

# La progettazione logica

Basi di dati per la gestione dell'informazione 2/ed

McGraw-Hill

Capitolo 3 (Paragrafo 3.5)

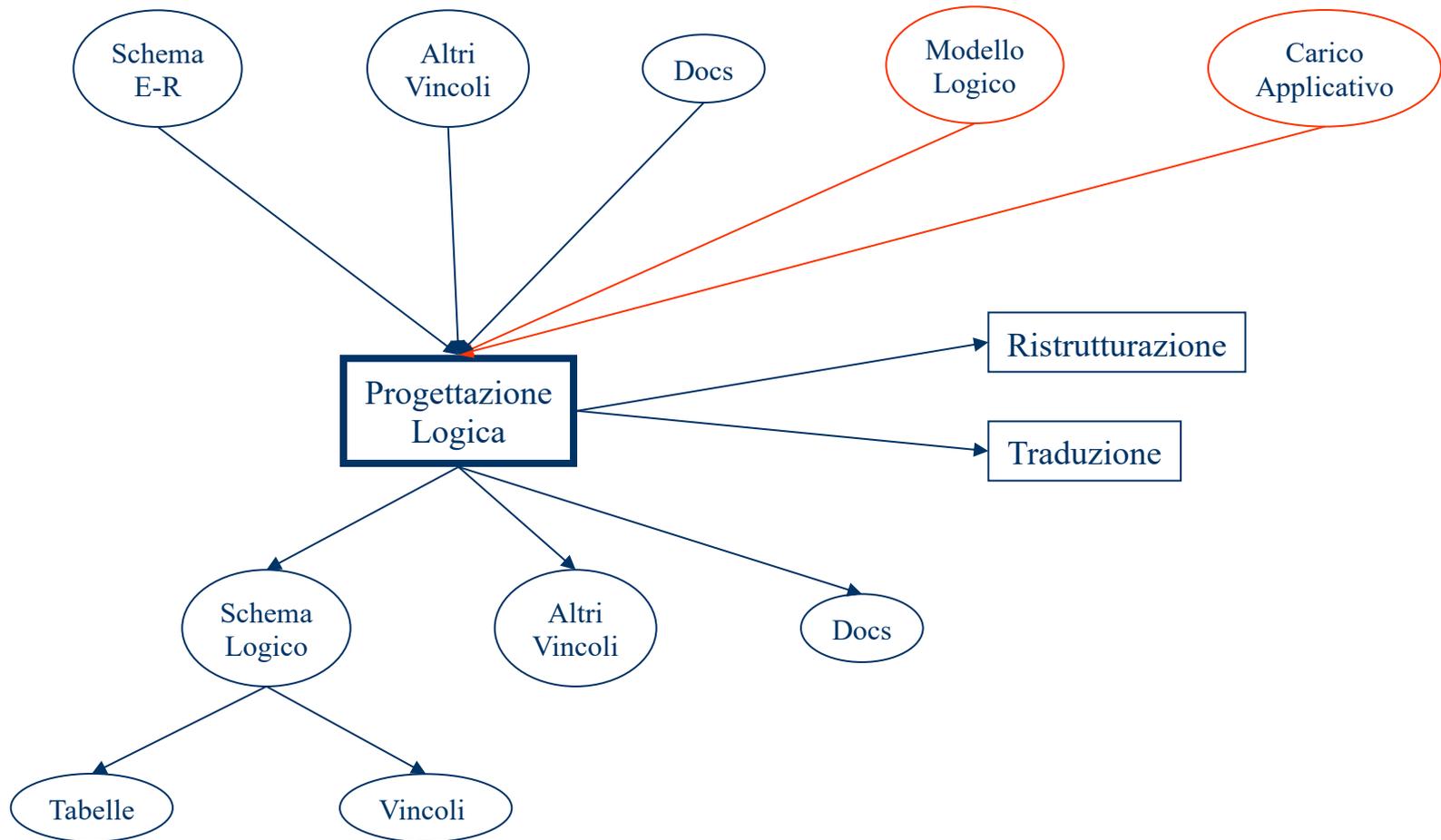
Capitolo 4 (Paragrafi 4.4)



# La progettazione Logica

- Scopo:
  - Costruire uno schema logico in cui siano rappresentate le stesse informazioni presenti nello schema E-R.
- Non è una semplice traduzione:
  - Semplificare la rappresentazione per facilitare la traduzione:
    - non tutti i costrutti del modello E-R sono supportati, ad esempio, dal modello relazionale.
  - Ottimizzare il progetto:
    - indici di prestazioni basati sulle caratteristiche delle operazioni

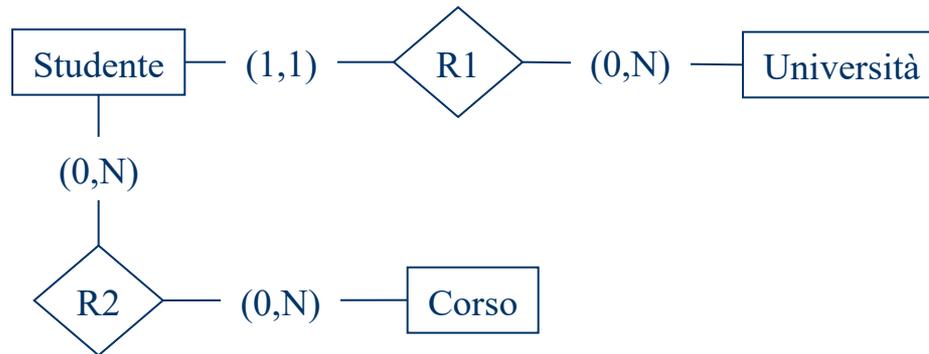
# La Progettazione Logica



# Carico Applicativo

- Volume dei dati.
  - Numero di occorrenze di entità ed associazioni;
  - Dimensione (media) degli attributi
- Operazioni.
- Caratteristiche delle operazioni.
  - Interattive o Batch
  - Frequenza
  - Dati

