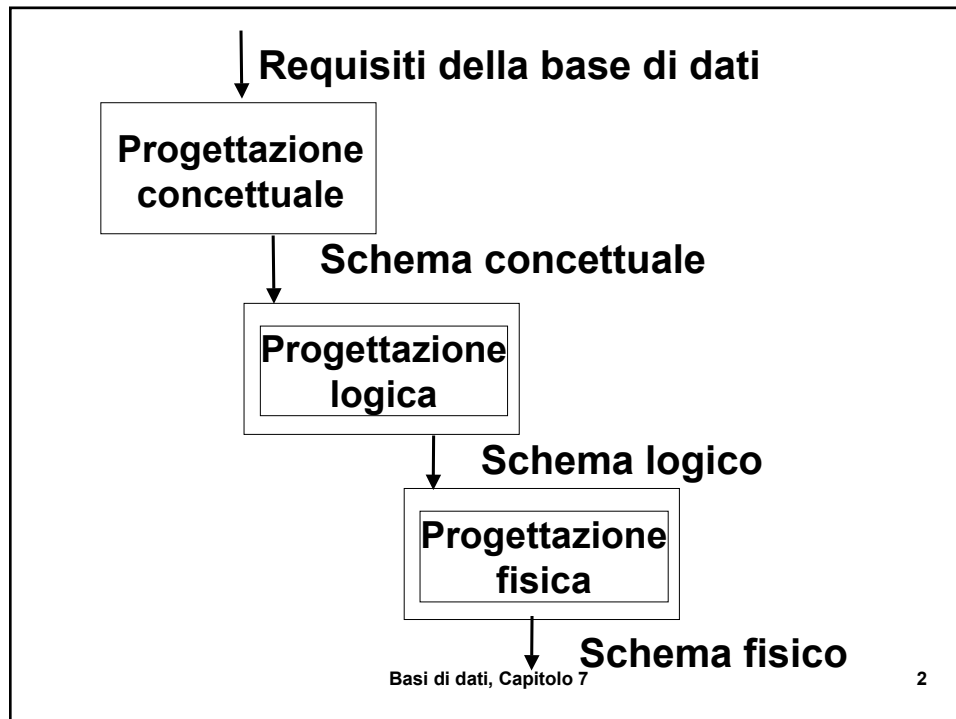


Capitolo 7:  
**Progettazione logica**



## **Obiettivo della progettazione logica**

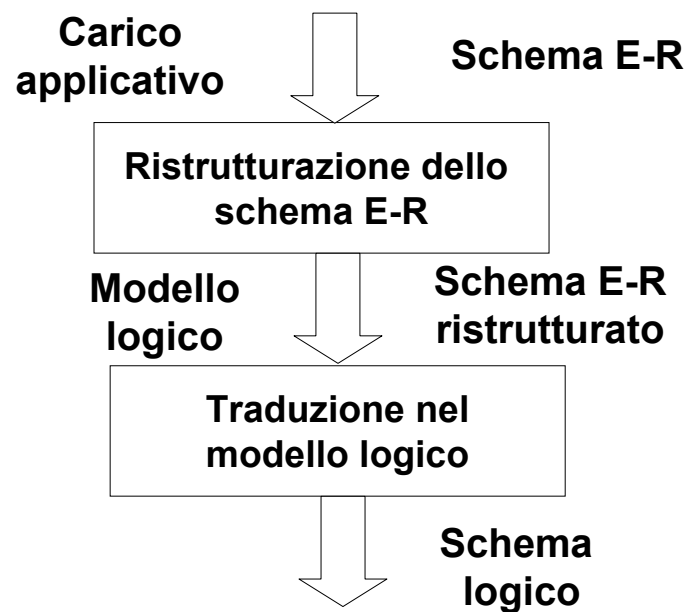
- **"tradurre" lo schema concettuale in uno schema logico che rappresenti gli stessi dati in maniera corretta ed efficiente**

## **Dati di ingresso e uscita**

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

## Non si tratta di una pura e semplice traduzione

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



## Ristrutturazione schema E-R

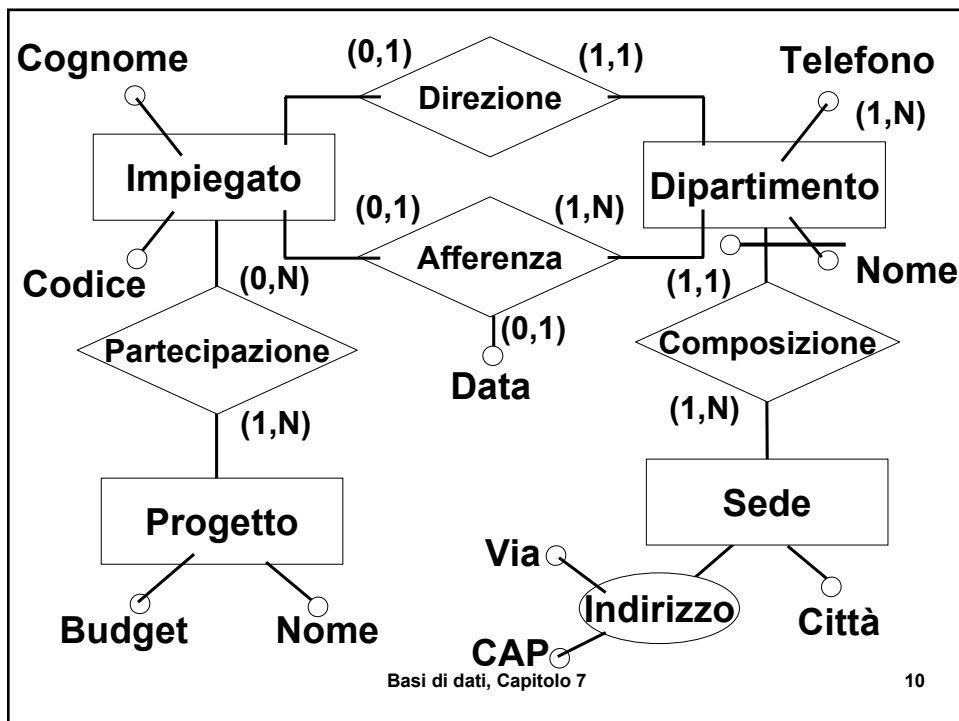
- **Motivazioni:**
  - **semplificare la traduzione**
  - **"ottimizzare" le prestazioni**
- **Osservazione:**
  - **uno schema E-R ristrutturato non è (più) uno schema concettuale nel senso stretto del termine**

