




Progettazione di basi di dati

Progettazione logica relazionale




1



Progettazione logica relazionale

- ▷ Introduzione
- ▷ Ristrutturazione dello schema ER
- ▷ Traduzione nel modello relazionale



2

2




Progettazione logica relazionale

Introduzione




3




Progettazione logica

- ▷ Richiede di scegliere il modello dei dati
 - modello relazionale
- ▷ Obiettivo
 - definizione di uno schema logico relazionale corrispondente allo schema ER di partenza
- ▷ Aspetti importanti
 - semplificazione dello schema per renderlo rappresentabile mediante il modello relazionale
 - ottimizzazione per aumentare l'efficienza delle interrogazioni





4

4



Passi della progettazione logica

Schema ER 



5

5



Passi della progettazione logica

Schema ER 

↓

Ristrutturazione dello schema

↓

Schema ER semplificato 



6

6



7

Progettazione logica relazionale

Ristrutturazione dello schema ER

8

Ristrutturazione dello schema ER

- ▷ Lo schema ER ristrutturato tiene conto di aspetti realizzativi
 - non è più uno schema concettuale
- ▷ Obiettivi
 - eliminazione dei costrutti per cui non esiste una rappresentazione diretta nel modello relazionale
 - trasformazioni volte ad aumentare l'efficienza delle operazioni di accesso ai dati

9

Ristrutturazione dello schema ER

▷ Modello ER ⇔ Modello relazionale
Corrispondenza dei concetti principali

<i>Modello ER</i>	<i>Modello relazionale</i>
Entità	⇔ Relazione (<i>tabella</i>)
Relazione	⇔ Riferimento (<i>chiave esterna</i>)
Generalizzazione	✗ Non presente
Attributo semplice	⇔ Attributo (<i>campo</i>)
Attributo multivalore	✗ Non presente

10

Ristrutturazione dello schema ER

1. Analisi delle ridondanze
2. Eliminazione delle generalizzazioni
3. Partizionamento di concetti
4. Eliminazione degli attributi multivalore
5. Scelta degli identificatori primari

11

Progettazione logica relazionale


Ristrutturazione dello schema ER

1 - Analisi delle ridondanze

12

Analisi delle ridondanze

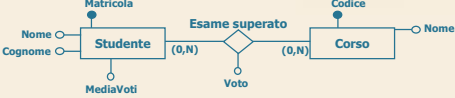
- ▷ Rappresentano informazioni significative, ma derivabili da altri concetti
 - decisione se conservarle
- ▷ Effetti delle ridondanze sullo schema logico
 - semplificazione e velocizzazione delle interrogazioni
 - maggiore complessità e rallentamento degli aggiornamenti
 - maggiore occupazione di spazio




13

13

Esempio di attributo ridondante



- ▷ L'attributo MediaVoti è ridondante
 - utile per velocizzare le interrogazioni relative al calcolo della media dei voti degli studenti
 - se conservato, occorre integrare lo schema relazionale con l'indicazione di ridondanza dell'attributo



14

14

Progettazione logica relazionale

Ristrutturazione dello schema ER
2 - Eliminazione delle generalizzazioni




15

15

Eliminazione delle generalizzazioni

- ▷ Non sono rappresentabili direttamente nel modello relazionale
 - sono sostituite da entità e relazioni
- ▷ Metodi di ristrutturazione
 - accorpamento delle entità figlie nell'entità padre
 - accorpamento dell'entità padre nelle entità figlie
 - sostituzione della gerarchia con relazioni



16

16

1 - Accorpamento nell'entità padre

Schema concettuale originale

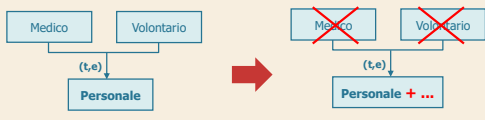




17

17

1 - Accorpamento nell'entità padre

▷ Entità figlie → entità padre

18

18

1 - Accorpamento nell'entità padre

▷ Attributi delle entità figlie → entità padre

Specializzazione (1,N) Medico, Associazione (0,1) Volontario, Specializzazione (0,N) Personale. Attributes: Nome, Cognome, Domicilio, CodFisc.

