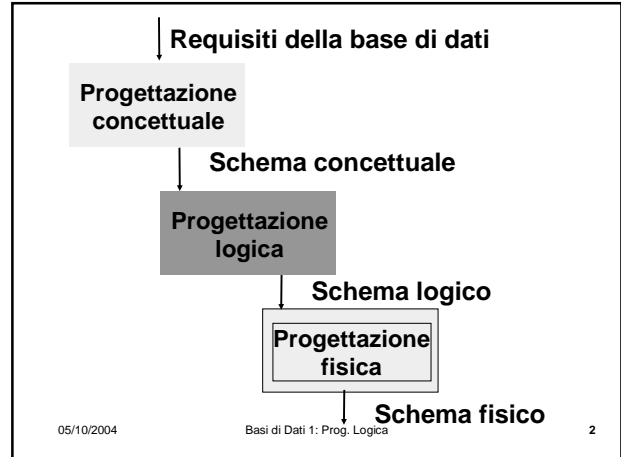


E. Giunchiglia
Basi di dati 1

(trasparenze basate su Atzeni, Ceri, Paraboschi, Torlone: Basi di dati, Capitolo 8)

Progettazione logica

05/10/2004



**Obiettivo della
progettazione logica**

- "tradurre" lo schema concettuale in uno schema logico che rappresenti gli stessi dati in maniera corretta ed efficiente

05/10/2004

Basi di Dati 1: Prog. Logica

3

Dati di ingresso e uscita

- Ingresso:
 - schema concettuale
 - informazioni sul carico applicativo
 - modello logico
- Uscita:
 - schema logico
 - documentazione associata

05/10/2004

Basi di Dati 1: Prog. Logica

4

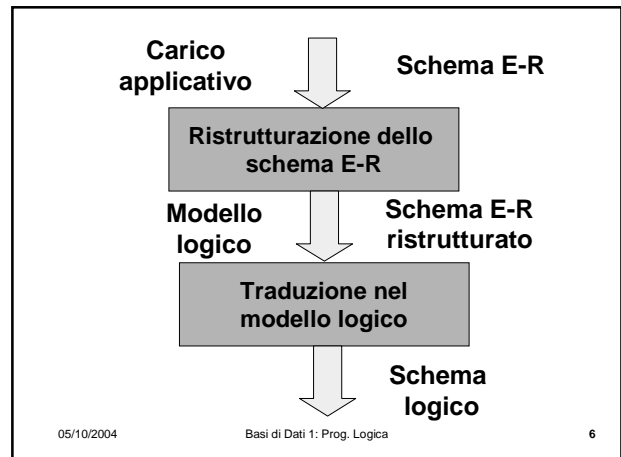
**Non si tratta di una pura e semplice
traduzione**

- alcuni aspetti non sono direttamente rappresentabili
- è necessario considerare le prestazioni

05/10/2004

Basi di Dati 1: Prog. Logica

5



Ristrutturazione schema E-R

- **Motivazioni:**
 - semplificare la traduzione
 - "ottimizzare" le prestazioni
- **Osservazione:**
 - uno schema E-R ristrutturato non è (più) uno schema concettuale nel senso stretto del termine

05/10/2004

Basi di Dati 1: Prog. Logica

7

Per ottimizzare il risultato abbiamo bisogno di analizzare le prestazioni a questo livello

- **Ma:**
 - le prestazioni non sono valutabili con precisione su uno schema concettuale!

05/10/2004

Basi di Dati 1: Prog. Logica

8

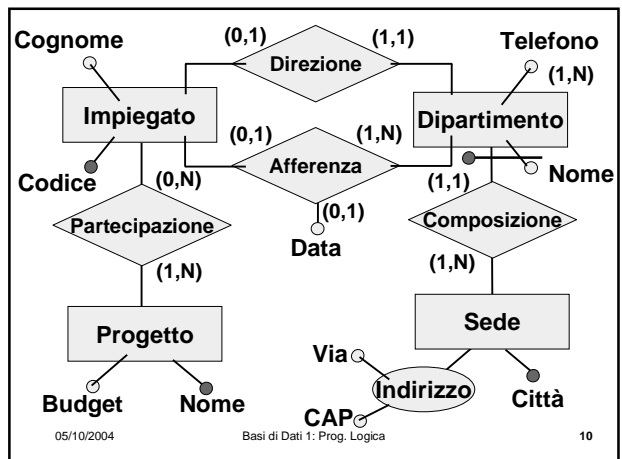
Consideriamo "indicatori" dei parametri che regolano le prestazioni

