



Progettazione di basi di dati

Progettazione logica relazionale

DBG

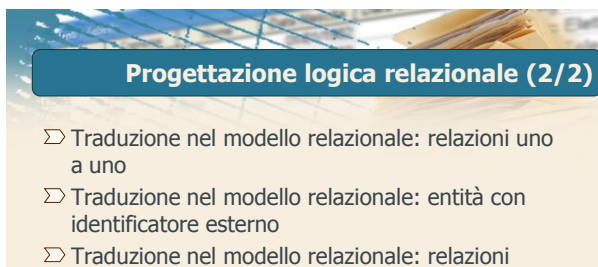


Progettazione logica relazionale (1/2)

- ▷ Introduzione
- ▷ Ristrutturazione dello schema ER
- ▷ Eliminazione delle gerarchie
- ▷ Partizionamento di concetti
- ▷ Eliminazione degli attributi multivalore
- ▷ Eliminazione degli attributi composti e scelta degli identificatori primari
- ▷ Traduzione nel modello relazionale: entità e relazioni molti a molti
- ▷ Traduzione nel modello relazionale: relazioni uno a molti

DBG

2



Progettazione logica relazionale (2/2)

- ▷ Traduzione nel modello relazionale: relazioni uno a uno
- ▷ Traduzione nel modello relazionale: entità con identificatore esterno
- ▷ Traduzione nel modello relazionale: relazioni ternarie

DBG

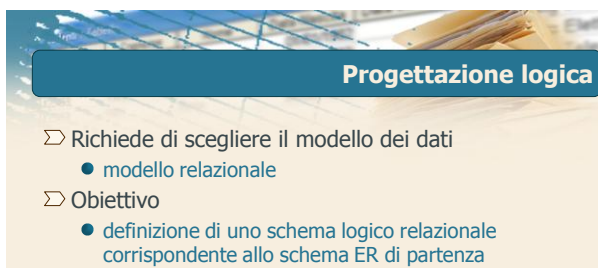
3



Progettazione logica relazionale

Introduzione

DBG




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

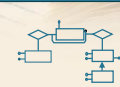
DBG

5



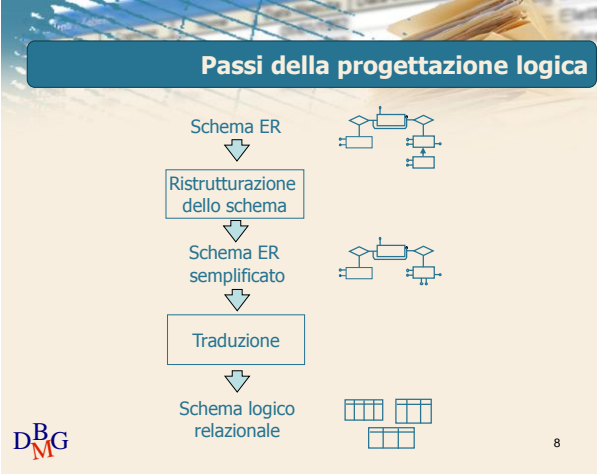
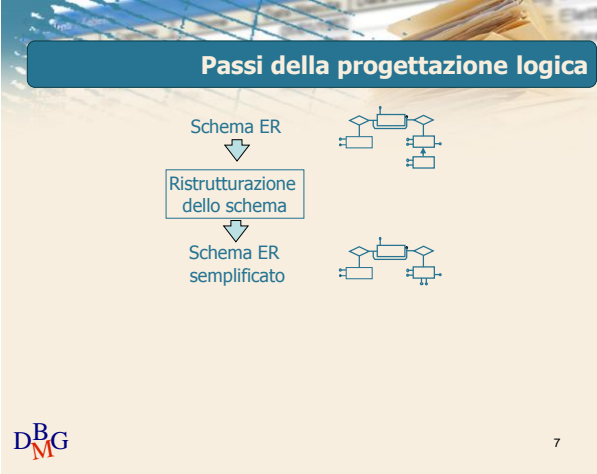
Passi della progettazione logica

Schema ER



DBG

6



Progettazione logica relazionale

Ristrutturazione dello schema ER

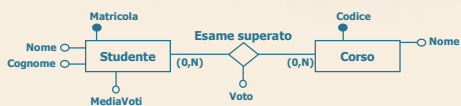
10

- ### 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
- 10

- ### Attività di ristrutturazione
- ▷ Analisi delle ridondanze
 - ▷ Eliminazione delle generalizzazioni
 - ▷ Partizionamento e accorpamento di entità e relazioni
 - ▷ Scelta degli identificatori primari
- 11