DBG 19

1 - Accorpamento nell'entità padre

▷ Relazioni con le entità figlie → entità padre

Esame specialistico (1,1) Effettuato da (1,N) Medico (1,N) Volontario (t,e) Personale. Reparto (1,N) Lavora in (1,1) Personale.

DBG 20

19

20

1 - Accorpamento nell'entità padre

▷ Attributo discriminante

- nell'esempio, l'attributo **Tipo** permette di distinguere a quale entità figlia appartiene ogni occorrenza
- se la generalizzazione è *sovrapposta*, sono possibili molte combinazioni come valori di **Tipo**

Esame specialistico (1,1) Effettuato da (0,N) Medico (0,N) Volontario (0,N) Personale (0,N). Attributes: Nome, Cognome, Domicilio, CodFisc. Discriminant: Tipo.

DBG 21

1 - Accorpamento nell'entità padre

▷ Tecnica sempre applicabile

▷ Tecnica sconsigliata in presenza di elevato numero di relazioni e/o attributi provenienti dalle entità figlie (→ vincoli di integrità dei dati)

Specializzazione (1,N) Medico (1,N) Volontario (t,e) Personale (0,N). Attribute: Tipo.

Specializzazione non può assumere un valore se **Tipo = Volontario** (*Specializzazione* è un attributo della sola entità figlia *Medico*)

DBG 22

21

22

2 - Accorpamento nelle entità figlie

Schema concettuale originale

Esame specialistico (1,1) Effettuato da (1,N) Medico (1,N) Volontario (t,e) Personale (0,N). Attributes: Nome, Cognome, Domicilio, CodFisc.

DBG 23

2 - Accorpamento nelle entità figlie

▷ Entità padre → entità figlie

Medico + ... Volontario + ... Personale (t,e) Personale (t,e)

DBG 24

23

24

2 - Accorpamento nelle entità figlie

▷ Attributi delle entità padre → entità figlie

DBM

25

25

2 - Accorpamento nelle entità figlie

▷ Relazioni con l'entità padre → entità figlie

DBM

26

26

2 - Accorpamento nelle entità figlie

▷ Tecnica sconsigliata per copertura **parziale**
(possibilità di trasformare la copertura *parziale* in copertura *totale* aggiungendo un'ulteriore entità figlia di tipo *Altri*)

▷ Tecnica non adatta per copertura **sovrapposta**
(difficile gestione degli **identificatori duplicati** relativi alle istanze comuni a più entità figlie)

DBM

27

27

3 - Aggiunta di relazioni

Schema concettuale originale

DBM

28

28

3 - Aggiunta di relazioni

▷ Generalizzazione ⇔ n relazioni
(una relazione per ogni entità figlia)

DBM

29

29

3 - Aggiunta di relazioni

▷ Identificatori della entità figlie

DBM

30

30

3 - Aggiunta di relazioni

⊃ Cardinalità delle nuove relazioni

(1,1) (1,1) (0,1) (0,1)

È un È un

(1,1)

DBM

31

31

3 - Aggiunta di relazioni

⊃ Soluzione più generale e sempre applicabile

- può essere dispendiosa per ricostruire l'informazione di partenza

DBM

32

32

Eliminazione delle generalizzazioni

Valutazione delle alternative

⊃ L'accorpamento delle entità figlie nell'entità padre è appropriato quando

- le entità figlie introducono differenziazioni non sostanziali (pochi valori nulli)
- le operazioni d'accesso non distinguono tra occorrenze dell'entità padre e delle figlie (accesso più efficiente)

DBM

33

33

Valutazione delle alternative

Valutazione delle alternative

⊃ L'accorpamento dell'entità padre nelle entità figlie è appropriato quando

- la generalizzazione è *totale*
- le operazioni di accesso sono *sostanzialmente diverse* per ciascuna entità figlia (l'accesso è più efficiente)

DBM

34

34

Valutazione delle alternative

Valutazione delle alternative

⊃ Sono possibili anche soluzioni "miste"

