



Basi Dati

La progettazione logica - Ristrutturazione

a.a. 2021/2022
Prof.ssa G. Tortora

Obiettivo

- Uno schema logico in grado di descrivere tutte le informazioni contenute nello schema ER prodotto nella fase di progettazione concettuale.

La ristrutturazione

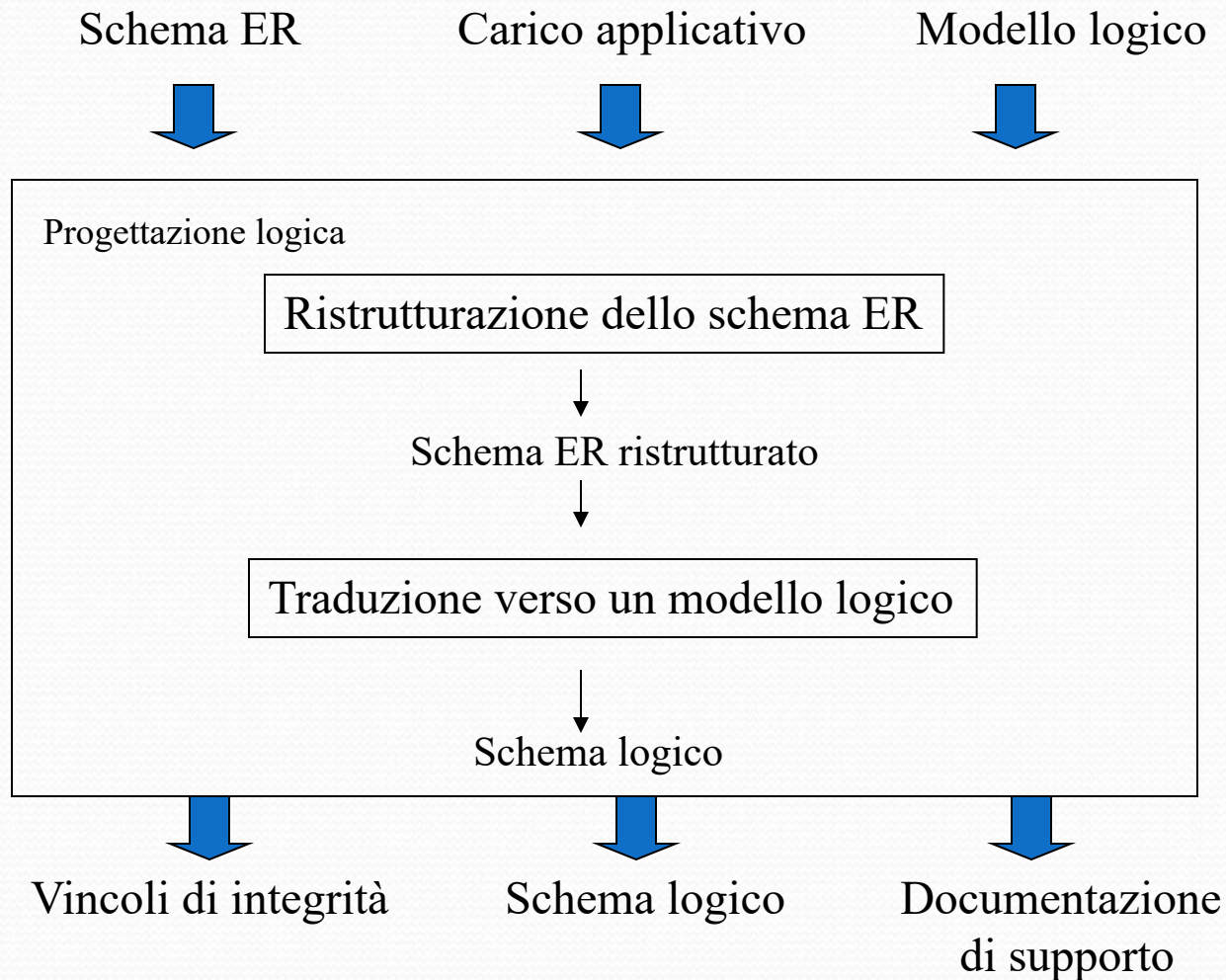
- Prima di passare allo schema logico è necessario ristrutturare lo schema ER per:
 - Semplificare la traduzione:
non tutti i costrutti del modello ER hanno una traduzione naturale nei modelli logici.
 - Ottimizzare il progetto:
rendere più efficiente l'esecuzione delle operazioni.



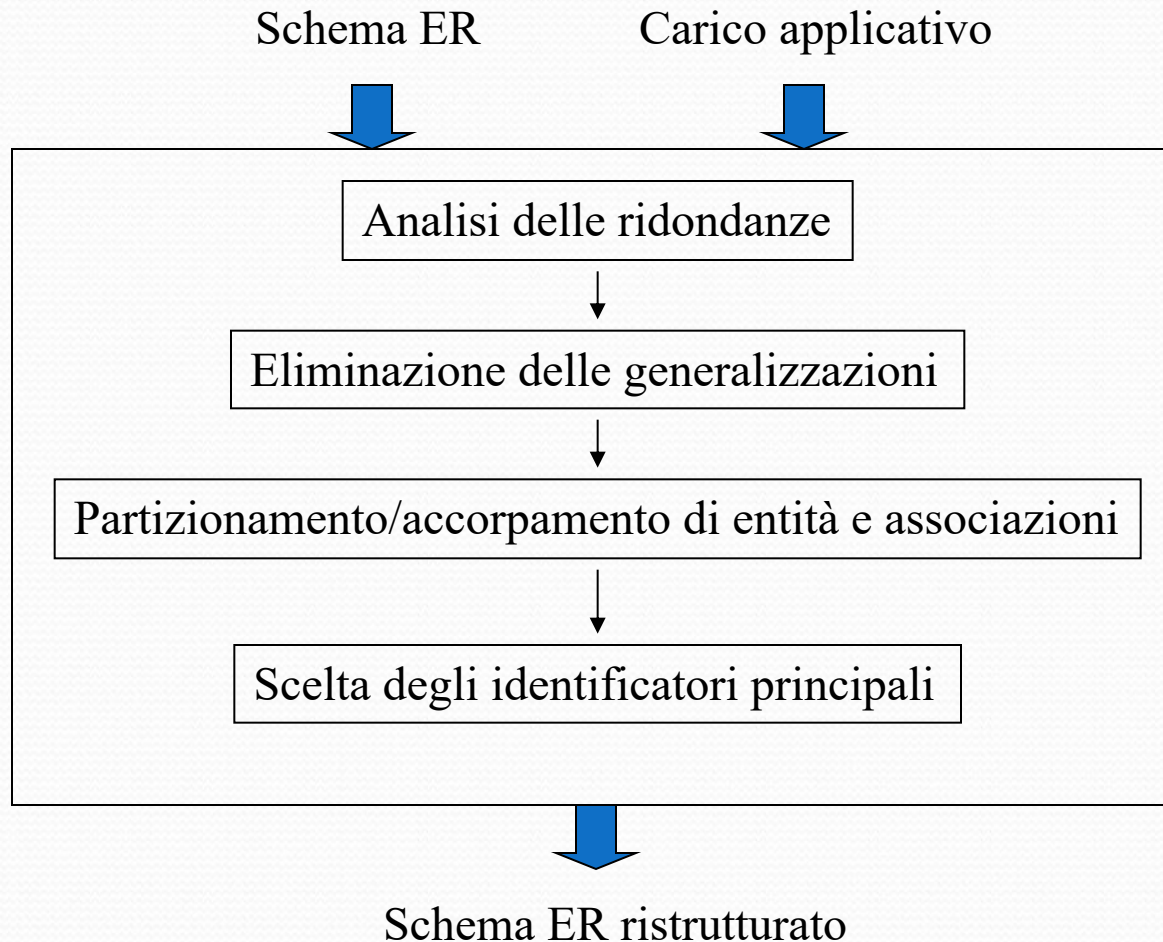
Attività di riorganizzazione e traduzione

- Ristrutturazione dello schema ER:
si basa su criteri di ottimizzazione dello schema.
- Traduzione verso il modello logico:
fa riferimento ad uno specifico modello logico.
- I dati ingresso sono lo schema concettuale ed il carico applicativo (dimensione dei dati e caratteristiche delle operazioni).

Progettazione logica di basi di dati



Ristrutturazione di schemi ER



Analisi delle prestazioni su schemi ER

- Lo schema ER ristrutturato non è uno schema concettuale in senso stretto:
tiene conto degli aspetti realizzativi per rendere più efficienti le operazioni previste.
- Lo schema ER va modificato per ottimizzare alcuni indici di prestazione del progetto.

Analisi delle prestazioni su schemi ER

- Indici di prestazioni (non valutabili in maniera precisa in sede di progettazione logica):
 - Costo di una operazione (# occorrenze di entità e relazioni visitate per rispondere a una operazione).
 - Occupazione di memoria.

Per studiare questi parametri è necessario conoscere:



- Volume dei dati:
 - # occorrenze di entità e relazioni.
 - Dimensioni di ciascun attributo.
- Caratteristiche delle operazioni:
 - Tipo (interattiva o batch).
 - Frequenza (# medio di esecuzioni in un t).
 - Dati coinvolti.

Tavole di analisi

- Tavola dei volumi:

concetto tipo volume

- Vengono riportati tutti i concetti dello schema (entità e relazioni) con il volume previsto a regime.

- Tavola delle operazioni:

operazioni tipo frequenza

- Viene riportata per ogni operazione la frequenza prevista, il tipo (interattiva o batch).

Tavole di analisi (2)

- Tavola degli accessi

concetto costruito accessi tipo

- Viene riportata, per ogni operazione, il numero di accessi ai concetti coinvolti ed il tipo di accesso (L/S).
- Le operazioni in scrittura (S) sono più onerose di quelle in lettura (L).

