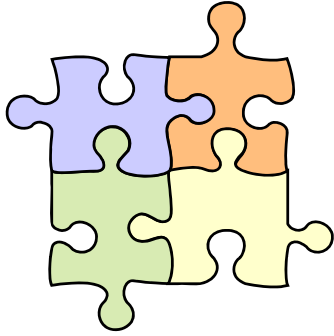


Luca Cabibbo



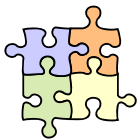
Dallo Schema Mapping all'Object-Relational Mapping

Luca Cabibbo
aprile 2008

1

Dallo Schema Mapping all'Object-Relational Mapping

Luca Cabibbo



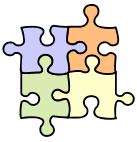
Sommario

- Introduzione all'Object-Relational Mapping
- Introduzione allo Schema Mapping
- Dallo Schema Mapping all'Object-Relational Mapping
- Discussione

2

Dallo Schema Mapping all'Object-Relational Mapping

Luca Cabibbo



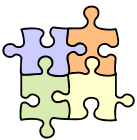
- Object-Relational Mapping

- Gli strumenti per l'**Object/Relational Mapping (ORM)**
 - forniscono un'API orientata agli oggetti per l'accesso a basi di dati relazionali
 - il programmatore non vede lo schema della base di dati
 - piuttosto, ha la possibilità di eseguire operazioni di gestione della persistenza per oggetti delle sue classi che devono essere gestiti in modo persistente – le cosiddette operazioni CRUD (Create, Read, Update, Delete)
 - ad es., **em.find(Product.class, id);**
 - semplificano l'accesso alla base di dati, risolvendo molte problematiche comuni
 - ad esempio, gestione delle corrispondenze, gestione della concorrenza e delle transazioni, caching di oggetti, ...
 - ad es., Java Persistence API, JDO, Hibernate, ...

3

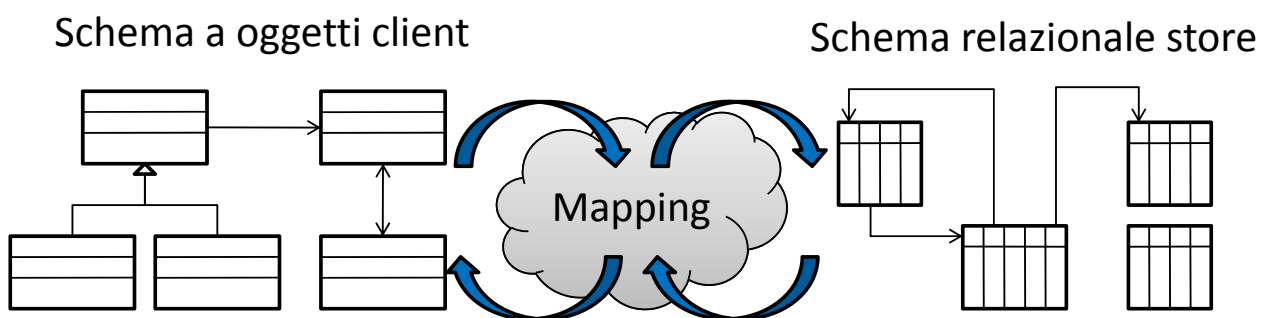
Dallo Schema Mapping all'Object-Relational Mapping

Luca Cabibbo



Mapping negli strumenti ORM

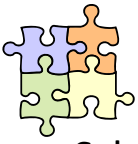
- Gli strumenti ORM richiedono che sia definito una mapping (ovvero, delle corrispondenze) tra le classi e lo schema della base di dati relazionale
 - lo strumento ORM traduce le operazioni CRUD in operazioni SQL proprio sulla base di questo mapping