**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!**

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

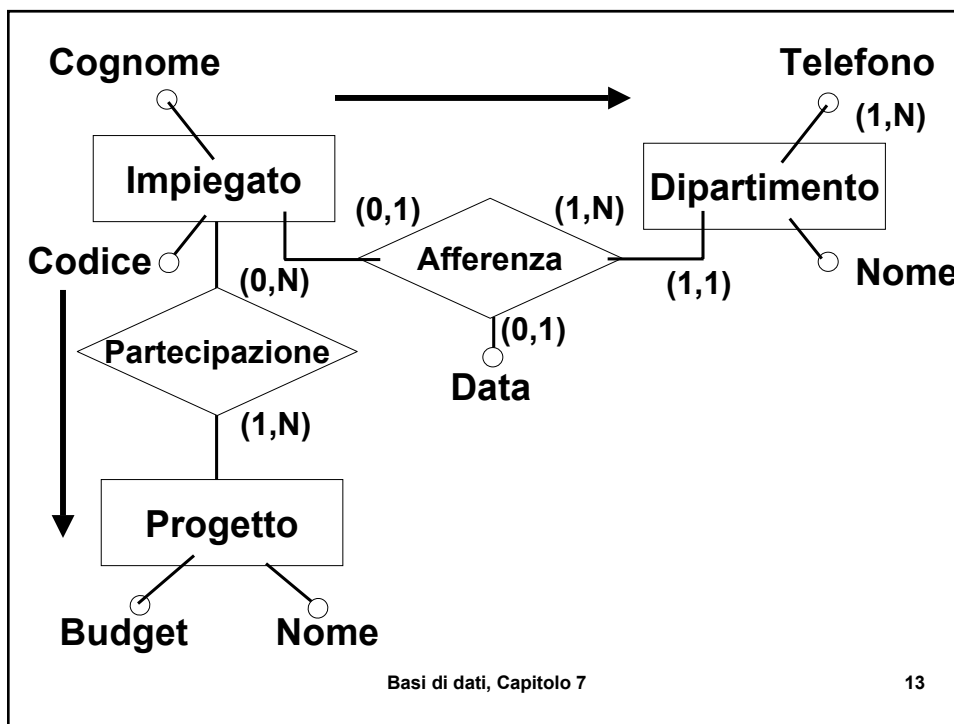


## Tavola dei volumi

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

## 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**



### 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

## **Attività della ristrutturazione**

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

## **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**

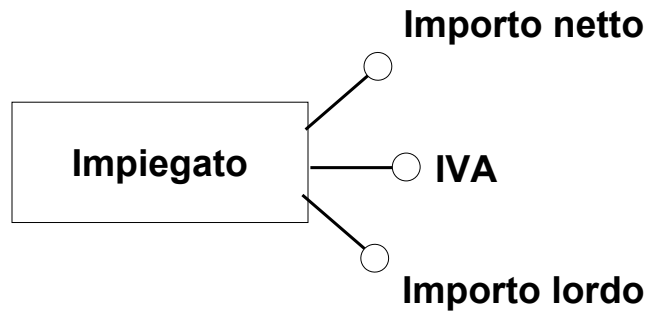
## Ridondanze

- **Vantaggi**
  - **semplificazione delle interrogazioni**
- **Svantaggi**
  - **appesantimento degli aggiornamenti**
  - **maggiore occupazione di spazio**

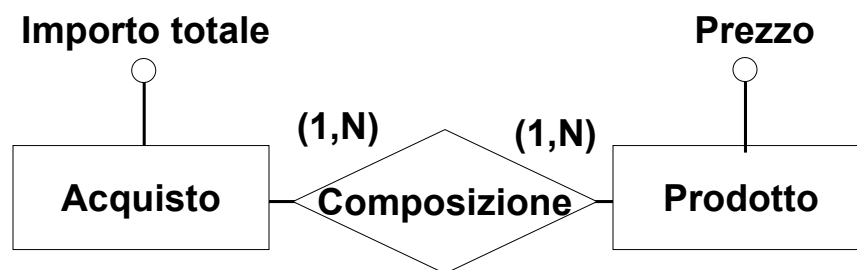
## Forme di ridondanza in uno schema E-R

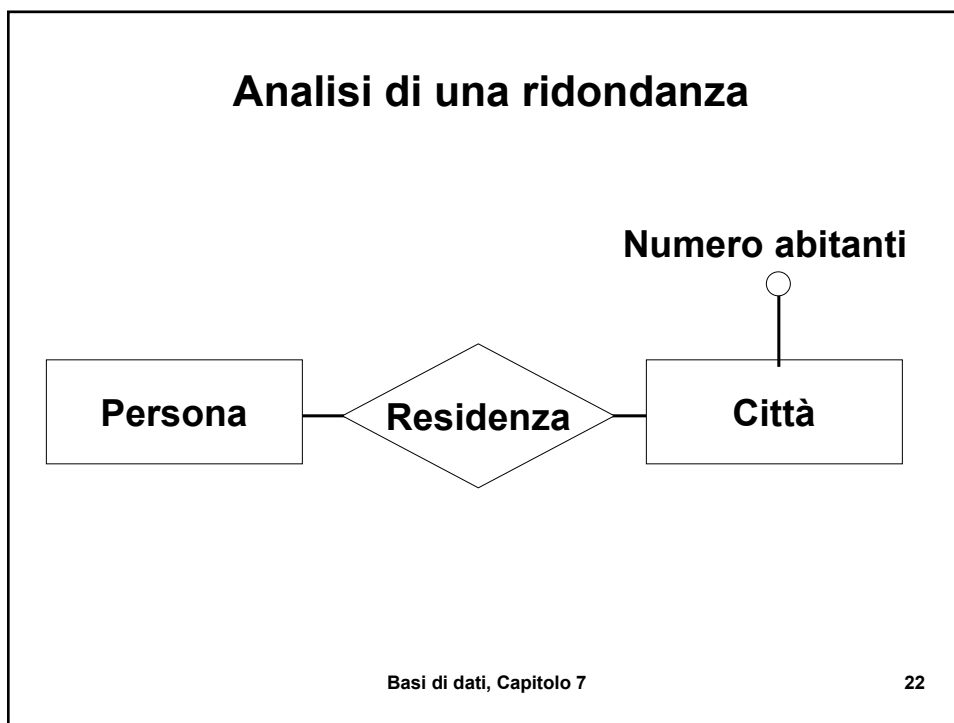
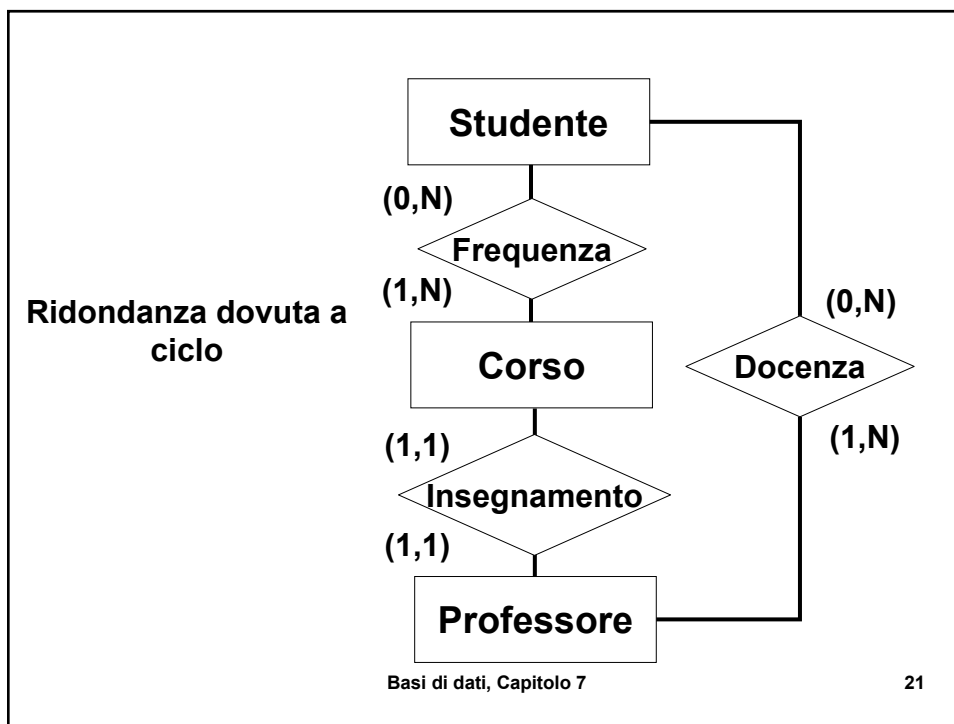
- **attributi derivabili:**
  - **da altri attributi della stessa entità (o relazione)**
  - **da attributi di altre entità (o relazioni)**
- **relazioni derivabili dalla composizione di altre relazioni in presenza di cicli**