Esempio

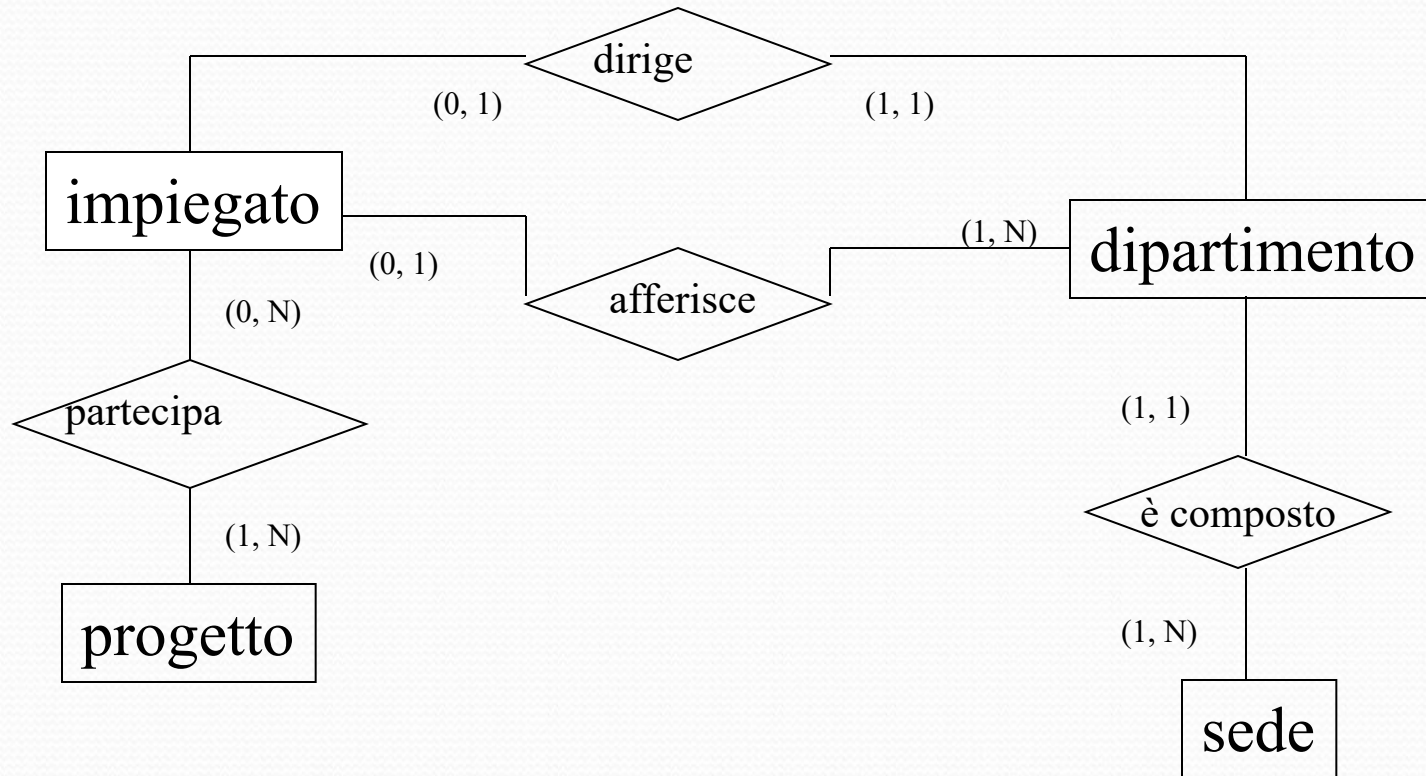


Tavola dei volumi

Concetto	Tipo	Volume
Sede	E	10
Dipartimento	E	80
Impiegato	E	2000
Progetto	E	500
E' composto	R	80
Afferisce	R	1900
Dirige	R	80
Partecipa	R	6000

Numero delle occorrenze delle associazioni

- Nella tavola dei volumi.
- Dipende da due parametri:
 - Numero delle occorrenze delle entità coinvolte nelle associazioni.
 - Numero medio di partecipazioni di un'occorrenza di entità alle occorrenze di associazioni.

Es: supponiamo che un impiegato lavori in media a tre progetti.

Operazioni

1. Assegna un impiegato ad un progetto.
2. Trova i dati di un impiegato, del dipartimento nel quale lavora e dei progetti ai quali partecipa.
3. Trova i dati di tutti gli impiegati di un certo dipartimento.
4. Per ogni sede, trova i suoi dipartimenti con il cognome del direttore e l'elenco degli impiegati del dipartimento.

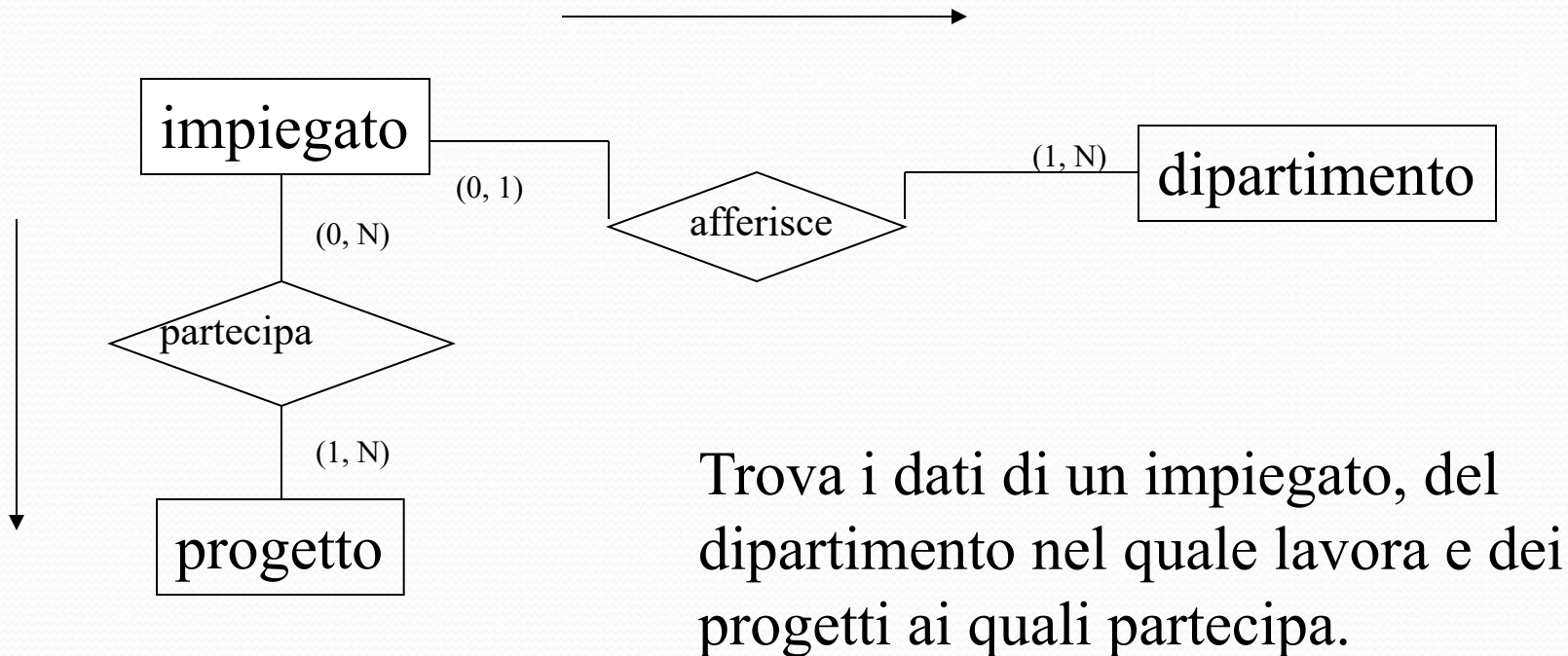
Tavola delle operazioni

Operazione	Tipo	Frequenza
Op. 1	I	50 al giorno
Op. 2	I	100 al giorno
Op. 3	I	10 al giorno
Op. 4	B	2 a settimana

Schema di operazione

- Per ogni operazione possiamo descrivere graficamente i dati coinvolti attraverso uno **schema di operazione**.
- È un frammento dello schema ER.
- Descrive il cammino logico da percorrere per accedere alle informazioni di interesse.

Schema di operazione (rif. Op. 2)



Costo di un'operazione

- Si stima il costo di un'operazione contando il numero degli accessi alle occorrenze di entità e di relazioni.

Tavola degli accessi (rif. Op. 2)

Concetto	Costrutto	Accessi	Tipo
Impiegato	E	1	L
Afferisce	R	1	L
Dipartimento	E	1	L
Partecipa	R	3	L
Progetto	E	3	L

Ristrutturazione di schemi ER

- La fase di ristrutturazione di uno schema ER è suddiviso:
 1. Analisi delle ridondanze.
 2. Eliminazione delle generalizzazioni.
 3. Partizionamento/accorpamento di entità e associazioni.
 4. Scelta degli identificatori primari.