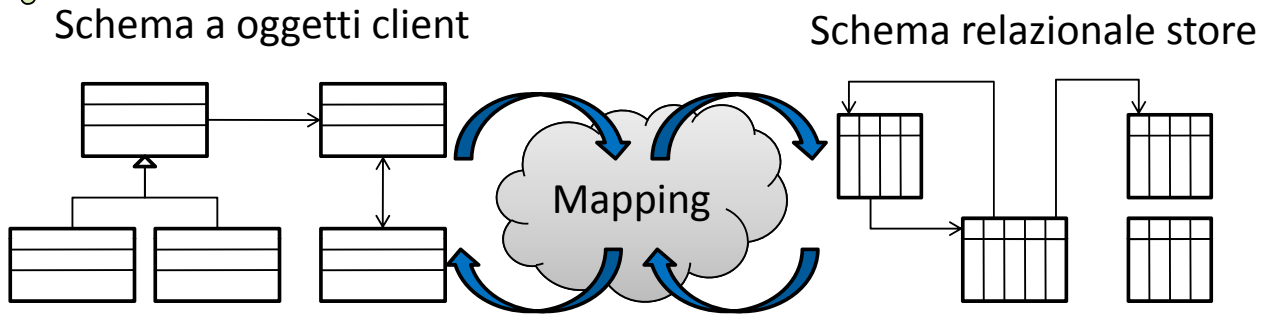
4

Dallo Schema Mapping all'Object-Relational Mapping

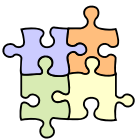
Luca Cabibbo



Mapping negli strumenti ORM

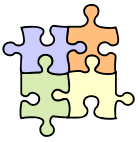


- Lo strumento ORM, sulla base del mapping, decide (in particolare)
 - Create – come memorizzare oggetti nella base di dati
 - update mapping
 - Read – come costruire oggetti a partire da tuple nella base di dati
 - query mapping



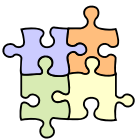
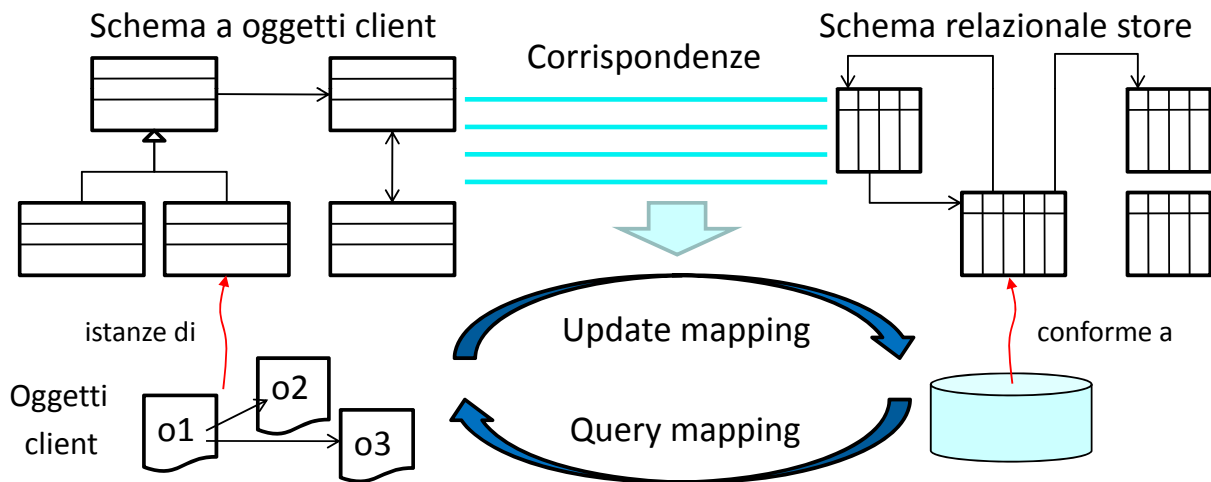
Specifica del mapping

- La modalità di specifica del mapping negli strumenti ORM attuali non è completamente soddisfacente
 - sarebbe preferibile
 - una modalità di specifica semplice e dichiarativa – in particolare, basata su corrispondenze tra attributi
 - la derivazione automatica del query mapping e dell'update mapping a partire da questa specifica dichiarativa