- le operazioni d'accesso distinguono tra occorrenze di alcune entità figlie (l'accesso è più efficiente)

⊃ Per le generalizzazioni a più livelli, si procede nello stesso modo, partendo dal livello inferiore

DBM

35

35

Progettazione logica relazionale

Ristrutturazione dello schema ER
3 - Partizionamento di concetti

DBM


36

36

Partizionamento di concetti

▷ Partizionamento di entità o relazioni

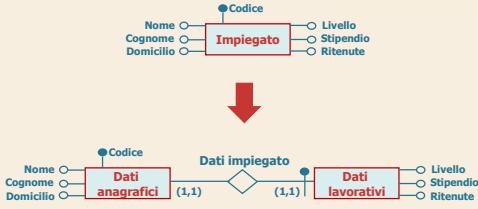

- rappresentazione migliore di concetti separati
- separazione di attributi di uno stesso concetto che sono utilizzati da operazioni diverse
- maggiore efficienza delle operazioni



37

37

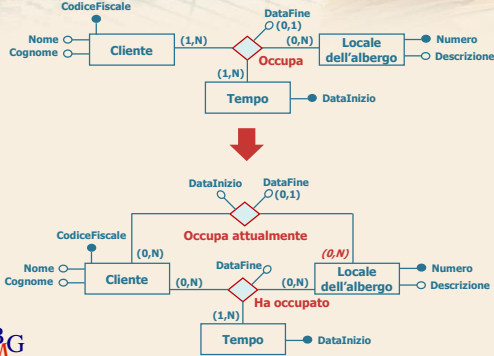

Partizionamento di entità

38

38

Partizionamento di relazioni


39

39

Progettazione logica relazionale

Ristrutturazione dello schema ER

4 - Eliminazione degli attributi multivaleore



40


40

Eliminazione degli attributi multivaleore

▷ Non sono rappresentabili nel modello relazionale

▷ L'attributo multivaleore è rappresentato mediante una nuova entità collegata da una relazione all'entità originale

- attenzione alla cardinalità della nuova relazione



41

41

Eliminazione degli attributi multivaleore

▷ Attributo multivaleore → 1 entità + 1 relazione N:N




42

42

Eliminazione degli attributi multivalore

▷ Attributo multivalore → 1 entità + 1 relazione 1:N

DBMG 43

43

Progettazione logica relazionale

Ristrutturazione dello schema ER
5 - Scelta degli identificatori primari

DBMG

44

Sceita degli identificatori primari

▷ Necessaria per definire la *chiave primaria* delle tabelle

▷ Un buon identificatore

- non assume mai valore nullo
- è costituito da pochi attributi (meglio 1)
- possibilmente è interno
- è utilizzato da molte operazioni d'accesso

▷ Può essere opportuno introdurre codici identificativi

DBMG 45

45

Progettazione logica relazionale

Traduzione nel modello relazionale

DBMG

46

Traduzione nel modello relazionale

▷ Si esegue sullo schema ER ristrutturato

- senza gerarchie e senza attributi multivalore

▷ Corrispondenze

- ad ogni entità corrisponde una tabella con gli stessi attributi
- per le relazioni occorre considerare la cardinalità massima

DBMG 47

47

Traduzione nel modello relazionale

▷ Modello ER ristrutturato ↔ Modello relazionale
Corrispondenza dei concetti principali

Modello ER		Modello ER ristrutturato		Modello relazionale
Entità	↔	Entità	↔	Relazione (tabella)
Relazione	↔	Relazione	↔	Riferimento (chiave esterna)
Generalizzazione	×	Non presente	×	Non presente
Attributo semplice	↔	Attributo semplice	↔	Attributo (campo)
Attributo multivalore	×	Non presente	×	Non presente

DBMG 48

48



Progettazione logica relazionale

Traduzione nel modello relazionale
1 - Entità (con identificatore interno)



49

Traduzione nel modello relazionale - Entità

Traduzione di **entità**

Ogni *entità* corrisponde a una *tabella* con gli stessi attributi dell'entità derivata