## Attributo derivabile



## Attributo derivabile da altra entità





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)**

## 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

## 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

## 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**

## **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**

## **Attività della ristrutturazione**

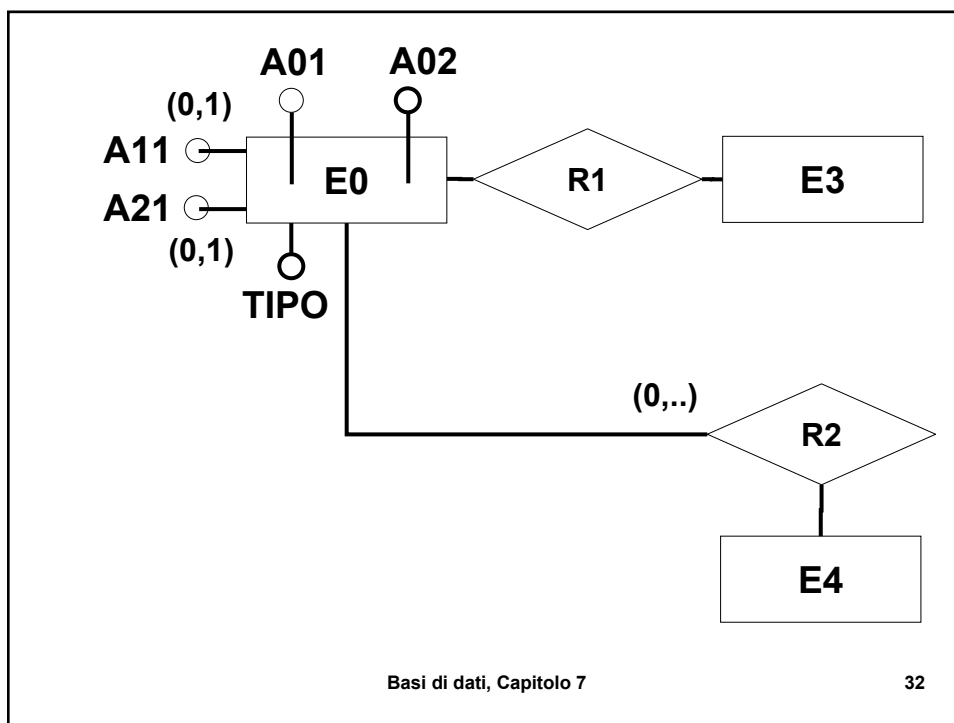
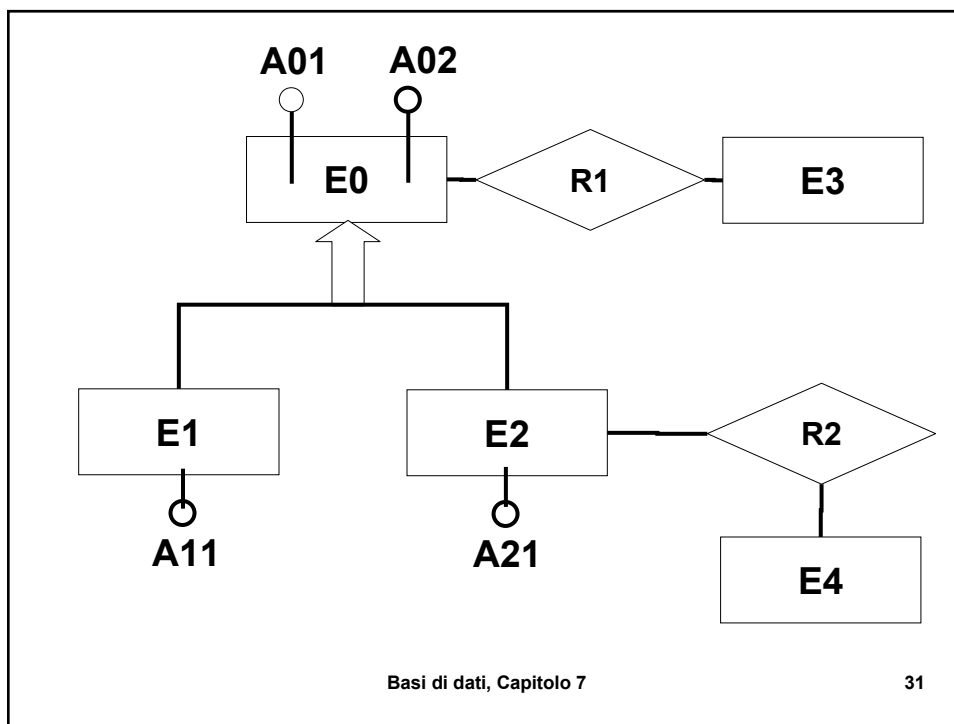
- **Analisi delle ridondanze**
- **Eliminazione delle generalizzazioni**
- **Partizionamento/accorpamento di entità e relazioni**
- **Scelta degli identificatori primari**

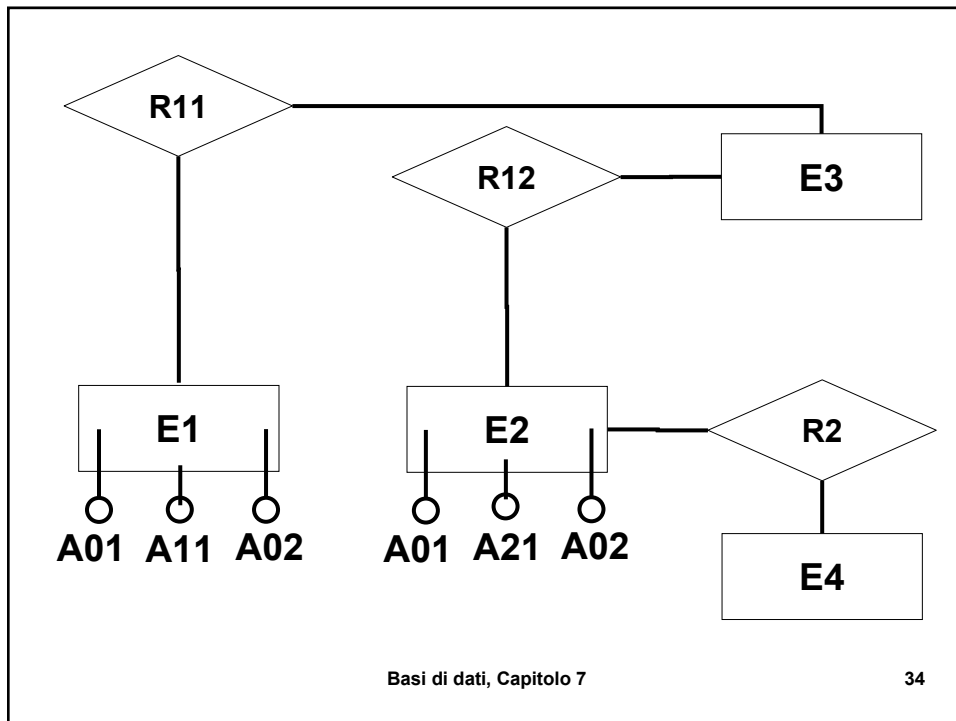
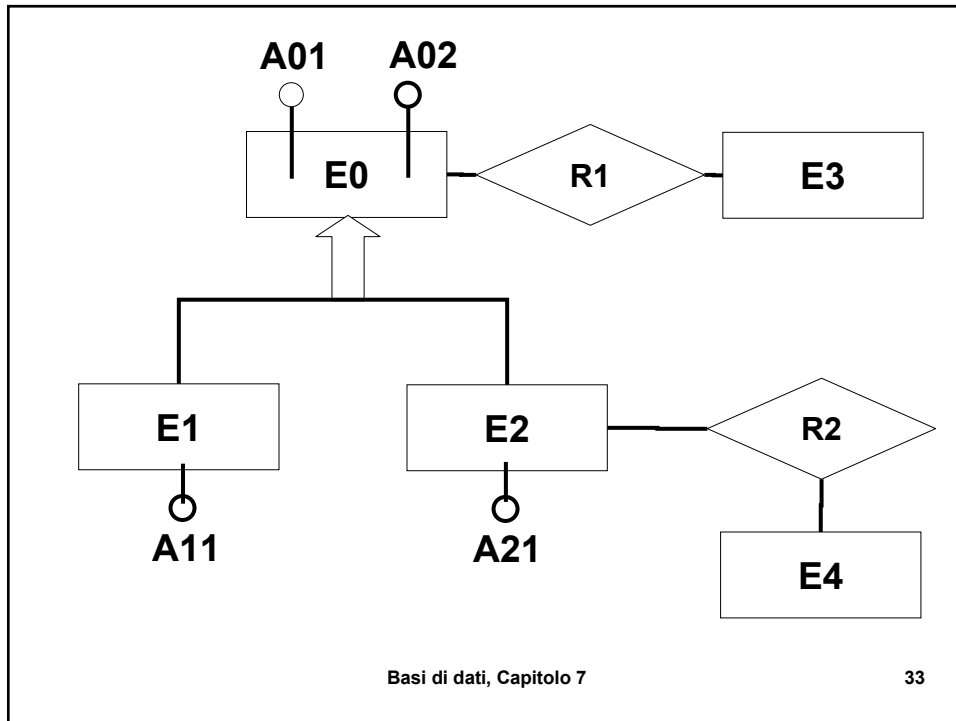
## **Eliminazione delle gerarchie**