Le ridondanze

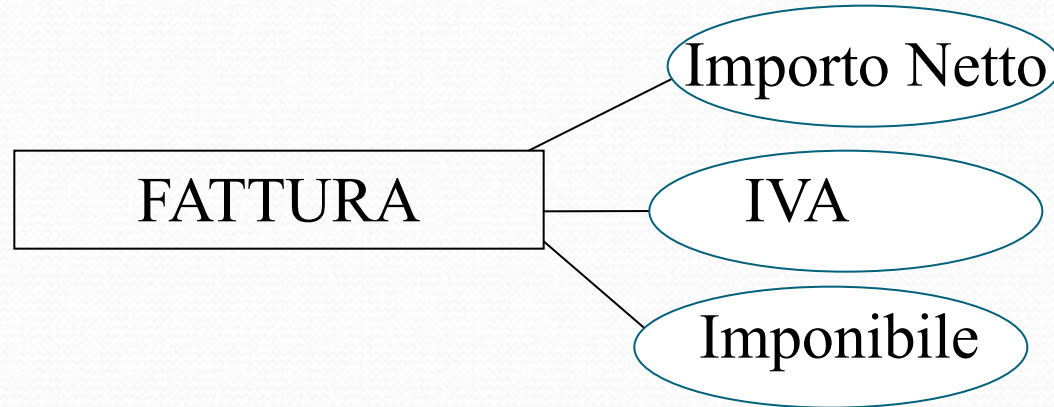
- Una **ridondanza** in uno schema concettuale corrisponde alla presenza di un dato che può essere derivato da altri dati.

Le ridondanze (2)

- Vantaggi
 - Semplificazione delle interrogazioni.
- Svantaggi
 - Appesantimento degli aggiornamenti.
 - Maggiore occupazione di spazio.

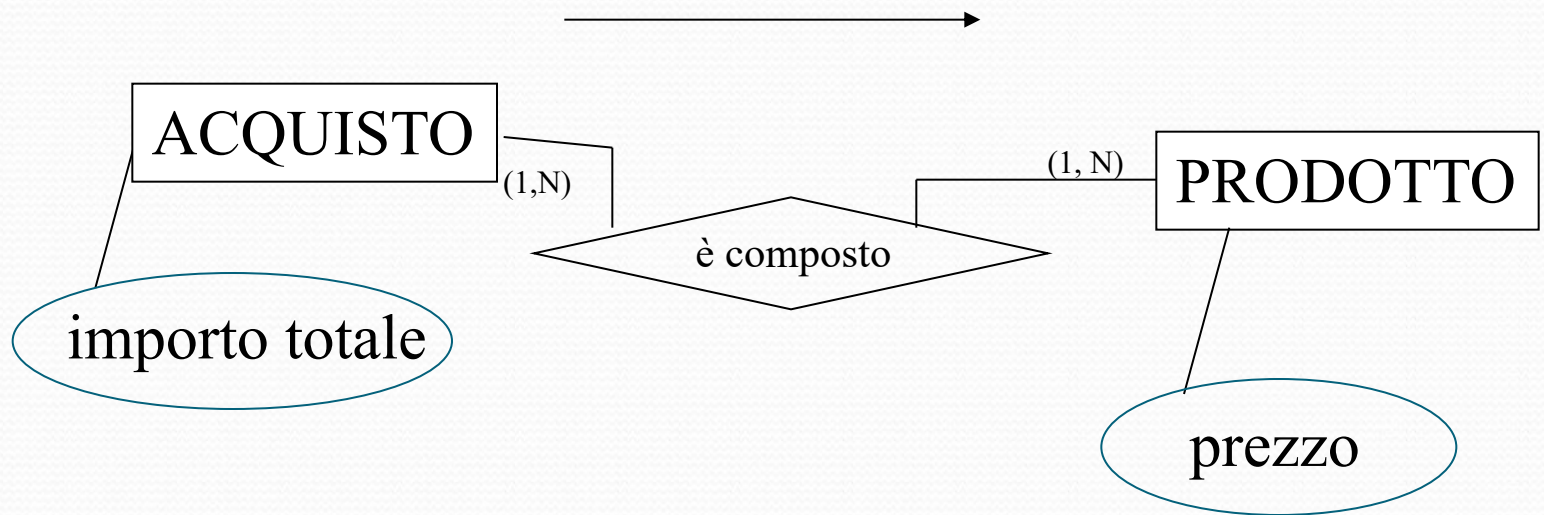
Attributi derivabili

- Da attributi della stessa entità / associazione:



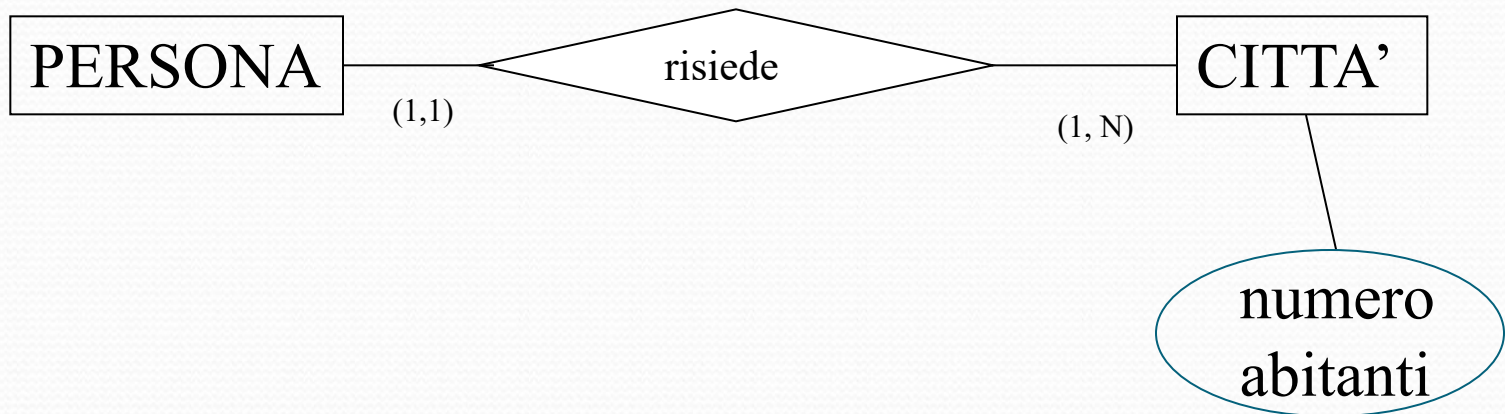
Attributi derivabili (2)

- Da attributi di altre entità / associazioni:



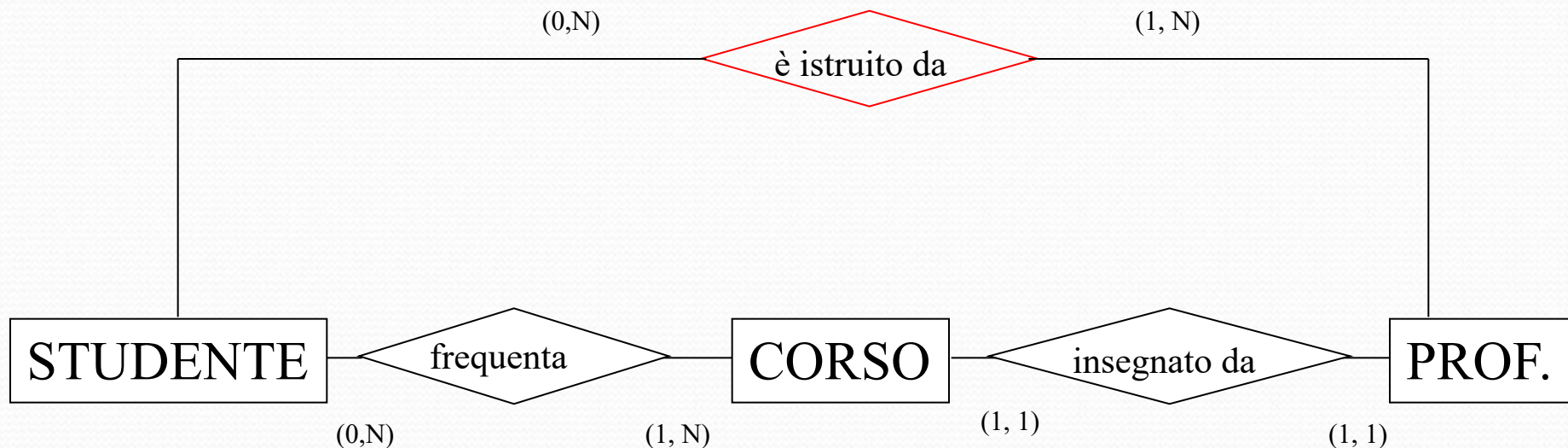
Attributi derivabili (3)

- Da operazioni di conteggio di occorrenze:



Attributi derivabili (4)

- Da associazioni in presenza di cicli:



- La presenza di cicli non genera necessariamente ridondanze:
 - La relazione **“è istruito da”** diventa **“è tesista di”**.

Inserire una ridondanza?

- La decisione va presa confrontando il costo delle operazioni che coinvolgono il dato ridondante e relativa occupazione di memoria nei casi di **presenza/assenza** di ridondanza.