- la chiave primaria è sottolineata
- gli attributi opzionali sono indicati con asterisco



`Persona(CodiceFiscale, Nome, Cognome, Professione*)`



50

Traduzione nel modello relazionale - Entità

Treatment of *attributi composti* (metodo 1):
utilizzo di **TIPI personalizzati**

(nella maggior parte dei DBMS è possibile definire **TIPI** personalizzati di variabili costituiti da un insieme di più variabili di *tipo elementare* e/o di *tipo personalizzato*)

Esempio:



`Persona(..., Indirizzo)`
TIPO Indirizzo(Via, Numero civico, CAP)



51

Traduzione nel modello relazionale - Entità


Treatment of *attributi composti* (metodo 2):
rappresentazione separata degli attributi componenti

(questa rappresentazione può essere conveniente se è necessario **accedere separatamente** a ciascun attributo)

Esempio:



`Persona(..., Via, Numero civico, CAP)`



52



Progettazione logica relazionale

Traduzione nel modello relazionale
2 - Relazioni



53

Traduzione nel modello relazionale - Relazioni

Relazione tra n entità nel modello E-R:
insieme di n -uple ordinate contenenti un'istanza I per ciascuna delle n entità partecipanti

Relazione nel modello relazionale:
insieme di n -uple ordinate contenenti un elemento E per ciascuno degli n attributi (dove E appartiene al dominio del corrispondente attributo)


- nel modello relazionale la relazione viene usualmente denominata *tabella* (che è in realtà la più diffusa *rappresentazione grafica* di una relazione)



54

Traduzione nel modello relazionale - Relazioni


- ▷ Nel modello relazionale **non esiste un concetto del tutto equivalente** al concetto di *relazione ER*
- ▷ Esiste però un **concetto corrispondente**, che può essere utilizzato nel modello relazionale per rappresentare le *relazioni ER*:
 il concetto di **riferimento**, realizzabile con lo strumento della **chiave esterna**



55

Traduzione nel modello relazionale - Relazioni


- ▷ Traduzione di **relazioni binarie 1:1**
 Sono possibili più traduzioni
 - dipende dal valore della cardinalità minima



56

Traduzione nel modello relazionale - Relazioni


- ▷ Traduzione di **relazioni binarie 1:1**
 Cardinalità minima 1 da entrambi i lati



↓

Rettore(Matricola, ..., NomeUniversità, DataElezione)
 ↓ chiave esterna
 Università(NomeUniversità, ...)


oppure: stesso procedimento applicato *simmetricamente* rispetto alle due entità Rettore, Università



57


Traduzione nel modello relazionale - Relazioni

- ▷ Traduzione di **relazioni binarie 1:1**
 Cardinalità minima 0 da uno dei due lati



↓


Università(NomeUniversità, ..., Matricola, DataElezione)
 ↓ chiave esterna
 Professore(Matricola, ...)



58

Traduzione nel modello relazionale - Relazioni


- ▷ Traduzione di **relazioni binarie 1:1**
 Cardinalità minima 0 da entrambi i lati



↓

Rettore(Matricola, NomeUniversità, DataElezione)
 ↓ anche chiave esterna
 Professore(Matricola, ...)
 Università(NomeUniversità, ...)

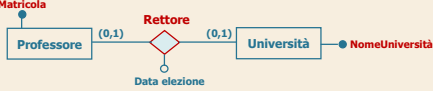
oppure: stesso procedimento applicato *simmetricamente* rispetto alle due entità Professore, Università



59

Traduzione nel modello relazionale - Relazioni


- ▷ Traduzione di **relazioni binarie 1:1**
 Cardinalità minima 0 da entrambi i lati



↓

Università(Nome, Città, Matricola*, DataElezione*)
 ↓ chiave esterna
 Professore(Matricola, Nome, Cognome)

oppure: stesso procedimento applicato *simmetricamente* rispetto alle due entità Professore, Università




60

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie 1:N**

Sono possibili due modalità di traduzione:


- mediante attributi
- mediante una nuova tabella



61

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie 1:N** mediante attributi




↓

Persona(..., NomeComune, DataTrasferimento)

↓ chiave esterna

Comune(NomeComune, ...)



