



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Progettazione Logica

Basi di Dati

Corso di Laurea in Informatica per il Management

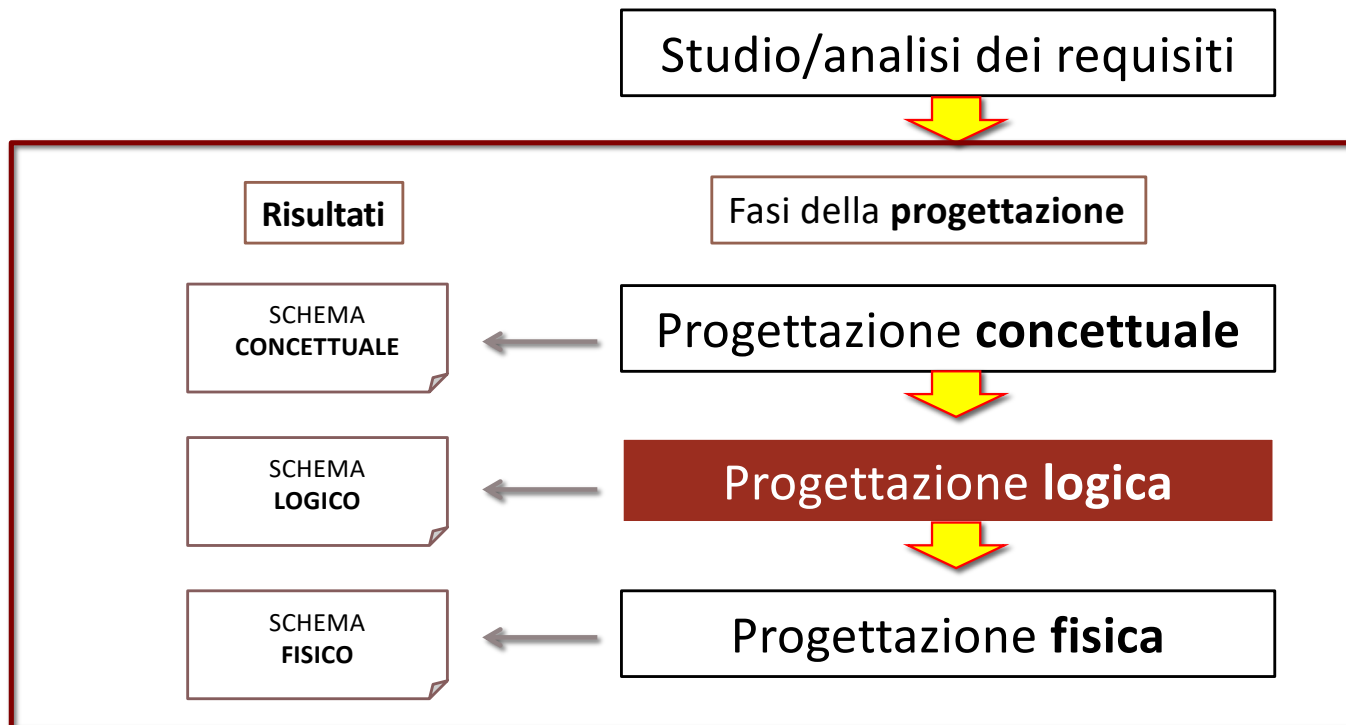
Alma Mater Studiorum - Università di Bologna

Prof. Marco Di Felice

Dipartimento di Informatica – Scienza e Ingegneria

marco.difelice3@unibo.it

Progettazione logica



Progettazione logica

L'obiettivo della **progettazione logica** è la **realizzazione del modello logico** (es. relazionale) a partire dalle informazioni del modello E-R.