# Carico Applicativo

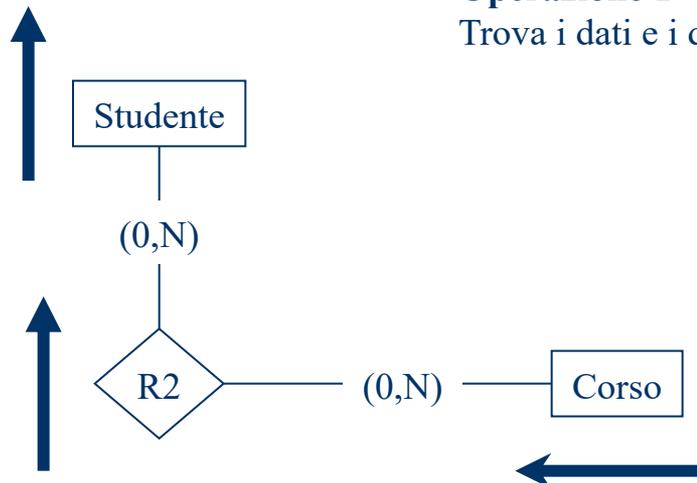


## Tavola dei Volumi

Studente	10000
Università	5
Corso	2000
R1	10000
R2	50000

## Operazione 1

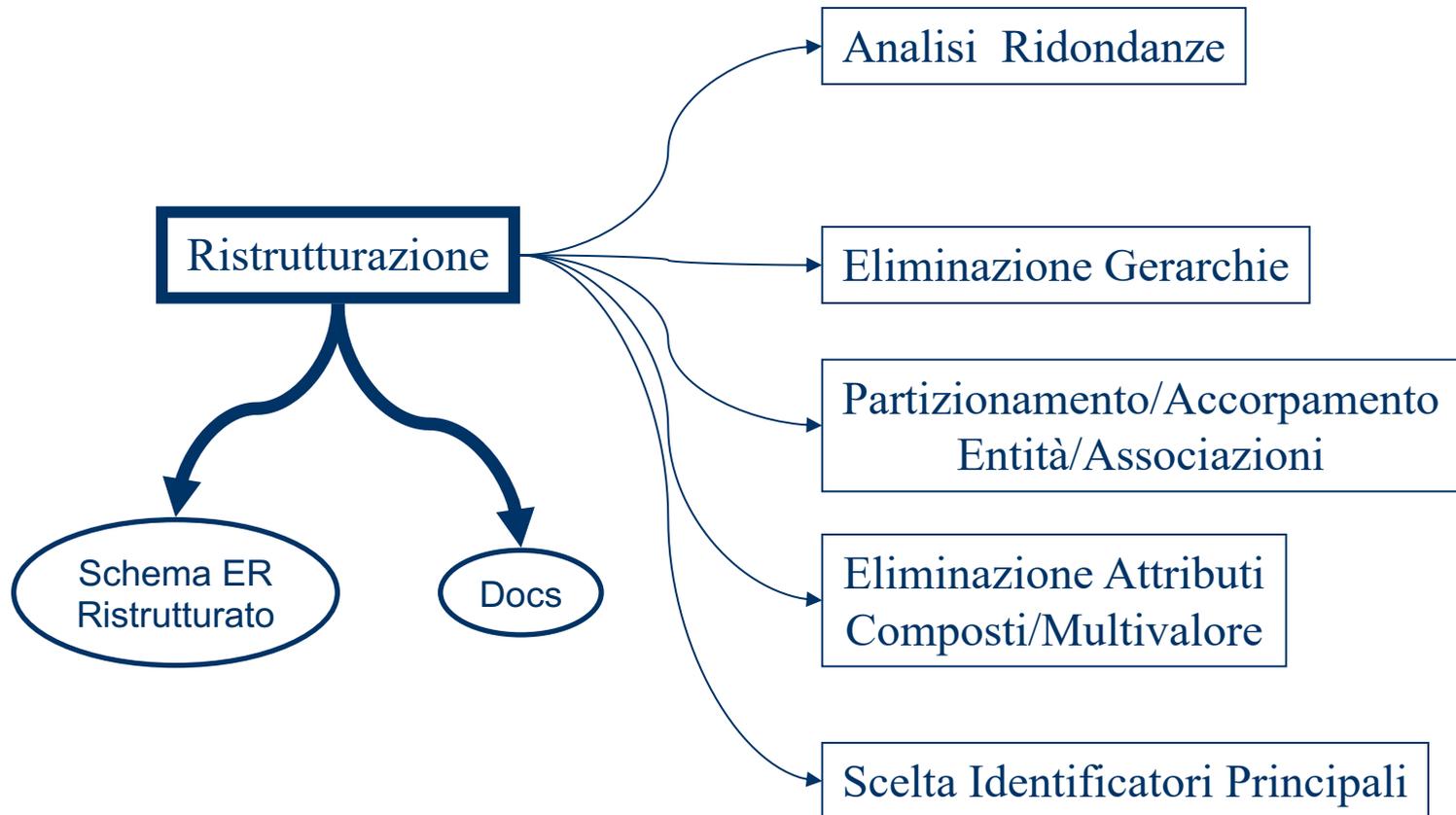
Trova i dati e i dati degli studenti di un corso, I, 3 V/G



## Tavola degli Accessi

Corso	E	1	L
R2	R	5	L
Studente	E	5	L

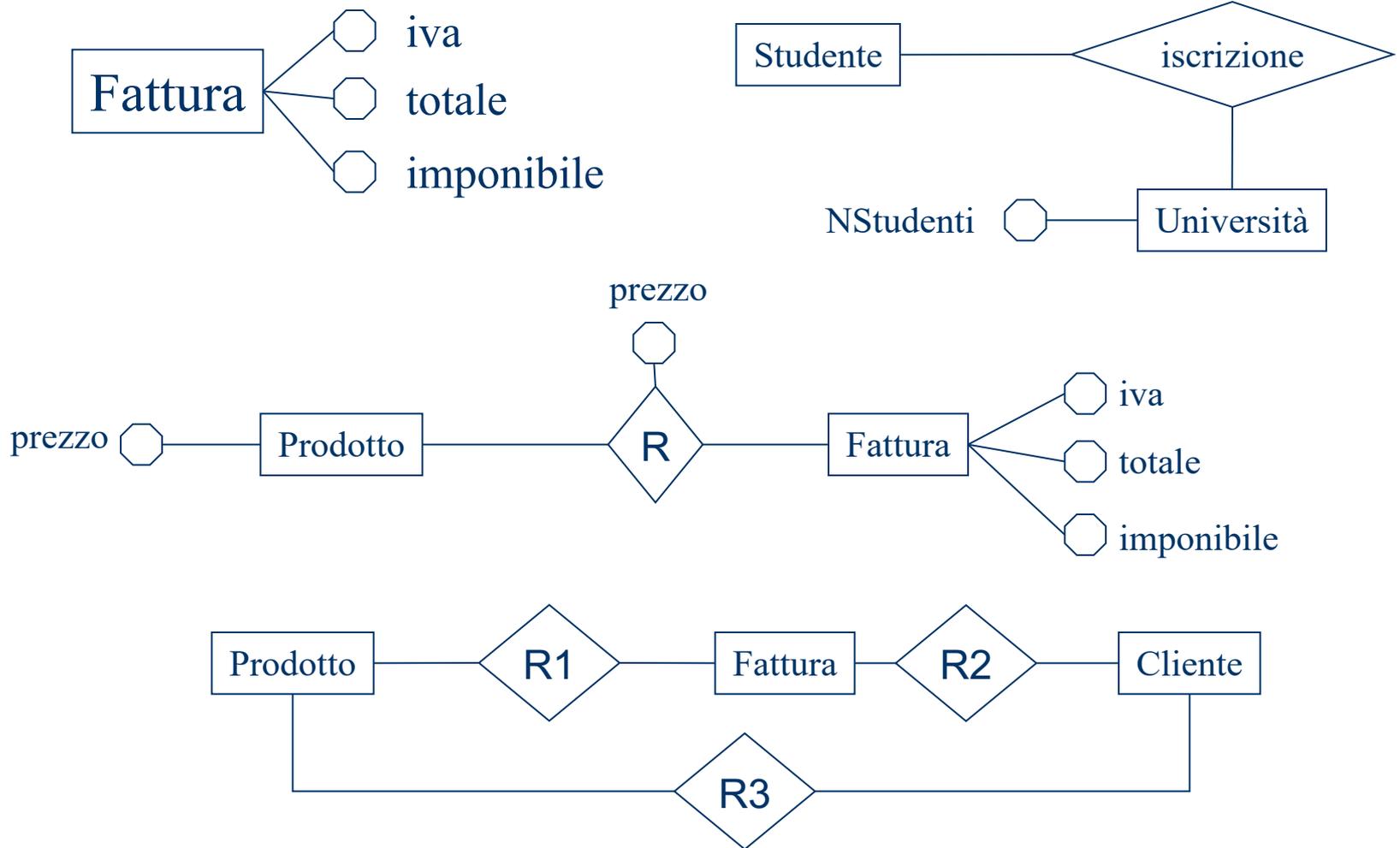
# Ristrutturazione



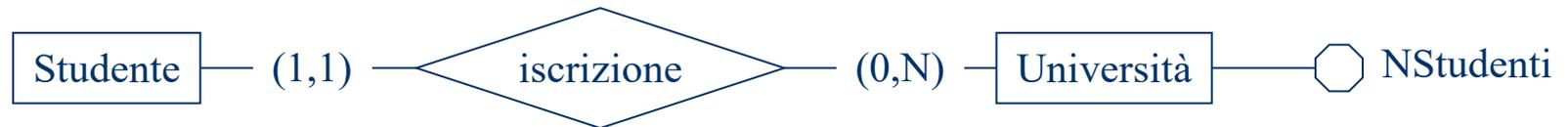
# Analisi delle Ridondanze

- Analisi e non Eliminazione.
- Ridondanza:
  - dato che può essere ottenuto (eventualmente con un costo computazionale) da altri dati.
- Pro di una ridondanza:
  - quando ci serve, il dato è già pronto
    - alcune operazioni sono velocizzate
- Contro di una ridondanza:
  - il dato ridondante occupa spazio (????);
  - il dato ridondante deve essere tenuto aggiornato
    - alcune operazioni sono rallentate.

