1. Progettazione della corrispondenza tra servizi richiedibili agli oggetti di dominio e quelli disponibili su altri sottosistemi da incapsulare nel Mapper
- 2. Progettazione delle Funzioni per la gestione del mapping : inserimento, variazione, cancellazione ed interrogazione delle corrispondenze tra servizi

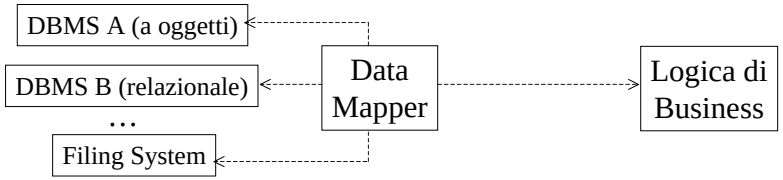
... Mapper : Soluzione

- 3. Funzioni per la distribuzione dei parametri nelle chiamate
- 4. Funzioni per l'adeguamento dei parametri provenienti da/dai serventi per generare i parametri che si aspetta il/i clients.

Data Mapper: Obiettivo

- Rendere indipendente la logica di business dalla organizzazione e dalla gestione dei dati
- La logica di business potrebbe considerare oggetti caratteristici del dominio; la base di dati potrebbe considerare oggetti più adeguati all'organizzazione dei data base o tavole o file; gli oggetti potrebbero essere realizzati in data base ad oggetti, relazionali od in sistemi di gestione di file

Data Mapper : Come funziona ...



- Una richiesta del client comporterà l'accesso a molti oggetti, tavole o file; gli oggetti, le tavole e i file possono avere molte interrelazioni tra loro, per decidere i dati da portare in memoria da ogni data set è necessario gestire tali interrelazioni; perciò è opportuno utilizzare il pattern Lazy Load.

... Data Mapper : Come funziona...

- Il mapper deve includere diverse strategie: classi corrispondenti a molte tavole, oppure a molti campi; classi che ereditano da altre classi;
- Una volta estratti i dati, il mapper deve congiungerli opportunamente per avere l'oggetto richiesto

... Data Mapper : Come funziona

- E' possibile avere uno o più Data Mapper dipendentemente dalla complessità del dominio. In tal caso è necessario avere una gerarchia di Data Mapper.
- Per collegare i dati degli oggetti di dominio ai campi nelle colonne del database è possibile utilizzare il Metadata Mapping
- Per motivi di test e per far lavorare lo stesso sistema con differenti data base si può sostituire il Data Mapper.

Data Mapper : Quando si usa

- Se si vuole che lo schema del database e quello degli oggetti di dominio evolvano indipendentemente. Sia nel progetto sia nel test del modello di dominio è possibile trascurare lo schema di data base perché gli oggetti di dominio non sanno niente della struttura del data base; la corrispondenza è tutta a carico del Data Mapper.

Data Mapper : Soluzione ...

- 1. Progettazione della base di dati per raccogliere i dati persistenti
- 2. Analizzare la corrispondenza tra schema del data base ed oggetti di business

... Data Mapper : Soluzione

- 3. Progettazione di uno o più Mapper, dipendentemente dalla complessità della corrispondenza; Se serve, progettare la gerarchia dei mappers
- 4. Realizzazione dei Data Mapper com'è previsto in Mapper

Metadata Mapping : Obiettivo

- Fare corrispondere ai dati di dominio degli oggetti in memoria i campi delle tavole nella base di dati
- La corrispondenza è espressa in forma tabulare e, quindi, può essere elaborata da programmi generici per eseguire i dettagli della lettura,inserimento e modifica dei dati

Metadata Mapping : Come funziona...

- La corrispondenza tra dati di dominio e campi di tavole si può manifestare con:
  - Generazione di codice
  - Programma riflessivo

... Metadata Mapping : Come funziona ...