DIAGRAMMA E-R

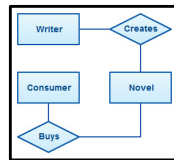


TAVOLA VOLUMI



ANALISI COSTI



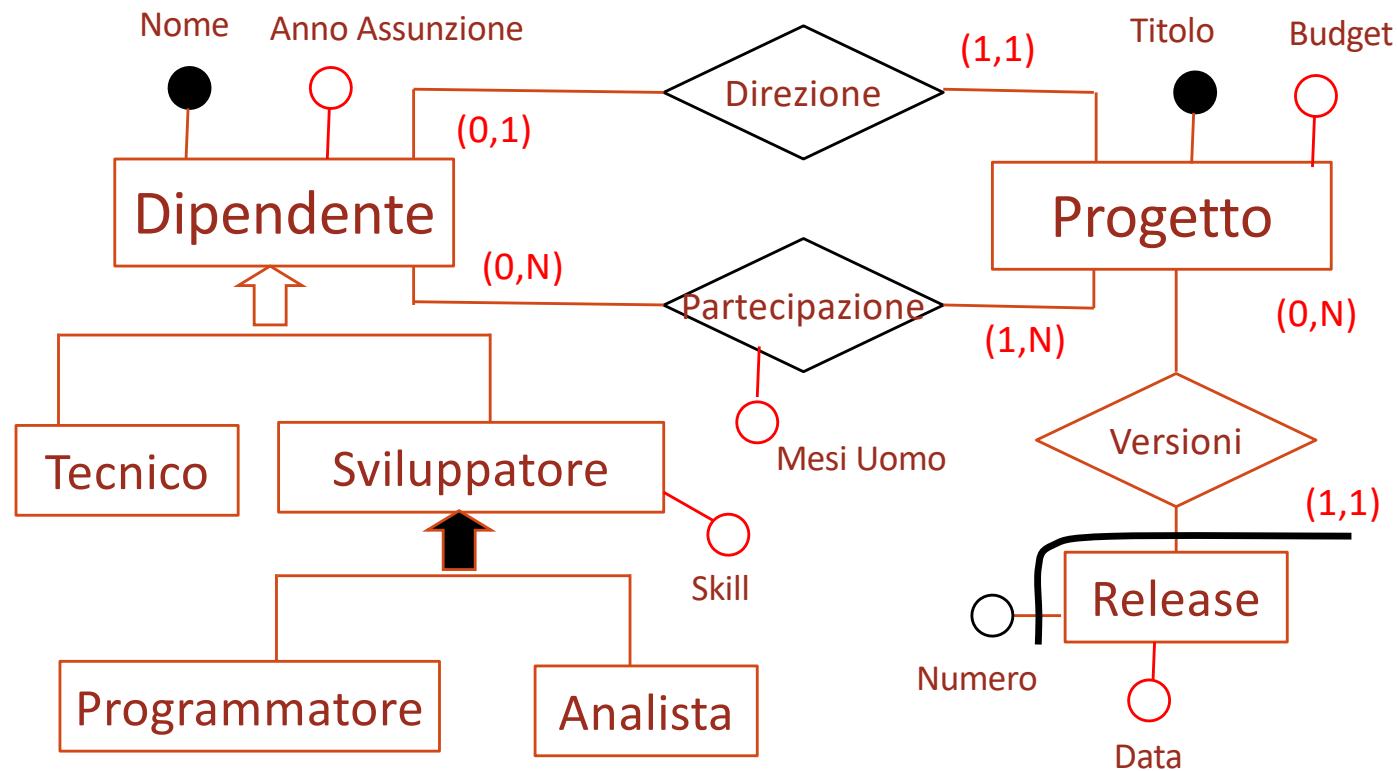
Modello Relazionale



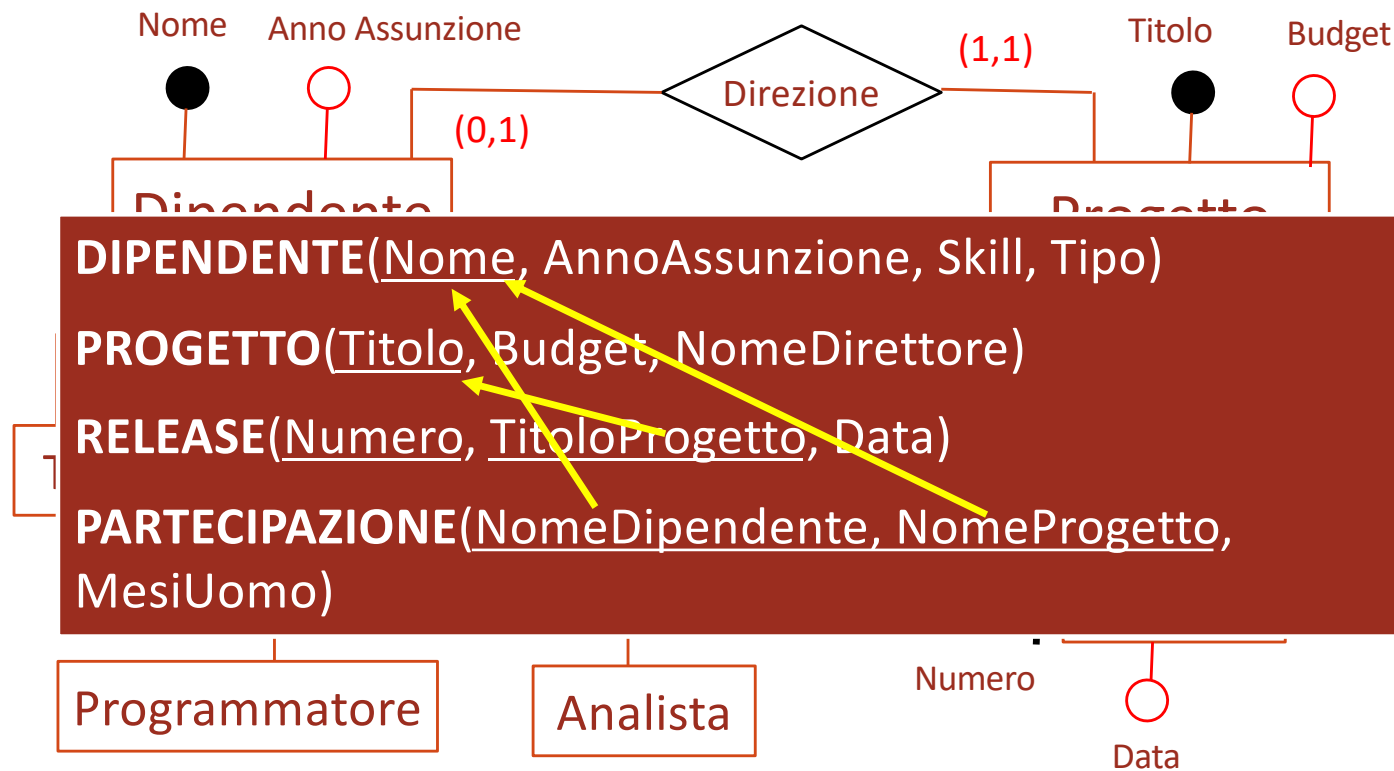
BASI DI DATI

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT

Progettazione logica



Progettazione logica



Progettazione logica

Una possibilità (**DA EVITARE**) è quella di tradurre ogni entità ed ogni relazione del modello E-R con una tabella corrispondente ...

PROBLEMI:

- **Efficienza** → Quante tabelle sono generate? Efficienza delle operazioni sui dati?
- **Correttezza** → Come si possono tradurre le generalizzazioni? Non esiste un costrutto equivalente nel modello E-R ...

