



# PROGETTAZIONE LOGICA

---

*Patrizio Dazzi*  
*a.a. 2017 - 2018*

# COMUNICAZIONI

---

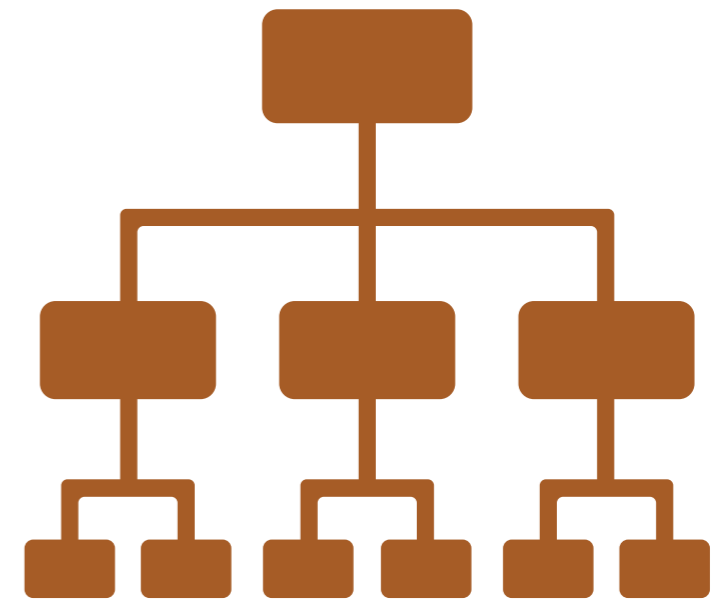
- Lezione odierna
  - Progettazione logica
  
- Nessun ricevimento questo mercoledì



# PICCOLO RIASSUNTO DELLA PUNTATA PRECEDENTE

---

- Concetti di progettazione
- Schemi entità - relazione
  - entità, relazioni, attributi, cardinalità, generalizzazione
- Strategie di Progettazione
  - Raccolta e analisi requisiti



**DALLA  
PROGETTAZIONE  
CONCETTUALE  
A QUELLA  
LOGICA**

---



# TRADUZIONE

---

*Tradurre. Traducere, cioè trans (al di là) e ducere (condurre):  
condurre, trasferire dei concetti, delle parole da una lingua all'altra.*

- Come ben sapete:
  - Essere fedeli al testo originale non è semplice e chi traduce sa che in molti casi fedeltà vuol dire semplicemente "l'equivalente".
  - Una traduzione deve riprodurre lo stile originale o semplicemente deve riprodurre lo stile attuale nella lingua in cui si traduce?
  - E l'ironia? Le metafore? Le circostanze politiche e storiche, le tensioni, alle quali si allude o che sono sottintese?
- Nel nostro caso, più che di una traduzione si tratta di una (ri-)progettazione

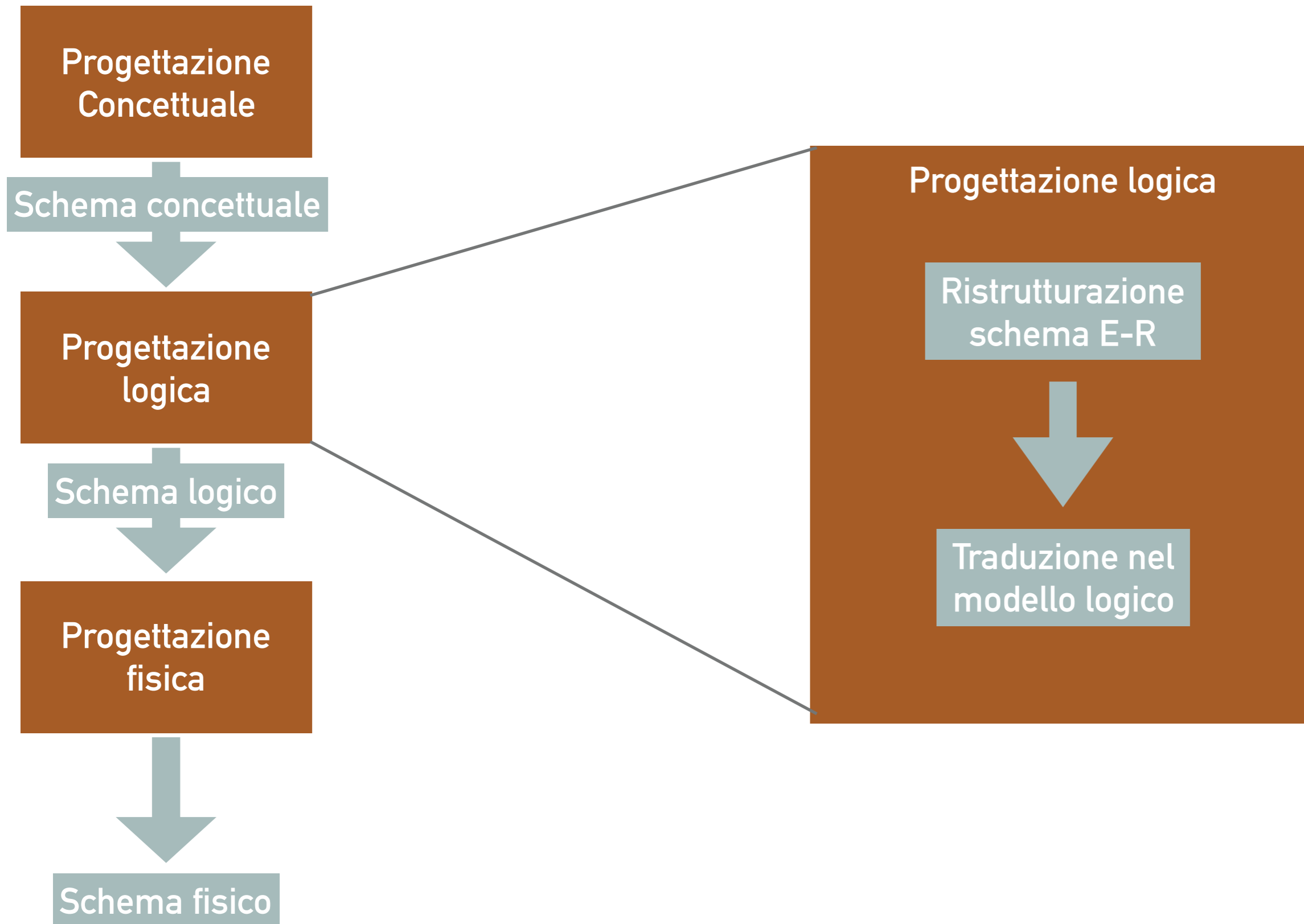
# PROCESSO DI TRADUZIONE

---

- obiettivo della progettazione logica e' la costruzione di uno schema logico in un determinato modello (ad esempio relazionale)
- volto a rappresentare i lo schema concettuale in modo:
  - corretto
  - efficiente
- due fasi:
  - ristrutturazione dello schema E-R
  - traduzione verso il modello logico (include possibili ottimizzazioni, quali la normalizzazione, che noi **NON** faremo)

# FASI DI TRADUZIONE

---





**RISTRUTTURAZIONE  
DELLO SCHEMA  
E-R:  
ANALISI  
QUANTITATIVA**

---





# INPUT ED OUTPUT DI QUESTA FASE

---

- Input:

- Schema Concettuale E-R iniziale, Carico Applicativo previsto (in termini di dimensione dei dati e caratteristica delle operazioni)

- Output :

- Schema E-R ristrutturato che rappresenta i dati e tiene conto degli aspetti realizzativi