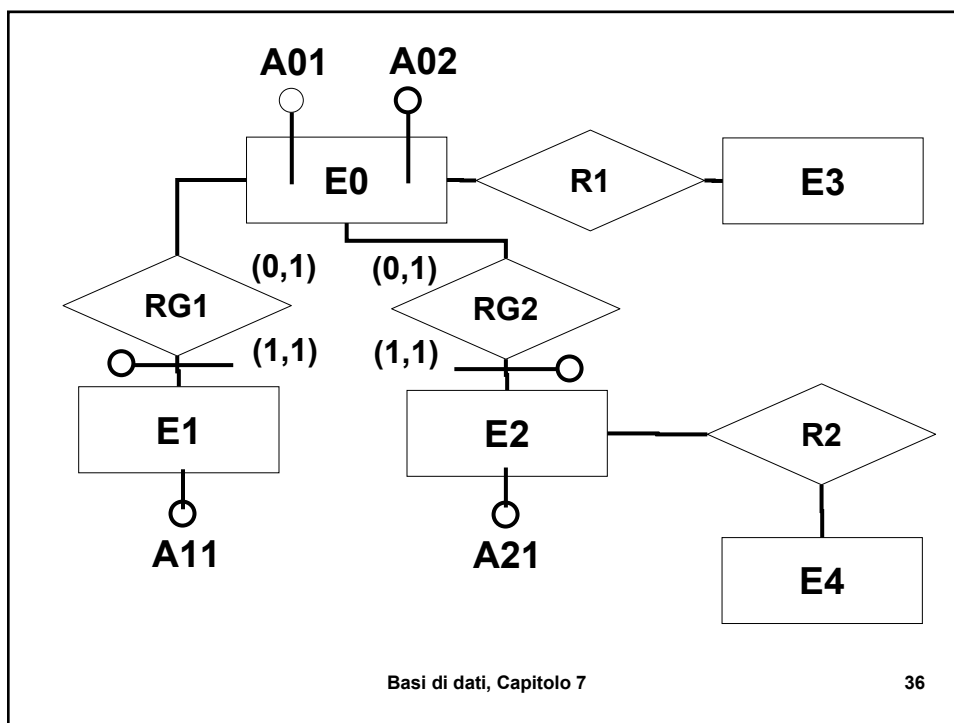
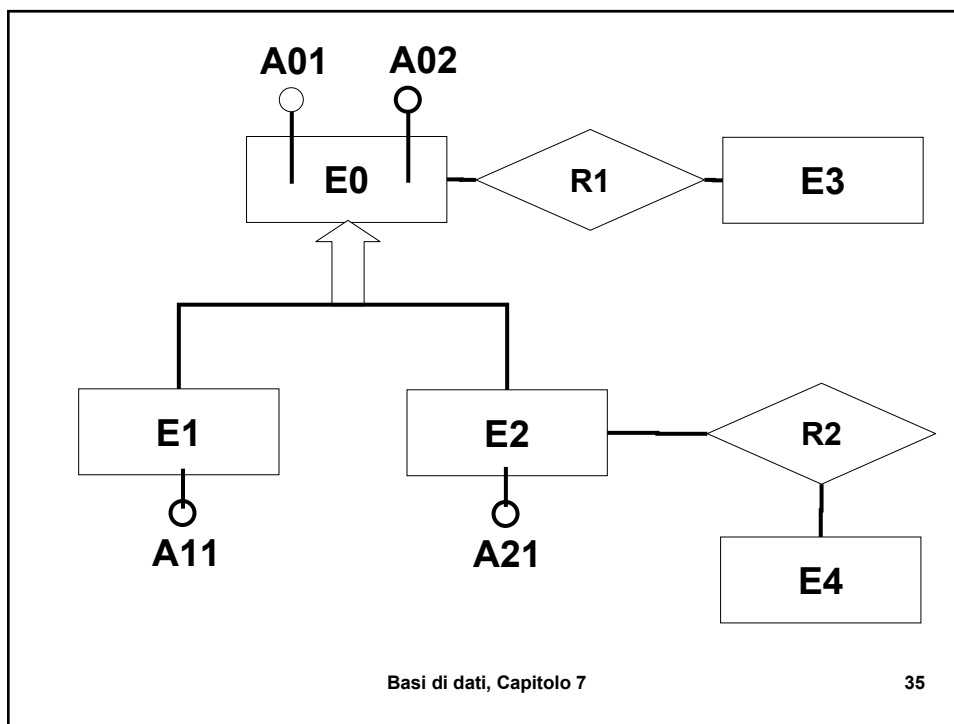
- **il modello relazionale non può rappresentare direttamente le generalizzazioni**
- **entità e relazioni sono invece direttamente rappresentabili**
  
- **si eliminano perciò le gerarchie, sostituendole con entità e relazioni**

## **Tre possibilità**

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

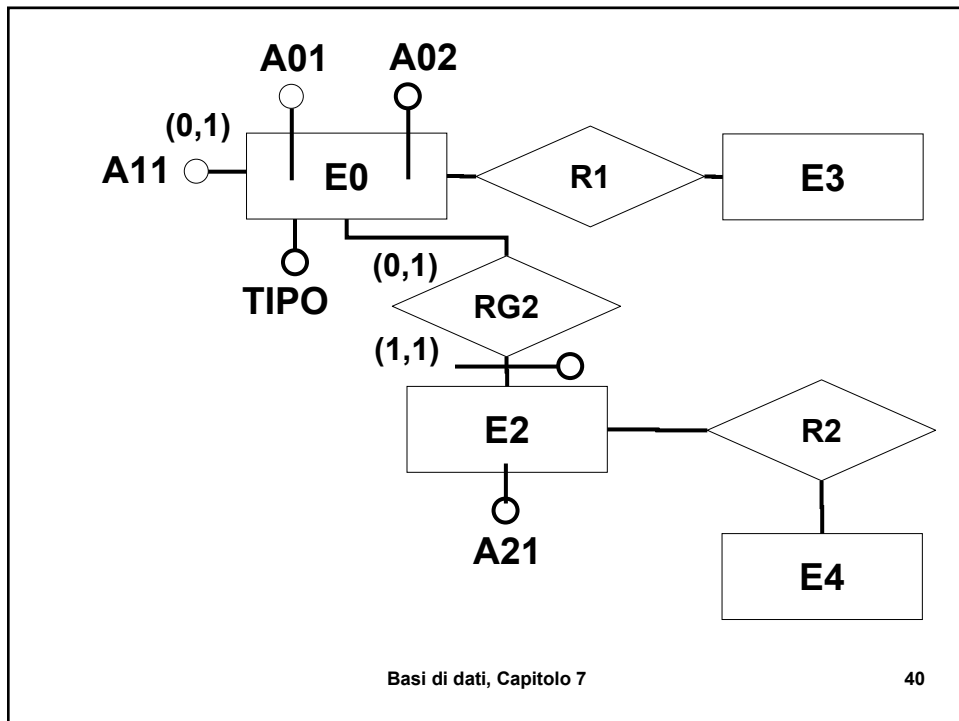
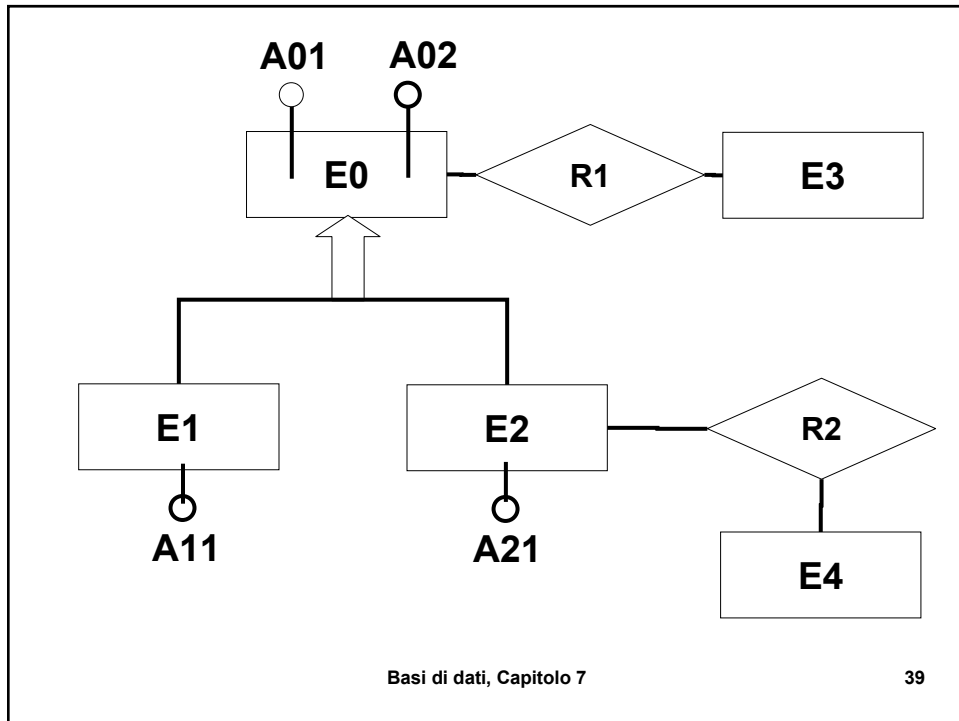