- ### 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
- 12

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



13

Progettazione logica relazionale

Eliminazione delle gerarchie



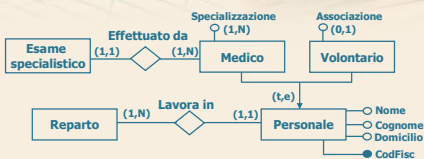
Eliminazione delle gerarchie

- ▷ 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



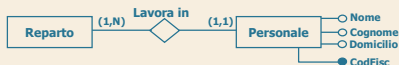
15

Esempio



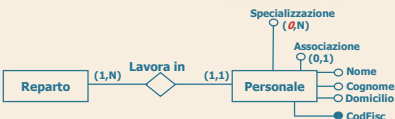
16

Accorpamento nel padre

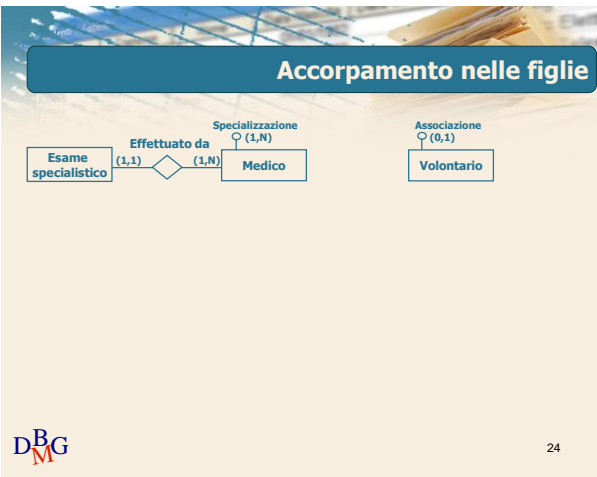
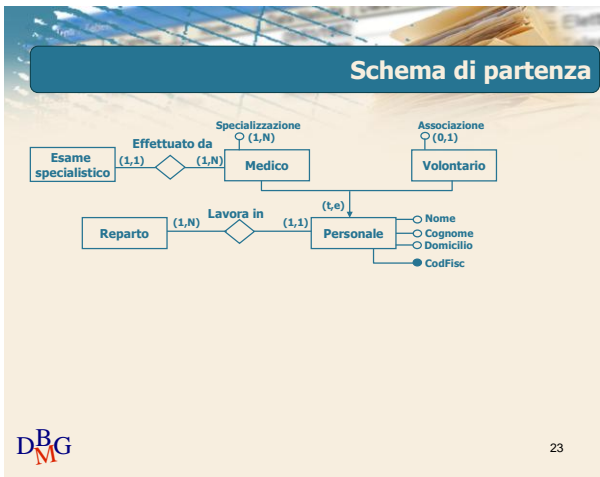
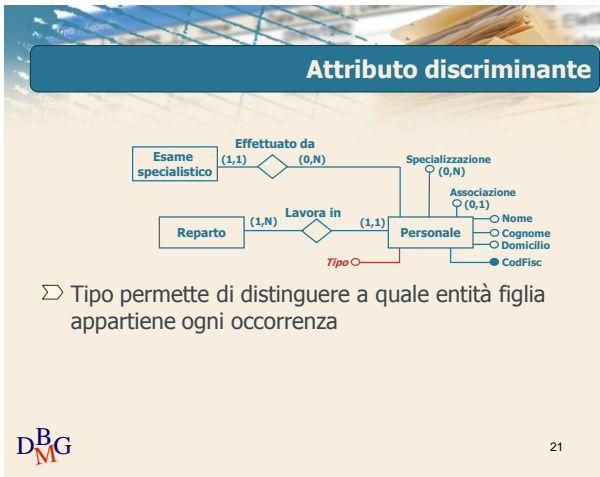