- **spazio:** numero di occorrenze previste
- **tempo:** numero di occorrenze (di entità e relationship) visitate durante un'operazione

05/10/2004

Basi di Dati 1: Prog. Logica

9



05/10/2004

Basi di Dati 1: Prog. Logica

10

Tavola dei volumi

Concetto	Tipo	Volume
Sede	E	10
Dipartimento	E	80
Impiegato	E	2000
Progetto	E	500
Composizione	R	80
Afferenza	R	1900
Direzione	R	80
Partecipazione	R	6000

05/10/2004

Basi di Dati 1: Prog. Logica

11

Esempio di valutazione di costo

- **Operazione:**
 - trova tutti i dati di un impiegato, del dipartimento nel quale lavora e dei progetti ai quali partecipa
- Si costruisce una tavola degli accessi basata su uno schema di navigazione

05/10/2004

Basi di Dati 1: Prog. Logica

12

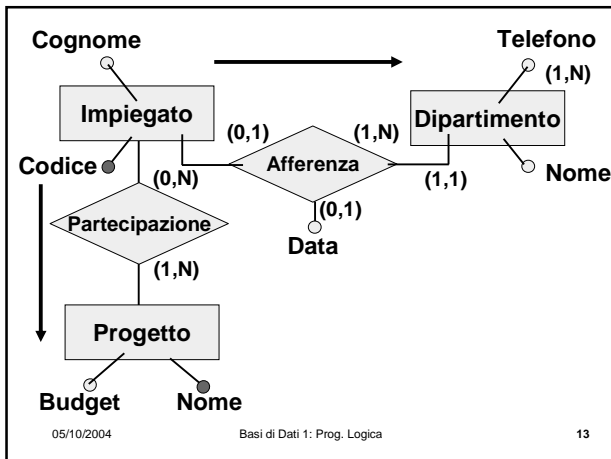


Tavola degli accessi

Concetto	Costrutto	Accessi	Tipo
Impiegato	Entità	1	L
Afferenza	Relazione	1	L
Dipartimento	Entità	1	L
Partecipazione	Relazione	3	L
Progetto	Entità	3	L

05/10/2004 Basi di Dati 1: Prog. Logica 14

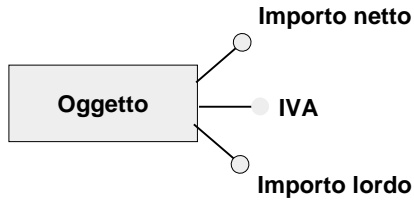
- ### Attività della ristrutturazione
- **Analisi delle ridondanze**
 - **Eliminazione delle generalizzazioni**
 - **Partizionamento/accorpamento di entità e relationship**
 - **Scelta degli identificatori primari**
- 05/10/2004 Basi di Dati 1: Prog. Logica 15

- ### Analisi delle ridondanze
- **Una ridondanza in uno schema E-R è una informazione significativa ma derivabile da altre**
 - **in questa fase si decide se eliminare le ridondanze eventualmente presenti o mantenerle**
- 05/10/2004 Basi di Dati 1: Prog. Logica 16

- ### Ridondanze
- **Vantaggi**
 - **semplificazione delle interrogazioni**
 - **Svantaggi**
 - **appesantimento degli aggiornamenti**
 - **maggiore occupazione di spazio**
- 05/10/2004 Basi di Dati 1: Prog. Logica 17

- ### Forme di ridondanza in uno schema E-R
- **attributi derivabili:**
 - **da altri attributi della stessa entità (o relazione)**
 - **da attributi di altre entità (o relazioni)**
 - **relazioni derivabili dalla composizione di altre relazioni in presenza di cicli**
- 05/10/2004 Basi di Dati 1: Prog. Logica 18