- **la scelta fra le alternative si può fare con metodo simile a quello visto per l'analisi delle ridondanze (però non basato solo sul numero degli accessi)**
- **è possibile seguire alcune semplici regole generali**

- 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**



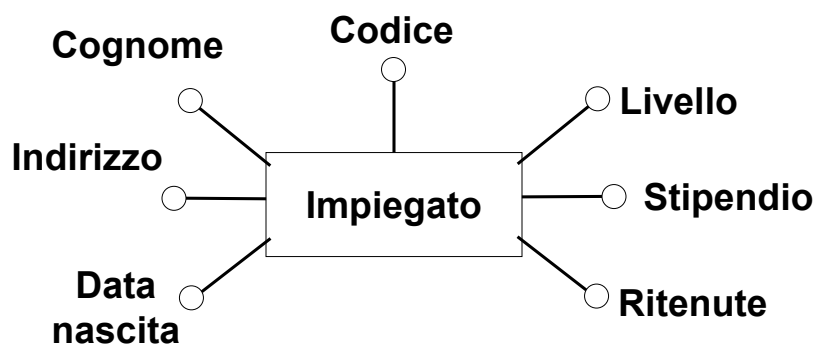
## **Attività della ristrutturazione**

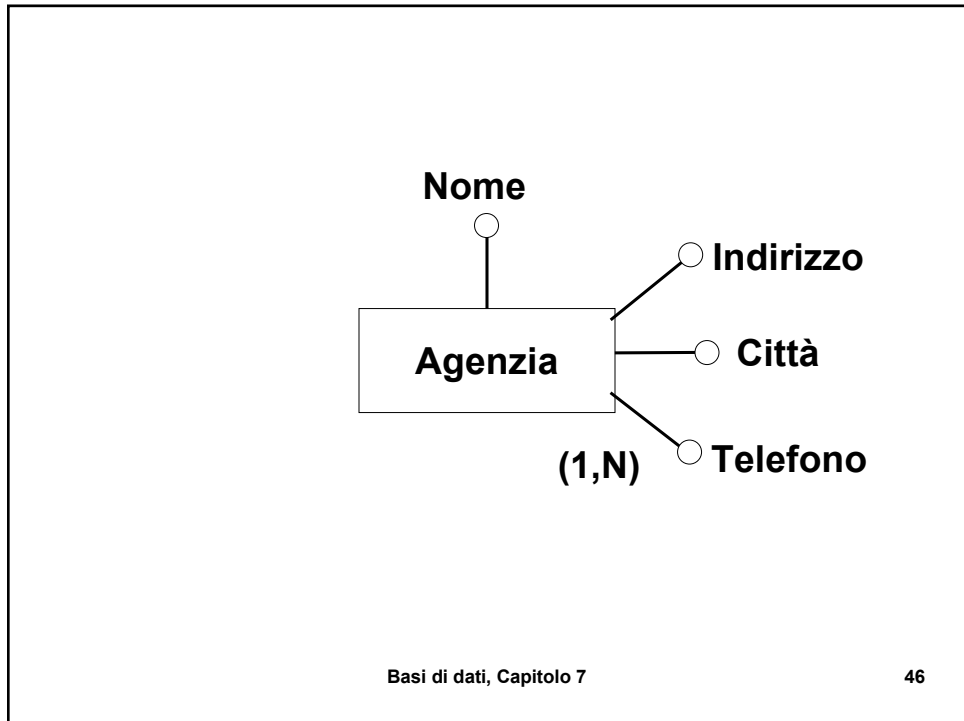
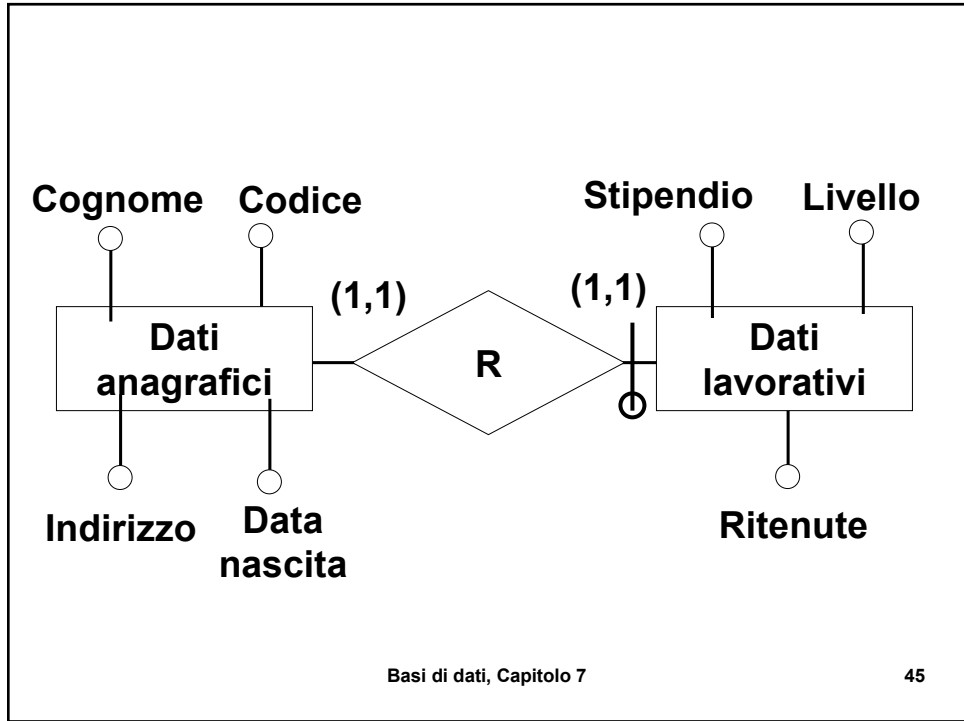
- **Analisi delle ridondanze**
- **Eliminazione delle generalizzazioni**
- **Partizionamento/accorpamento di entità e relazioni**
- **Scelta degli identificatori primari**