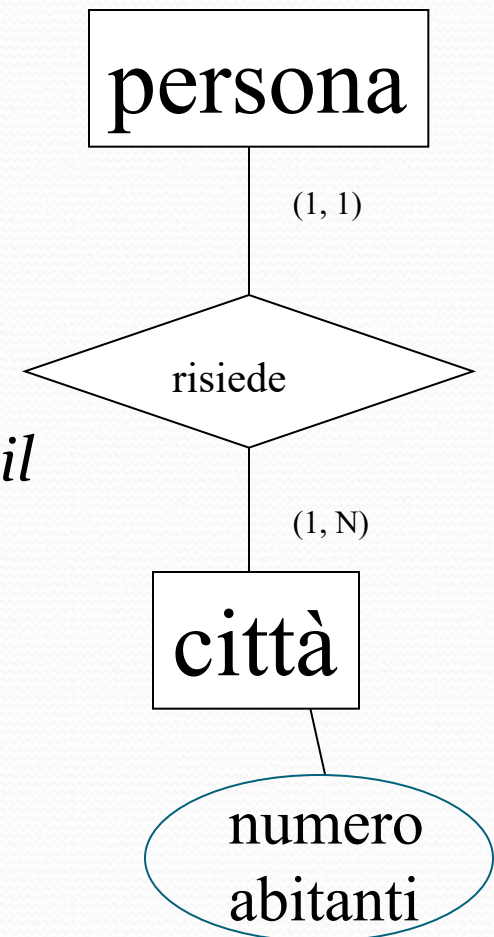
Tavole di analisi

- Tavola dei volumi:

• Città	E	200
• Persona	E	1000000
• Risiede	R	1000000

- Op_1 : *memorizza una nuova persona con la relativa città di residenza.*
- Op_2 : *stampa tutti i dati di una città (incluso il numero di abitanti).*
- Tavola delle operazioni:

• Op_1	I	500/giorno
• Op_2	I	2/giorno



“Numero di abitanti” → Città

- Proviamo a valutare gli indici di prestazione in caso di presenza del dato ridondante:
- $\text{Mem}(\text{num_abit}) \cong 4 \text{ byte} \rightarrow \text{dato ridondante} \cong 4 \times 200 = 800 \text{ byte} (\cong 1 \text{ kbyte})$
- Costo:
 - *Un accesso in scrittura ha un costo doppio rispetto ad un accesso in lettura.*

Tavole degli accessi (con ridondanza)

• Operazione 1

- Persona E 1 S
(per memorizzare una nuova persona)
- Risiede R 1 S
(per memorizzare una nuova coppia città-persona)
- Città E 1 L
(per cercare la città di interesse)
- Città E 1 S
(incrementa di 1 il numero di abitanti)

- $3S * 500/\text{giorno} + 1L * 500/\text{giorno}$



- 3500 accessi/giorno

(un accesso in scrittura vale il doppio di un accesso in lettura)

• Operazione 2

- Città E 1 L

- Trascurabile: $1L * 2/\text{giorno}$

Tavole degli accessi (senza ridondanza)

• Operazione 1

- Persona E 1 S
- Risiede R 1 S

(non si accede a Città per aggiornare il dato derivato)

- $2S * 500/\text{giorno}$
 \Downarrow
- 2000 accessi/giorno
(contano doppio gli accessi in scrittura)

• Operazione 2

- Città E 1 L
- Risiede R 5000¹ L

- $1L + 1L * 5000 * 2/\text{giorno}$
 \Downarrow
- 10000 accessi/giorno

¹ (#abitanti/città=1000000/200)

- **12.000 accessi vs 3.500 accessi + 1Kbyte**
- Gli accessi in lettura necessari per calcolare il dato derivato sono molti di più degli accessi in scrittura per mantenerlo aggiornato.

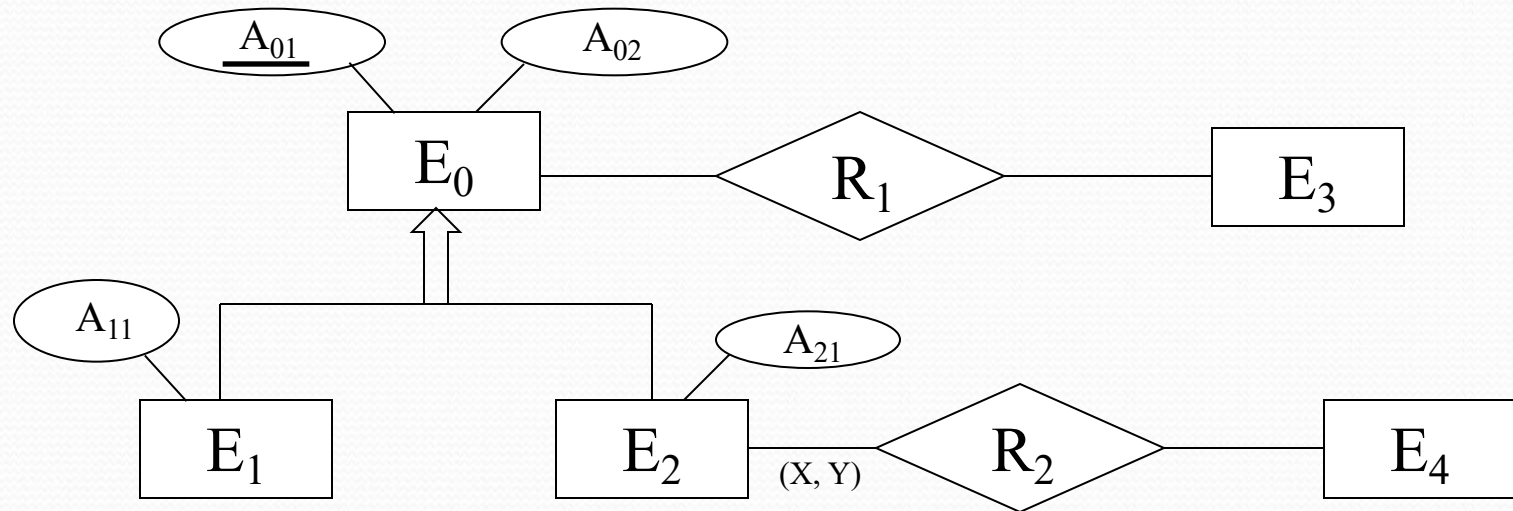
Eliminazione delle generalizzazioni

- I sistemi tradizionali per la gestione di basi di dati non consentono di rappresentare direttamente una generalizzazione.
- È necessario tradurre le generalizzazioni usando altri costrutti del modello ER (entità, relazioni).

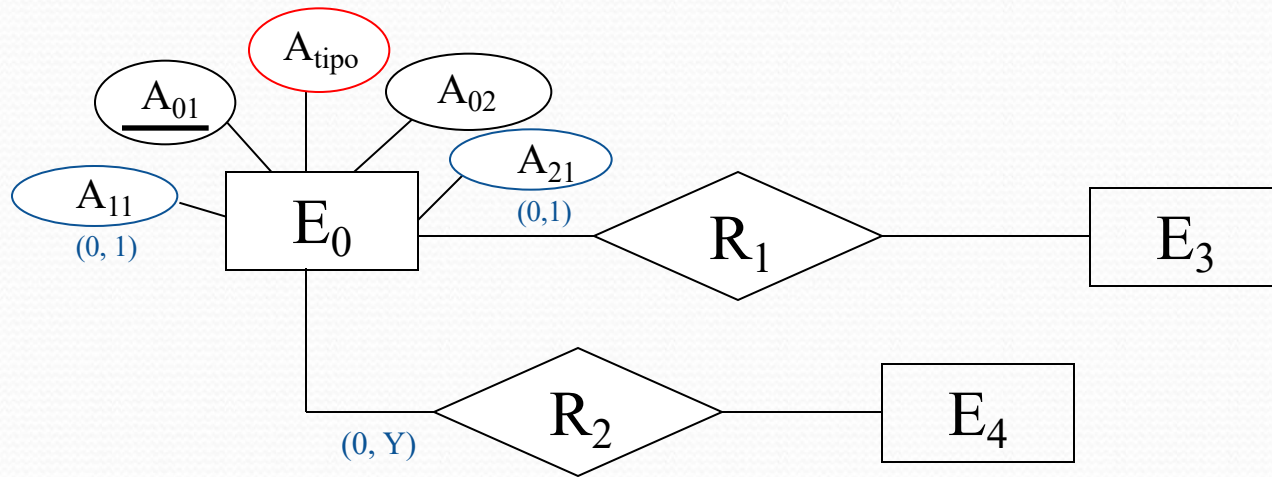
Eliminazione delle generalizzazioni

- a) Accorpamento delle figlie della generalizzazione nel padre.
- b) Accorpamento del padre della generalizzazione nelle figlie.
- c) Sostituzione della generalizzazione con associazioni.