# ANALISI DELLE PRESTAZIONI SU SCHEMI E-R

---

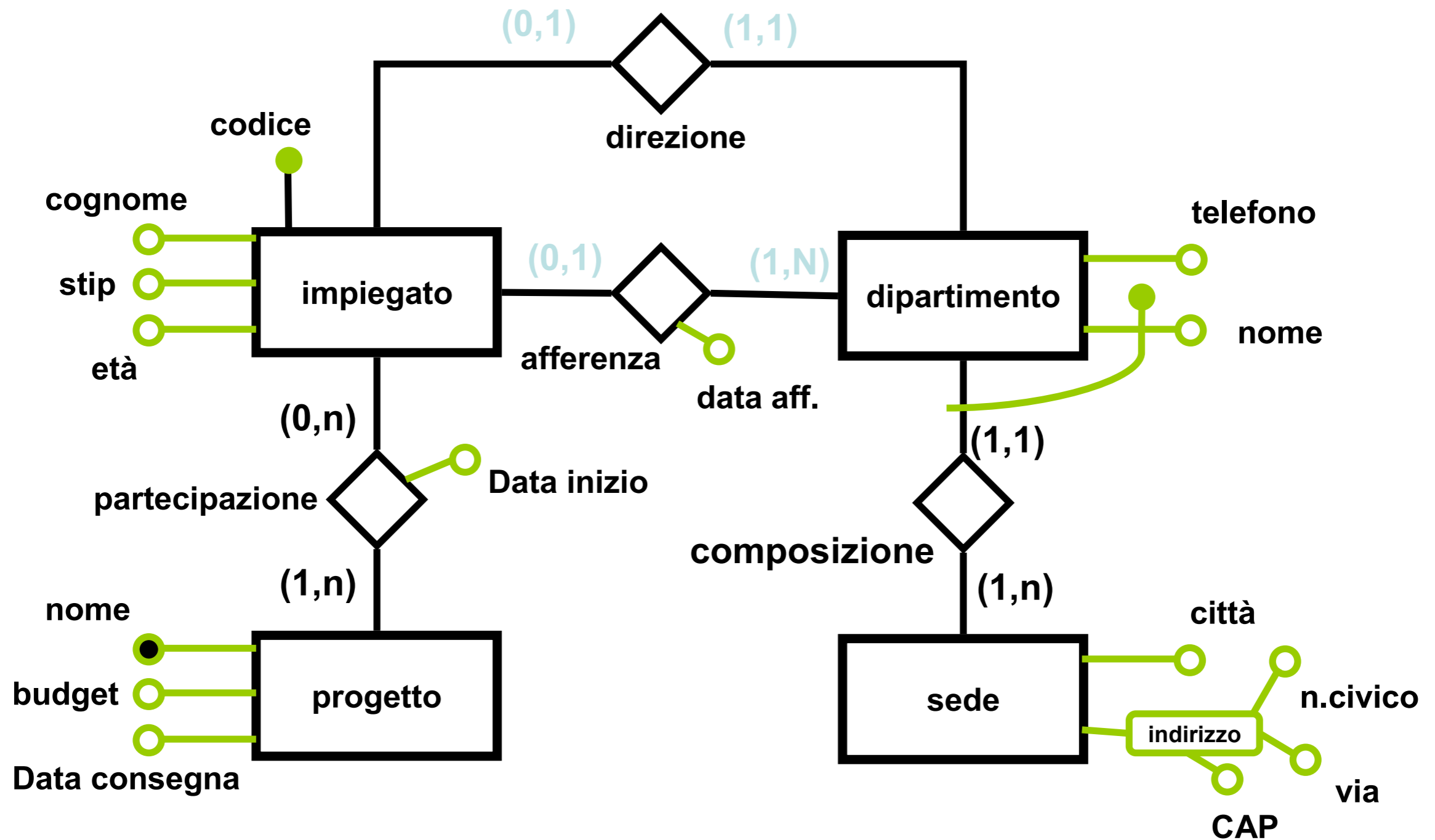
- Due indici di prestazione per la valutazione di schemi E-R
  - **Costo di un'operazione:** in termini di numero di occorrenze di entità ed associazioni da visitare per rispondere ad una data operazione sulla base di dati
  - **Occupazione di memoria:** in termini di spazio di memoria necessario per memorizzare i dati del sistema.

# VOLUMI E CARATTERISTICHE DEI DATI

---

- Per studiare questi due parametri abbiamo bisogno di conoscere:
  - Volume dei dati:
    - numero (medio) di occorrenze di ogni entità ed associazione
    - dimensioni di ciascun attributo
  - Caratteristiche delle operazioni:
    - tipo di operazione (interattiva o batch)
    - frequenza (esecuzioni/tempo)
    - dati coinvolti (entità e/o associazioni)

# ESEMPIO: DITTA CON SEDI IN CITTÀ DIVERSE



# OPERAZIONI DELL'ESEMPIO

---

- Operazione 1: assegna un impiegato ad un progetto
- Operazione 2: trova i dati di un impiegato, del dipartimento nel quale lavora e dei progetti in cui e' coinvolto
- Operazione 3: trova i dati di tutti gli impiegati di un certo dipartimento
- Operazione 4: per ogni sede, trova i dipartimenti con il cognome del direttore e l'elenco degli impiegati.

# OBIETTIVO: DERIVARE VOLUMI, OPERAZIONI E ACCESSI

---

- Volumi: quantità attese per ogni entità e relazione
- Operazioni: frequenze stimate di accesso alle entità e alle relazioni
- Accessi: rapporto dimensionale relativo tra entità e relazioni riguardo agli accessi effettuati per lo svolgimento di una data operazione



# VOLUMI E OPERAZIONI

## 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

## OPERAZIONI

concetto	tipo	volume
op1	I	50/giorno
op2	I	100/giorno
op3	I	10/giorno
op4	B	2/settimana