# Esempi di Ridondanze



# Analisi delle Ridondanze



## Tavola dei Volumi

Studente	10000
Università	5
Iscrizione	10000

- Operazioni

- op1: inserisci nuovo studente, 300 V/G, I
- op2: trova numero studenti Università, 2V/S, I

# Con Ridondanza

- Memoria aggiuntiva rispetto all'assenza di ridondanza:
  - 4\*5 Byte (int su 4 byte)

<b>Tavola degli Accessi</b>		<b>op1</b>
Studente 1	S	
Iscrizione	1	S
Università	1	L
Università	1	S

<b>Tavola degli Accessi</b>		<b>op2</b>
Università	1	L
Costo = 1L		
Costo Settimanale = 2L		

Costo = (3S+1L) = 7L  
Costo Giornaliero = 2100 L  
Costo Settimanale = 10500L

**Costo Totale = 10502L**

# Senza Ridondanza



Tavola degli Accessi		op1
Studente	1	S
Iscrizione	1	S

Costo =  $2S = 4L$   
Costo Giornaliero = 1200L  
Costo Settimanale = 7000L

Tavola degli Accessi		op2
Università	1	L
Iscrizione	2000	L

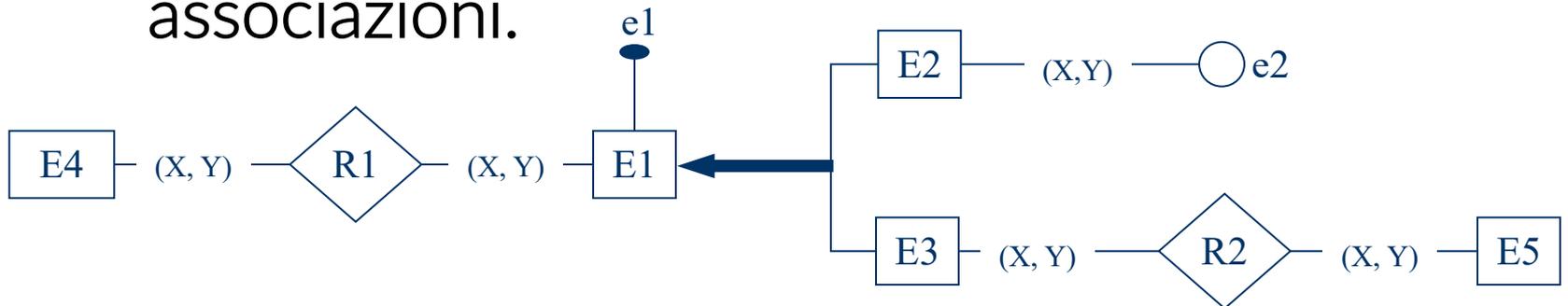
Costo = 2001L  
Costo Settimanale = 4002L

**Costo Totale = 11002L**

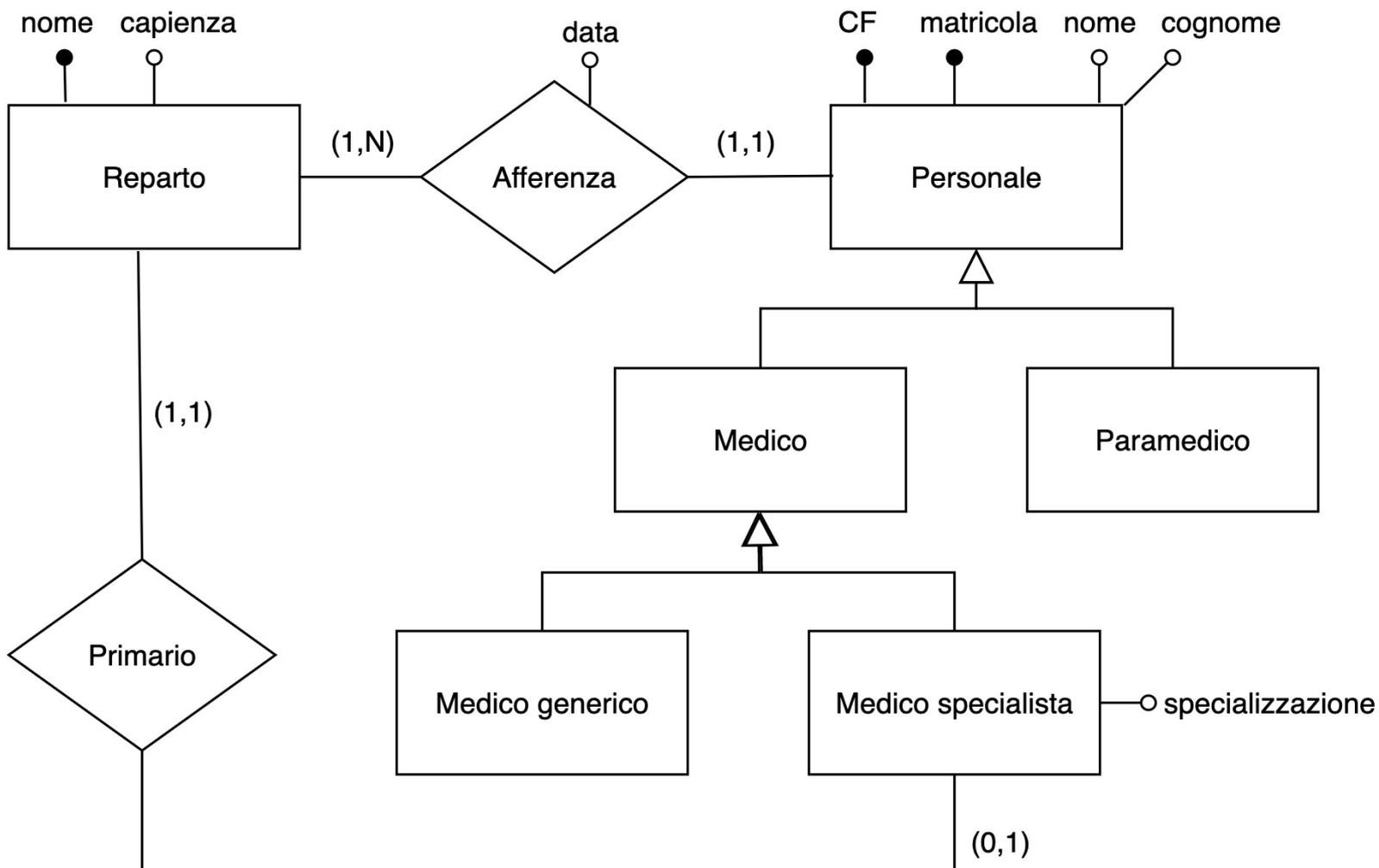
**Se op2 diventa 1 V/S, conviene la soluzione senza ridondanza**

# Eliminazione delle gerarchie

- Tre soluzioni possibili
  - Accorpamento delle entità figlie nell'entità padre
  - Accorpamento dell'entità padre nelle entità figlie
  - Sostituzione della generalizzazione con associazioni.



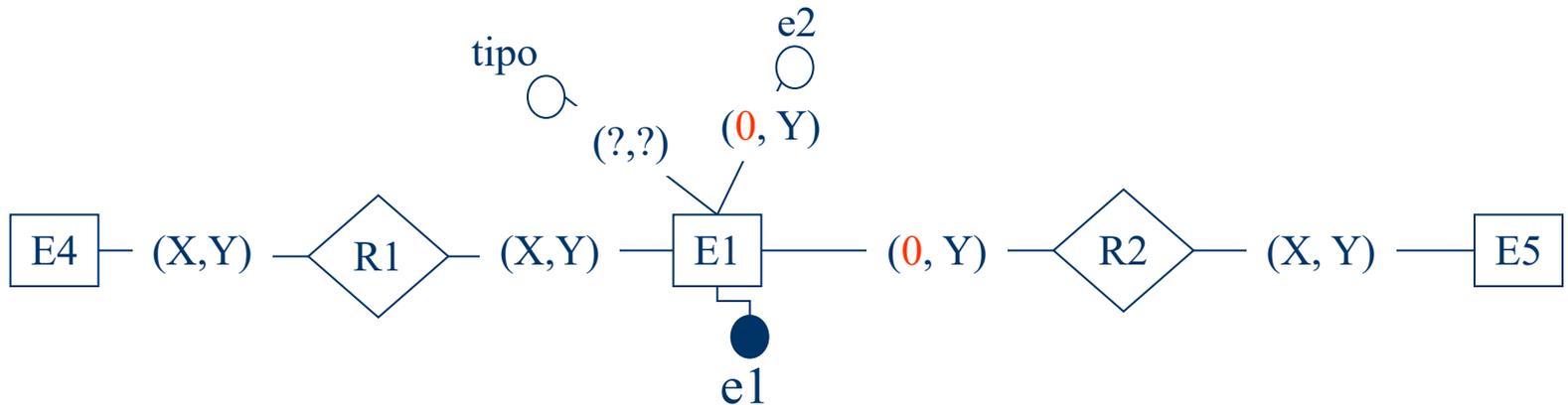
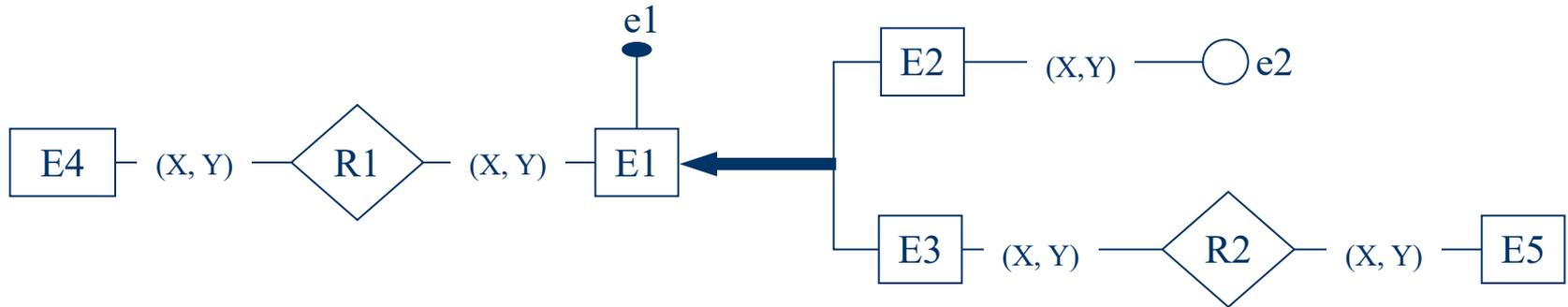
# Esempio