17

Attributi delle entità figlie



18



Attributi del padre

25

Relazioni con il padre

⇒ Occorre sdoppiare le relazioni con l'entità padre

26

Cardinalità della relazione Lavora in

⇒ Occorre sdoppiare le relazioni con l'entità padre

27

Accorpamento nelle figlie

⇒ Non adatta per copertura parziale o sovrapposta

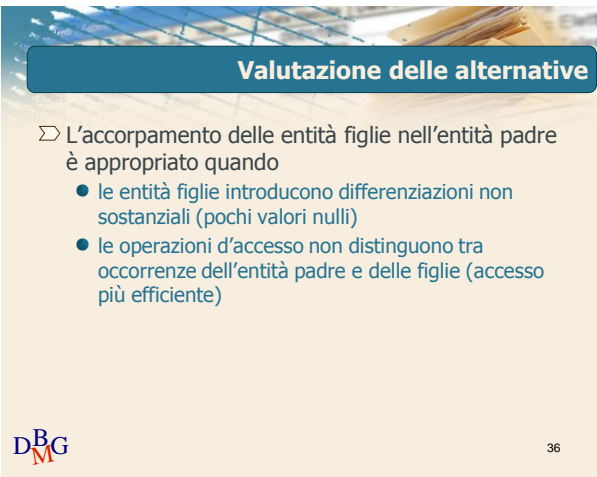
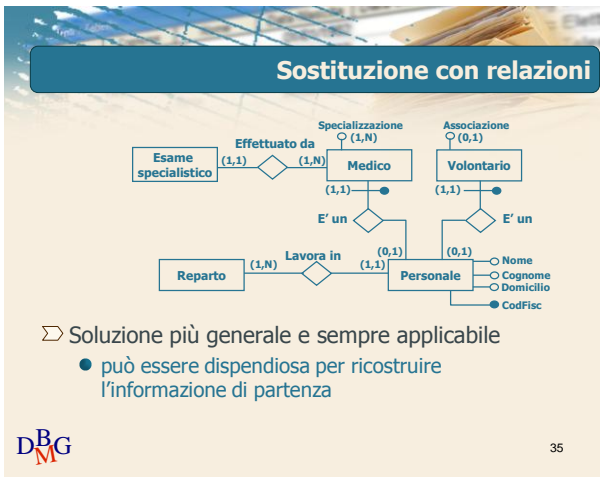
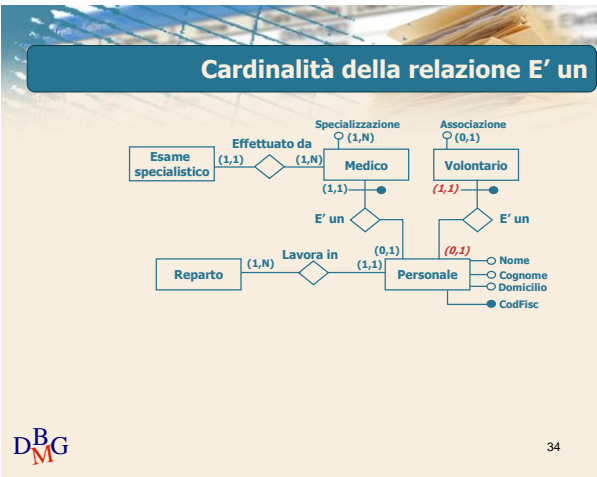
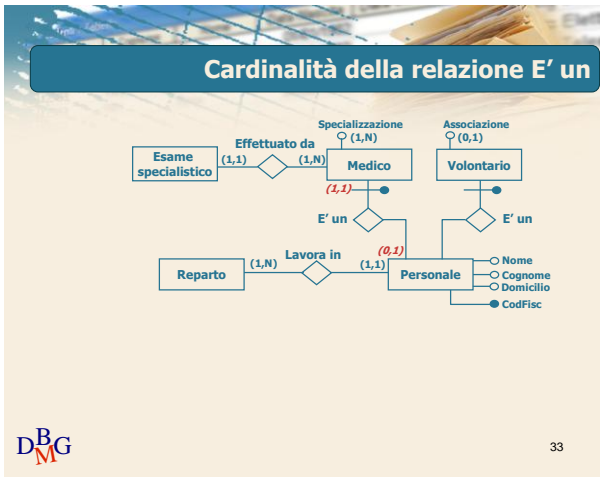
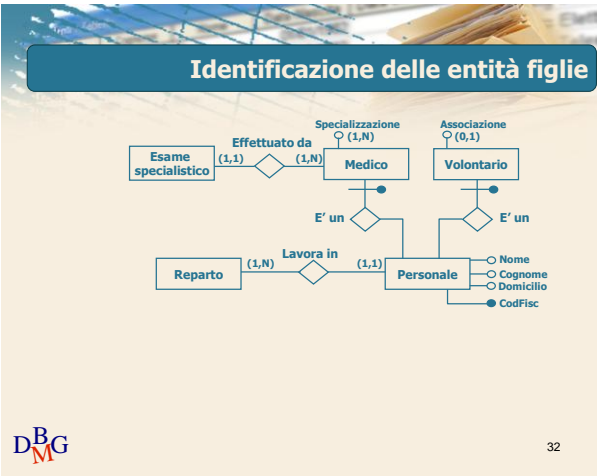
28

Schema di partenza

29

Sostituzione con relazioni

30



Valutazione delle alternative

- ▷ L'accorpamento dell'entità padre nelle entità figlie è appropriato quando
 - la generalizzazione è totale
 - le operazioni d'accesso distinguono tra occorrenze delle diverse entità figlie (accesso più efficiente)

Valutazione delle alternative

- ▷ Sono possibili anche soluzioni "miste"
 - le operazioni d'accesso distinguono tra occorrenze di alcune entità figlie (accesso più efficiente)

Valutazione delle alternative

- ▷ Sono possibili anche soluzioni "miste"
 - le operazioni d'accesso distinguono tra occorrenze di alcune entità figlie (accesso più efficiente)
- ▷ Per le generalizzazioni a più livelli, si procede nello stesso modo, partendo dal livello inferiore

