



SAPIENZA
UNIVERSITÀ DI ROMA

Facoltà di Ing. dell'Informazione, Informatica e Statistica
Laurea in Informatica

Basi di Dati, Modulo 2

Prof. Toni Mancini, Prof. Federico Mari

Dipartimento di Informatica

<http://tmancini.di.uniroma1.it>

<http://mari.di.uniroma1.it>

Slides B.3.2 (S.B.3.2)

Basi di Dati Relazionali

La Fase di Progettazione

Progettazione delle Funzionalità

Progettazione delle Funzionalità

Lo schema concettuale contiene una specifica per ogni use-case che a sua volta definisce ogni operazione dello use case in termini di:

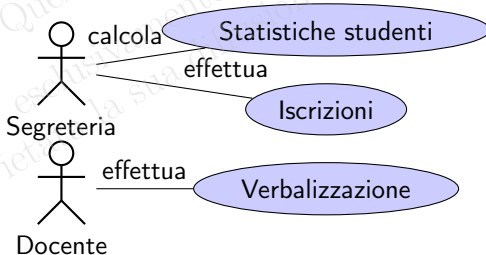
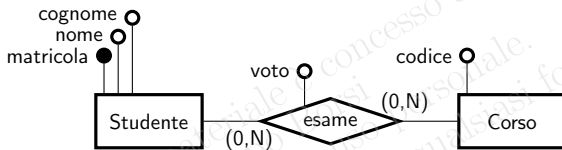
- ▶ **segnatura**: nome dell'operazione, nome e tipo degli argomenti, tipo dell'eventuale valore di ritorno
- ▶ **precondizioni**
- ▶ **postcondizioni**.

Durante la fase di progettazione, per ogni operazione di use-case, va definita una **specifica realizzativa** in termini di:

- ▶ **segnatura**: nome dell'operazione, nome e dominio degli argomenti, dominio dell'eventuale valore di ritorno (i domini sono i domini supportati dal DBMS fatti corrispondere in fase di progettazione ai tipi concettuali)
- ▶ **algoritmo** in pseudo-codice contenente eventuali comandi SQL per l'interazione con la base dati.

Esempio

Schema concettuale: Diagrammi ER e degli use-case



Esempio (2)

Progettazione della base dati

Dominio Voto: si decide di definire un dominio utente Voto. Il dominio è basato sul dominio integer; i valori ammessi sono gli interi tra 18 e 30.

Studente(matricola:integer)

Corso(codice:integer)

esame(studente:integer, corso:integer, voto:Voto)

vincolo **foreign key**: studente references Studente(matricola)

vincolo **foreign key**: corso references Corso(codice)

Esempio (3)

Schema concettuale: Specifica concettuale di use-case

Specifica use-case Verbalizzazione

verbalizzaEsame(s : **Studente**, c : **Corso**, v : [18,31])

precondizioni: L'istanza s non è coinvolta in alcuna istanza della relationship **esame** con l'istanza c:

$$\neg \exists c \text{ esame}(s, c).$$

postcondizioni:

Modifica del Livello Estensionale dei Dati: Il livello estensionale dei dati al termine dell'esecuzione della funzione differisce da quello di partenza come segue:

Elementi del dominio di interpretazione : invariati

Variazioni nelle ennuple di predicati: **esame**(s, c), **voto**(s, c, v).

Valore di Ritorno: nessuno.

End

Esempio (4)

Specifica realizzativa di use-case

Specifica realizzativa use-case Verbalizzazione

verbalizzaEsame(s : integer, c : integer, v : Voto)

algoritmo:

1. Esegui il seguente comando SQL:
insert into esame(studente, corso, voto) **values** (PAR_1, PAR_2, PAR_3)
dopo aver rimpiazzato 'PAR_1', 'PAR_2', 'PAR_3' con i valori dei parametri attuali, rispettivamente, s, c, v.
2. Se il comando precedente restituisce un errore di **vincolo di chiave violato**, allora genera l'errore '**Lo studente di matricola s ha già superato l'esame per il corso di codice c**'.

End

Esempio (5)

Schema concettuale: Specifica concettuale di use-case

Specifica use-case **StatisticheStudenti**

mediaVoti(s : Studente) : reale in [18,31]

precondizioni: L'istanza s è coinvolta in almeno un'istanza della relationship **esame**:

$\exists c \text{ esame}(s, c)$.

postcondizioni:

Modifica del Livello Estensionale dei Dati: nessuna

Valore di Ritorno: Sia $C = \{(c, v) \mid \text{esame}(s, c) \wedge \text{voto}(s, c, v)\}$. Si ha:

$$\text{result} = \frac{\sum_{(c,v) \in C} v}{|C|}$$

End

Esempio (6)

Specifica realizzativa di use-case

Specifica realizzativa use-case StatisticheStudenti

mediaVoti(s : integer) : Reale_18_30

algoritmo:

1. Esegui la seguente query SQL e memorizzarne il risultato nella variabile *Q*:
select avg(voto) as mediaVoto
from esame **where** studente = PAR_1
dopo aver rimpiazzato 'PAR_1' con il valore del parametro attuale *s*.
(La query restituisce un reale o **NULL**.)
2. Se il risultato della query è **NULL**, generare l'errore 'Lo studente di matricola *s* non esiste oppure non ha sostenuto esami' e terminare.
3. Altrimenti, restituire *Q*.

End

Nota: Il dominio **Reale_18_30** è stato scelto, all'inizio della Fase di Progettazione, come corrispondente del dominio concettuale **reale [18,30]**.

Esempio (7)

Schema concettuale: Specifica concettuale di use-case

Specifica use-case StatisticheStudenti (continua)

numMedioEsami() : reale ≥ 0

precondizioni: Il livello estensionale dei dati definisce almeno una istanza di entità **Studente**:
 $\exists s \text{ Studente}(s)$.

postcondizioni:

Modifica del Livello Estensionale dei Dati: nessuna

Valore di Ritorno: **result** è pari al numero di istanze di relationship **esame** definite nel livello estensionale diviso per il numero di istanze di entità **Studente**. Formalmente, siano:

$$E = \{(s, c) \mid \text{esame}(s, c)\} \quad \text{e} \quad S = \{s \mid \text{Studente}(s)\}$$

gli insiemi, rispettivamente, di tutte le coppie (s, c) istanze della relationship **esame** e di tutte le istanze dell'entità **Studente**. Si ha: $\text{result} = \frac{|E|}{|S|}$.

End

Esempio (8)

Specifica realizzativa di use-case

Specifica realizzativa use-case StatisticheStudenti (continua)

numMedioEsami() : RealeNonNeg

algoritmo:

1. Esegui la seguente query SQL e memorizzarne il risultato nella variabile *Q*:

```
select e.numEsami/s.numStudenti as numMedioEsami  
from ( select count(*) as numEsami from esame ) e,  
      ( select count(*) as numStudenti from Studente ) s
```

(La query restituisce un reale o un'eccezione di 'divisione per zero'.)

2. Se il risultato della query è un'eccezione di 'divisione per zero', generare l'errore '**Non esistono studenti**' e terminare.
3. Altrimenti, restituire *Q*.

End

Nota: Il dominio **RealeNonNeg** è stato scelto, all'inizio della Fase di Progettazione, come corrispondente del dominio concettuale **reale** ≥ 0 .