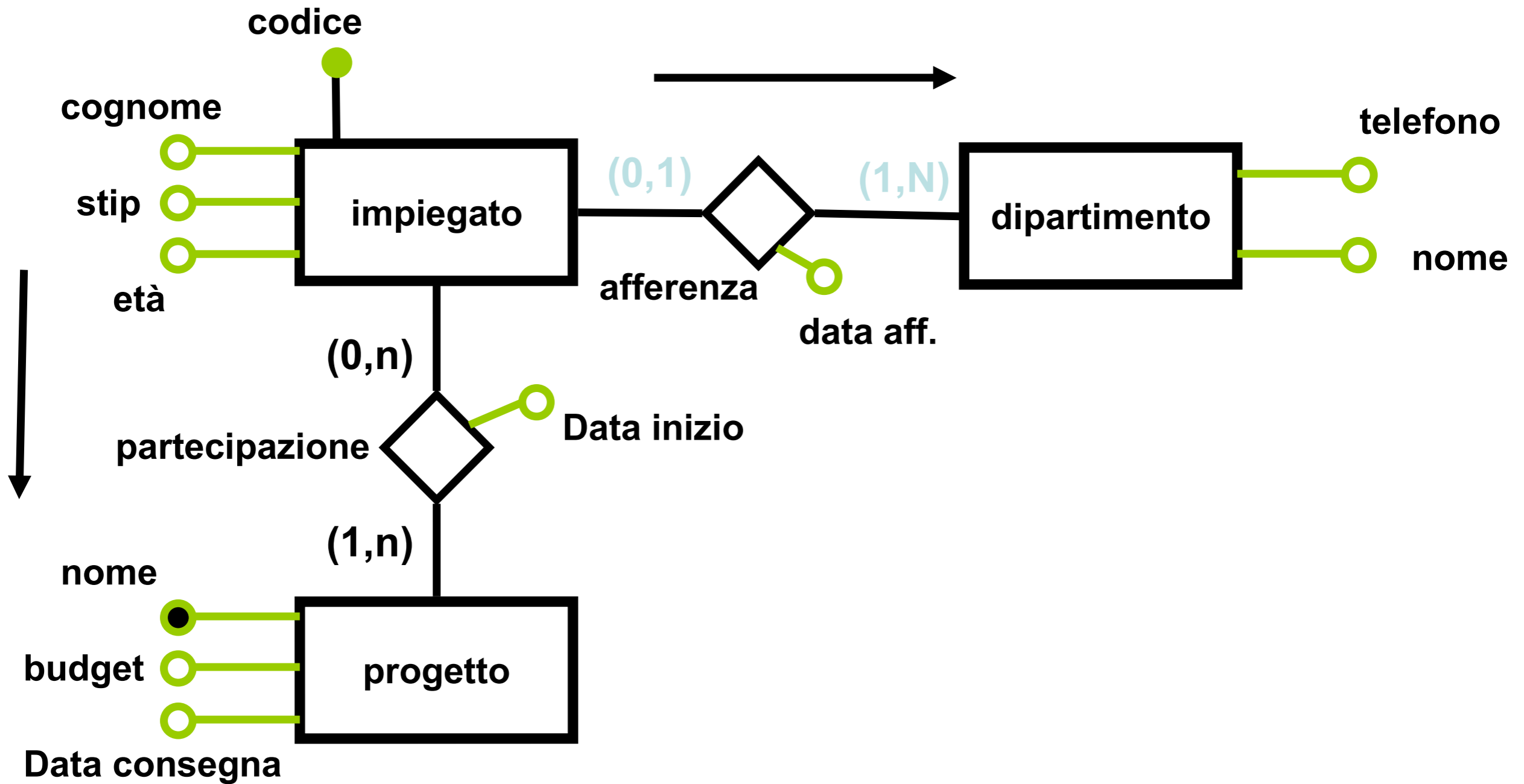
# STIMA DEI COSTI

---

- Avendo a disposizione questi dati è possibile:
  - stimare i costi di ogni operazione
  - contando il numero di accessi alle occorrenze di entità e relazioni necessario per eseguire l'operazione.
- Prendiamo per esempio Operazione 2:
  - trova i dati di un impiegato, del dipartimento nel quale lavora e dei progetti in cui e' coinvolto
  - facciamo poi riferimento allo schema di operazione

# ESEMPIO: OPERAZIONE 2

---



## STIMA DEL COSTO DELL'OPERAZIONE 2 – COSTRUZIONE TAVOLA DEGLI ACCESSI

---

- Dobbiamo accedere ad:
  - un'occorrenza di **Impiegato** e di **Afferenza** e quindi di **Dipartimento**;
  - Successivamente, per avere i dati dei progetti a cui lavora, dobbiamo accedere (in media) a tre occorrenze di **Partecipazione** e quindi a tre entità **Progetto** (6000/2000).
  - Tutto viene riassunto nella tavola degli accessi

# TAVOLA DEGLI ACCESSI

---

## ACCESSI

concetto	costrutto	accesso	tipo
impiegato	entita	1	L
afferenza	relazione	1	L
dipartimento	entita	1	L
partecipazione	relazione	3	L
progetto	entita	3	L

L lettura, S scrittura. In genere la scrittura e' piu' onorosa che la lettura

**RISTRUTTURAZIONE  
DELLO SCHEMA  
E-R:  
OPERAZIONI  
DISPONIBILI**

---





# HO LE INFORMAZIONI QUANTITATIVE, E ADESSO ?

---

- Tabella dei volumi, delle operazioni e degli accessi
  - raccolte o derivate
- Utilizzabili per guidare il processo di ristrutturazione dello schema E-R
- Vediamo quali sono le operazioni a nostra disposizione
  - ridondanze, generalizzazioni, partizionamento, aggregazione, identificazione delle chiavi

# RISTRUTTURAZIONE SCHEMA E-R: OPERAZIONI A NOSTRA DISPOSIZIONE

---

- **Analisi delle Ridondanze:** si decide se eliminare o no eventuali ridondanze.
- **Eliminazione delle Generalizzazioni:** tutte le generalizzazioni vengono analizzate e sostituite da altro.
- **Partizionamento/Accorpamento** di entita' ed associazioni: si decide se partizionare concetti in piu' parti o viceversa accorpare.
- **Scelta degli identificatori primari:** si sceglie un identificatore per quelle entita' che ne hanno piu' di uno

# ANALISI RIDONDANZE

---

- Attributi derivabili da altri attributi della stessa entità  
(importo fattura : importo lordo)
- Attributi derivabili da attributi di altre entità (o associazioni)  
(Importo Acquisto: Importo totale da Prezzo )
- Attributi derivabili da operazioni di conteggio (Abitanti Città:  
Numero abitanti contando il numero di Residenza )
- Associazioni derivabili dalla composizione di altre  
associazioni in presenza di cicli. (Docenza da Frequenza ed  
Insegnamento). Tuttavia i cicli non necessariamente generano  
ridondanze.

# DATO DERIVABILE

---

- Vantaggi: riduce gli accessi per calcolare il dato derivato.
- Svantaggi: occupazione di memoria e necessita' di effettuare operazioni aggiuntive per mantenere il dato aggiornato.
- Decisione: mantenere o eliminare?
  - È necessario confrontare i costi di esecuzione delle operazioni sull'oggetto

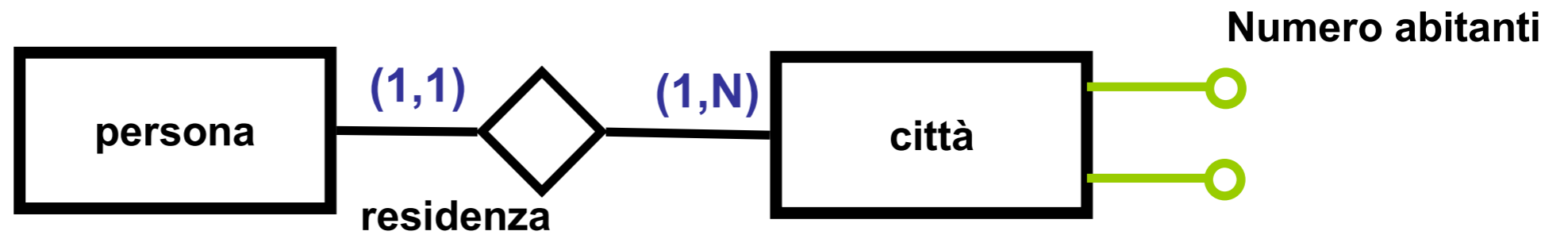
# ESEMPIO

---

- Consideriamo l'esempio Città-Persona per l'anagrafica di una regione.
  - Operazione 1: memorizza una persona nuova con la relativa città.
  - Operazione 2: stampa tutti i dati di una città (incluso il numero di abitanti).
- Valutiamo gli indici di prestazione per l'attributo Numero Abitanti

# ESEMPIO

---



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

Operazione	Tipo	Frequenza
Città	E	200
Persona	E	1000000
Residenza	R	1000000



# SOLUZIONE CON RIDONDANZA: ANALISI

---

- Assumendo che il numero di abitanti richieda 4 byte (dimensione tipica di un intero: 32bit) il dato richiede  $4 \times 200 = 800$  byte.
- Operazione 1 richiede un accesso in scrittura a Persona uno in scrittura a Residenza ed uno in lettura ed uno in scrittura (per incrementare il numero di abitanti) a Città ripetuto 500 volte si hanno 1500 accessi in scrittura e 500 in lettura.
- Operazione 2 richiede un solo accesso in lettura a Città 2 volte al giorno.
- Supponendo che la scrittura ha un costo doppio rispetto ad una lettura si hanno 3500 accessi al giorno in presenza della ridondanza.