Approccio desiderato

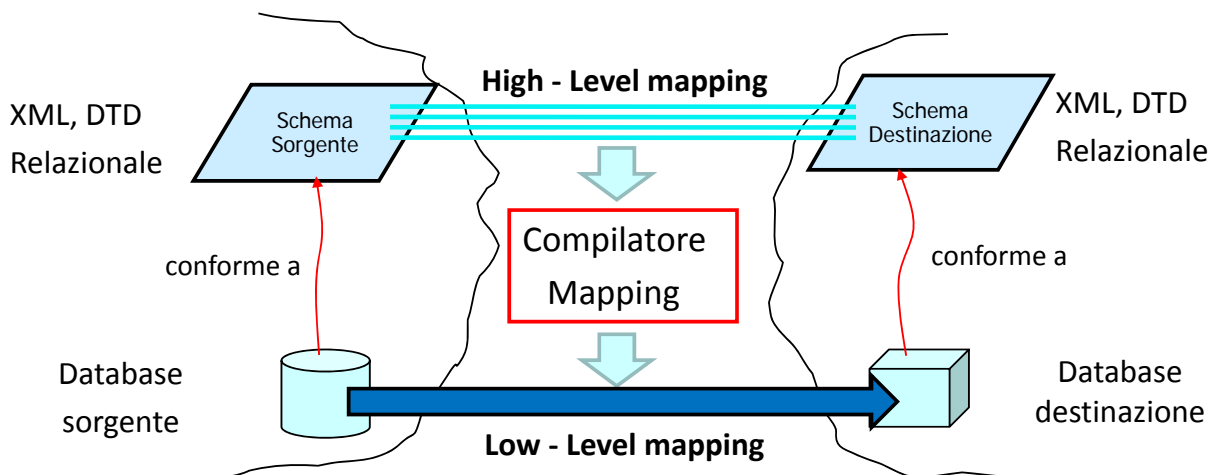
□ L'approccio desiderato

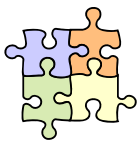


- Schema Mapping

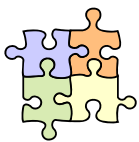
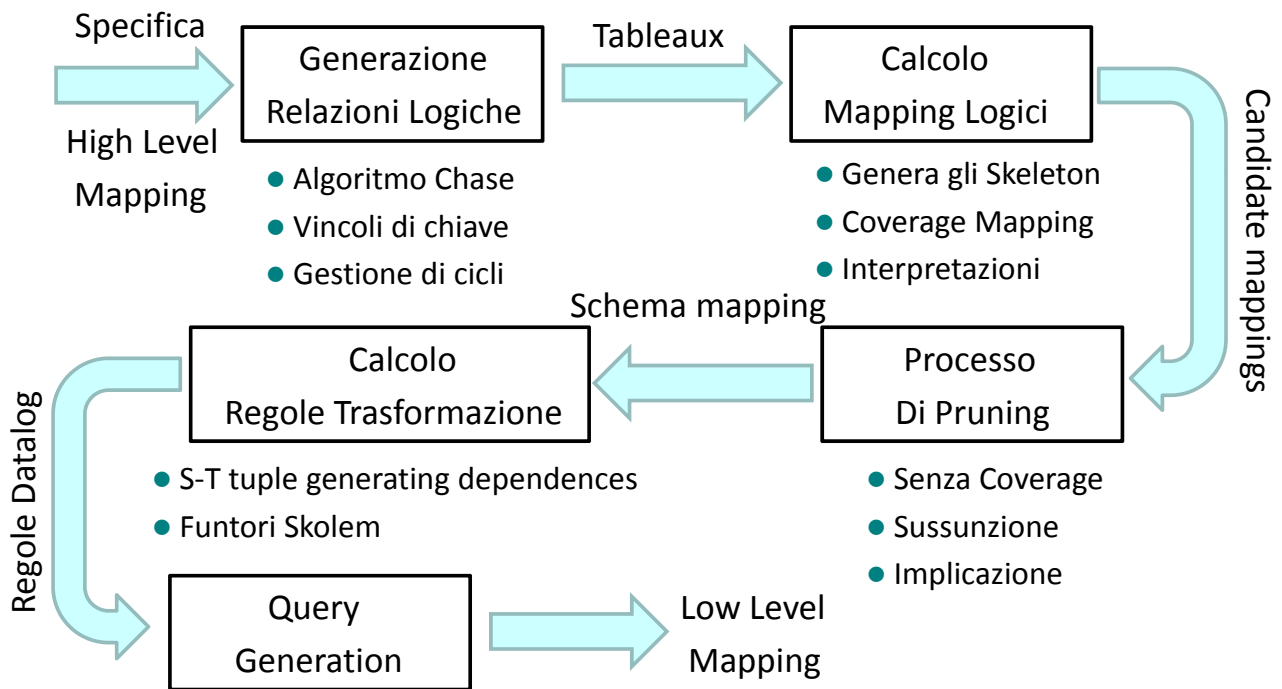
□ Lo Schema Mapping