Attributo derivabile

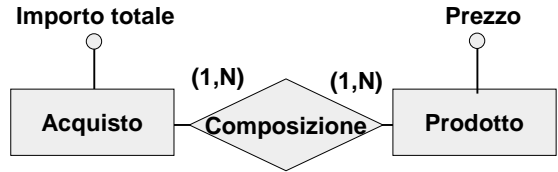


05/10/2004

Basi di Dati 1: Prog. Logica

19

Attributo derivabile da altra entità

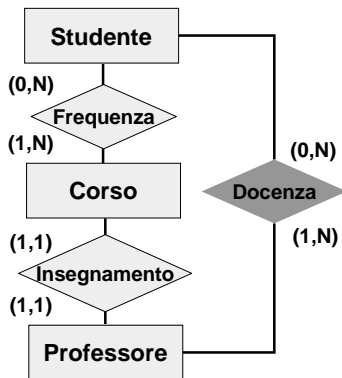


05/10/2004

Basi di Dati 1: Prog. Logica

20

Ridondanza dovuta a ciclo

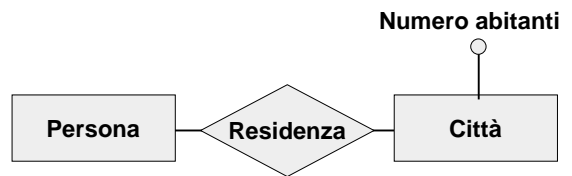


05/10/2004

Basi di Dati 1: Prog. Logica

21

Analisi di una ridondanza



05/10/2004

Basi di Dati 1: Prog. Logica

22

Concetto	Tipo	Volume
Città	E	200
Persona	E	1000000
Residenza	R	1000000

- Operazione 1: memorizza una nuova persona con la relativa città di residenza (500 volte al giorno)
- Operazione 2: stampa tutti i dati di una città (incluso il numero di abitanti) (2 volte al giorno)

05/10/2004

Basi di Dati 1: Prog. Logica

23

Presenza di ridondanza

Operazione 1

Concetto	Costrutto	Accessi	Tipo
Persona	Entità	1	S
Residenza	Relazione	1	S
Città	Entità	1	L
Città	Entità	1	S

Operazione 2

Concetto	Costrutto	Accessi	Tipo
Città	Entità	1	L

05/10/2004

Basi di Dati 1: Prog. Logica

24

Assenza di ridondanza

Operazione 1

Concetto	Costrutto	Accessi	Tipo
Persona	Entità	1	S
Residenza	Relazione	1	S

Operazione 2

Concetto	Costrutto	Accessi	Tipo
Città	Entità	1	L
Residenza	Relazione	5000	L

05/10/2004

Basi di Dati 1: Prog. Logica

25

Presenza di ridondanza

- **Costi:**
 - **Operazione 1: 1500 accessi in scrittura e 500 accessi in lettura al giorno**
 - **Operazione 2: trascurabile.**
- **Contiamo doppi gli accessi in scrittura**
 - **Totale di 3500 accessi al giorno**

05/10/2004

Basi di Dati 1: Prog. Logica

26

Assenza di ridondanza

- **Costi:**
 - **Operazione 1: 1000 accessi in scrittura**
 - **Operazione 2: 10000 accessi in lettura al giorno**
- **Contiamo doppi gli accessi in scrittura**
 - **Totale di 12000 accessi al giorno**

05/10/2004

Basi di Dati 1: Prog. Logica

27

Attività della ristrutturazione

- **Analisi delle ridondanze**
- **Eliminazione delle generalizzazioni**
- **Partizionamento/accorpamento di entità e relazioni**
- **Scelta degli identificatori primari**

05/10/2004

Basi di Dati 1: Prog. Logica

28

Eliminazione delle gerarchie

- **il modello relazionale non può rappresentare direttamente le generalizzazioni**
- **entità e relazioni sono invece direttamente rappresentabili**