# SOLUZIONE SENZA RIDONDANZA: ANALISI

---

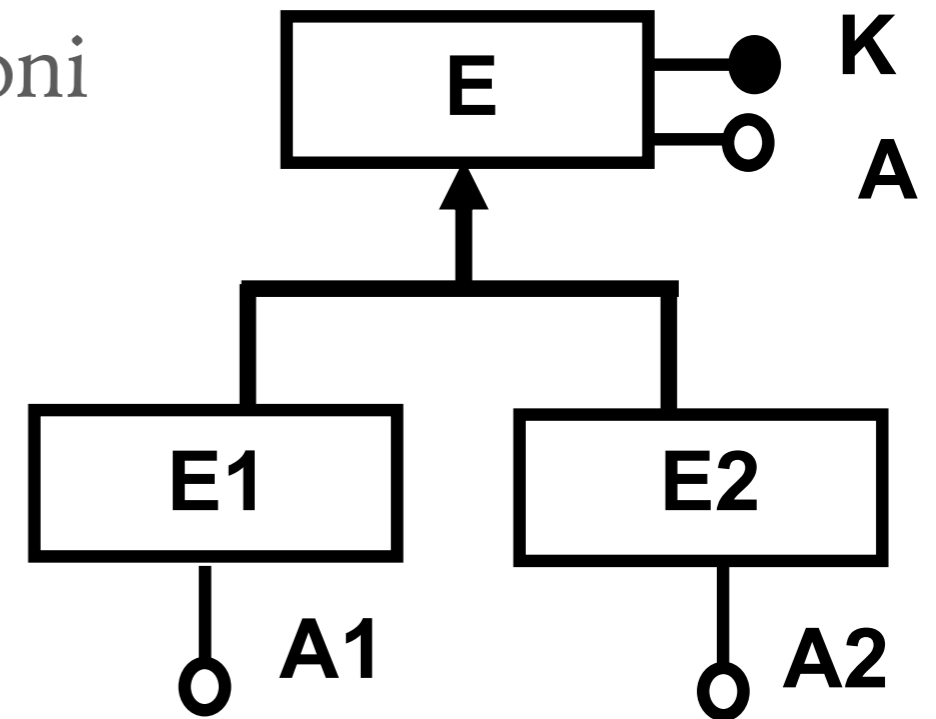
- Per l'operazione 1, un accesso in scrittura a Persona ed uno in scrittura a Residenza per un totale di 1000 accessi in scrittura al giorno.
- Per l'operazione 2 abbiamo bisogno di un acceso in lettura a Città (possiamo trascurare) e 5000 accessi in lettura a Residenza in media (persone/città) per un totale di 10.000 accessi in lettura al giorno.
- Il totale e' di 12000 accessi in lettura al giorno. Quindi 8500 in più rispetto al caso di ridondanza contro meno di un solo Kilobyte di memoria in più.
- D'altra parte se l'operazione 2 fosse stata richiesta solo 1 volta ogni 4 settimane avremmo avuto  $3500 * 24 = 84000$  accessi ogni 4 settimane con ridondanza contro 58000 in assenza.

# GERARCHIE E MODELLO RELAZIONALE

---

➤ il modello relazionale non rappresenta le gerarchie, le sostituisce mediante entità e associazioni:

- Mantenendo le entità con associazioni
- Collasso verso l'alto
- Collasso verso il basso

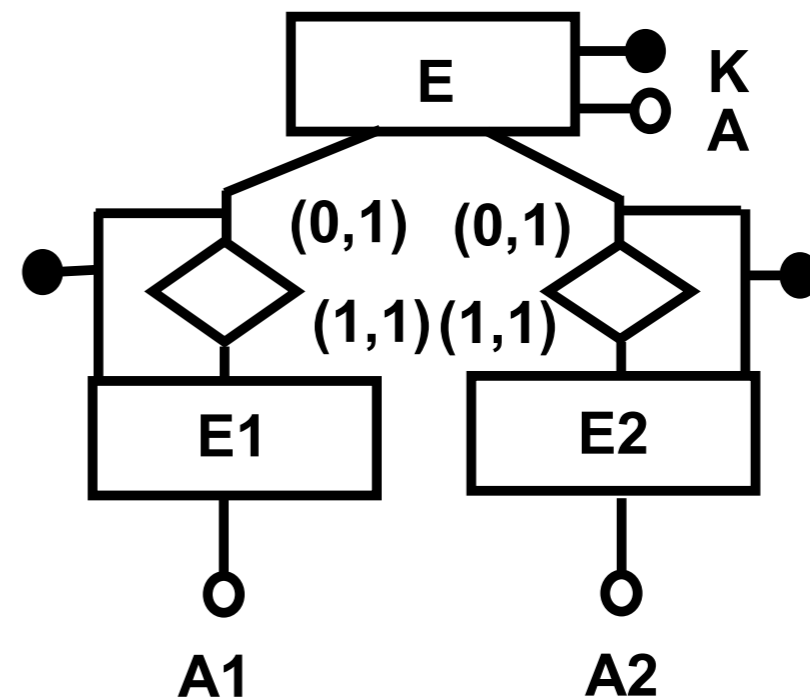


➤ l'applicabilità e la convenienza delle soluzioni dipendono dalle proprietà di copertura e dalle operazioni previste

# MANTENIMENTO DELLE ENTITÀ

---

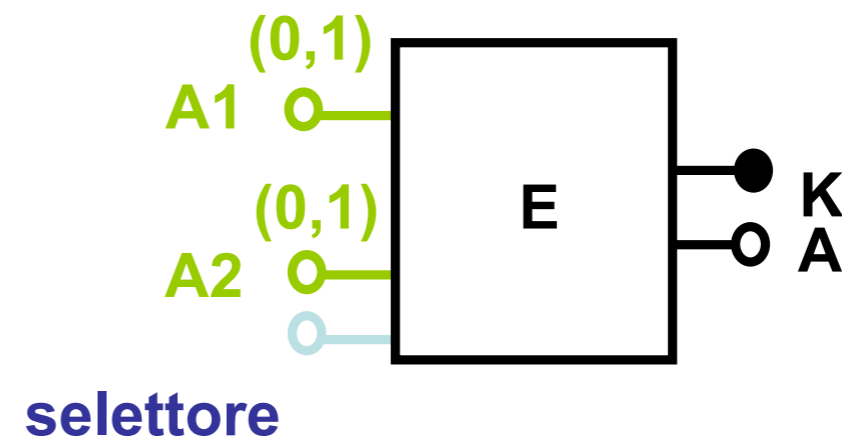
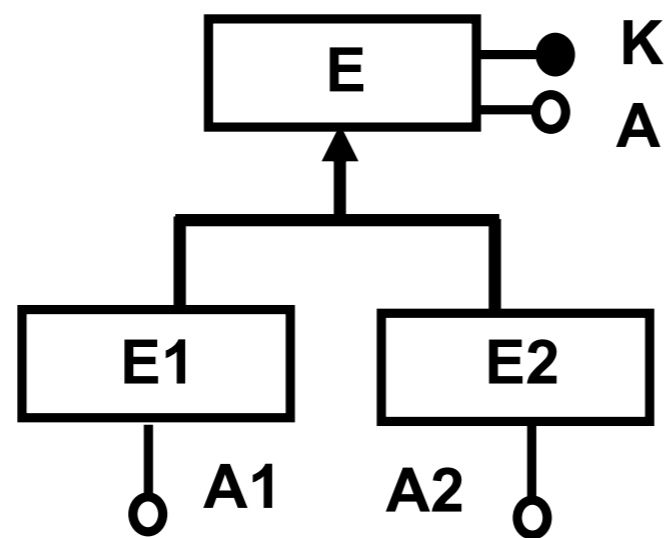
- tutte le entità vengono mantenute
- le entità figlie sono in associazione con l'entità padre
- le entità figlie sono identificate esternamente tramite l'associazione
- questa soluzione è sempre possibile, indipendentemente dalla copertura