- Generazione di codice: i dati richiesti sono l'input per la generazione di un programma che li mette in corrispondenza con i campi; il codice sembra scritto manualmente ma è generato automaticamente
- Programma riflessivo:un generico programma a cui è chiesto un dato o un metodo e questo legge il nome del campo o del metodo dai metadata e quindi accede ad esso mettendolo a disposizione

... Metadata Mapping : Come funziona

- La generazione è poco dinamica perché ogni modifica di corrispondenza comporta la generazione di nuovi programmi e la loro ricompilazione e la ricostruzione di una parte dell'applicazione
- Approccio riflessivo è più dinamico ma rallenta l'esecuzione dei programmi
- Entrambi sono farraginosi nella correzione. Se si usa adeguatamente la documentazione è meglio l'approccio riflessivo; invece se si usa leggere i programmi sono più espliciti quelli generati.

Metadata Mapping : Quando si usa

- Invece che i metadati si può utilizzare codice manoscritto per far corrispondere i dati ai campi; quando la scrittura di questo codice è molto onerosa rispetto alla gestione di una tavola di metadati o di un programma generatore allora è opportuno utilizzare i metadati;
- I metadati possono rendere più facile la reingegnerizzazione di una base di dati; infatti i metadati sono un'interfaccia dello schema del data base.

Metadata Mapping : Soluzione ...

- 1. Analizzare gli attributi necessari per descrivere i campi in modo che possano essere collegati ai dati degli oggetti;
- 2. Valutare se qualche attributo è specifico per pochi casi; se sì conviene eliminarli per diminuire la complessità dei metadati

... Metadata Mapping : Soluzione

- 3. Se si vuole utilizzare il programma riflessivo, costruire il parser dei metadati
- 4. Se si vogliono generare programmi di collegamento, costruire il programma generatore
- 5. Prevedere le modifiche al parser oppure ai programmi generati per comprendere i casi specifici non previsti dagli attributi presenti nei metadati.

Presentazione

Strato di presentazione

- Sarà realizzato con tecnologie per la costruzione di interfacce.
- In questo corso non sono trattate e si rimanda ad altro corso.

Verifica delle architettura

Motivazione della Verifica …

- L'architettura è un manufatto critico nella configurazione di un sistema software. Un errore nell'architettura si ripercuote e si amplifica in tutti gli altri manufatti, richiedendo enormi risorse economiche per ripararlo

… Motivazione della Verifica

- La verifica serve a rilevare il più presto possibile, durante la costruzione di un sistema software, se le componenti hanno le proprietà e le relazioni attese dai requisiti del sistema software

Scopo della Verifica

- Il metodo proposto è il Software Architecture Analysis Method (SAAM).
- Particolarmente adatto alla verifica di attributi di qualità del software quali:
  - Manutenibilità
  - Modificabilità
  - Robustezza
  - Flessibilità
  - ….

Input a SAAM …

- L'architettura da verificare
- Un insieme di scenari di verifica;
  - ogni scenario descrive l'interazione di una parte interessata con il sistema;
  - essi servono a rendere operative le caratteristiche del sistema
  - Uno scenario si compone di:
    - Interesse della parte interessata al sistema;
    - Ambiente che descrive le parti di sistema che impattano la caratteristica di interese;
    - Risposta che descrive il risultato atteso.

… Input a SAAM

- Le caratteristiche di qualità da verificare, devono essere collegati alle finalità organizzative del software.
  - Per esempio: la modificabilità è critica per il successo del software perché deve essere capace di rispondere rapidamente alle evoluzioni per essere competitivi in un mercato turbolento. Allora la modificabilità può essere misurata con il tempo necessario per: sostituire un COTS; per cambiare l'interfaccia; per aggiungere una funzione…

Output da SAAM

- Individuazione delle aree dell'architettura più problematiche per i cambiamenti prospettati dagli scenari
- Comprensione della capacità di limitare le modifiche necessarie per la realizzazione dei cambiamenti prospettati dai scenari

Parti Interessate …

Parte Interessata	Definizione	Interesse
Architetto Software	Responsabile della architettura del sistema e della mediazione tra le diverse necessità	Moderazione e mediazione tra le parti interessate riguardo i requisiti di qualità
Sviluppatore	Progettista/ Programmatore	Chiarezza e completezza della descrizione dell'architettura, alta coesione delle parti, chiari meccanismi di connessione
Manutentore	Responsabile modifiche successive al rilascio iniziale	Manutenibilità, capacità di individuare le componenti da modificare a seguito di richieste di manutenzione