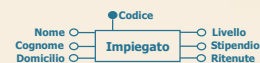
Progettazione logica relazionale

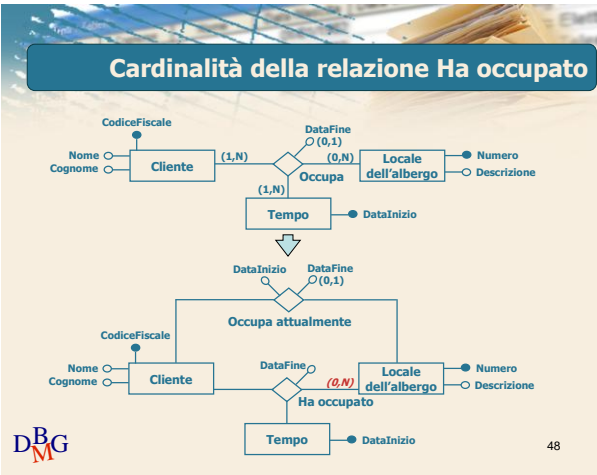
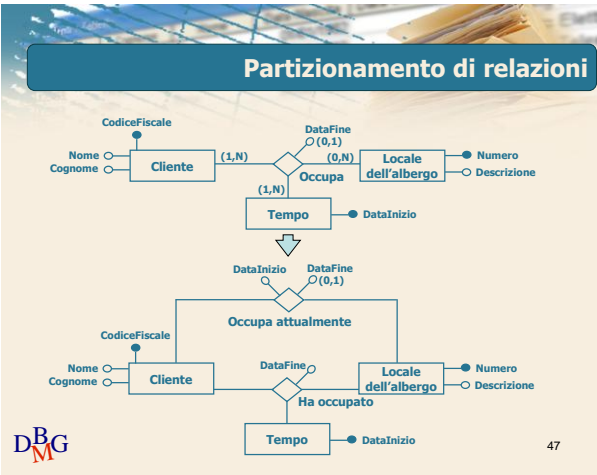
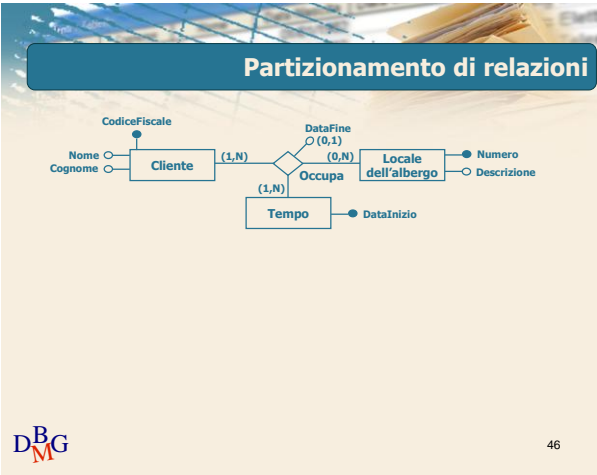
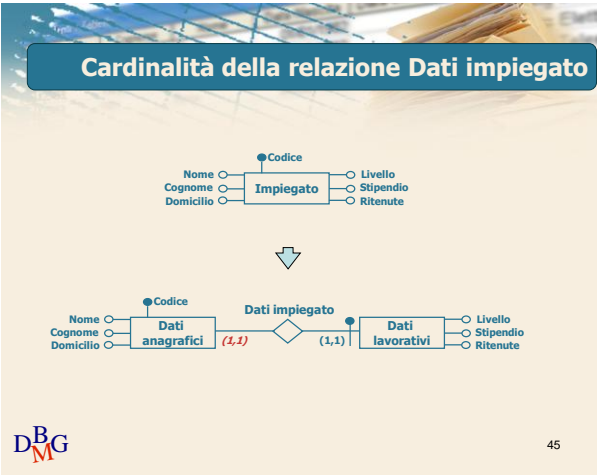
Partizionamento dei concetti