- affronta il problema di determinare interrogazioni per muovere dati da una sorgente ad un'altra
- sulla base di un mapping espresso mediante corrispondenze tra attributi



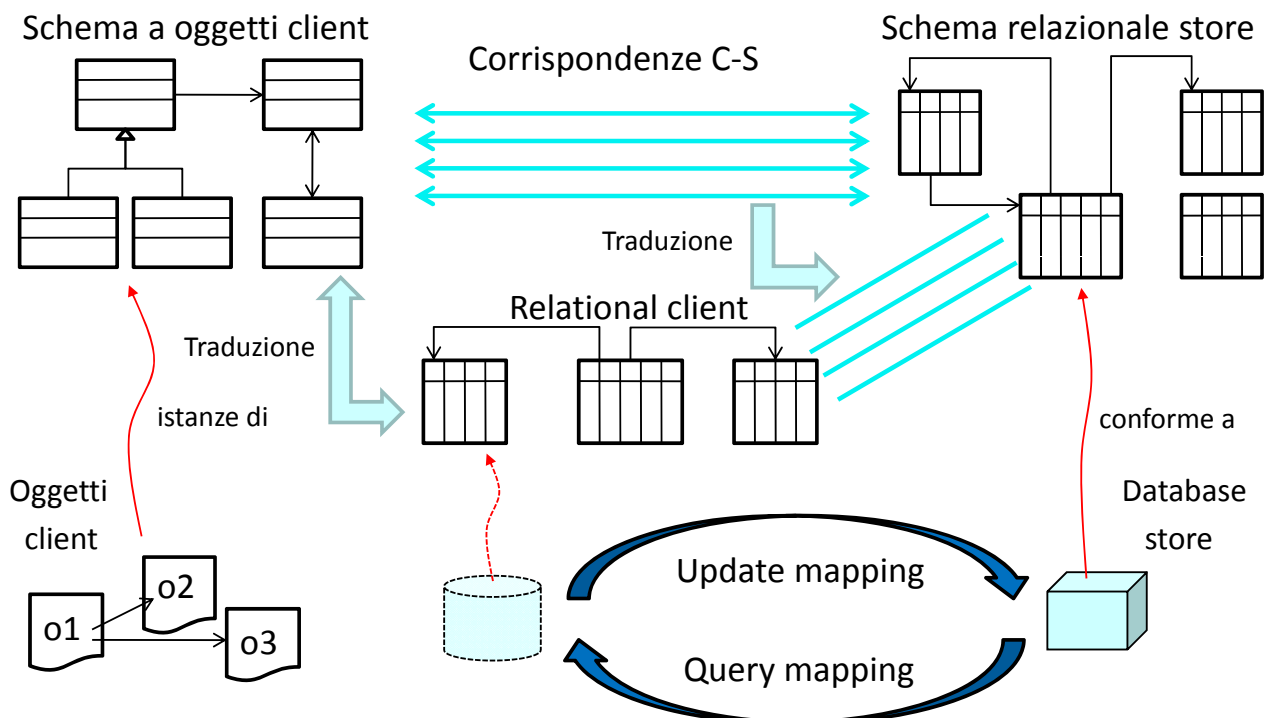


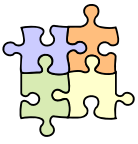
Fasi nello Schema Mapping



- Dallo SM all'ORM

- L'idea è risolvere il problema dell'Object-Relational Mapping sulla base delle tecniche di Schema Mapping