… Parti Interessate …

Parte Interessata	Definizione	Interesse
Integratore	Sviluppatore responsabile assemblaggio componenti	Analogo a quello dello sviluppatore
Collaudatore	Sviluppatore responsabile test sistema	Protocolli integrati per la gestione errori; basso accoppiamento e alta coesione tra componenti; integrità concettuale
Esperto standard	Sviluppatore responsabile della conoscenza degli standard (correnti e futuri) a cui il sistema deve conformarsi	Separazione dei concetti trattati, manutenibilità, interoperabilità

... Parti Interessate

Parte Interessata	Definizione	Interesse
Ingegnere prestazioni	Persona che analizza gli artefatti per verificare il soddisfacimento delle prestazioni e del throughput	Comprensibilità, integrità concettuale prestazioni, affidabilità
Esperto sicurezza	Persona responsabile della verifica dei requisiti di sicurezza del sistema	Sicurezza
Responsabile di progetto	Persona che alloca le risorse ai team, responsabile del budget e interfaccia verso il cliente	Chiara strutturazione dell'architettura, struttura modulare, scadenze

Passi previsti da SAAM

- Descrizione degli Scenari
- Presentazione dell'Architettura
- Assegnazione delle Priorità agli Scenari
- Valutazione degli Scenari Indiretti
- Valutazione dell'Interazione tra Scenari
- Produzione del Report di Verifica

Descrizione degli Scenari

- Gli scenari devono illustrare i cambiamenti prevedibili per il sistema. Pertanto è necessario esprimere negli scenari tutti gli usi più importanti del sistema e per ognuno di essi le qualità che il sistema deve esprimere per soddisfare i suoi utilizzatori.
- Questa fase è rilevante e si esegue in due o tre riunioni delle parti interessate in cui si estraggono gli scenari più significativi

Esempio di Scenario

Scenario # 12	**Descrizione**. Cambiare il data base utilizzato da un file ad un DBMS relazionale SQL Standard
Interesse. Portabilità del sistema in sistemi operativi molto diffusi	
Ambiente. Il sistema è già in esercizio ma è necessario estendere il bacino dei potenziali utilizzatori.	
Risposta.Il cambiamento deve comportare la modifica di poche classi (< = 3) con uno sforzo < 20gg uomo	

Presentazione dell'Architettura

- L'architettura o le architetture candidate sono descritte alle parti interessate in modo da poter essere messe in collegamento con gli scenari.
- La descrizione dell'architettura è aiutata dai formalismi grafici oltre che dalle descrizioni testuali
- La descrizione dell'architettura potrebbe suggerire qualche altro scenario a qualcuna delle parti interessate partecipante alla verifica

Assegnazione delle Priorità agli Scenari

- Gli scenari sono classificati
  - Diretti, riscontrano la loro realizzazione nell'attuale architettura
  - Indiretti, richiedono modifiche all'architettura per la loro realizzazione
- Tutti gli scenari sono messi in ordine di importanza, secondo le parti interessate che partecipano alla verifica

Esempio di Classificazione e Priorità

Scenario #	**Classificazione**	**Priorità** (N. di P. I. che lo ritengono importante
25	Diretto	3
26	Diretto	5
27	Indiretto	4
28	Diretto	2

Valutazione degli Scenari Indiretti

- Per gli scenari indiretti devono essere valutati i cambiamenti necessari per la loro realizzazione
- Per ogni scenario devono essere rilevati i cambiamenti necessari ed i costi previsti per la loro valutazione
- Un cambiamento può essere:
  - Modifica di una componente
  - Aggiunta di una componente
  - Aggiunta di una relazione tra componenti esistenti

Esempi di Valutazione di Scenari Indiretti

Scenario #	Richiesta di modifica	# Comp. da Modif.	Sforzo Stimato
7	Interfaccia Verso il Data Base	2	1,5 mesi uomo
8	Modello di vista dei dati	3	5 gg uomo

Valutazione dell'Interazione tra Scenari

- Due scenari si dicono interagenti se sono collegati o richiedono la modifica di una o più componenti uguali
- Quando due scenari interagiscono potrebbe voler dire che
  - Ci sono componenti che realizzano concetti o requisiti differenti e che, quindi, daranno problemi nella manutenzione

Produzione del Report di Verifica

- Produrre il report che raccoglie i risultati di tutte le attività precedenti