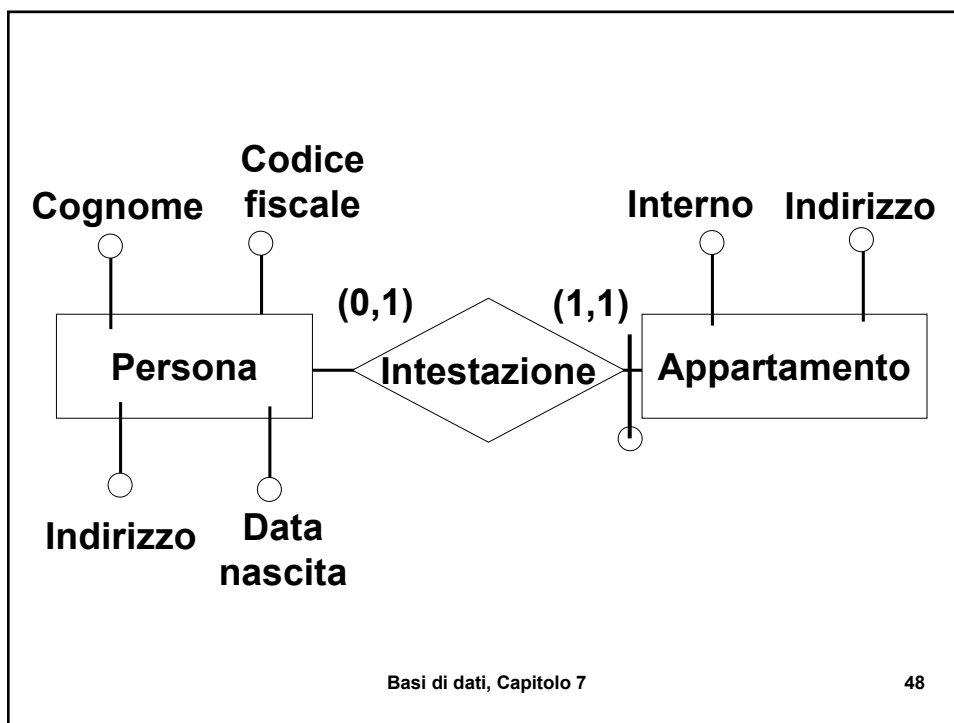
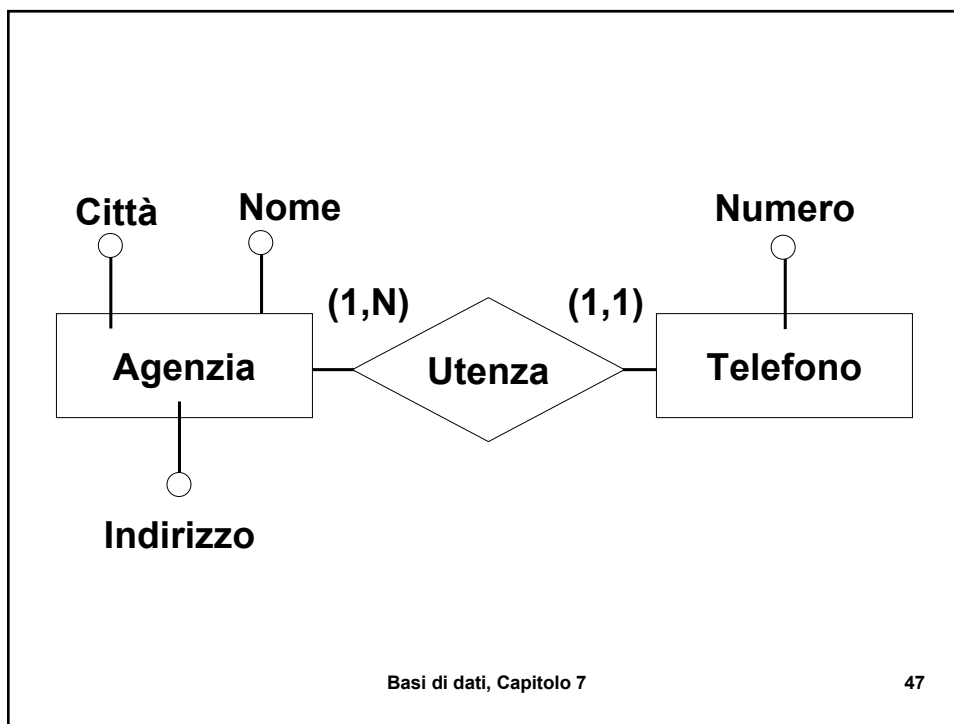
- **Ristrutturazioni effettuate per rendere più efficienti le operazioni in base a un semplice principio**
- **Gli accessi si riducono:**
  - **separando attributi di un concetto che vengono acceduti separatamente**
  - **raggruppando attributi di concetti diversi acceduti insieme**

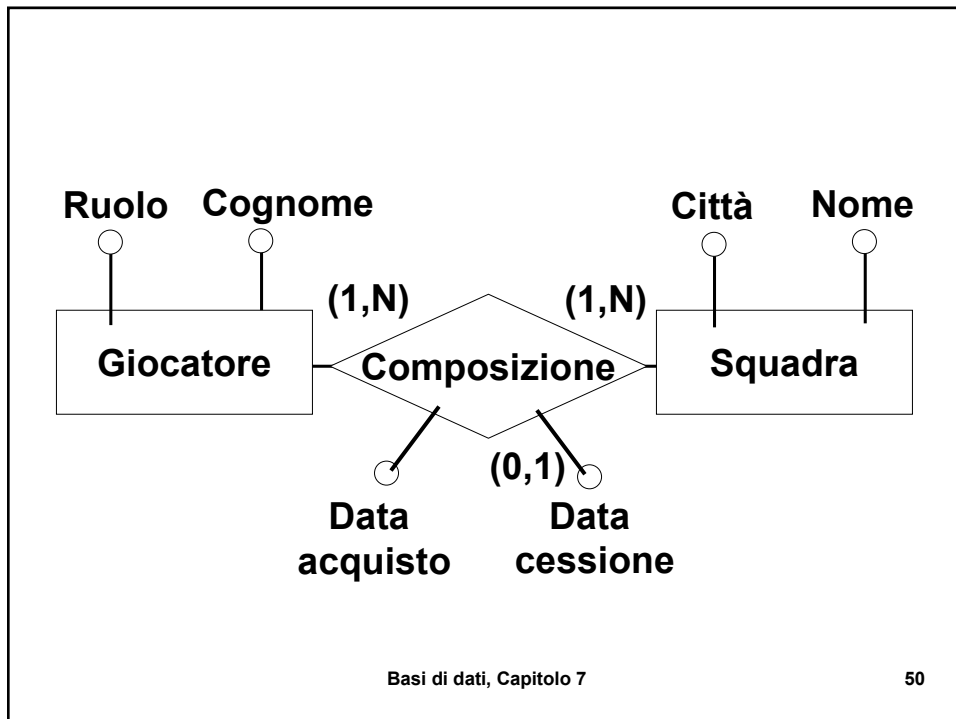
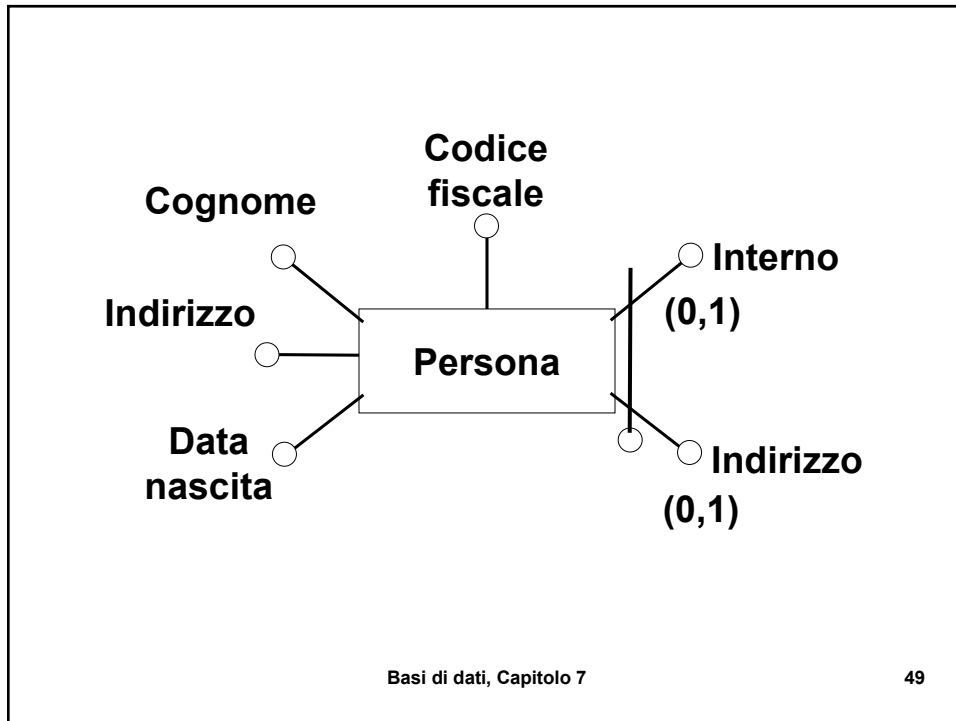
## Ristrutturazioni, casi principali