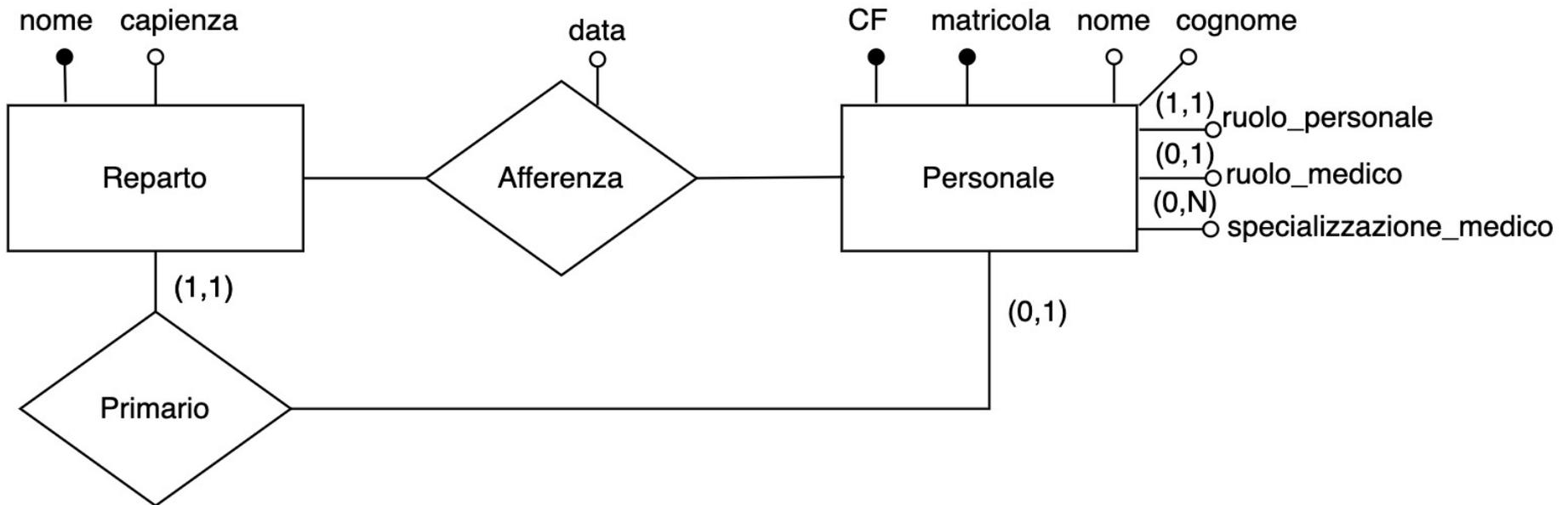
# Prima Soluzione

- Accorpamento dei figli con il padre
  - Si introduce, in E1, un attributo che identifica il tipo di entità figlia di appartenenza.
  - Sulla cardinalità di tale attributo bisogna fare molta attenzione.
  - Conviene se le operazioni importanti non fanno molta differenza tra entità padre e figlie.

# Prima Soluzione



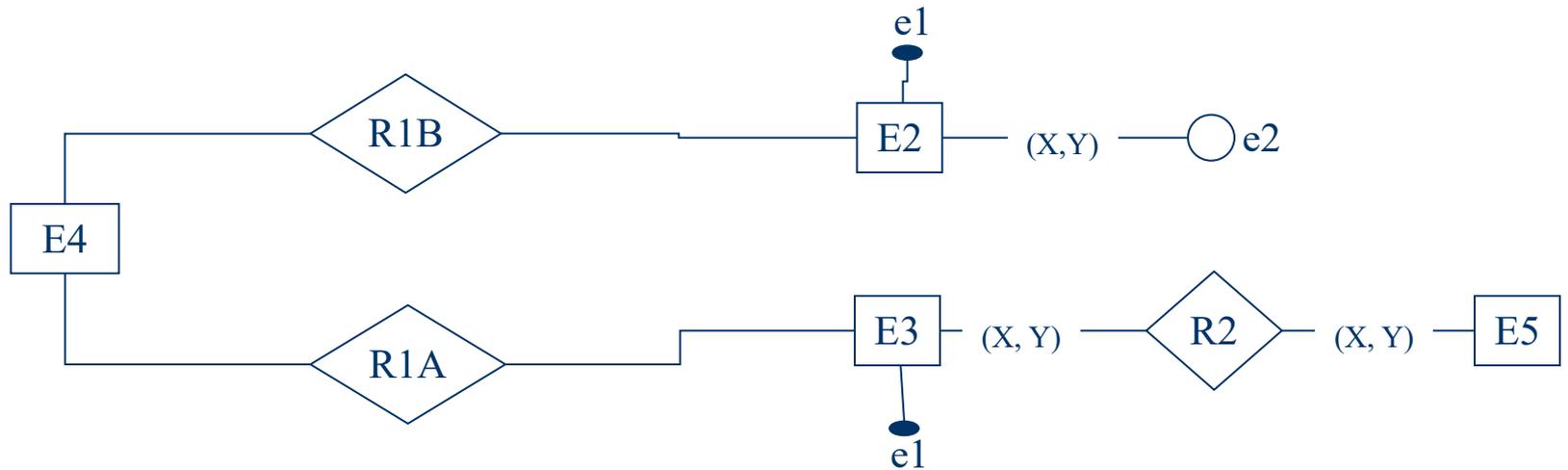
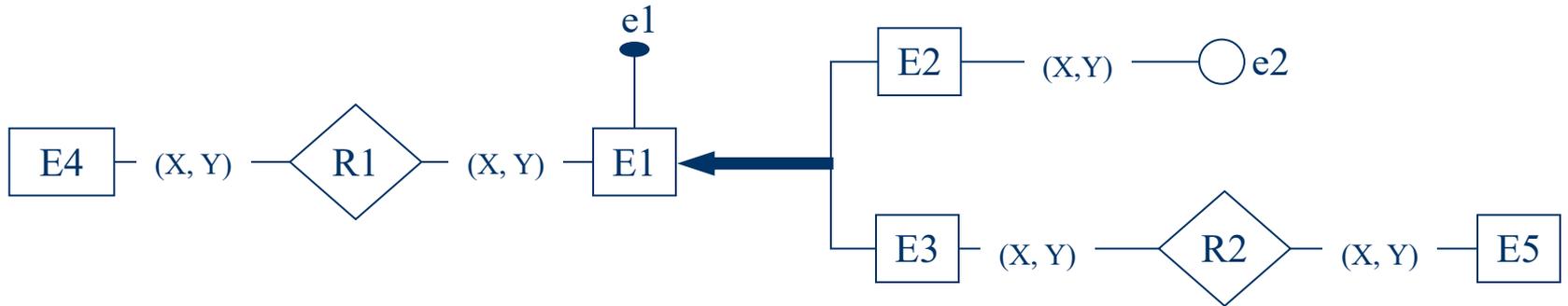
# Esempio