Schema iniziale

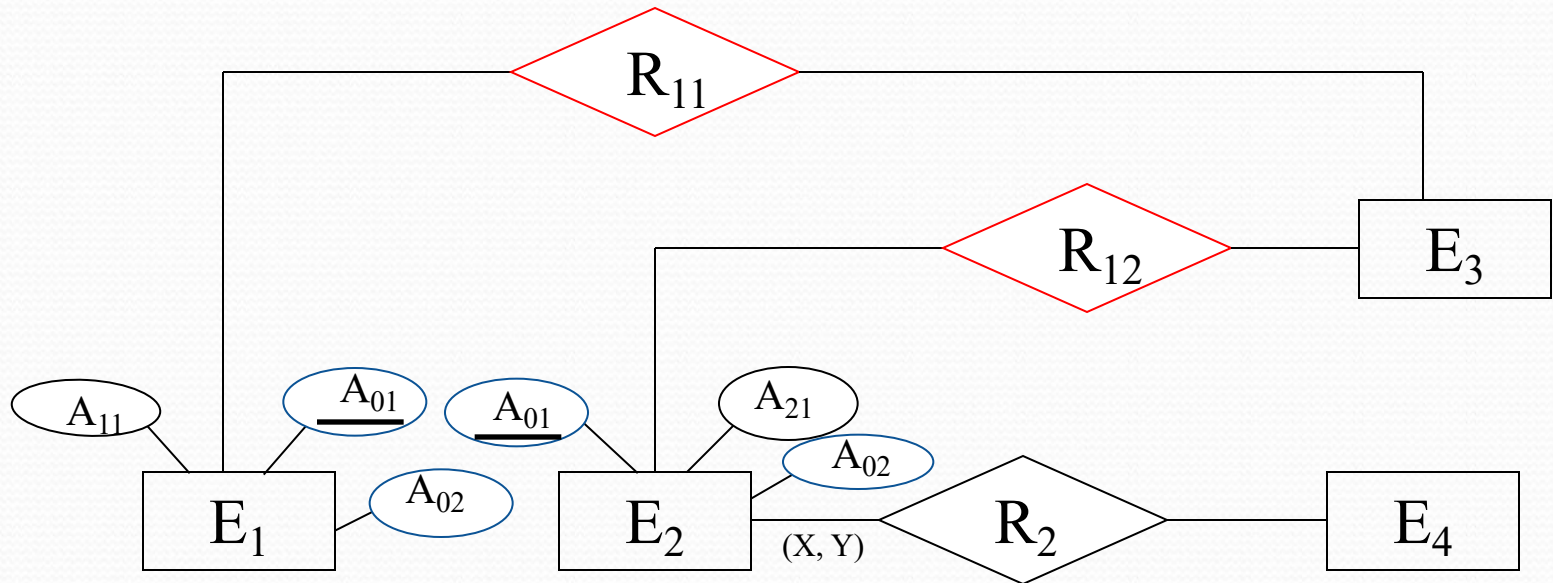


(a) Accorpamento delle figlie della generalizzazione nel padre



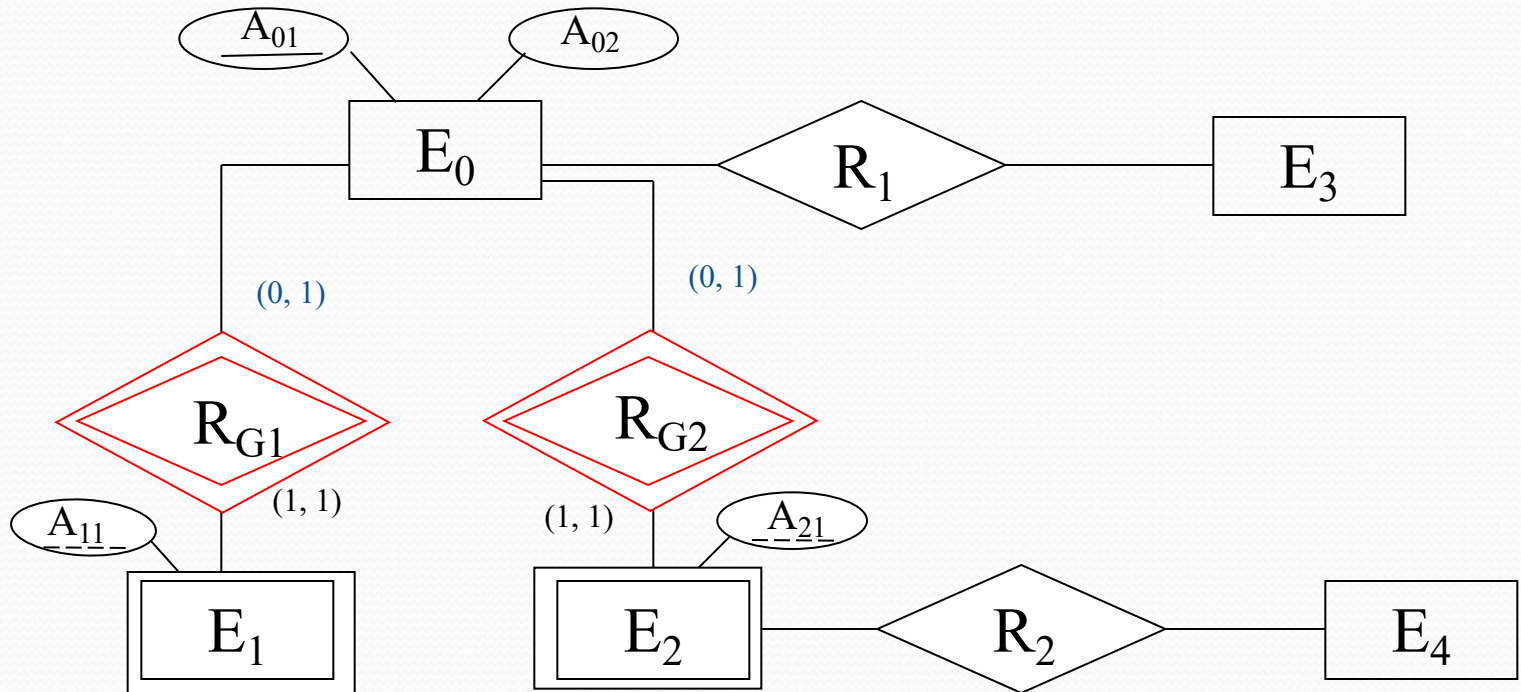
Le entità E_1 ed E_2 vengono eliminate e le loro proprietà vengono aggiunte al padre E_0 .

(b) Accorpamento del padre della generalizzazione nelle figlie



(c) Sostituzione della generalizzazione con associazioni

- Le entità E_1 e E_2 sono identificate esternamente dall'entità E_0 :



- Vincoli:** ogni occorrenza di E_0 non può partecipare contemporaneamente a R_{G1} ed R_{G2} se la generalizzazione è totale, ogni occorrenza di E_0 deve partecipare o ad un'occorrenza di R_{G1} o ad un'occorrenza di R_{G2}

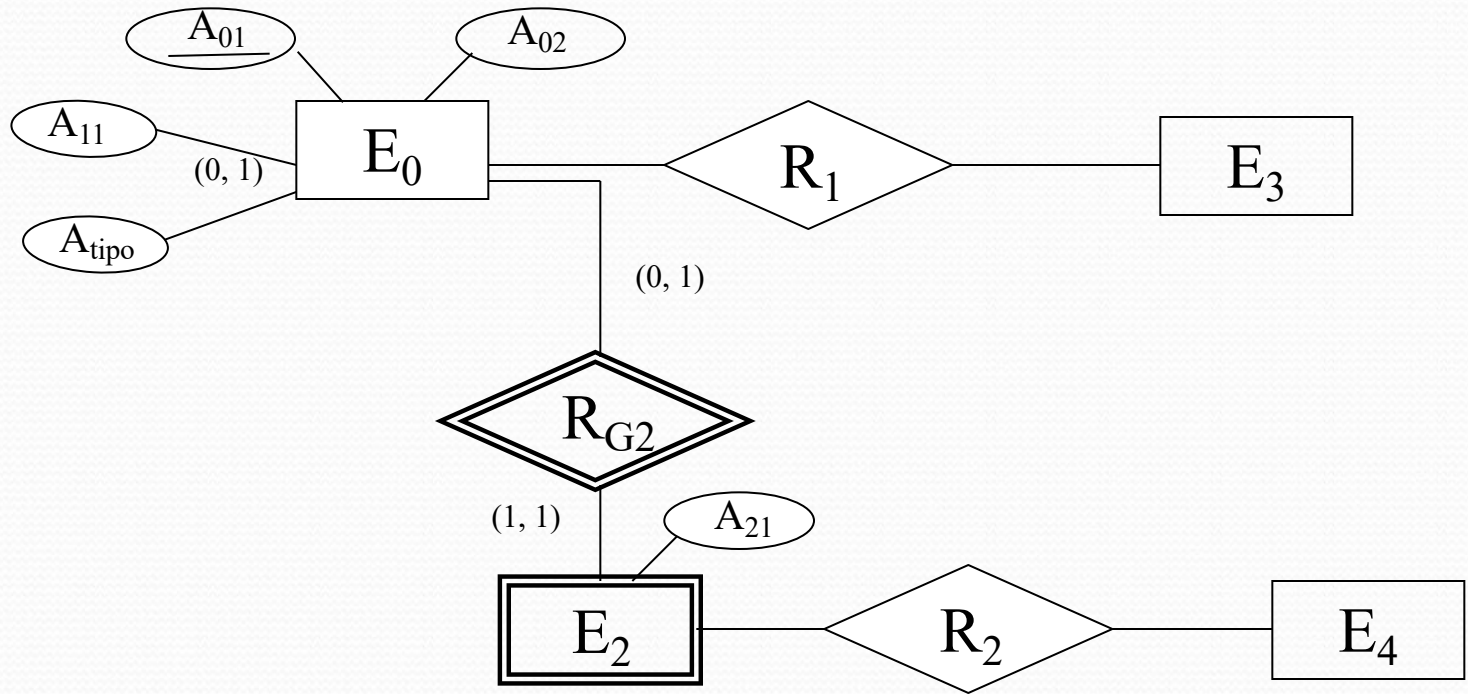
Come scegliere tra le diverse alternative

- L'alternativa **(a)** conviene quando le operazione non fanno molta distinzione tra le occorrenze e tra gli attributi di E_0 , E_1 , ed E_2 . Introduciamo valori nulli, ma abbiamo un minor numero di accessi.
- L'alternativa **(b)** è applicabile quando la generalizzazione è totale. Altrimenti le occorrenze di E_0 che non sono occorrenze né di E_1 , né di E_2 non sarebbero rappresentate.
- L'alternativa **(c)** è applicabile quando la generalizzazione non è totale, e ci sono operazioni che fanno distinzione tra entità padre ed entità figlie. Assenza di valori nulli e incremento degli accessi.

Come scegliere tra le diverse alternative (2)

- La ristrutturazione delle generalizzazioni è un tipico caso per cui il conteggio delle istanze e degli accessi non permette sempre di scegliere la migliore alternativa.
- Infatti, sembrerebbe che l'alternativa (c) non conviene quasi mai perché richiede molti più accessi per eseguire le operazioni sui dati.
- Questo tipo di ristrutturazione ha il grosso vantaggio di generare entità con pochi attributi:
 - A livello pratico questo si traduce in strutture logiche di piccole dimensioni per cui un accesso fisico permette di recuperare molti dati (tuple) in una sola volta.