- **si eliminano perciò le gerarchie, sostituendole con entità e relazioni**

05/10/2004

Basi di Dati 1: Prog. Logica

29

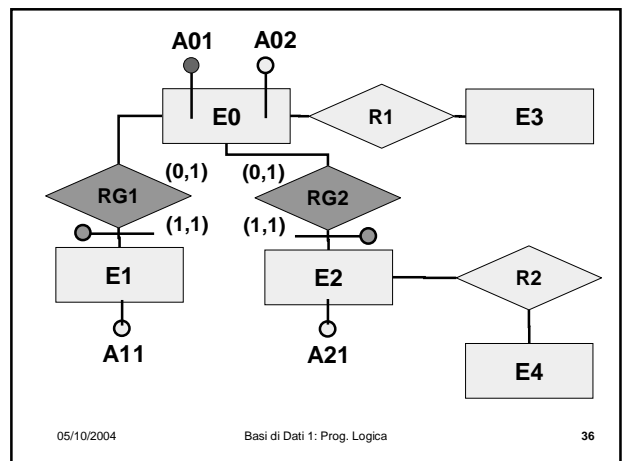
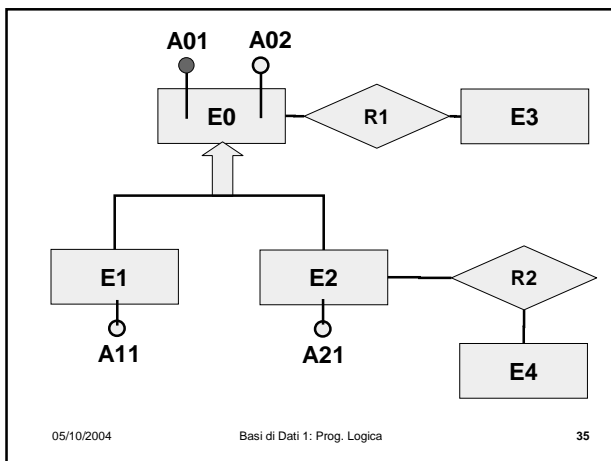
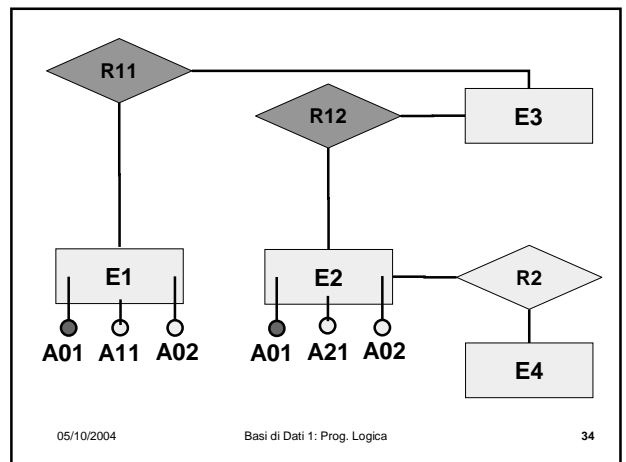
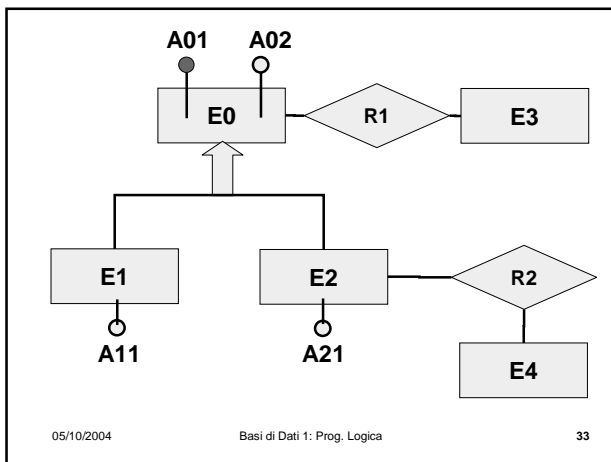
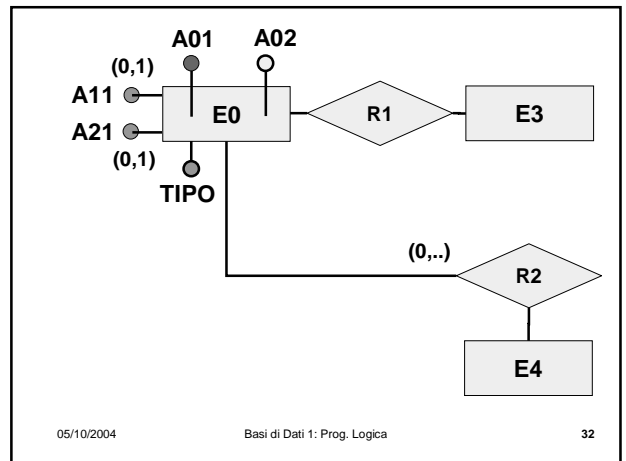
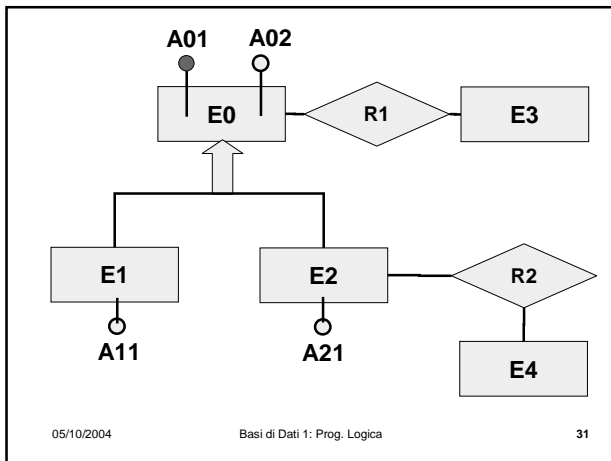
Tre possibilità

1. **accorpamento delle figlie della generalizzazione nel genitore**
2. **accorpamento del genitore della generalizzazione nelle figlie**
3. **sostituzione della generalizzazione con relazioni**