Dallo SM all'ORM

□ Idea

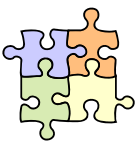
- esprimere corrispondenze tra classi e relazioni – mediante corrispondenze tra attributi
- risolvere due problemi di schema mapping
 - uno in cui lo schema source corrisponde allo schema store e lo schema target corrisponde allo schema client – genera il query mapping
 - uno nella direzione opposta – genera l'update mapping

□ Problema

- le tecniche di schema mapping sono definite tra schemi relazionali

□ Soluzione

- tradurre lo schema ad oggetti in uno relazionale



Dallo SM all'ORM

□ Idea

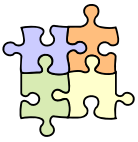
- tradurre lo schema client (ad oggetti) in uno schema relazionale – sulla base di una traduzione semplice
- tradurre le corrispondenze tra classi e relazioni in corrispondenze relazionali tra la traduzione relazionale dello schema client e lo schema store

□ Problema

- alcune corrispondenze tra attributi client-store non si traducono in corrispondenze tra attributi per lo schema mapping

□ Soluzione

- estendere le tecniche di schema mapping per gestire nuovi tipi di corrispondenze



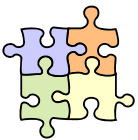
Dallo SM all'ORM

Altri problemi

- le tecniche di schema mapping non considerano o considerano solo parzialmente alcuni vincoli di integrità
 - in particolare, possibilità di avere valori nulli, chiavi, chiavi esterne

Soluzione

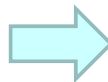
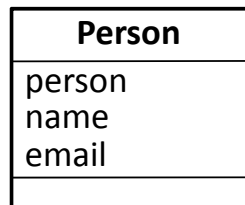
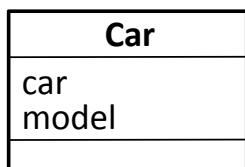
- estendere le tecniche di schema mapping per gestire i vincoli di interesse



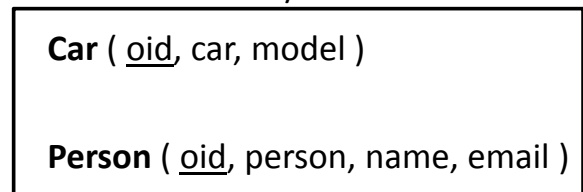
Traduzione dello schema ad oggetti

- Usata una traduzione molto semplice per rappresentare lo schema client in uno schema relazionale corrispondente
 - obiettivo – semplificare i ragionamenti

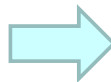
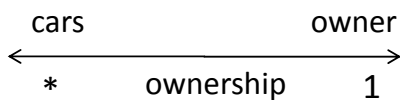
● Classe