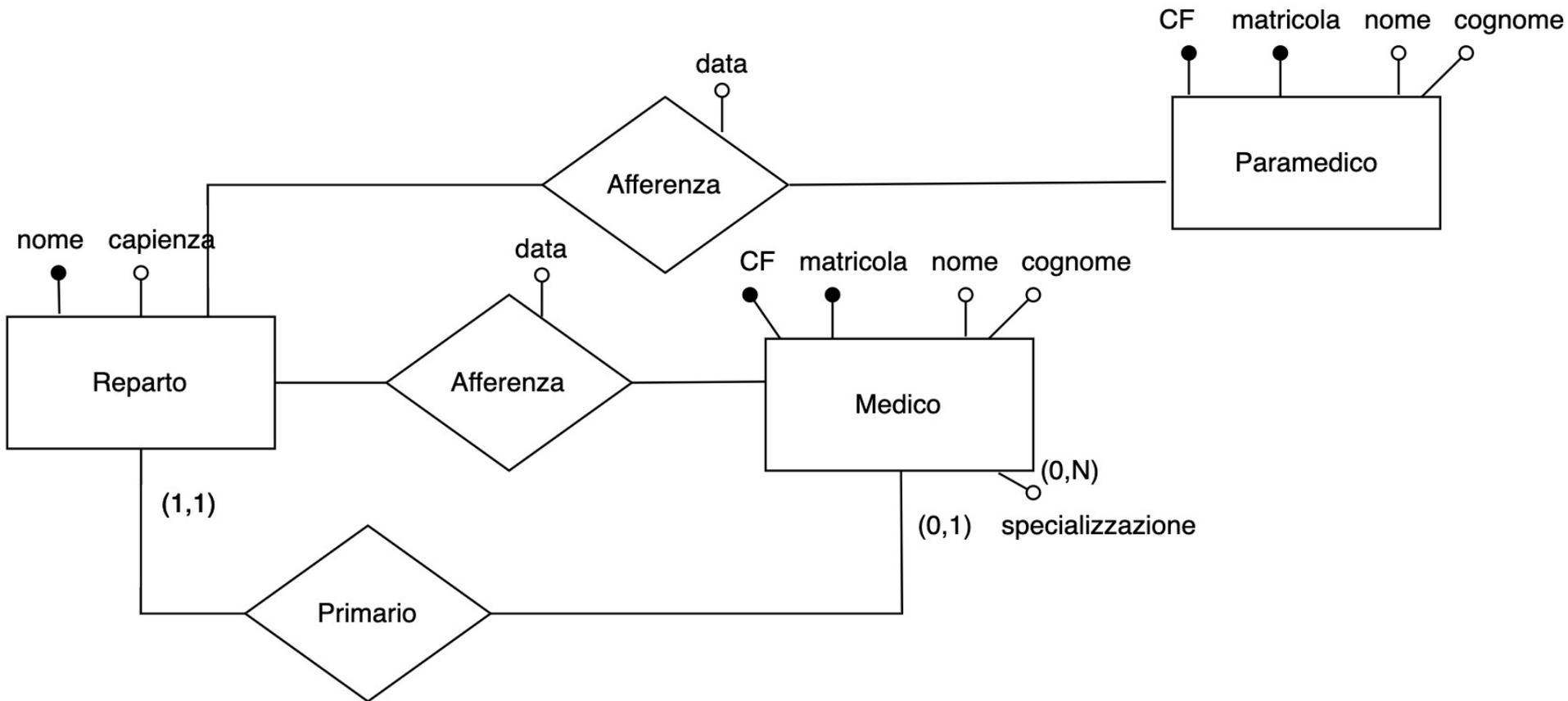
# Seconda Soluzione

- Accorpamento del padre con i figli.
  - È possibile solo se la generalizzazione è totale.
  - È vantaggiosa se le operazioni importanti fanno molta differenza tra le entità figlie.

# Seconda Soluzione



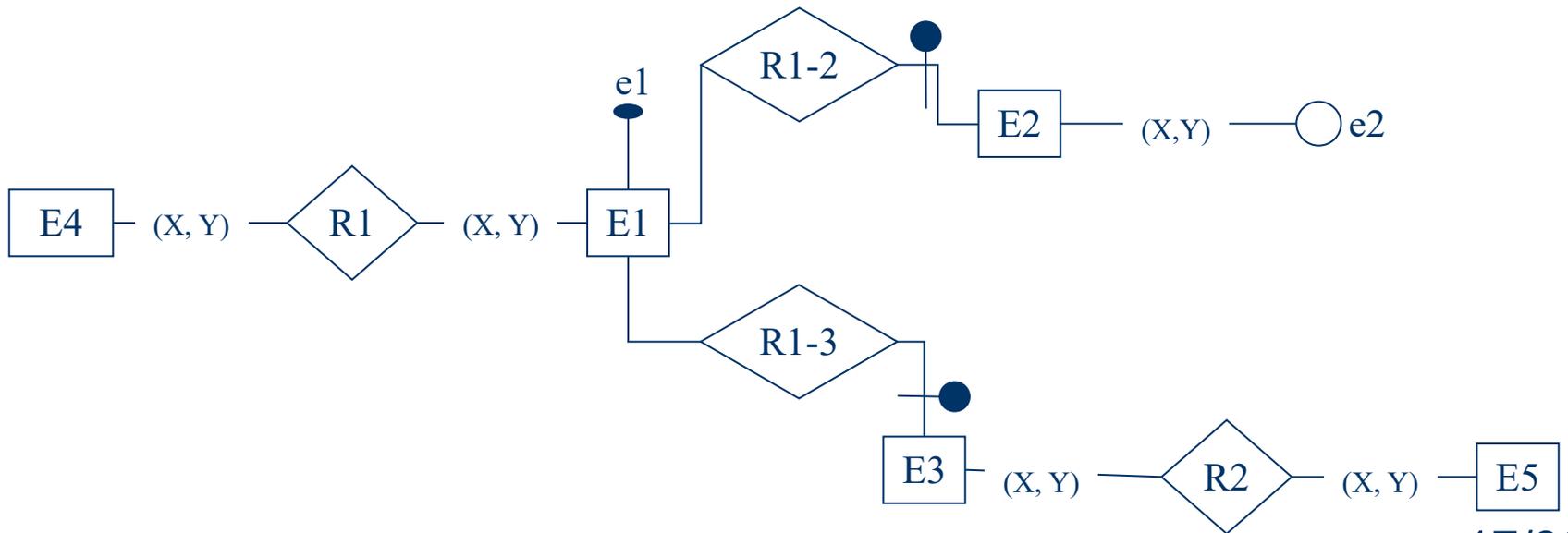
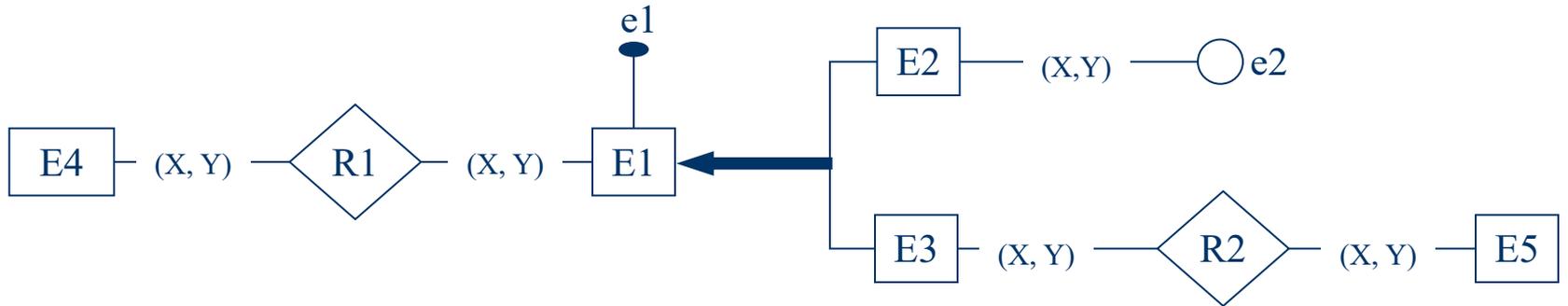
# Esempio