05/10/2004

Basi di Dati 1: Prog. Logica

30



- la scelta fra le alternative si può fare con metodo simile a quello visto per l'analisi delle ridondanze (però non basato solo sul numero degli accessi)
- è possibile seguire alcune semplici regole generali

05/10/2004

Basi di Dati 1: Prog. Logica

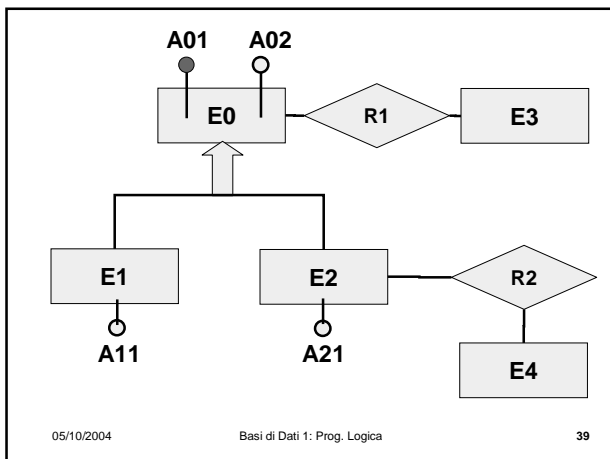
37

1. conviene se gli accessi al padre e alle figlie sono contestuali
 2. conviene se gli accessi alle figlie sono distinti
 3. conviene se gli accessi alle entità figlie sono separati dagli accessi al padre
- sono anche possibili soluzioni "ibride", soprattutto in gerarchie a più livelli

05/10/2004

Basi di Dati 1: Prog. Logica

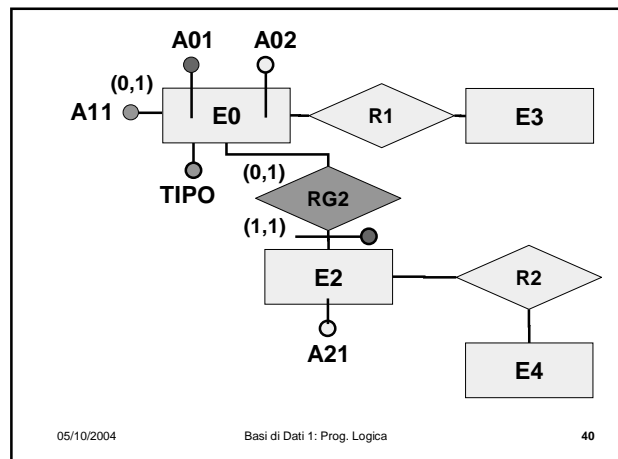
38



05/10/2004

Basi di Dati 1: Prog. Logica

39



05/10/2004

Basi di Dati 1: Prog. Logica

40

Attività della ristrutturazione

- Analisi delle ridondanze
- Eliminazione delle generalizzazioni
- Partizionamento/accorpamento di entità e relazioni
- Scelta degli identificatori primari

05/10/2004

Basi di Dati 1: Prog. Logica

41

- Ristrutturazioni effettuate per rendere più efficienti le operazioni in base a un semplice principio
- Gli accessi si riducono:
 - separando attributi di un concetto che vengono acceduti separatamente
 - raggruppando attributi di concetti diversi acceduti insieme

05/10/2004

Basi di Dati 1: Prog. Logica

42

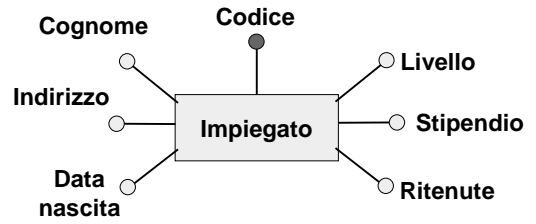
Ristrutturazioni, casi principali

- partizionamento verticale di entità
- partizionamento orizzontale di relationship
- eliminazione di attributi multivalore
- accorpamento di entità/ relationship