Ristrutturazione alternativa



Generalizzazioni a più livelli

- Per le generalizzazioni a più livelli si può procedere analizzando una generalizzazione alla volta a partire dal fondo della gerarchia.

Partizionamento / accorpamento di concetti

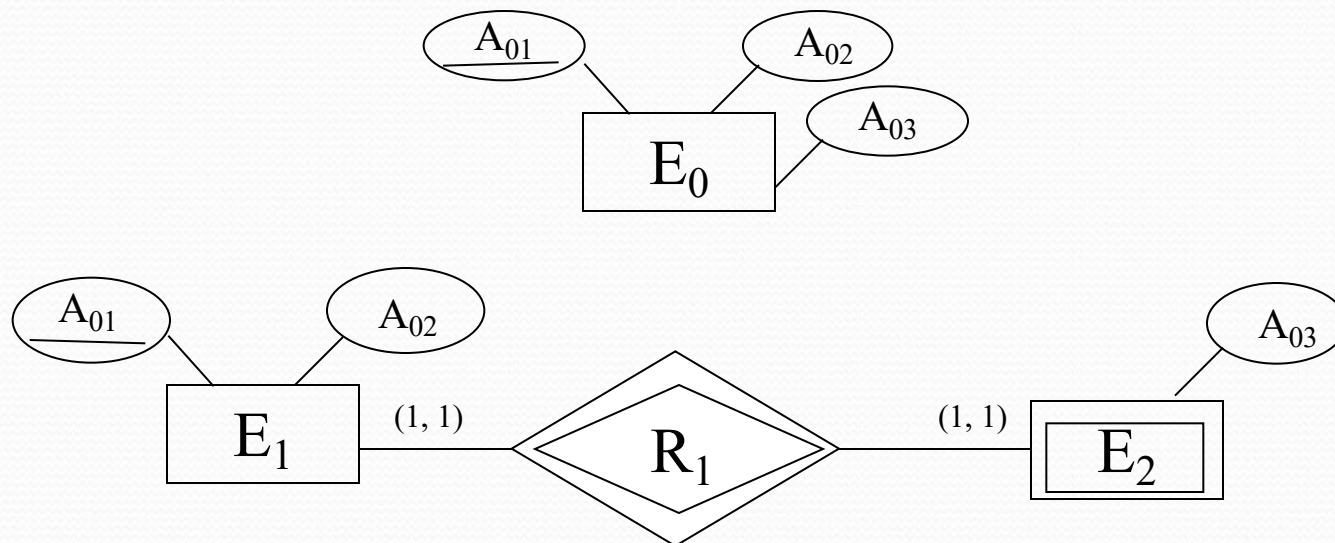
- Gli accessi si riducono separando attributi di uno stesso concetto che vengono acceduti da operazioni diverse e raggruppando attributi di concetti diversi che vengono acceduti dalle medesime operazioni.

Partizionamento / accorpamento di concetti

1. Partizionamento di entità:
 - Decomposizione verticale (*attributi*)
 - Decomposizione orizzontale (*entità*)
2. Eliminazione di attributi multivalore.
3. Accorpamento di entità.

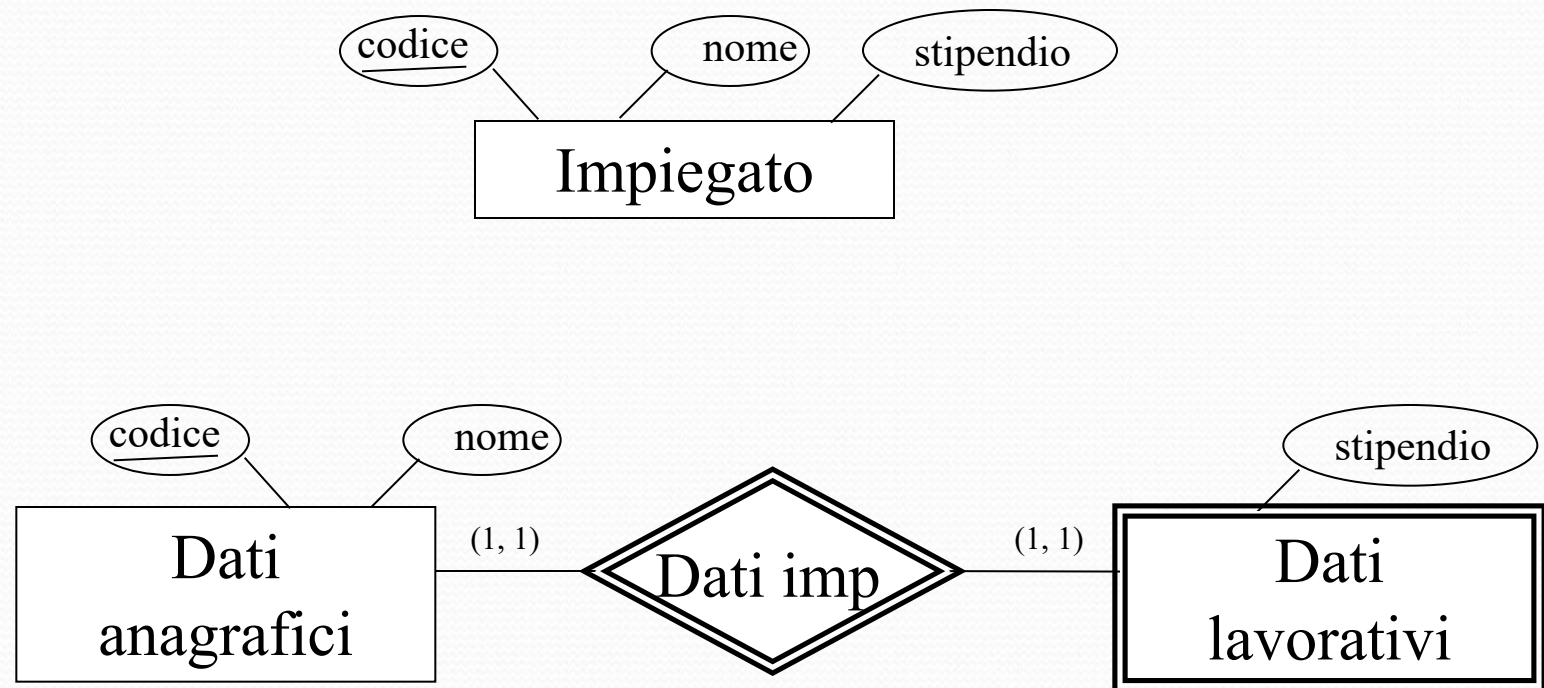
Decomposizione verticale

- Suddivisione del concetto operando sulla struttura.
- È conveniente quando lo stesso insieme di operazioni accede solo ad un sottoinsieme di attributi.



Decomposizione verticale (2) - *Esempio*

- L'entità *impiegato* può essere sostituita da due entità: *dati anagrafici* e *dati lavorativi* collegate da una associazione 1:1 :