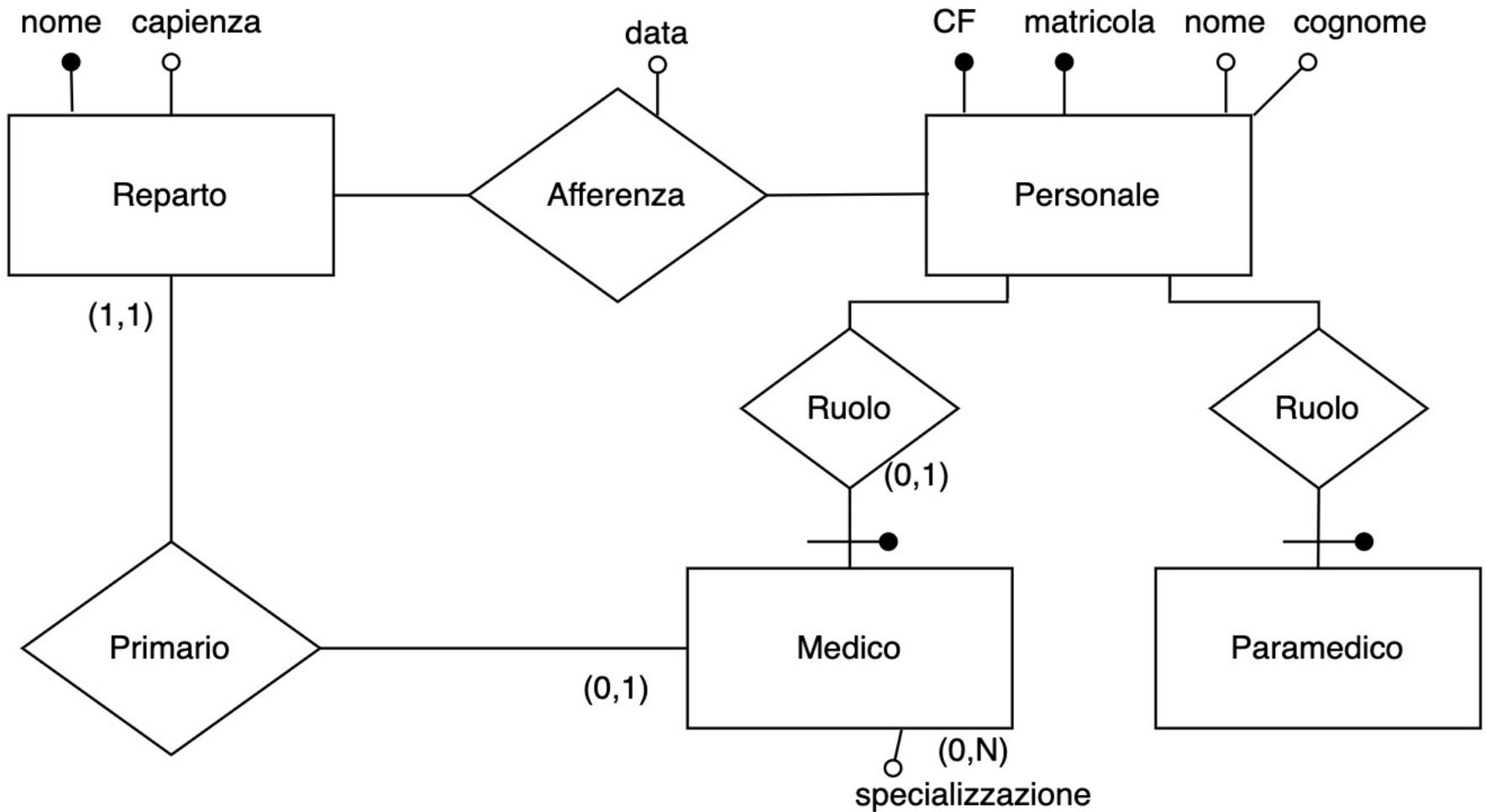
# Terza Soluzione

- Sostituzione della generalizzazione con associazioni.
  - È la soluzione più generale.
  - Utile se le operazioni importanti fanno differenza tra entità padre ed entità figlie.
  - Se la generalizzazione è totale, c'è un vincolo che non riesco ad esprimere.
  - Anche il vincolo sulla esclusività della generalizzazione non è esprimibile direttamente.

# Terza Soluzione

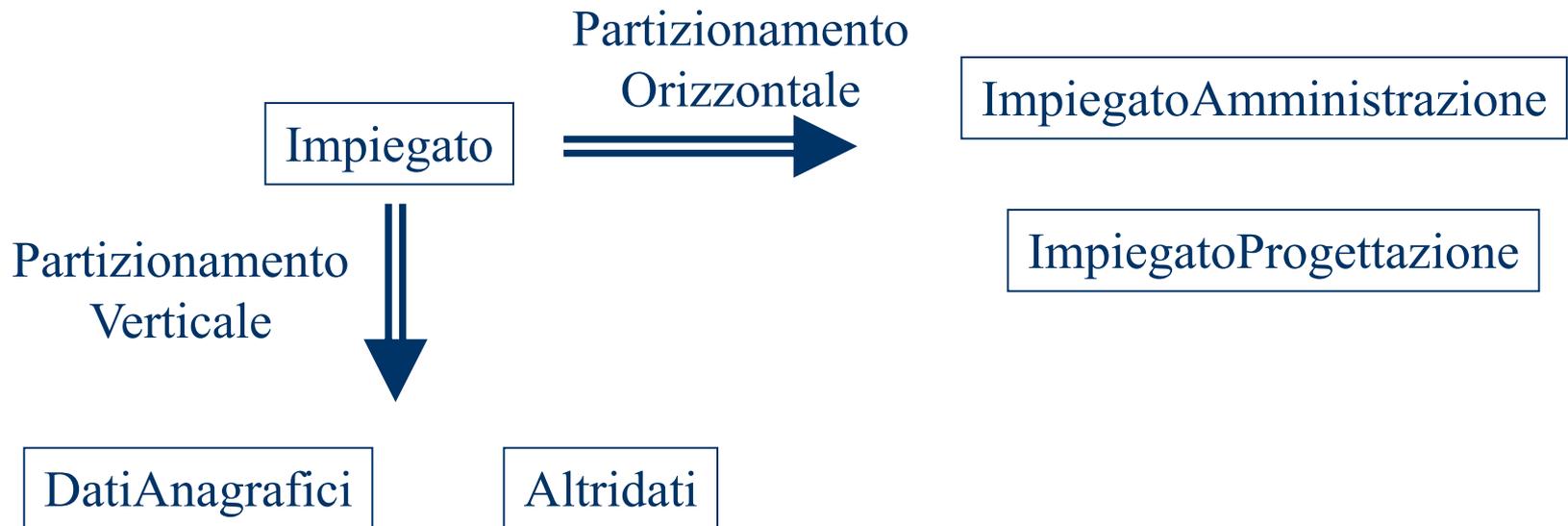


# Esempio



# Accorpamento/Partizionamento di Entità

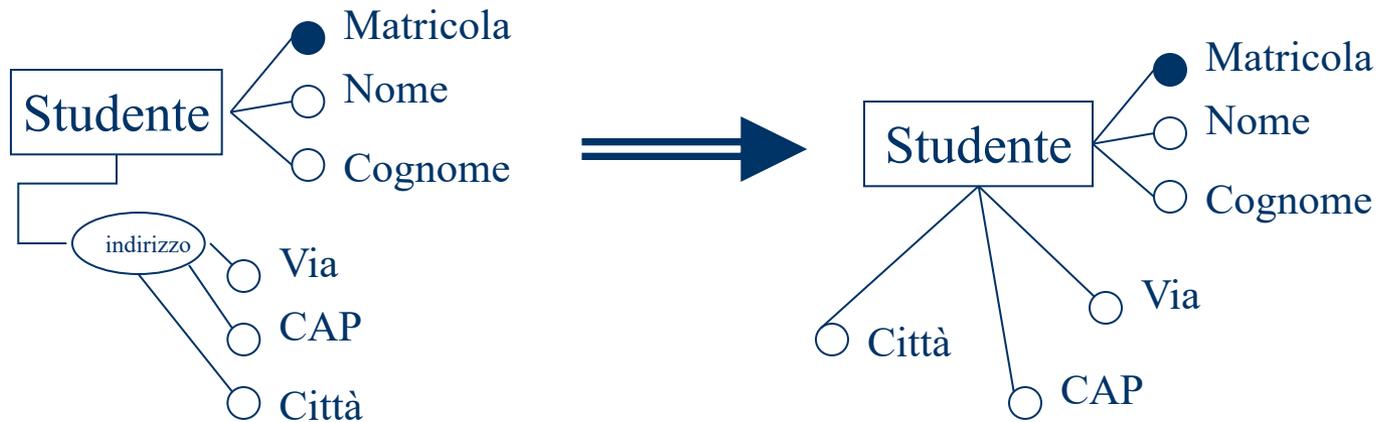
- Per migliorare l'efficienza può essere utile partizionare una entità.