05/10/2004

Basi di Dati 1: Prog. Logica

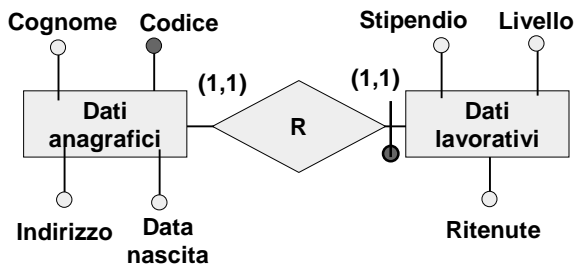
43



05/10/2004

Basi di Dati 1: Prog. Logica

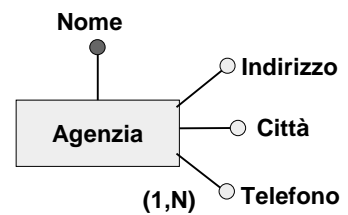
44



05/10/2004

Basi di Dati 1: Prog. Logica

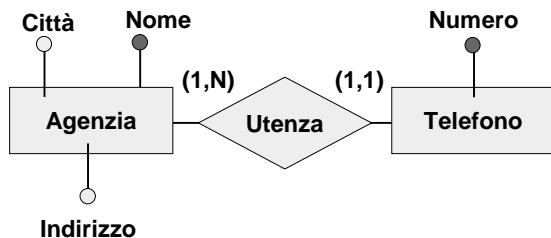
45



05/10/2004

Basi di Dati 1: Prog. Logica

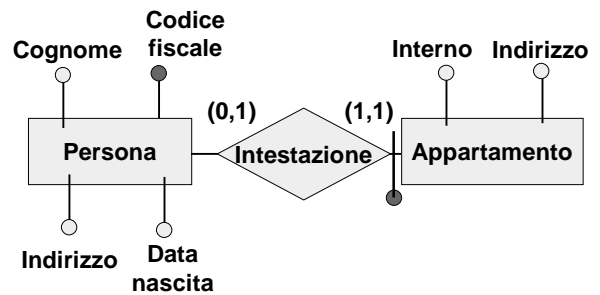
46



05/10/2004

Basi di Dati 1: Prog. Logica

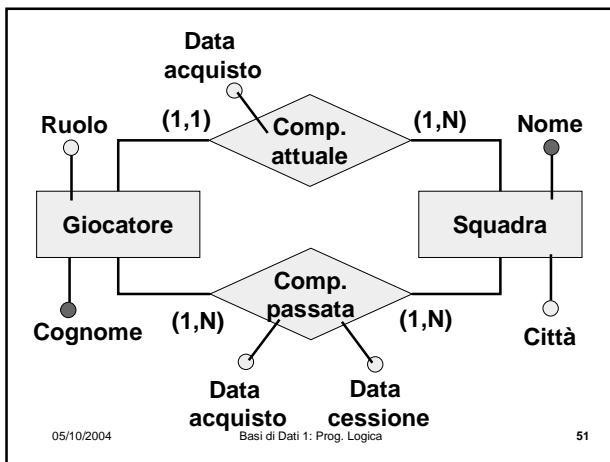
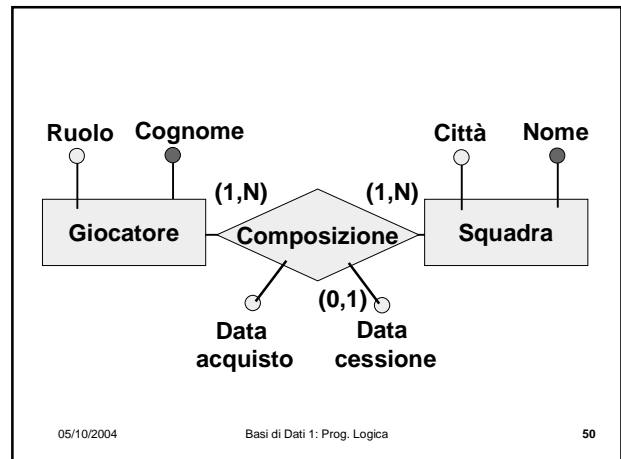
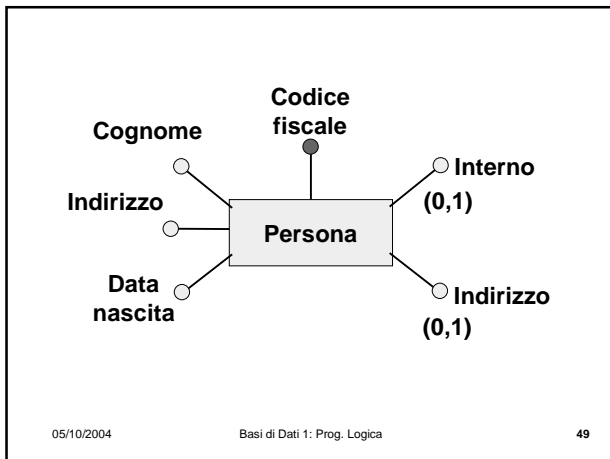
47



05/10/2004

Basi di Dati 1: Prog. Logica

48



Attività della ristrutturazione

- Analisi delle ridondanze
- Eliminazione delle generalizzazioni
- Partizionamento/accorpamento di entità e relazioni
- Scelta degli identificatori primari

05/10/2004 Basi di Dati 1: Prog. Logica 52

Scelta degli identificatori principali

- operazione indispensabile per la traduzione nel modello relazionale
- Criteri
 - assenza di opzionalità
 - semplicità
 - utilizzo nelle operazioni più frequenti o importanti

05/10/2004 Basi di Dati 1: Prog. Logica 53

Se nessuno degli identificatori soddisfa i requisiti visti?