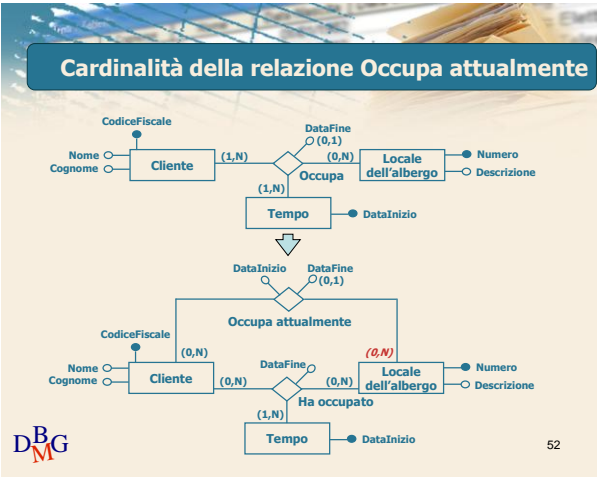
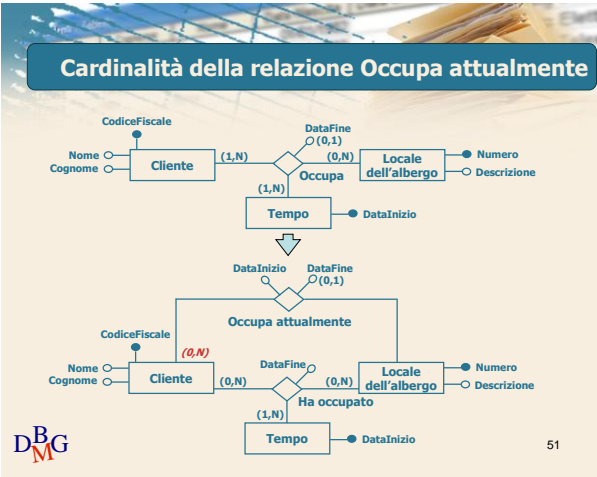
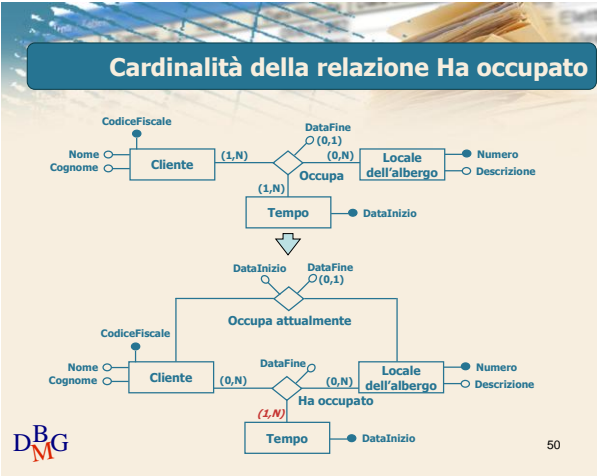
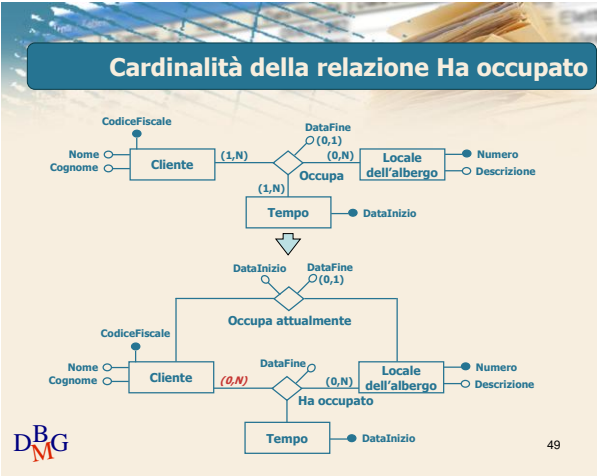
Partizionamento di concetti

- ▷ 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

Partizionamento di entità







Progettazione logica relazionale

Eliminazione degli attributi multivalore

DBM

Eliminazione degli attributi multivalore

Non sono rappresentabili nel modello relazionale

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

- attenzione alla cardinalità della nuova relazione

DBM

54

Eliminazione degli attributi multivale

Diagram illustrating the elimination of multivalued attributes from a database schema. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Titolo di studio** (1,N). The transformation process is indicated by a downward arrow.

DBG

55

Eliminazione degli attributi multivale

Diagram illustrating the elimination of multivalued attributes from a database schema. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Titolo di studio** (1,N). The transformation process is indicated by a downward arrow. The resulting schema shows a relationship **Ha conseguito** between **Persona** and **Titolo di studio** (Titolo).

DBG

56

Cardinalità della relazione Ha conseguito

Diagram illustrating the cardinality of the relationship **Ha conseguito**. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Titolo di studio** (1,N). The transformation process is indicated by a downward arrow. The resulting schema shows a relationship **Ha conseguito** between **Persona** (1,N) and **Titolo di studio** (Titolo).

DBG

57

Cardinalità della relazione Ha conseguito

Diagram illustrating the cardinality of the relationship **Ha conseguito**. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Titolo di studio** (1,N). The transformation process is indicated by a downward arrow. The resulting schema shows a relationship **Ha conseguito** between **Persona** (1,N) and **Titolo di studio** (1,N) (Titolo).

DBG

58

Eliminazione degli attributi multivale

Diagram illustrating the elimination of multivalued attributes from a database schema. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Telefono** (1,N). The transformation process is indicated by a downward arrow.