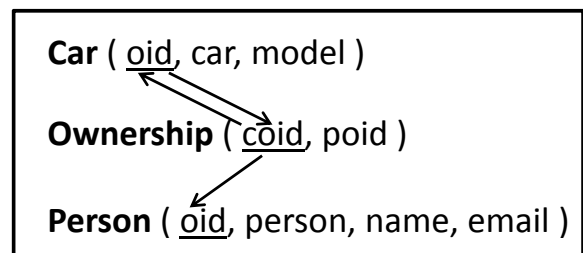
● Una relazione / Cluster relazioni

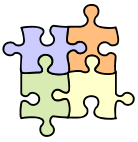


● Associazione



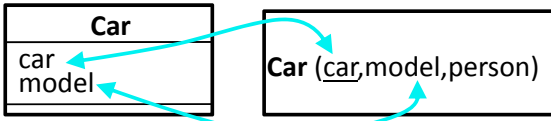
● Join Table e chiavi esterne



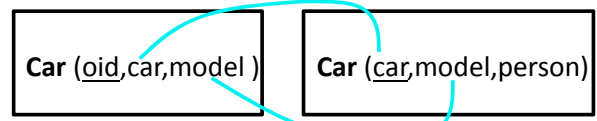


Traduzione delle corrispondenze

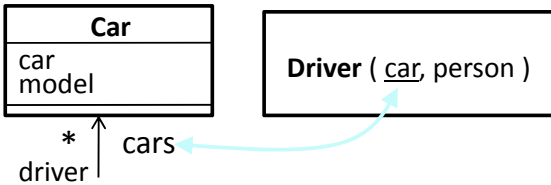
● Corrispondenza attributi orientata



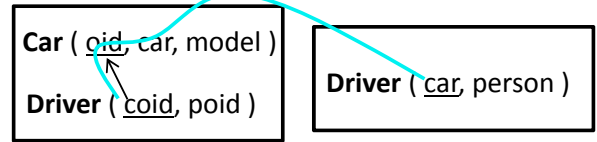
● Corrispondenza attributi



● Corrispondenza associazione

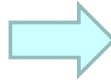


● Corrispondenza generalizzata



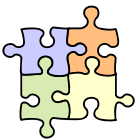
● Orientamento

- Bidirezionale
- Client → Store
- Store → Client



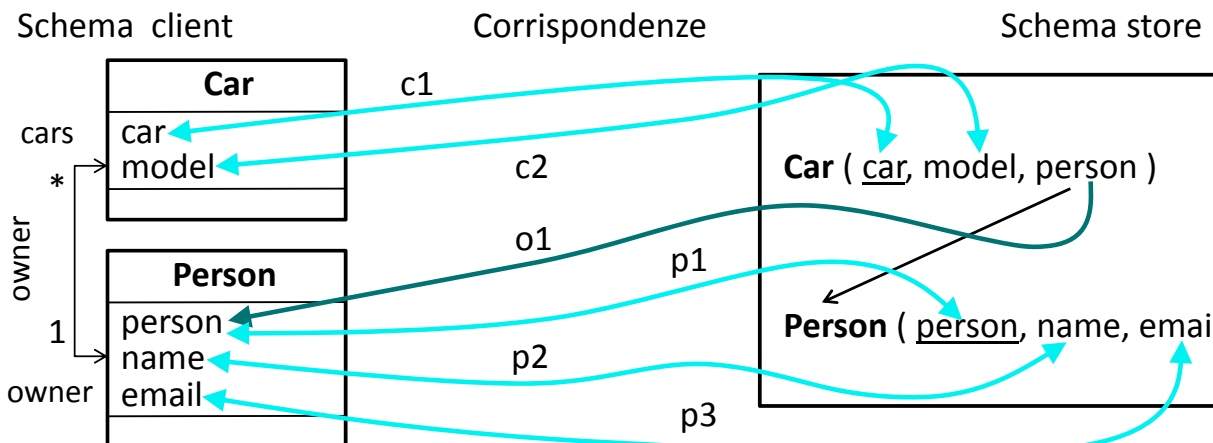
● Problema di riferimento

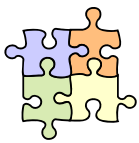
- Update e Query Mapping
- Update Mapping
- Query Mapping



Corrispondenze orientate

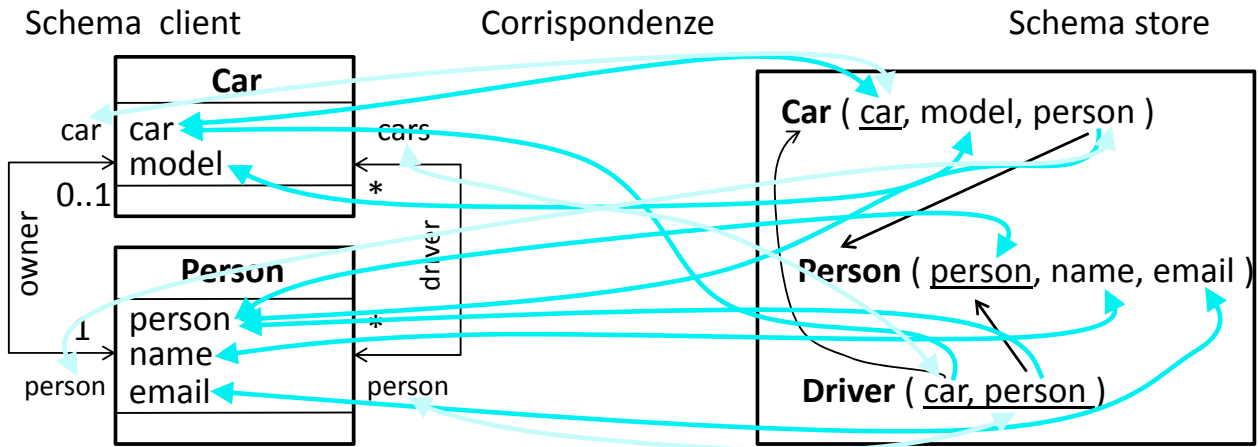
- Alcune corrispondenze vanno applicate solo in uno dei due problemi di schema mapping
 - utile associare un verso alle corrispondenze