Si introducono nuovi attributi (codici) contenenti valori speciali generati appositamente per questo scopo

05/10/2004 Basi di Dati 1: Prog. Logica 54

Traduzione verso il modello relazionale

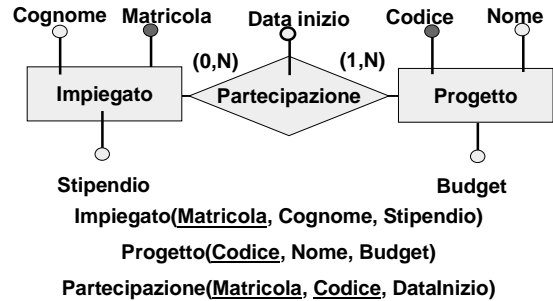
- **idea di base:**
 - le entità diventano relazioni sugli stessi attributi
 - le associazioni (ovvero le relazioni E-R) diventano relazioni sugli identificatori delle entità coinvolte (più gli attributi propri)

05/10/2004

Basi di Dati 1: Prog. Logica

55

Entità e relationship molti a molti



05/10/2004

Basi di Dati 1: Prog. Logica

56

Entità e relationship molti a molti

Impiegato(Matricola, Cognome, Stipendio)
Progetto(Codice, Nome, Budget)
Partecipazione(Matricola, Codice, DataInizio)

- con vincoli di integrità referenziale fra
 - Matricola in Partecipazione e (la chiave di) Impiegato
 - Codice in Partecipazione e (la chiave di) Progetto

05/10/2004

Basi di Dati 1: Prog. Logica

57

Nomi più espressivi per gli attributi della chiave della relazione che rappresenta la relationship

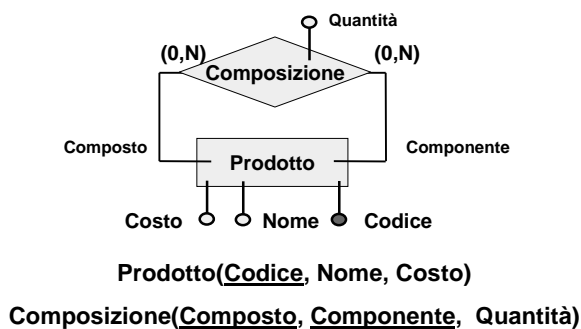
Impiegato(Matricola, Cognome, Stipendio)
Progetto(Codice, Nome, Budget)
Partecipazione(Matricola, Codice, DataInizio)
Partecipazione(Impiegato, Progetto, DataInizio)

05/10/2004

Basi di Dati 1: Prog. Logica

58

Relationship ricorsive

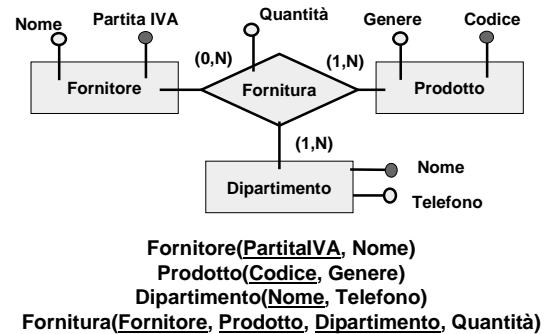


05/10/2004

Basi di Dati 1: Prog. Logica

59

Relationship n-arie

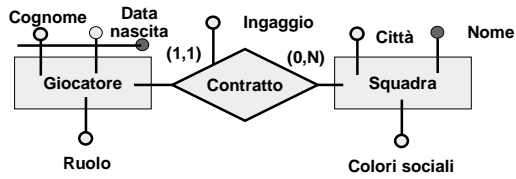


05/10/2004

Basi di Dati 1: Prog. Logica

60

Relationship uno a molti



Giocatore(Cognome, DataNascita, Ruolo)
Contratto(CognGiocatore, DataNascG, Squadra, Ingaggio)
Squadra(Nome, Città, ColoriSociali)

- corretto?