DBG

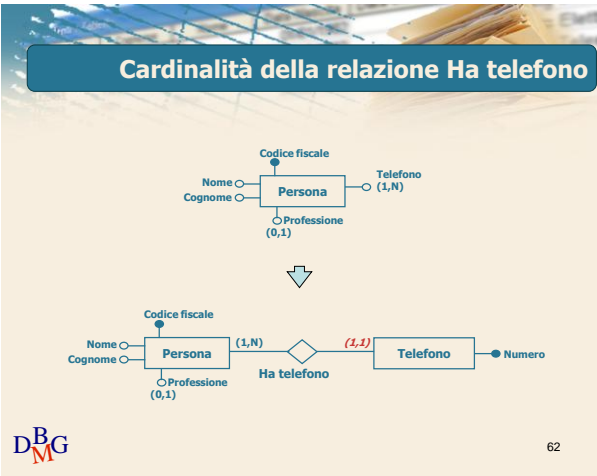
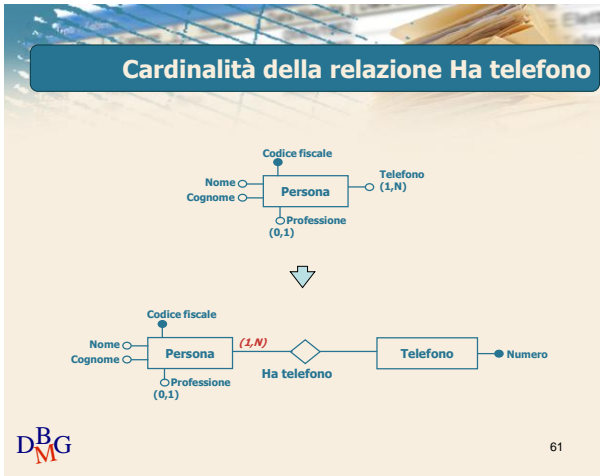
59

Eliminazione degli attributi multivale

Diagram illustrating the elimination of multivalued attributes from a database schema. The initial schema shows an entity **Persona** with attributes: **Codice fiscale** (primary key), **Nome**, **Cognome**, **Professione** (0,1), and **Telefono** (1,N). The transformation process is indicated by a downward arrow. The resulting schema shows a relationship **Ha telefono** between **Persona** and **Telefono** (Numero).

DBG

60



Progettazione logica relazionale

Eliminazione degli attributi composti
e scelta degli identificatori primari

DBG

Eliminazione degli attributi composti

- Non sono rappresentabili nel modello relazionale
- Due alternative

DBG

64

Eliminazione degli attributi composti

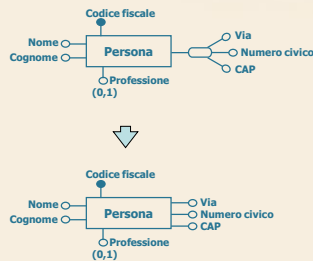
- Non sono rappresentabili nel modello relazionale
- Due alternative
 - si rappresentano in modo separato gli attributi componenti
 - adatto se è necessario accedere separatamente a ciascun attributo

DBG

65



Rappresentazione separata degli attributi



DBG

67

Eliminazione degli attributi composti

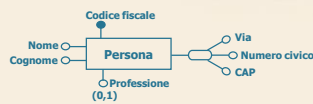
- ⊃ Non sono rappresentabili nel modello relazionale
- ⊃ Due alternative

- si rappresentano in modo separato gli attributi componenti
 - adatta se è necessario accedere separatamente a ciascun attributo
- si introduce un unico attributo che rappresenta la concatenazione degli attributi componenti
 - adatta se è sufficiente l'accesso all'informazione complessiva

DBG

68

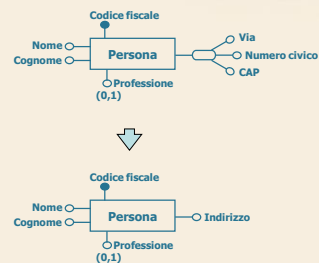
Rappresentazione con un attributo unico



DBG

69

Rappresentazione con un attributo unico



DBG

70

Scelta degli identificatori primari

- ⊃ Necessaria per definire la chiave primaria delle tabelle
- ⊃ Un buon identificatore
 - non assume 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

DBG

71

Progettazione logica relazionale

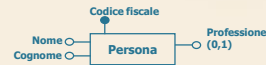
Traduzione nel modello relazionale:
entità e relazioni molti a molti

DBG

Traduzione nel modello relazionale

- ▷ Si esegue sullo schema ER ristrutturato
 - senza gerarchie, attributi multivalore e composti
- ▷ Trasformazioni
 - ad ogni entità corrisponde una tabella con gli stessi attributi
 - per le relazioni occorre considerare la cardinalità massima

Traduzione di entità



Traduzione di entità



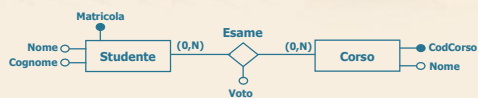
Persona(CodiceFiscale, Nome, Cognome, Professione*)