# COLLASSO VERSO L'ALTO

---

- Il collasso verso l'alto riunisce tutte le entità figlie nell'entità padre
- selettore è un attributo che specifica se una istanza di E appartiene a una delle sotto-entità

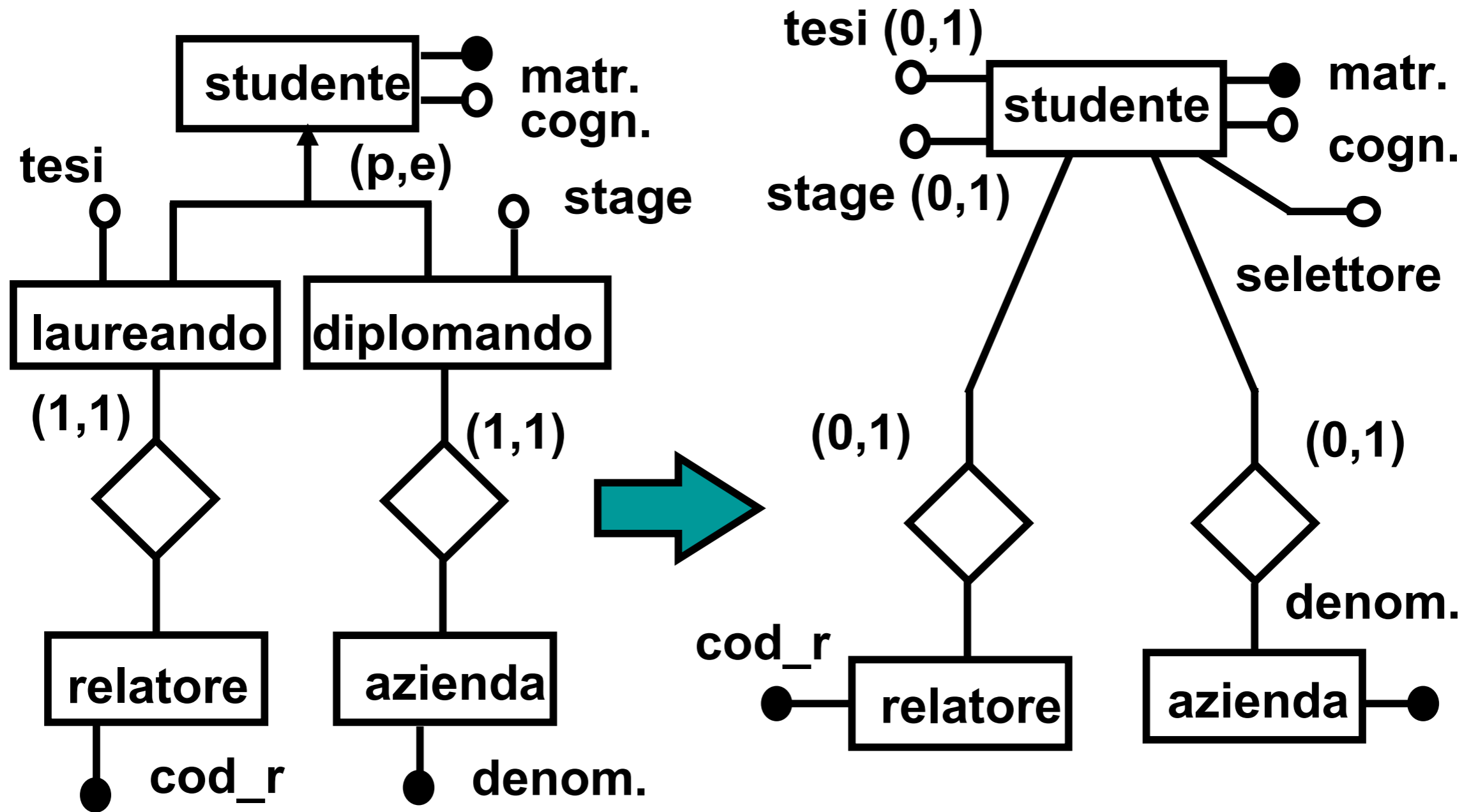


## COLLASSO VERSO L'ALTO (2)

---

- Il collasso verso l'alto favorisce operazioni che consultano insieme gli attributi dell'entità padre e quelli di una entità figlia:
  - in questo caso si accede a una sola entità, anziché a due attraverso una associazione
  - gli attributi obbligatori per le entità figlie divengono opzionali per il padre
  - si avrà una certa percentuale di valori nulli

# COLLASSO VERSO L'ALTO: ESEMPIO



# COLLASSO VERSO IL BASSO

---

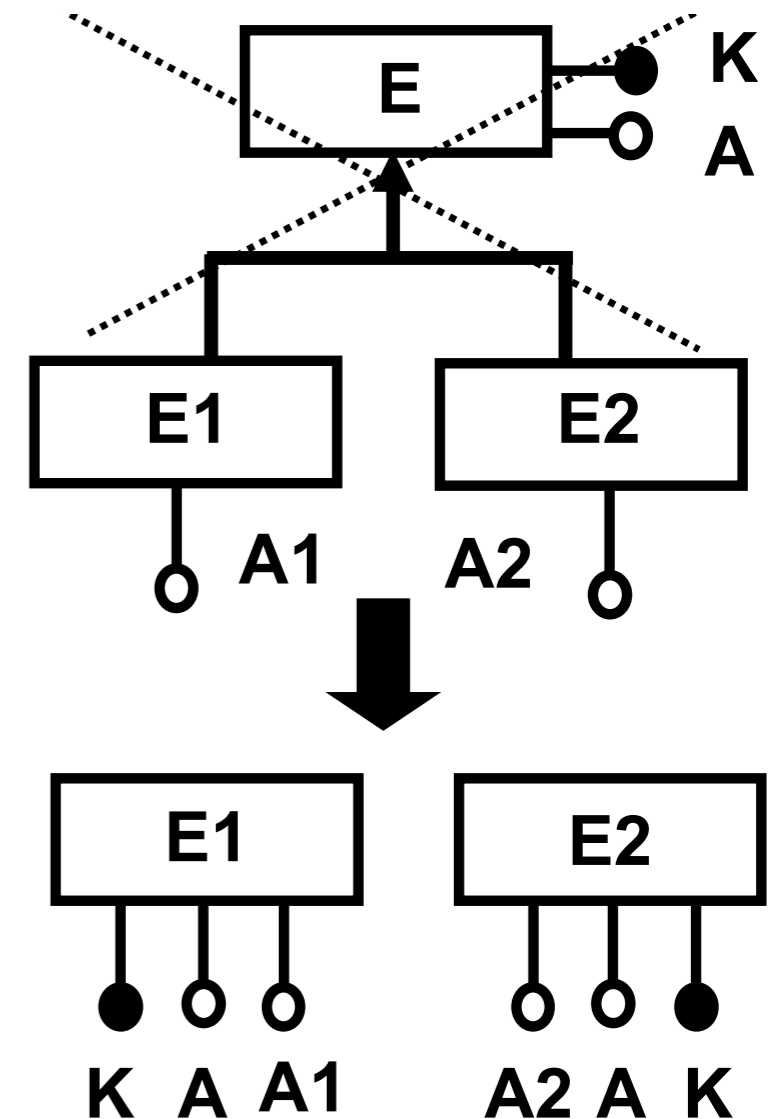
- si elimina l'entità padre trasferendone gli attributi su tutte le entità figlie
  - una associazione del padre è replicata, tante volte quante sono le entità figlie
  - la soluzione è interessante in presenza di molti attributi di specializzazione (con il collasso verso l'alto si avrebbe un eccesso di valori nulli)
  - favorisce le operazioni in cui si accede separatamente alle entità figlie