05/10/2004

Basi di Dati 1: Prog. Logica

61

Soluzione più compatta

Giocatore(Cognome, DataNascita, Ruolo)
Contratto(CognGiocatore, DataNascG, Squadra, Ingaggio)
Squadra(Nome, Città, ColoriSociali)

Giocatore(Cognome, DataNasc, Ruolo, Squadra, Ingaggio)
Squadra(Nome, Città, ColoriSociali)

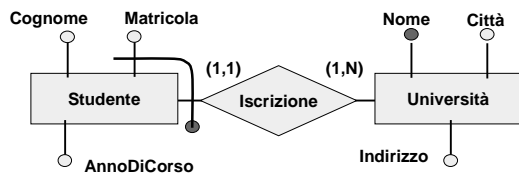
- con vincolo di integrità referenziale fra Squadra in Giocatore e la chiave di Squadra
- se la cardinalità minima della relationship è 0, allora Squadra e Ingaggio in Giocatore deve ammettere valore nullo

05/10/2004

Basi di Dati 1: Prog. Logica

62

Entità con identificazione esterna



Studente(Matricola, Università, Cognome, AnnoDiCorso)
Università(Nome, Città, Indirizzo)

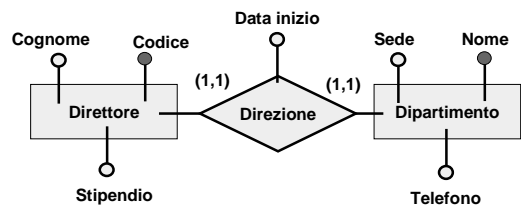
- con vincolo ...

05/10/2004

Basi di Dati 1: Prog. Logica

63

Relationship uno a uno



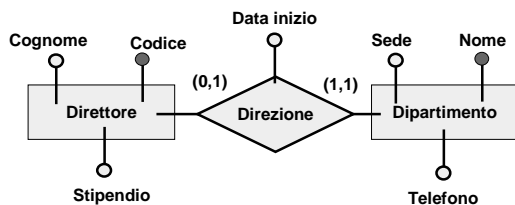
- varie possibilità:
 - fondere da una parte o dall'altra
 - fondere tutto?

05/10/2004

Basi di Dati 1: Prog. Logica

64

Una possibilità privilegiata



Impiegato(Codice, Cognome, Stipendio)
Dipartimento(Nome, Sede, Telefono, Direttore, InizioD)

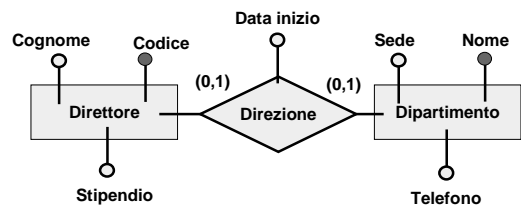
- con vincolo di integrità referenziale, senza valori nulli

05/10/2004

Basi di Dati 1: Prog. Logica

65

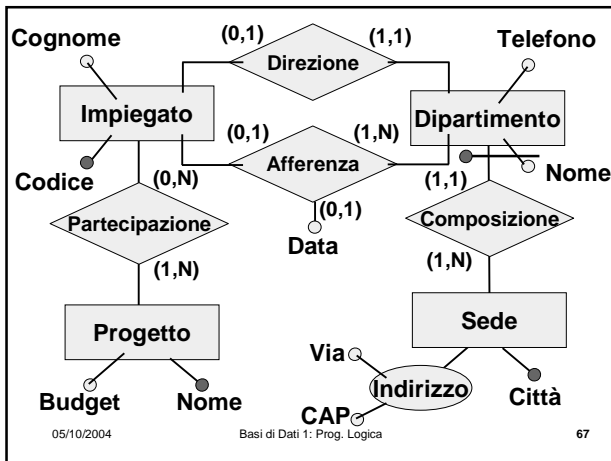
Un altro caso



05/10/2004

Basi di Dati 1: Prog. Logica

66



Schema finale

Impiegato(Codice, Cognome, Dipartimento*, Sede*, Data*)

Dipartimento(Nome, Città, Telefono, Direttore)

Sede(Città, Via, CAP)

Progetto(Nome, Budget)

Partecipazione(Impiegato, Progetto)

05/10/2004 Basi di Dati 1: Prog. Logica 68

Strumenti di supporto

- Esistono sul mercato prodotti CASE che forniscono un supporto a tutte le fasi della progettazione di basi di dati

05/10/2004 Basi di Dati 1: Prog. Logica 69