- ▷ Chiave primaria sottolineata
- ▷ Attributi opzionali indicati con asterisco

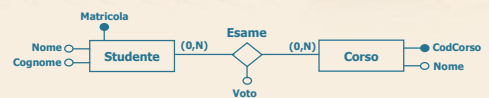
Traduzione di relazioni binarie molti a molti

- ▷ Ogni relazione molti a molti corrisponde a una tabella
 - la chiave primaria è la combinazione degli identificatori delle due entità collegate
 - è possibile ridenominare gli attributi della tabella che corrisponde alla relazione (necessario in caso di relazioni ricorsive)

Relazione binaria molti a molti

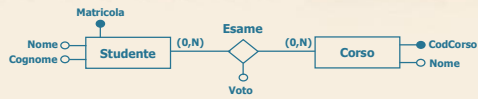


Relazione binaria molti a molti: entità



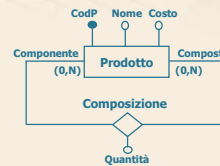
Studiante(Matricola, Nome, Cognome)
Corso(CodCorso, Nome)

Relazione binaria molti a molti

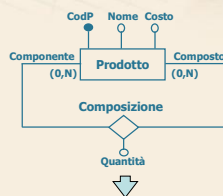


Studente(Matricola, Nome, Cognome)
Corso(CodCorso, Nome)
Esame(Matricola, CodCorso, Voto)

Relazione binaria molti a molti ricorsiva

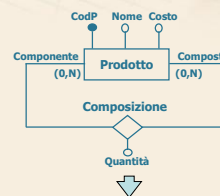


Relazione binaria molti a molti ricorsiva



Prodotto(CodP, Nome, Costo)

Relazione binaria molti a molti ricorsiva



Prodotto(CodP, Nome, Costo)
Composizione(CodComposto, CodComponente, Quantità)

Progettazione logica relazionale

Traduzione nel modello relazionale:
relazioni uno a molti

Relazione binaria uno a molti

- ⇒ Sono possibili due modalità di traduzione
- mediante attributi
 - mediante una nuova tabella

Relazione binaria uno a molti

Persona (Nome, Cognome, Codice fiscale) —(1,1) Residenza (1,N) Comune (NomeComune, Provincia)

Data trasferimento

DBG

85

Relazione binaria uno a molti: entità

Persona (Nome, Cognome, Codice fiscale) —(1,1) Residenza (1,N) Comune (NomeComune, Provincia)

Data trasferimento

↓

Persona(CodiceFiscale, Nome, Cognome)

Comune(NomeComune, Provincia)

DBG

86

Relazione binaria uno a molti

Persona (Nome, Cognome, Codice fiscale) —(1,1) Residenza (1,N) Comune (NomeComune, Provincia)

Data trasferimento

↓

Persona(CodiceFiscale, Nome, Cognome, NomeComune)

Comune(NomeComune, Provincia)

DBG

87

Relazione binaria uno a molti

Persona (Nome, Cognome, Codice fiscale) —(1,1) Residenza (1,N) Comune (NomeComune, Provincia)

Data trasferimento

↓

Persona(CodiceFiscale, Nome, Cognome, NomeComune, DataTrasferimento)

Comune(NomeComune, Provincia)

DBG

88

Relazione binaria uno a molti

Studente (Nome, Cognome, Matricola) —(0,1) Laurea (0,N) Facoltà (NomeFacoltà, Città)

Data laurea

DBG

89

Relazione binaria uno a molti: alternativa n.1

Studente (Nome, Cognome, Matricola) —(0,1) Laurea (0,N) Facoltà (NomeFacoltà, Città)

Data laurea

↓

Studente(Matricola, Nome, Cognome)

Facoltà(NomeFacoltà, Città)

DBG

90

Relazione binaria uno a molti: alternativa n.1

Studente(Matricola, Nome, Cognome)
Facoltà(NomeFacoltà, Città)
Laurea(Matricola, NomeFacoltà, DataLaurea)

DBG

91

Relazione binaria uno a molti: alternativa n.2

Studente(Matricola, Nome, Cognome, NomeFacoltà*, DataLaurea*)
Facoltà(NomeFacoltà, Città)

DBG

92

Progettazione logica relazionale

Traduzione nel modello relazionale:
relazioni uno a uno

DBG

Relazione binaria uno a uno

⊃ Sono possibili più traduzioni

- dipende dal valore della cardinalità minima

DBG

94

Relazione binaria uno a uno: caso 1

⊃ Partecipazione obbligatoria da entrambi i lati

Rettore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

DBG

95

Relazione binaria uno a uno: alternativa n.1

⊃ Partecipazione obbligatoria da entrambi i lati

Rettore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

DBG

96

Relazione binaria uno a uno: alternativa n.1

Partecipazione obbligatoria da entrambi i lati

```
graph LR
    Rettore[Matricola, Nome, Cognome] --- "(1,1)" E_Rettore{E' Rettore}
    E_Rettore --- "(1,1)" Università[NomeUniversità, Città]
    E_Rettore --> Data_elezione[Data elezione]
```