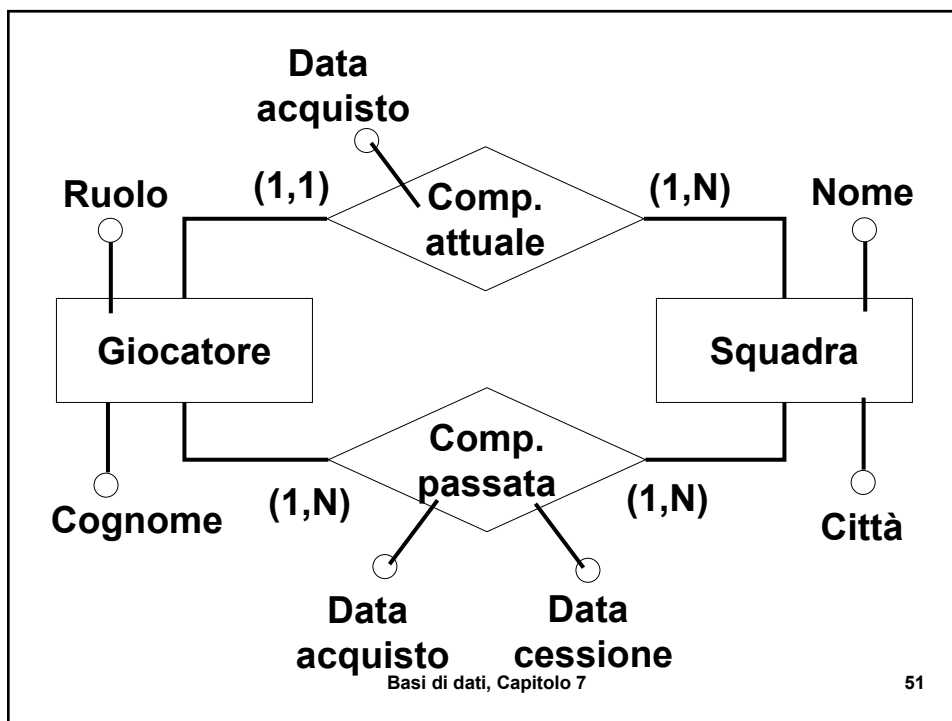
- **partizionamento verticale di entità**
- **partizionamento orizzontale di relationship**
- **eliminazione di attributi multivalore**
- **accorpamento di entità/ relationship**











## Attività della ristrutturazione

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

## **Scelta degli identificatori principali**

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

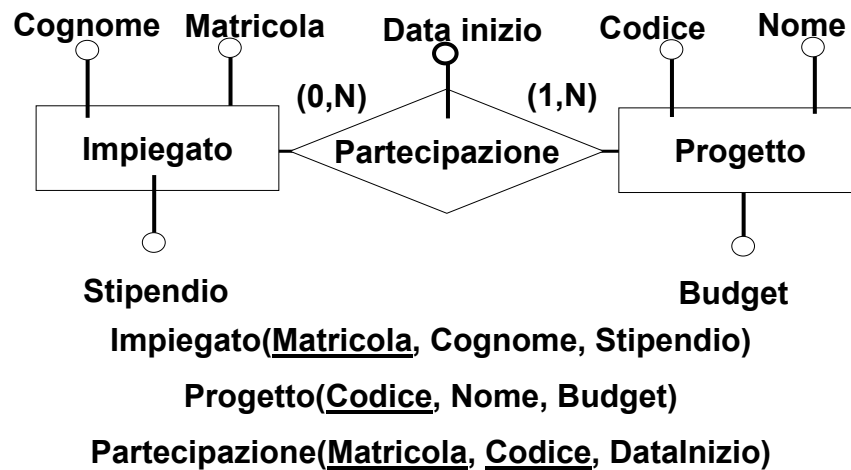
**Se nessuno degli identificatori soddisfa i requisiti visti?**

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

## 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)

## Entità e relationship molti a molti



## 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

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

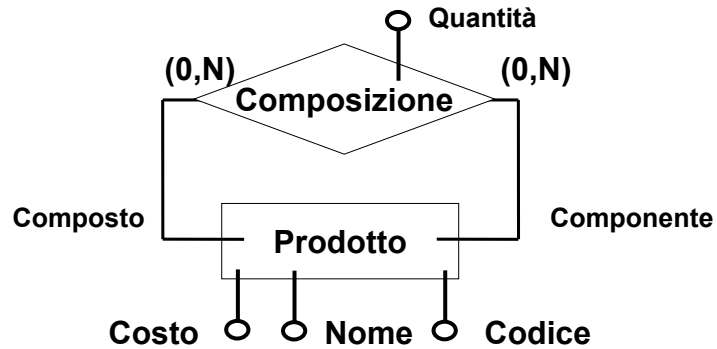
Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

Partecipazione(Matricola, Codice, DataInizio)

Partecipazione(Impiegato, Progetto, DataInizio)

## Relationship ricorsive



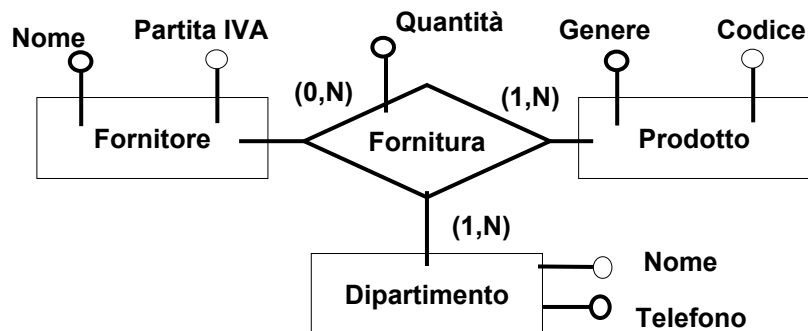
**Prodotto**(Codice, Nome, Costo)

**Composizione**(Composto, Componente, Quantità)

Basi di dati, Capitolo 7

59

## Relationship n-arie



**Fornitore**(PartitaIVA, Nome)

**Prodotto**(Codice, Genere)

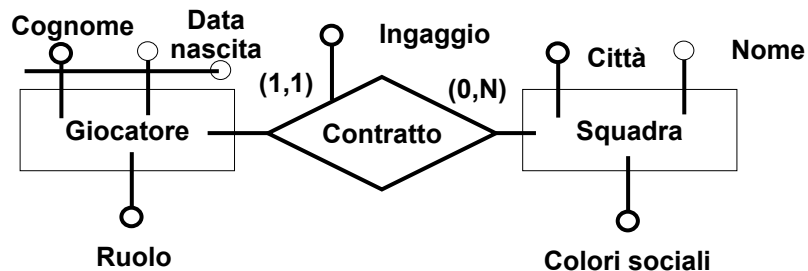
**Dipartimento**(Nome, Telefono)

**Fornitura**(Fornitore, Prodotto, Dipartimento, Quantità)

Basi di dati, Capitolo 7

60

## Relationship uno a molti



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

- corretto?

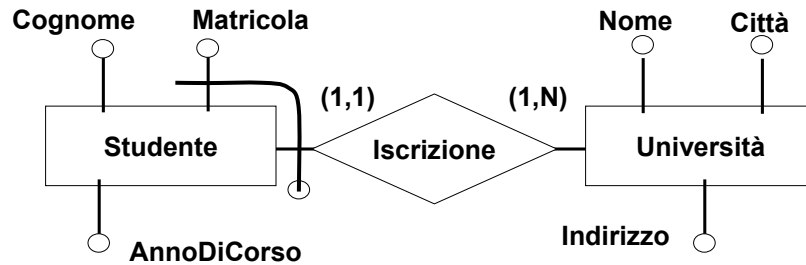
## Soluzione più compatta

**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 in Giocatore deve ammettere valore nullo

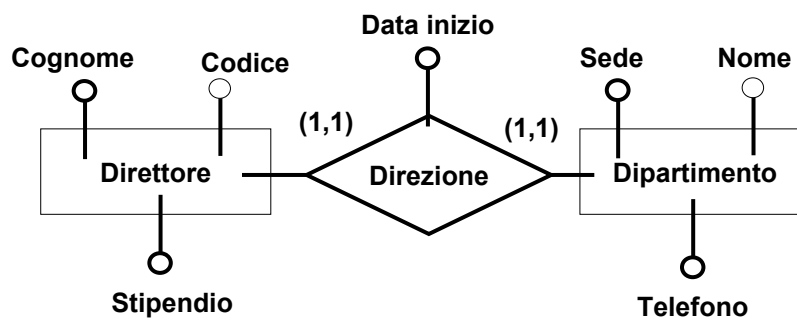
## Entità con identificazione esterna



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

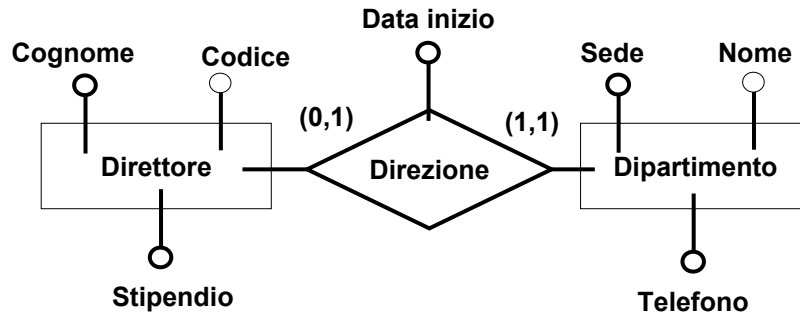
- con vincolo ...