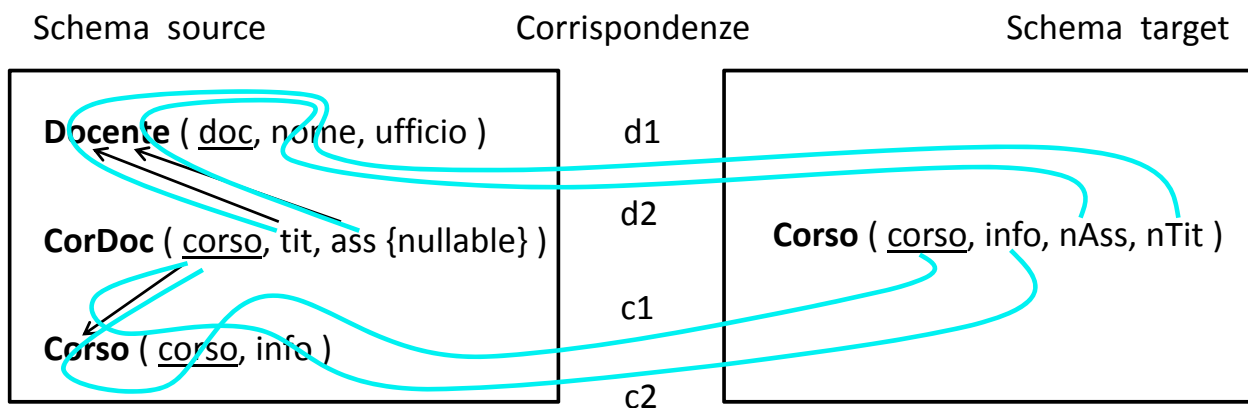
Corrispondenze per associazioni

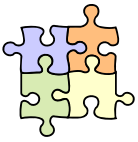
- A livello di specifica del mapping O-R, utile avere corrispondenze per le associazioni
 - andranno tradotte in corrispondenze generalizzate (vedi dopo)



Corrispondenze generalizzate

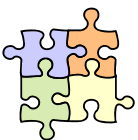
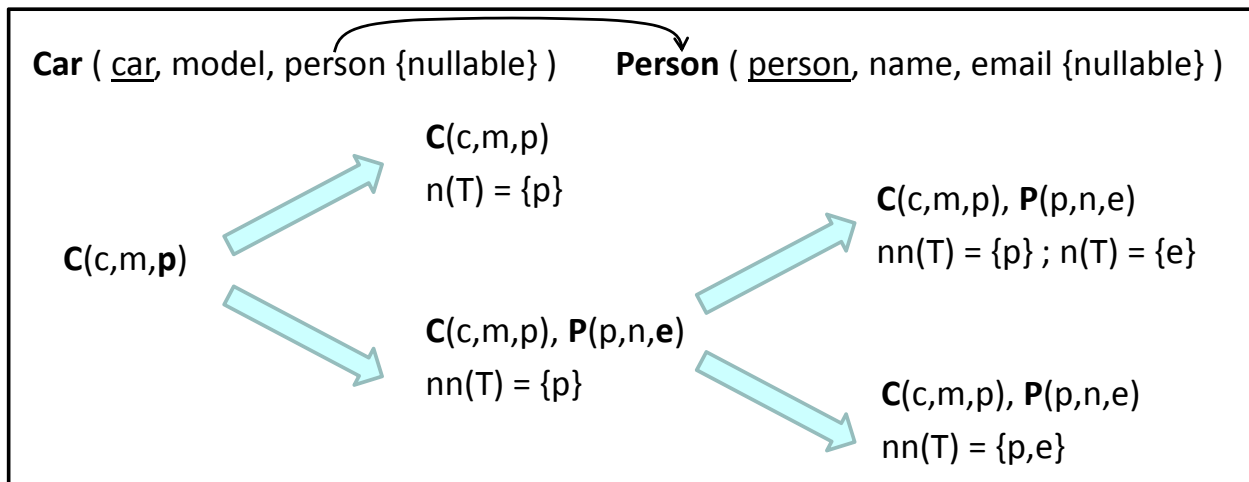
- La traduzione di corrispondenze che coinvolgono associazioni può essere supportata dalle corrispondenze generalizzate
 - le corrispondenze generalizzate “implicano” anche dei join