# COLLASSO VERSO IL BASSO

---

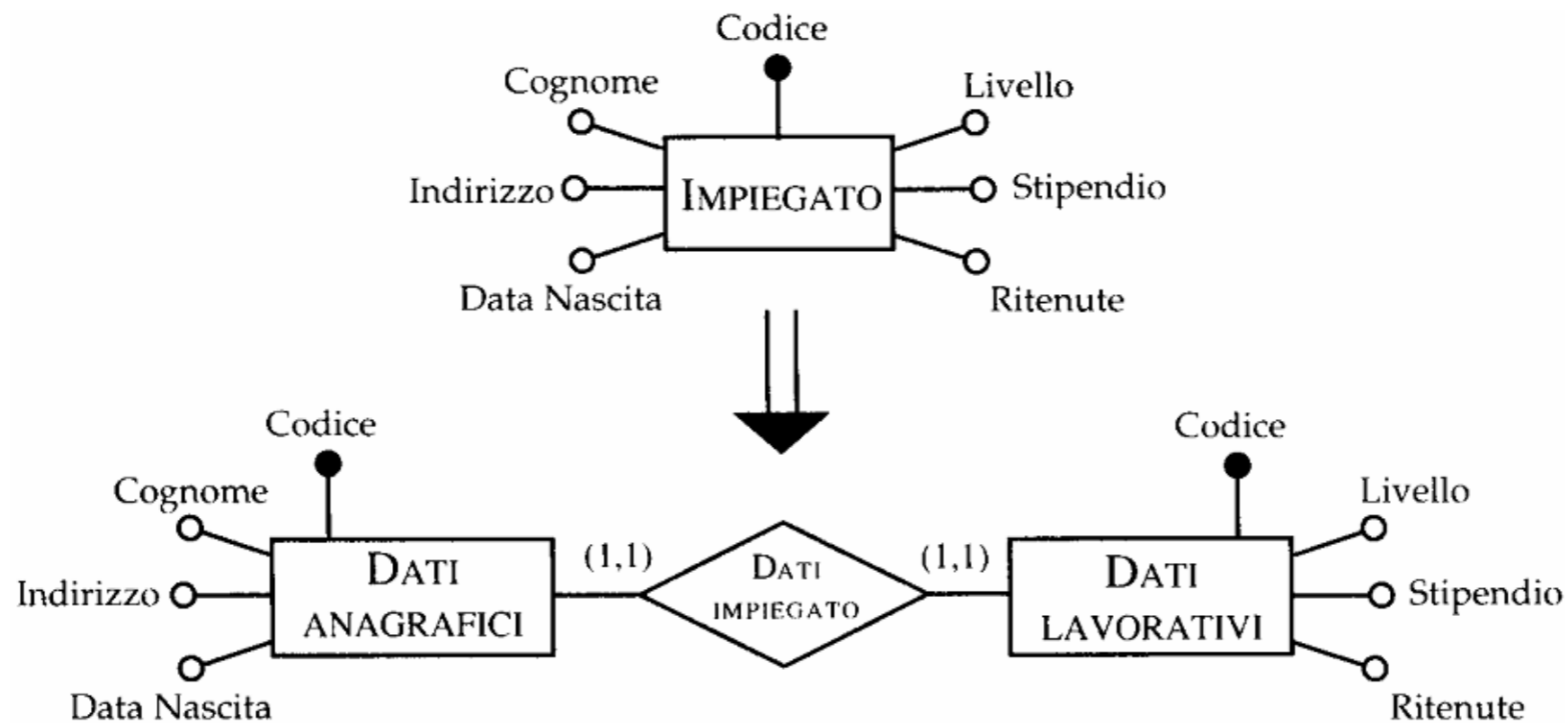
- ▶ limiti di applicabilità:
  - ▶ se la copertura non è totale non si può fare: dove mettere gli E che non sono né E1, né E2 ?
  - ▶ se la copertura non è esclusiva introduce ridondanza: per una istanza presente sia in E1 che in E2 si rappresentano due volte gli attributi di E



# PARTIZIONAMENTO/ACCORPAMENTO

---

- Il principio generale e' il seguente: gli accessi si riducono
  - separando attributi di uno stesso concetto che vengono acceduti da operazioni diverse
  - raggruppando attributi di concetti diversi che vengono acceduti dalle medesime operazioni



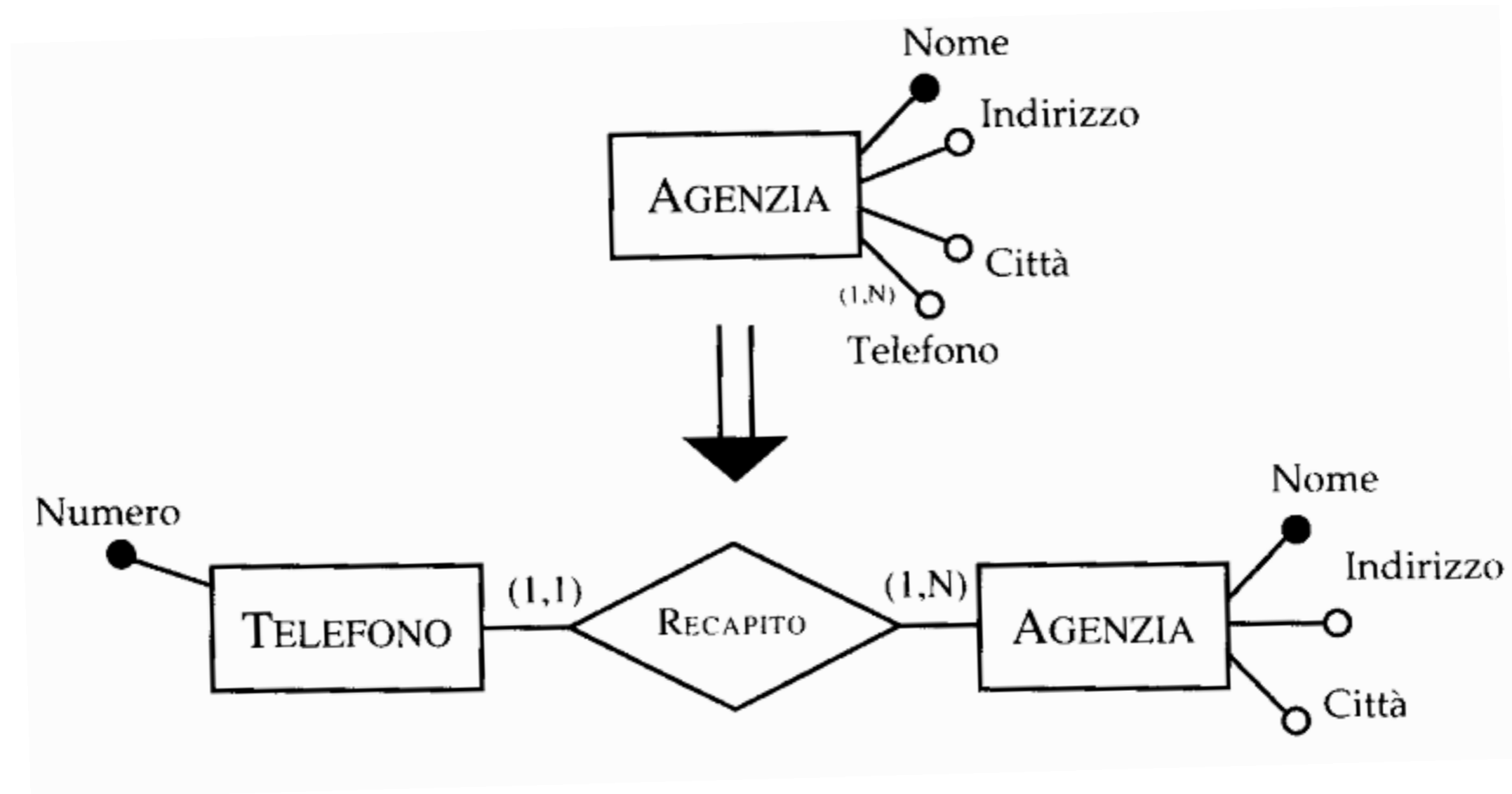
# PARTIZIONAMENTO VERTICALE E ORIZZONTALE