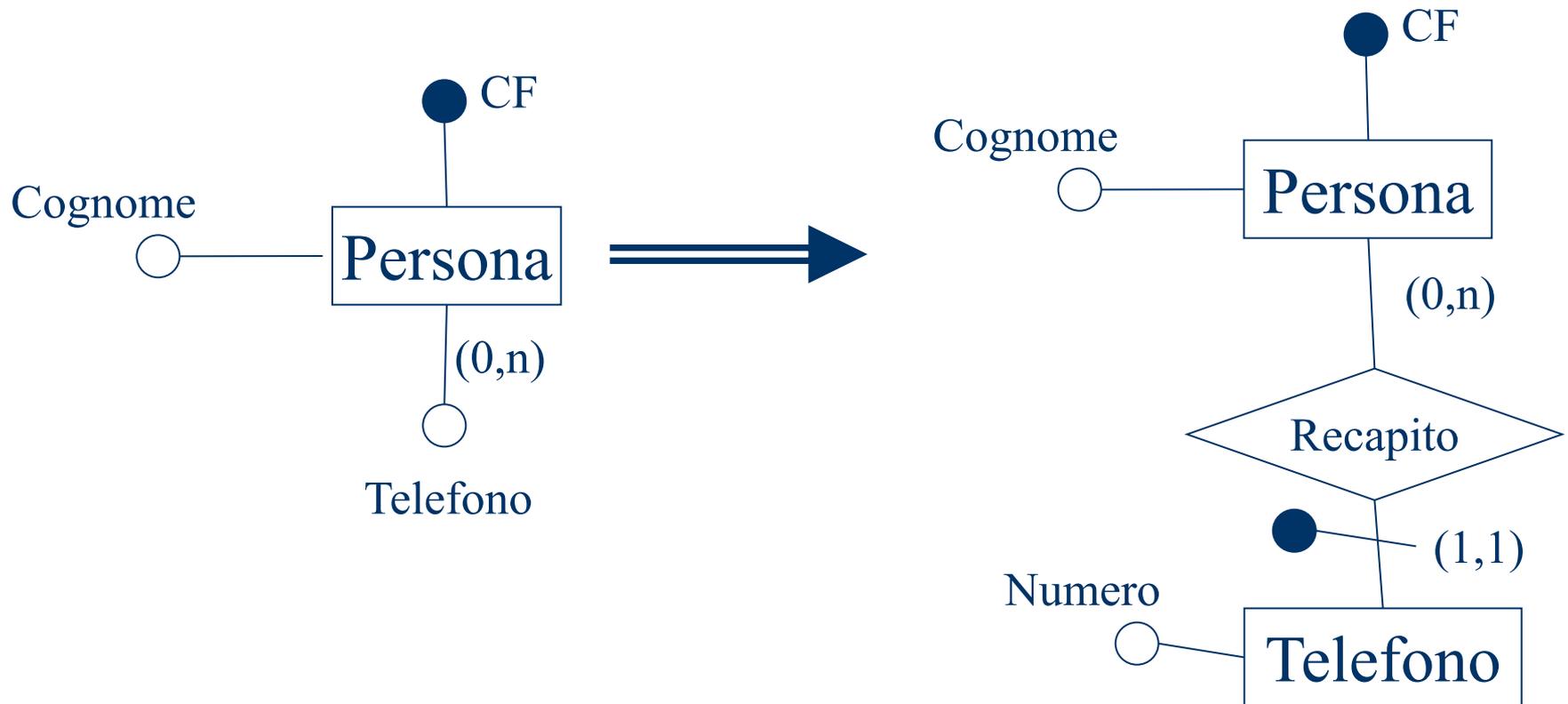
# Accorpamento/Partizionamento

- Nel partizionamento verticale è necessario duplicare la chiave.
- Tutte le associazioni che fanno riferimento all'entità vanno replicate.
- In alcuni casi potrebbe essere utile accorpare.
- Le tecniche di valutazione sono simili a quelle usate per le ridondanze.
- Discorsi analoghi possono essere applicati alle associazioni.