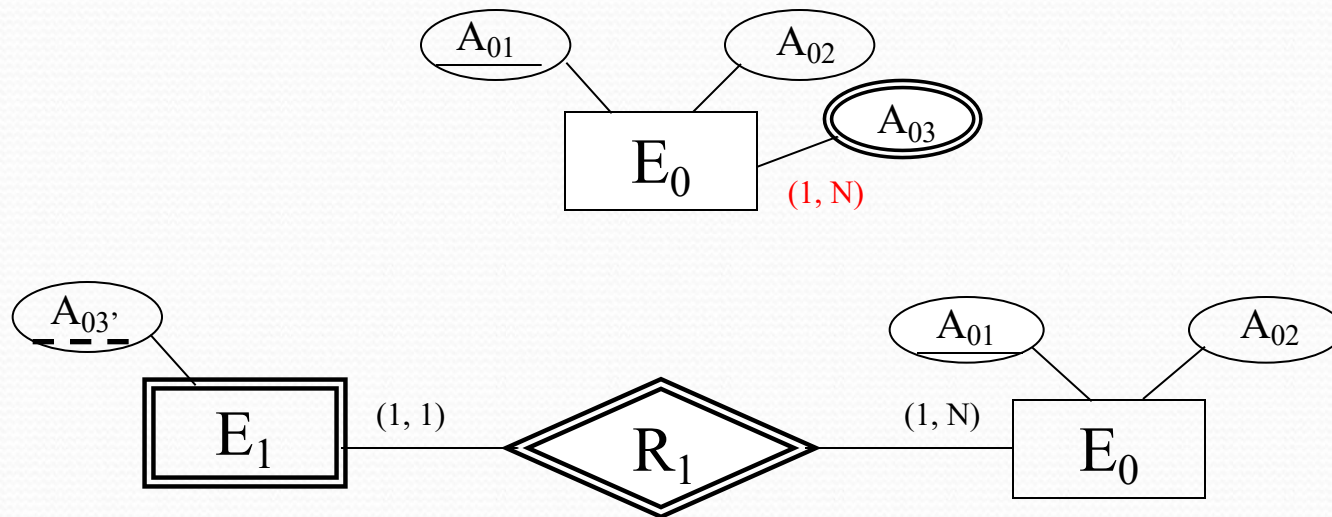


Decomposizione orizzontale

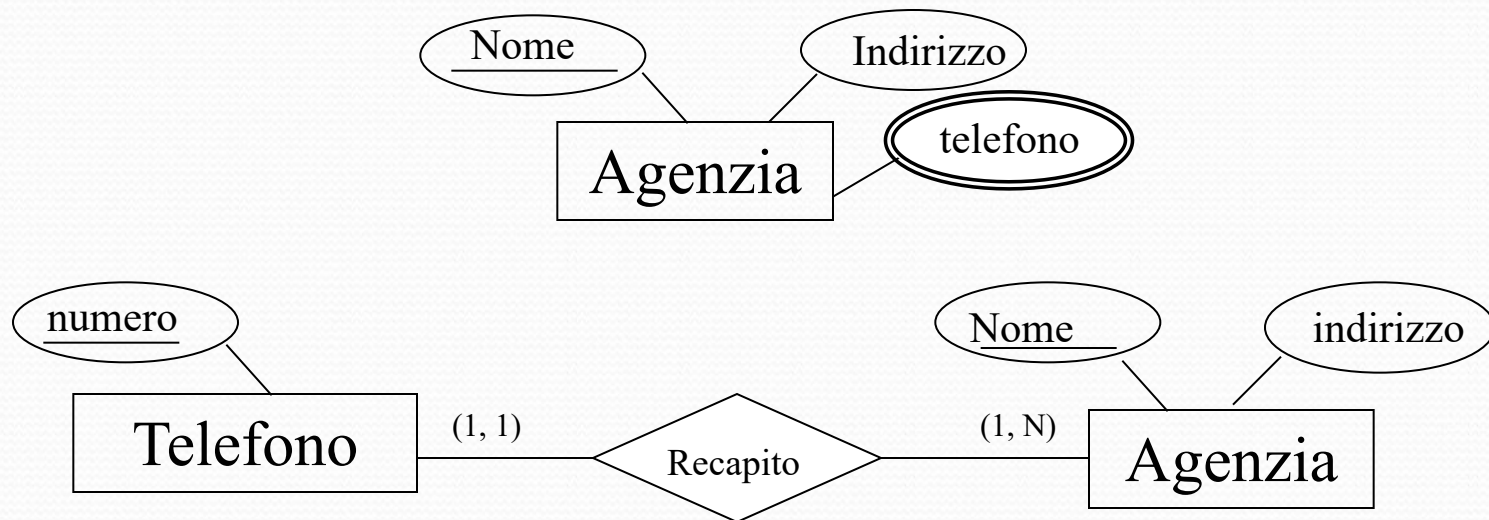
- Agisce sulle occorrenze delle entità.
- La struttura rimane invariata.
- Occorre duplicare tutte le associazioni a cui l'entità originaria partecipava.
- **Esempio:** IMPIEGATO può essere partizionato in *ANALISTA* e *VENDITORE*.
- Le entità ottenute hanno gli stessi attributi dell'entità di partenza.

Eliminazione Attributi multivalore

- Il modello relazionale non consente questa rappresentazione:



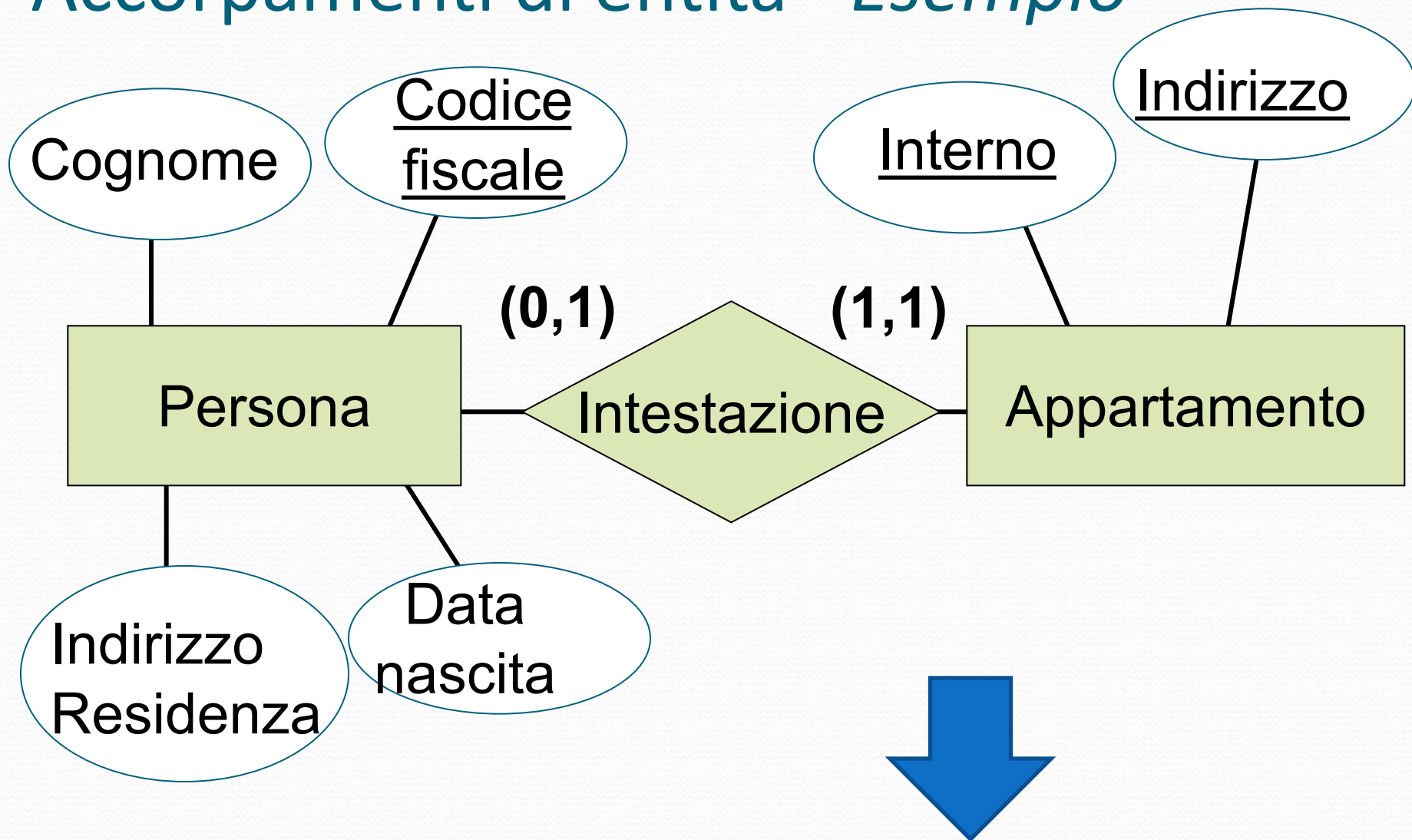
Eliminazione Attributi multivalori (2) - *Esempio*



Accorpamenti di entità

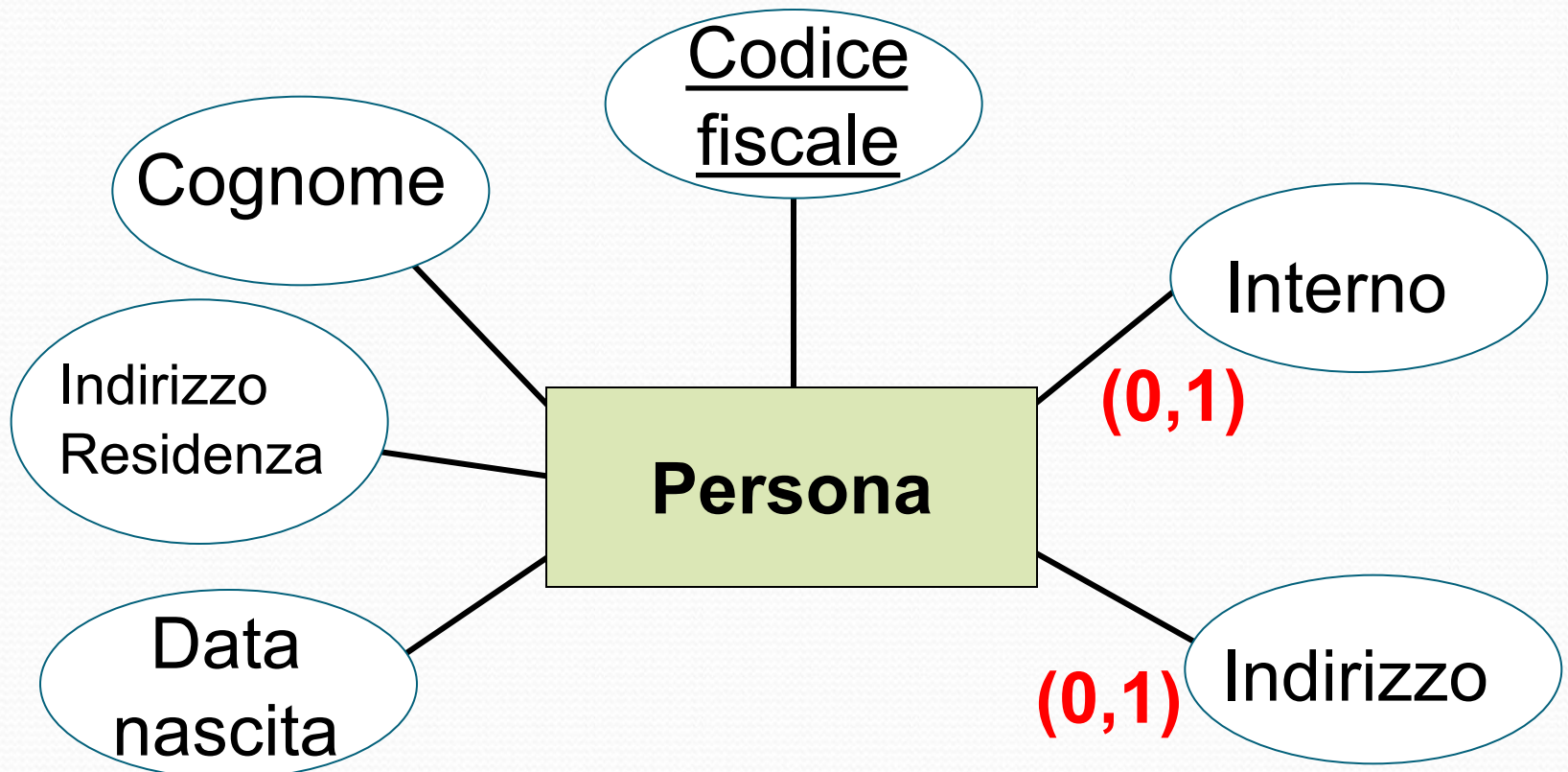
- Accorpamento dei concetti operando sulla struttura.
- È conveniente quando le operazioni accedono a dati presenti su entrambe le entità.
- Si effettua generalmente su associazioni di tipo 1:1.
- Presenza di valori nulli.