Gestione di valori nulli

- Gli algoritmi esistenti di Schema Mapping adottano delle tecniche basate su tableaux e chase
- Per la gestione di valori nulli è necessario
 - considerare tableaux “parziali”
 - introdurre un nuovo tipo di passo nel chase



Gestione di valori nulli

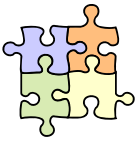
- Gli algoritmi esistenti di Schema Mapping adottano delle tecniche di pruning
- Per la gestione di valori nulli è necessario
 - introdurre nuove regole di pruning

Tabella regole kill

S \ T	None	Null	Not Null	Mand
None	-	Ok	Kill	Skol
Null	Ok	Ok	Kill	Skol
Not Null	Ok	Kill	Ok	Ok
Mand	Ok	Kill	Ok	Ok

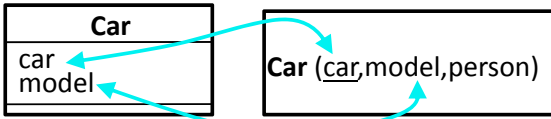
Ordine di applicazione (Generale)

1. No Coverage
2. (Null Foreign Key Joined)
3. Sussunzione
4. Implicazione
5. **Regole di kill**

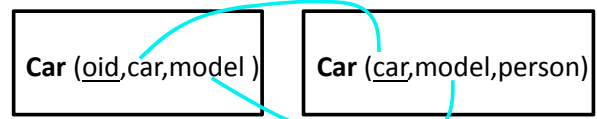


Traduzione delle corrispondenze

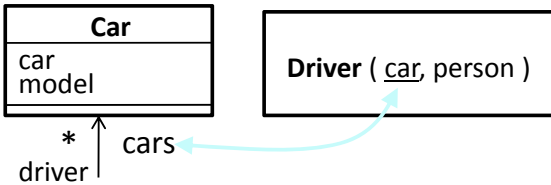
● Corrispondenza attributi orientata



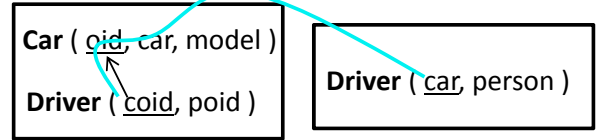
● Corrispondenza attributi



● Corrispondenza associazione

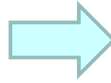


● Corrispondenza generalizzata



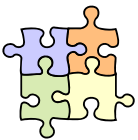
● Orientamento

- Bidirezionale
- Client → Store
- Store → Client



● Problema di riferimento

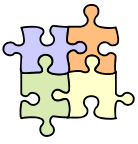
- Update e Query Mapping
- Update Mapping
- Query Mapping



Gestione delle operazioni CRUD

□ Necessario dire

- come implementare le operazioni CRUD
- in termini del query mapping e dell'update mapping



- Discussione

- Quanto mostrato finora è stato realizzato nell'ambito di una tesi svolta da Gabriele Rendina (che ringrazio)
 - tuttavia, ci sono ancora vari punti in sospeso
 - “gruppi” di corrispondenze – e calcolo delle coperture delle corrispondenze basata su gruppi
 - gestione delle gerarchie di classi – anche con riferimento alla modalità della specifica delle corrispondenze
 - in generale, è un problema aperto definire un buon modo per esprimere corrispondenze tra schemi