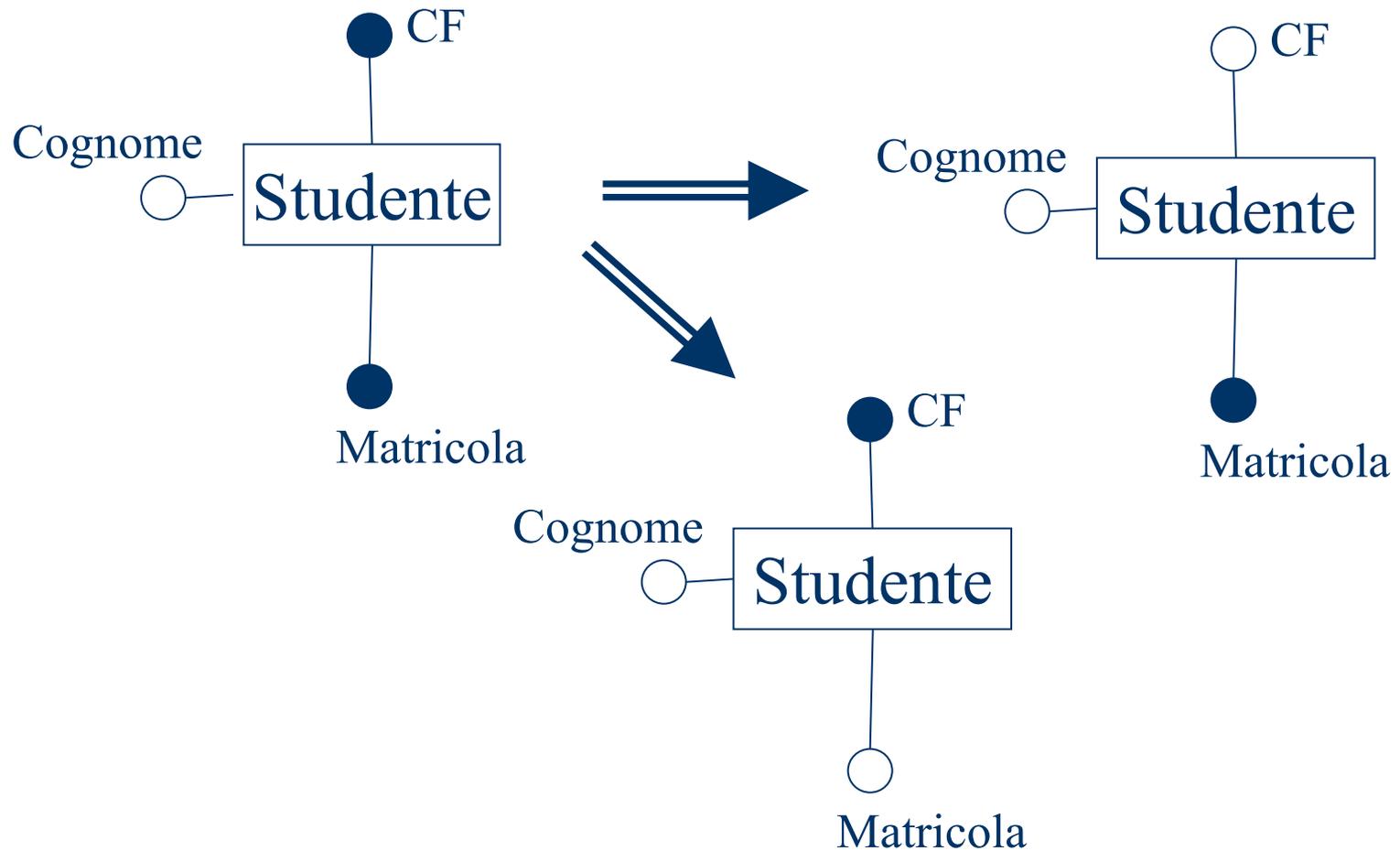
# Eliminazione Attributi Composti



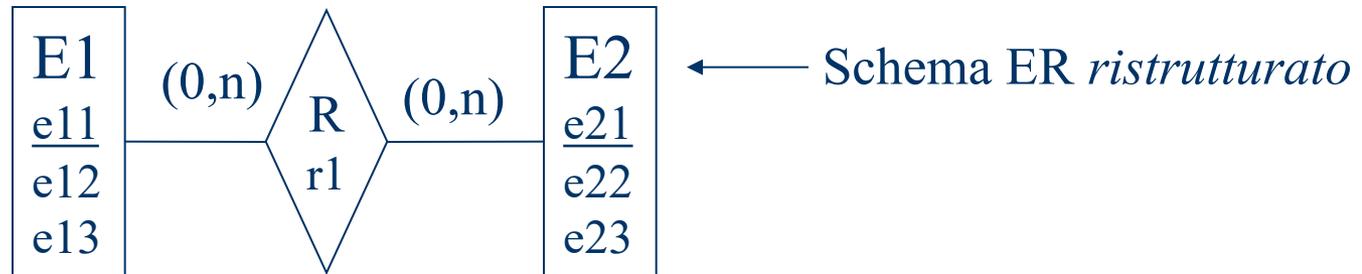
# Eliminazione Attributi Multivalore



# Scelta Identificatori Principali



# Traduzione



$E1(\underline{e11}, e12, e13)$

$E2(\underline{e21}, e22, e23)$

$R(\underline{e11}, \underline{e21}, r1)$

Vincoli di integrità referenziale tra

- $e11$  in  $R$  ed  $e11$  in  $E1$

- $e21$  in  $R$  ed  $e21$  in  $E2$

## Studente

Matricola Cognome

123456 Eramo

123467 Di Egidio

## Esame

Matricola Codice

123456 A12

123456 A13

123467 A13

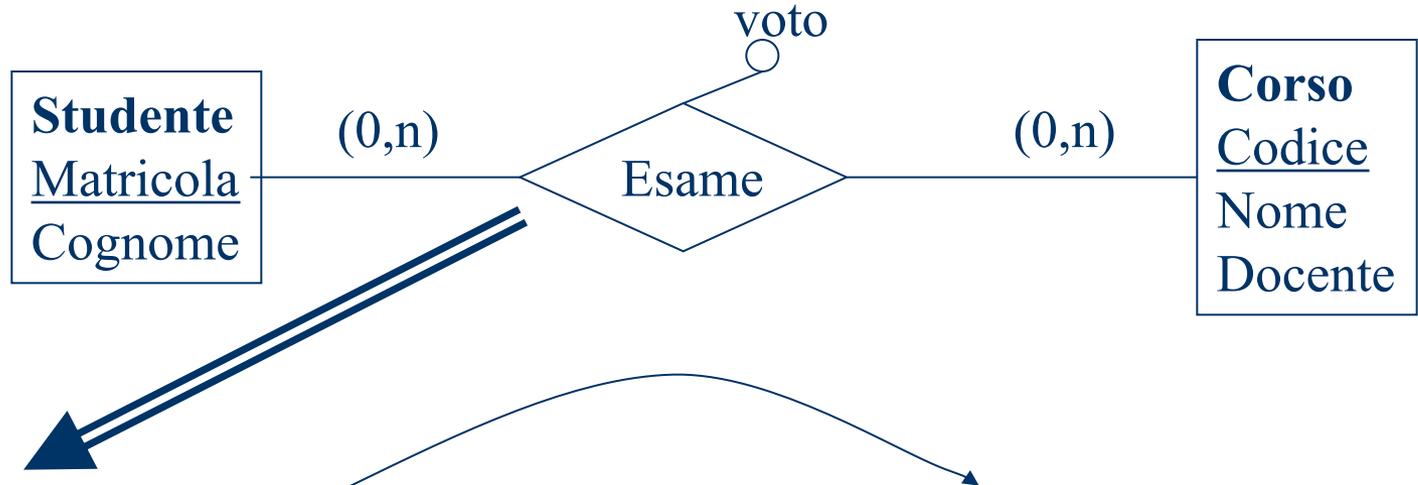
## Corso

Codice Nome

A12 ICT

A13 Semiotica

# Traduzione



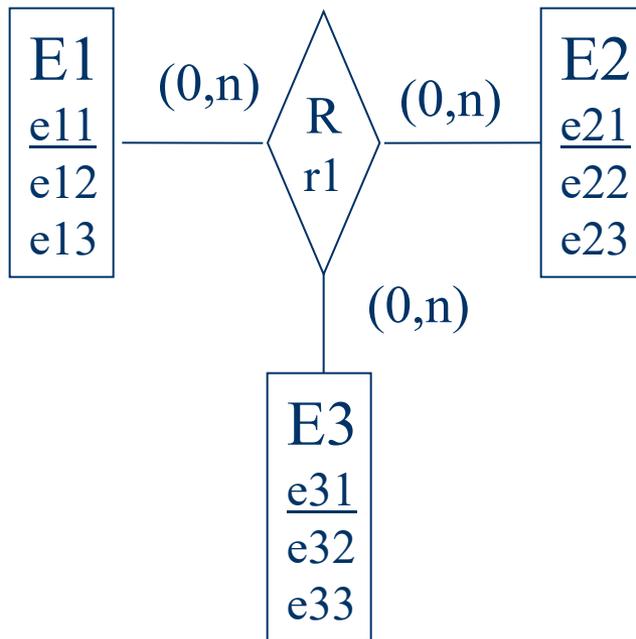
Studenti(Matricola, Cognome)  
Corsi(Codice, Nome, Docente)  
Esami(Matricola, Codice, voto)  
Vincoli di integrità referenziale tra  
• Matricola in Esami e Matricola in Studenti  
• Codice in Esami e Codice in Corsi

Studenti(Matricola, Cognome)  
Corsi(Codice, Nome, Docente)  
Esami(Studente, Corso, voto)  
Vincoli di integrità referenziale tra  
• Studente in Esami e Matricola in Studenti  
• Corso in Esami e Codice in Corsi

# Traduzione



# Traduzione



E1(e11,e12,e13)

E2(e21,e22,e23)

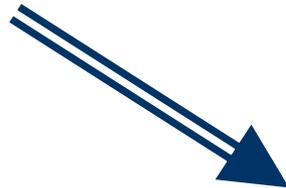
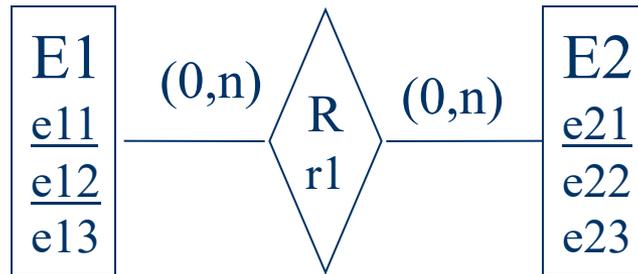
E3(e31,e32,e33)

R(e11,e21,e31,r1)

Vincoli di integrità referenziale tra

- e11 in R ed e11 in E1
- e21 in R ed e21 in E2
- e31 in R ed e31 in E3

# Traduzione



$E1(\underline{e11}, \underline{e12}, e13)$

$E2(\underline{e21}, e22, e23)$

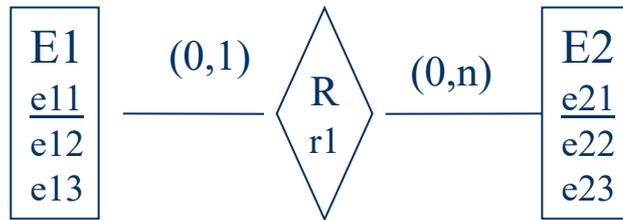
$R(\underline{e11}, \underline{e12}, \underline{e21}, r1)$

Vincoli di integrità referenziale tra

- $(e11, e12)$  in R ed  $(e11, e12)$  in E1

- $e21$  in R ed  $e21$  in E2

# Traduzione



$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23)$   
 $R(\underline{e11}, \underline{e21}, r1)$   
Vincoli di integrità referenziale tra

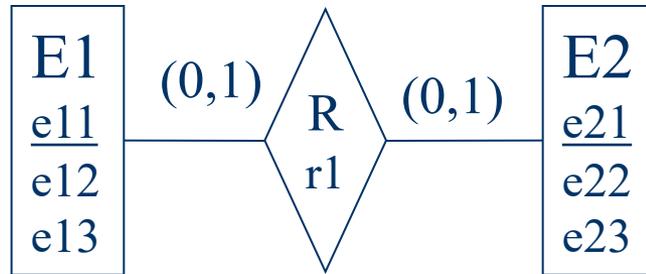
- $e11$  in R ed  $e11$  in E1
- $e21$  in R ed  $e21$  in E2

$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23)$   
 $R(\underline{e11}, \underline{e21}, r1)$   
Vincoli di integrità referenziale tra

- $e11$  in R ed  $e11$  in E1
- $e21$  in R ed  $e21$  in E2

$E1(\underline{e11}, e12, e13, \underline{e21}, r1)$   
 $E2(\underline{e21}, e22, e23)$   
Vincolo tra

- $e21$  in E1 ed  $e21$  in E2



$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23)$   
 $R(\underline{e11}, \underline{e21}, r1)$   
 Vincoli di integrità referenziale tra
 

- $e11$  in  $R$  ed  $e11$  in  $E1$
- $e21$  in  $R$  ed  $e21$  in  $E2$



$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23)$   
 $R(\underline{e11}, \underline{e21}, r1)$   
 Vincoli tra
 

- $e11$  in  $R$  ed  $e11$  in  $E1$
- $e21$  in  $R$  ed  $e21$  in  $E2$



$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23)$   
 $R(\underline{e11}, \underline{e21}, r1)$   
 Vincoli tra
 

- $e11$  in  $R$  ed  $e11$  in  $E1$
- $e21$  in  $R$  ed  $e21$  in  $E2$



$E1(\underline{e11}, e12, e13, \underline{e21}, r1)$   
 $E2(\underline{e21}, e22, e23)$   
 Vincolo tra
 

- $e21$  in  $E1$  ed  $e21$  in  $E2$



$E1(\underline{e11}, e12, e13)$   
 $E2(\underline{e21}, e22, e23, \underline{e11}, r1)$   
 Vincolo tra
 

- $e11$  in  $E2$  ed  $e11$  in  $E1$

# Traduzione