Rettore(Matricola, Nome, Cognome, *NomeUniversità*, *DataElezione*)
Università(NomeUniversità, Città)

DBG

97

Relazione binaria uno a uno: alternativa n.2

Partecipazione obbligatoria da entrambi i lati

```
graph LR
    Rettore[Matricola, Nome, Cognome] --- "(1,1)" E_Rettore{E' Rettore}
    E_Rettore --- "(1,1)" Università[NomeUniversità, Città]
    E_Rettore --> Data_elezione[Data elezione]
```

Rettore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

DBG

98

Relazione binaria uno a uno: alternativa n.2

Partecipazione obbligatoria da entrambi i lati

```
graph LR
    Rettore[Matricola, Nome, Cognome] --- "(1,1)" E_Rettore{E' Rettore}
    E_Rettore --- "(1,1)" Università[NomeUniversità, Città, Matricola]
    E_Rettore --> Data_elezione[Data elezione]
```

Rettore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città, *Matricola*, *DataElezione*)

DBG

99

Relazione binaria uno a uno: caso 2

Partecipazione opzionale da un lato

```
graph LR
    Professore[Matricola, Nome, Cognome] --- "(0,1)" Rettore{Rettore}
    Rettore --- "(1,1)" Università[NomeUniversità, Città]
    Rettore --> Data_elezione[Data elezione]
```

DBG

100

Relazione binaria uno a uno: entità

Partecipazione opzionale da un lato

```
graph LR
    Professore[Matricola, Nome, Cognome] --- "(0,1)" Rettore{Rettore}
    Rettore --- "(1,1)" Università[NomeUniversità, Città]
    Rettore --> Data_elezione[Data elezione]
```

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

DBG

101

Relazione binaria uno a uno

Partecipazione opzionale da un lato

```
graph LR
    Professore[Matricola, Nome, Cognome] --- "(0,1)" Rettore{Rettore}
    Rettore --- "(1,1)" Università[NomeUniversità, Città, Matricola]
    Rettore --> Data_elezione[Data elezione]
```

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città, *Matricola*, *DataElezione*)

DBG

102

Relazione binaria uno a uno: caso 3

Partecipazione opzionale da entrambi i lati

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

Relazione binaria uno a uno: alternativa n.1

Partecipazione opzionale da entrambi i lati

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)

Relazione binaria uno a uno: alternativa n.1

Partecipazione opzionale da entrambi i lati

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)
Rettore(Matricola, NomeUniversità, DataElezione)

Relazione binaria uno a uno: alternativa n.2

Partecipazione opzionale da entrambi i lati

Professore(Matricola, Nome, Cognome)
Università(NomeUniversità, Città)
Rettore(Matricola, NomeUniversità, DataElezione)

Relazione binaria uno a uno: alternativa n.3

Partecipazione opzionale da entrambi i lati

Professore(Matricola, Nome, Cognome)
Università(Nome, Città)

Relazione binaria uno a uno: alternativa n.3

Partecipazione opzionale da entrambi i lati


Professore(Matricola, Nome, Cognome)
Università(Nome, Città, Matricola*, DataElezione*)



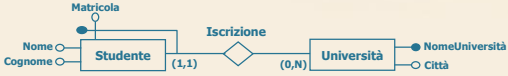
Progettazione logica relazionale


Traduzione nel modello relazionale:
entità con identificatore esterno





Entità con identificatore esterno

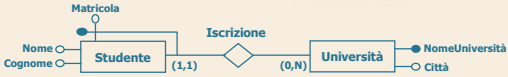




110




Entità con identificatore esterno



↓

Università(NomeUniversità, Città)
Studiante(Matricola, NomeUniversità, Nome, Cognome)



111



Entità con identificatore esterno



↓

Università(NomeUniversità, Città)
Studiante(Matricola, NomeUniversità, Nome, Cognome)

⊃ La relazione è rappresentata insieme all'identificatore




112



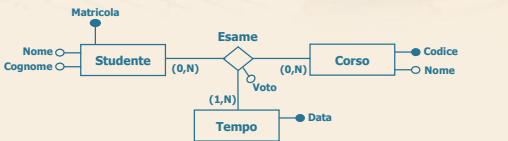
Progettazione logica relazionale


Traduzione nel modello relazionale:
relazioni ternarie





Relazione ternaria





114

Relazione ternaria: entità

Studente(Matricola, Nome, Cognome)
Corso(Codice, Nome)
Tempo(Data)

Relazione ternaria: identificatore

Studente(Matricola, Nome, Cognome)
Corso(Codice, Nome)
Tempo(Data)
Esame(Matricola, Codice, Data)

Relazione ternaria: attributi

Studente(Matricola, Nome, Cognome)
Corso(Codice, Nome)
Tempo(Data)
Esame(Matricola, Codice, Data, Voto)