Accorpamenti di entità - *Esempio*

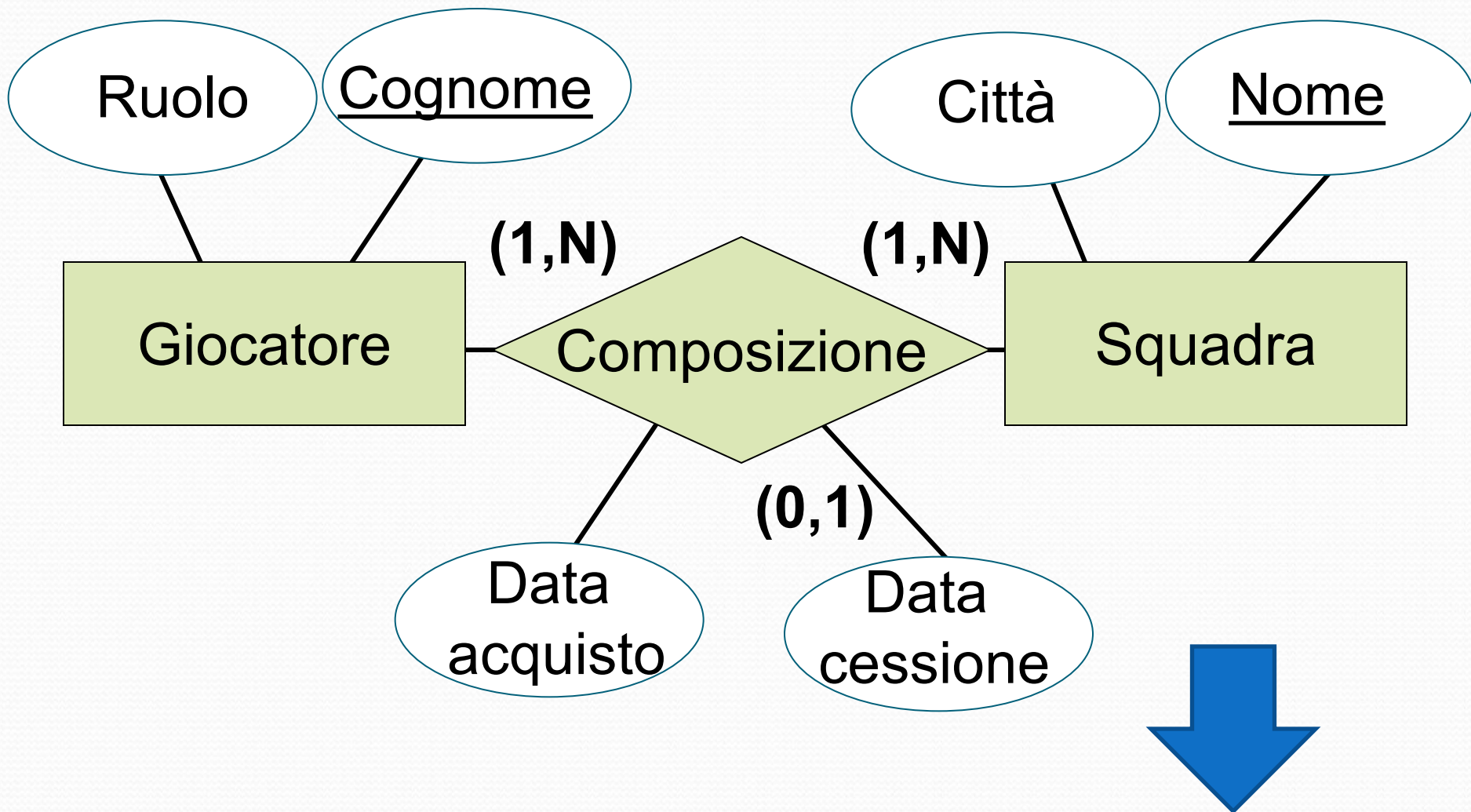


Accorpamenti di entità – *Esempio (2)*

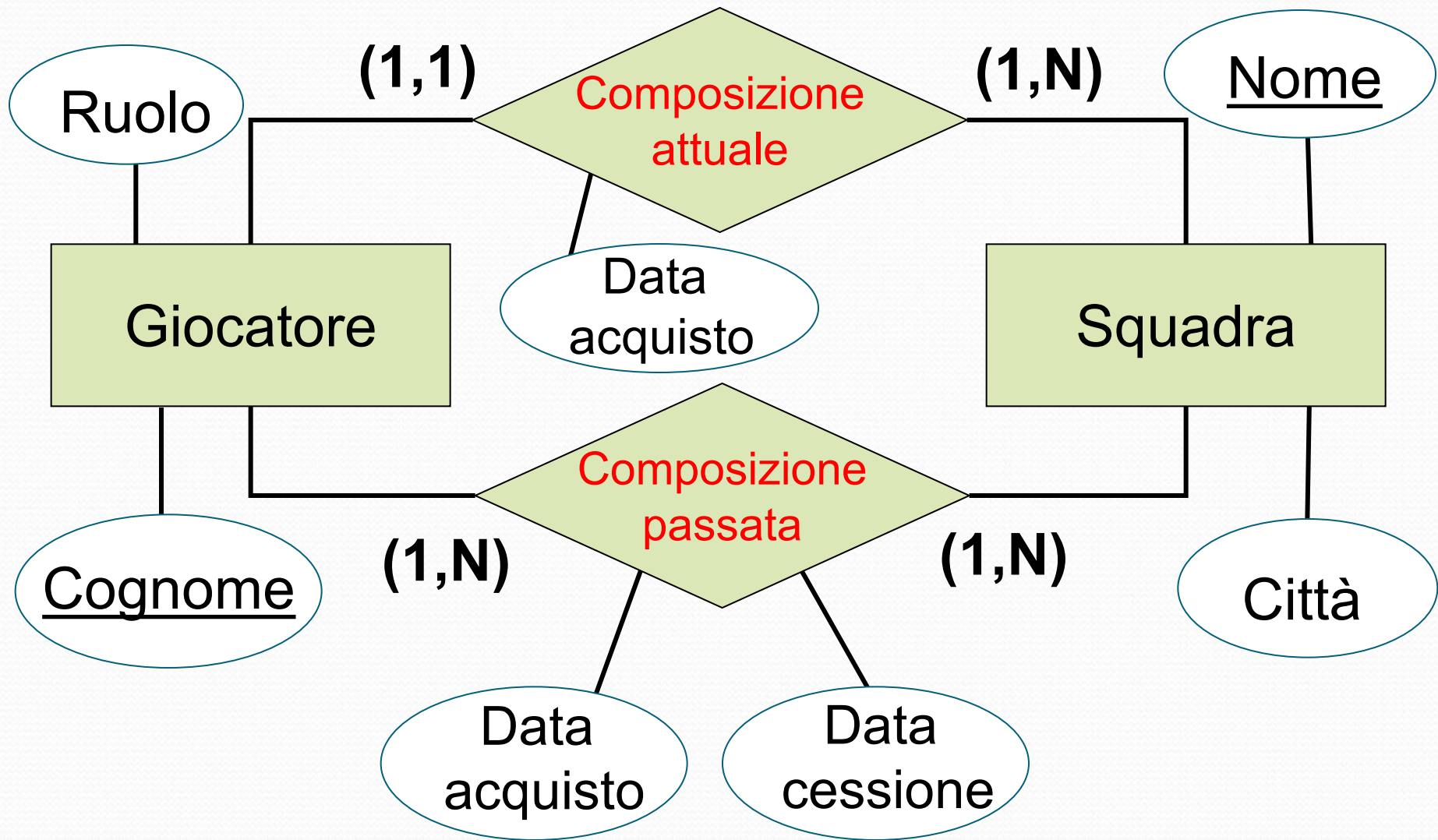
- Le operazioni più frequenti su persona richiedono sempre dati relativi all'appartamento che occupa:



Decomposizione di Associazioni - *Esempio*



Decomposizione di Associazioni – *Esempio (2)*



Identificatori

- Scelta della chiave primaria per la costruzione degli indici per il recupero efficiente dei dati.
- Criteri generali:
 - Escludere attributi con valori nulli.
 - Numero minimo di attributi (dimensioni ridotte degli indici).
 - Identificatore coinvolto in molte operazioni.
 - Velocità di accesso all'indice.
 - Tempo minimo per confrontare i valori.
- Se nessuno degli identificatori candidati soddisfa tali criteri si introduce un ulteriore attributo all'entità:
 - Questo attributo conterrà valori speciali (**codici**) generati appositamente per identificare le occorrenze delle entità.

Identificatori: Esempio

- Nel caso in cui l'entità **Studente** abbia **due** possibili chiavi:
 1. Matricola (10 caratteri, numerici).
 2. Codice fiscale (16 caratteri, alfanumerici).
- *Perché è opportuno selezionare l'attributo **Matricola** come identificatore primario?*
 - *La matricola richiede un tempo minore per confrontare due valori tra loro: la velocità di accesso all'indice è ridotta*