---

- Nell'esempio precedente vengono create due entità e gli attributi vengono divisi: **partizionamento verticale**
- Nel caso in cui si suddivida in due entità con gli stessi attributi (ad esempio Analista e Venditore) con operazioni distinte sulle due si ha il **partizionamento orizzontale**

# ELIMINAZIONE ATTRIBUTI MULTIPLI

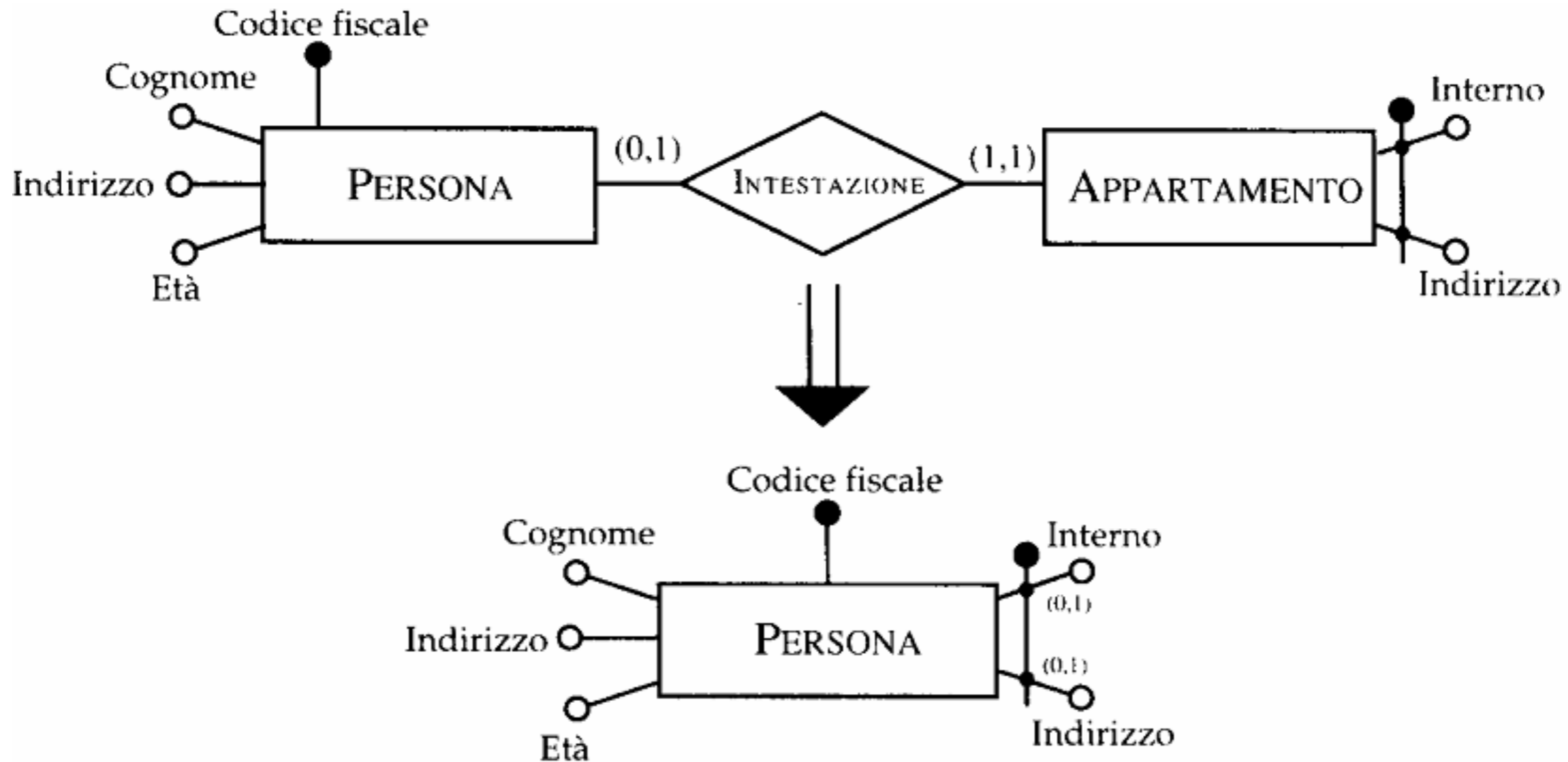
---



# ACCORPAMENTO DI ENTITÀ

---

- Operazione logico inversa del partizionamento

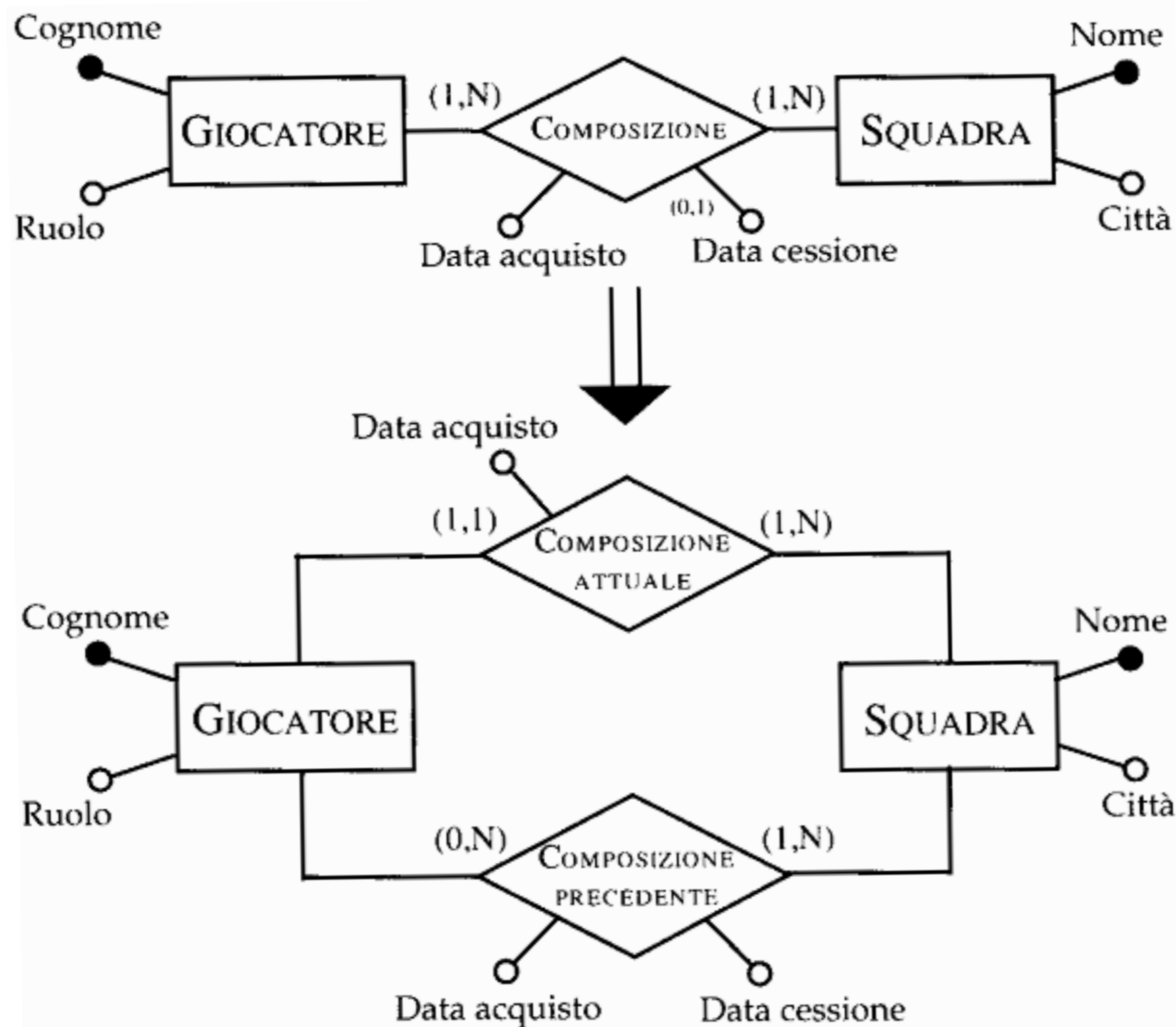


# QUANDO SI ACCORPA ?

---

- L'accorpamento è giustificato se
  - le operazioni più frequenti su Persona richiedono sempre i dati relativi all'appartamento
  - quindi vogliamo risparmiare gli accessi alla relazione che li lega.
- Normalmente gli accorpamenti si fanno su relazioni uno ad uno, raramente su uno a molti mai su molti a molti.