## Relationship uno a uno



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

## Una possibilità privilegiata

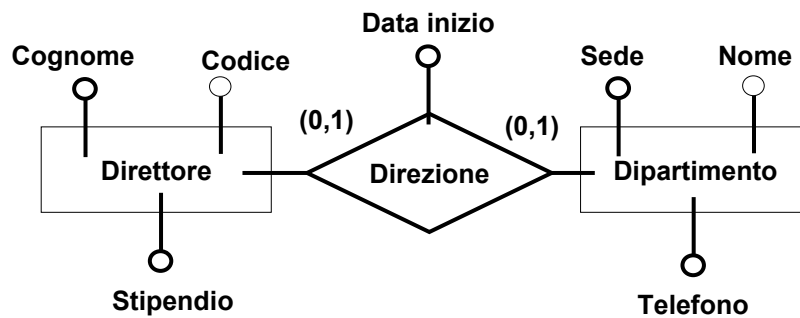


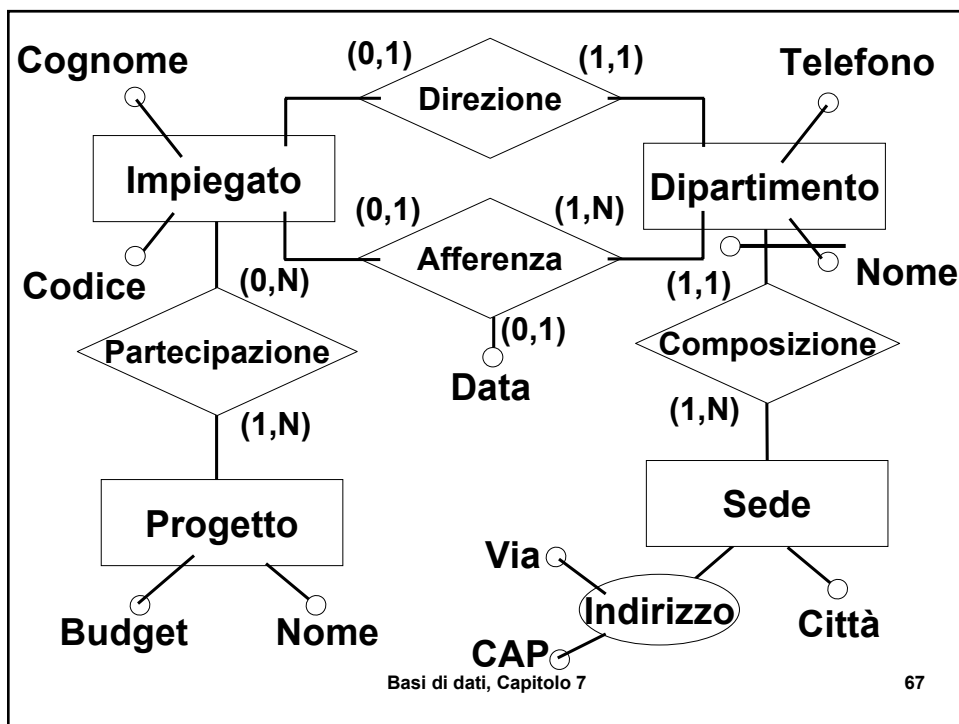
**Impiegato** (Codice, Cognome, Stipendio)

**Dipartimento** (Nome, Sede, Telefono, Direttore, InizioD)

- con vincolo di integrità referenziale, senza valori nulli

## Un altro caso





### Schema finale

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

**Dipartimento**(Nome, Città, Telefono, Direttore)

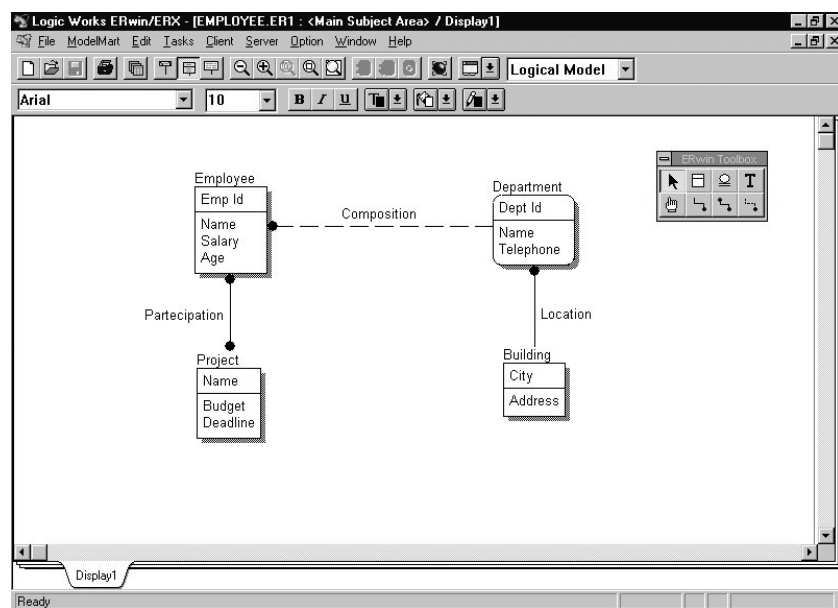
**Sede**(Città, Via, CAP)

**Progetto**(Nome, Budget)

**Partecipazione**(Impiegato, Progetto)

## Strumenti di supporto

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



Logic Works ERwin/ERX - [EMPLOYEE.ER1 - <Main Subject Area> / Display1]

File ModelMat Edit Tasks Client Server Option Window Help

Physical Model

Arial 10 B / U

ORACLE Schema Generation Report : <Main Subject

Report: Schema Generation Report

ORACLE 7.0 Schema Generation

Referential Integrity:  Primary Key (FK)  Foreign Key (FK)  UNIQUE (AK)

Schema Option:  Pre-Script  Create Proced  TABLESPACE  DATABASE

ORACLE Schema Generation Report Preview

```

CREATE TABLE Employee (
  Emp_Id NUMBER NOT NULL,
  Dept_Id NUMBER NOT NULL,
  Name VARCHAR2(20) NULL,
  Salary NUMBER NULL,
  Age NUMBER NULL,
  PRIMARY KEY (Emp_Id) );

CREATE TABLE Project (
  Name VARCHAR2(20) NOT NULL,
  Budget NUMBER NULL,
  Deadline DATE NULL,
  PRIMARY KEY (Name) );

CREATE TABLE Employee_Project (
  Emp_Id NUMBER NOT NULL,
  Name VARCHAR2(20) NOT NULL,
  PRIMARY KEY (Emp_Id, Name) );

```

Employee\_Project: Emp\_Id: NUMBER, Name: VARCHAR2(20)

Employee: Emp\_Id: NUMBER, Dept\_Id: NUMBER, Name: VARCHAR2(20), Salary: NUMBER, Age: NUMBER

Department: Dept\_Id: NUMBER, Name: VARCHAR2(20), Telephone: NUMBER

Project: Name: VARCHAR2(20), Budget: NUMBER, Deadline: DATE

Manager: Emp\_Id: NUMBER, Dept\_Id: NUMBER, Room: VARCHAR2(20), Name: VARCHAR2(20)

Building: Name: VARCHAR2(20), City: VARCHAR2(20), Address: VARCHAR2(20)

Display1

Ready

Basi di dati, Capitolo 7

71