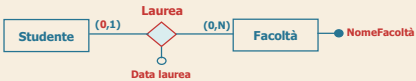
62

61

62

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie 1:N** mediante attributi




↓

Studente(..., NomeFacoltà*, DataLaurea*)

↓ chiave esterna


Facoltà(NomeFacoltà, Città)



63

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie 1:N** mediante una nuova tabella




↓

Studente(...)

Laurea(Matricola, NomeFacoltà, DataLaurea)

↓ chiave esterna

Facoltà(NomeFacoltà, ...)



64

63


64

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie N:N**

Ogni *relazione N:N* corrisponde a una *tabella*

- la *chiave primaria* è la *lista degli identificatori* delle due entità collegate
- è possibile *ridenominare* gli attributi della tabella che corrisponde alla relazione (necessario in caso di relazioni *ricorsive*)



65

Traduzione nel modello relazionale - Relazioni

▷ Traduzione di **relazioni binarie N:N**



↓

Studente(Matricola, ...)

Corso(CodCorso, ...)

Esame(Matricola, CodCorso, Voto)



66

65

66

Traduzione nel modello relazionale - Relazioni

Traduzione di **relazioni binarie ricorsive N:N**

↓

Prodotto(CodP, ...)
 Composizione(CodComposto, CodComponente, Quantità)

DBG 67

67

Traduzione nel modello relazionale - Relazioni

Traduzione di **relazioni ternarie**

↓

Studente(Matricola, ...)
 Corso(Codice, ...)
 Tempo(Data)
 Esame(Matricola, Codice, Data, Voto)

DBG 68

68

Progettazione logica relazionale

Traduzione nel modello relazionale
3 - Entità con identificatore esterno

DBG 69

69

Traduzione nel modello relazionale Entità con identificatore esterno

↓

Studente(Matricola, NomeUniversità, ...)
 ↳ chiave esterna
 Università(NomeUniversità, ...)

La relazione **Iscrizione** è realizzata all'interno dell'identificatore di Studente con la chiave esterna **NomeUniversità**

DBG 70

70

Traduzione nel modello relazionale Entità con identificatore esterno

Identificatori esterni a più livelli

↓

Studente(Matricola, NomeUniversità, NomeNazione, ...)
 ↳ + ↳ chiave esterna
 Università(NomeUniversità, NomeNazione, ...)
 ↳ ↳ chiave esterna
 Nazione(NomeNazione, ...)

DBG 71

71

Progettazione logica relazionale

Traduzione nel modello relazionale
4 - Vincoli d'integrità referenziale

DBG 72

72

Traduzione nel modello relazionale Integrità referenziale

Le relazioni rappresentano *vincoli di integrità referenziale*

```

    graph LR
      S[Studente] ---|"(0,N)"| E{Esame}
      E ---|"(0,N)"| C[Corso]
      S --- Nome1[Nome]
      S --- Cognome[Cognome]
      C --- CodCorso[CodCorso]
      C --- Nome2[Nome]
      E --- Voto[Voto]
  
```

Studente(Matricola, Nome, Cognome)
 Corso(CodCorso, Nome)
 Esame(Matricola, CodCorso, Voto)

Vincoli d'integrità referenziale

Esame(Matricola) REFERENCES Studente(Matricola)
 Esame(CodCorso) REFERENCES Corso(CodCorso)

DBG 73

73

Integrità referenziale: relazione Esame

Schema relazionale

Studente(Matricola, Nome, Cognome)
 Corso(CodCorso, Nome)
 Esame(Matricola, CodCorso, Voto)

Vincoli d'integrità referenziale

Esame(Matricola) REFERENCES Studente(Matricola)
 Esame(CodCorso) REFERENCES Corso(CodCorso)

Studente(Matricola, Nome, Cognome)
 ↑
 Esame(Matricola, CodCorso, Voto)
 ↓
 Corso(CodCorso, Nome)

DBG 74

74