# PARTIZIONAMENTO O RAGGRUPPAMENTO DI ASSOCIAZIONI



# SCELTA DELLA CHIAVE PRIMARIA

---

- Gli attributi con valori nulli non possono essere identificatori principali
- Identificatori composti da pochi attributi sono da preferire a quelli con molti attributi
- Identificatori composti da pochi attributi sono da preferire a identificatori esterni
- Un identificatore che viene utilizzato da molte operazioni per accedere alle occorrenze di una entità è da preferire agli altri



# TRADUZIONE CANONICA

---

- Regola Generale
  - Per ogni entità si definisce una tabella con lo stesso nome aventi per attributi gli stessi attributi e per chiave l'identificatore
  - Per ogni associazione si definisce una tabella con lo stesso nome avente per attributi gli stessi attributi e per chiave gli identificatori delle entità coinvolte (legate con vincolo di integrità referenziale)
  - Nel seguito applicheremo la regola per i tre tipi di relazioni esaminati in precedenza

## TRADUZIONE CANONICA (2)

---

- La traduzione canonica è sempre possibile ed è l'unica possibilità per le associazioni N a M
- Altre forme di traduzione delle associazioni sono possibili per altri casi di cardinalità (1 a 1, 1 a N)
- Le altre forme di traduzione fondono in una stessa relazione entità e associazioni

# TRADUZIONE ASSOCIAZIONE MOLTI A MOLTI

---



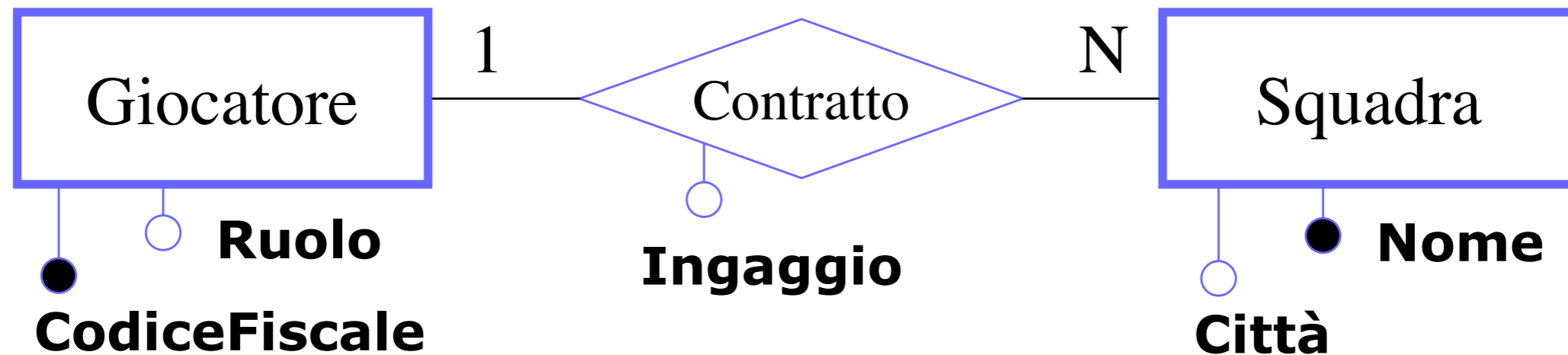
IMPIEGATO (Matricola, Nome)

PROGETTO(Codice, Nome)

PARTECIPAZIONE(Matricola, Codice, Data Inizio)

# UNO A MOLTI

---



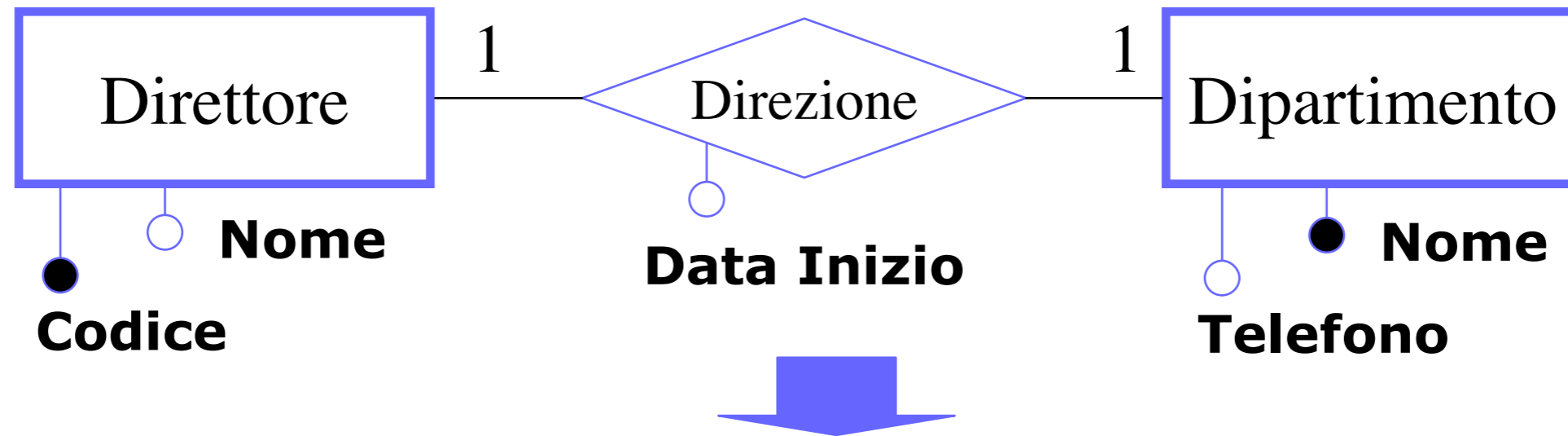
GIOCATORE (CodiceFiscale , Ruolo)  
SQUADRA(Nome, Città)  
CONTRATTO(CodiceFiscale, **Nome**, Ingaggio)



GIOCATORE (CodiceFiscale , Ruolo, Nome, Ingaggio)  
SQUADRA(Nome, Città)

# UNO A UNO

---



DIRETTORE (Codice, Nome, Data Inizio, NomeDipartimento)

DIPARTIMENTO(Nome, Telefono)

oppure

DIRETTORE (Codice, Nome,)

DIPARTIMENTO(Nome, Telefono, Data Inizio, Direttore)

# ENTITÀ CON IDENTIFICATORE ESTERNO

---



STUDENTE (Matricola, NomeUniversità, NomeStudente)

UNIVERSITA' (Nome, Indirizzo)

- Rappresentando l'identificatore esterno come chiave nell'entità studente si rappresenta direttamente anche l'associazione
- Questo è sempre vero perché un identificatore esterno si può avere solo con associazioni aventi cardinalità (1,1)

**FINE DELLA LEZIONE**