Progettazione logica

Per garantire la qualità dello schema prodotto, la **progettazione logica** tipicamente include **due passaggi**:

- **Ristrutturazione del modello concettuale** → modificare lo schema E-R al fine di abilitare la traduzione nel modello logico e di ottimizzare il progetto nel suo complesso.
- **Traduzione nel modello logico** → traduzione dei costrutti del modello E-R nei costrutti equivalenti del modello relazionale ...

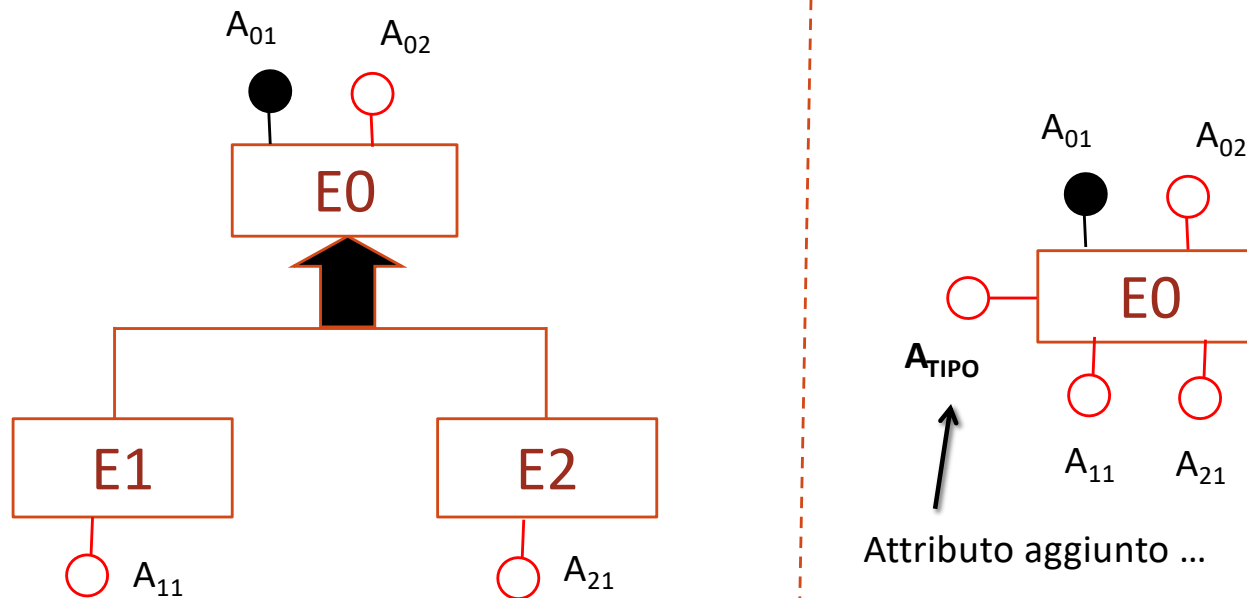
Progettazione logica

Prima di tradurre il modello E-R, è necessario **ristrutturarlo** per motivi di **correttezza/efficienza**:

FASI (F) PREVISTE (alcune di esse potrebbero non essere necessarie)

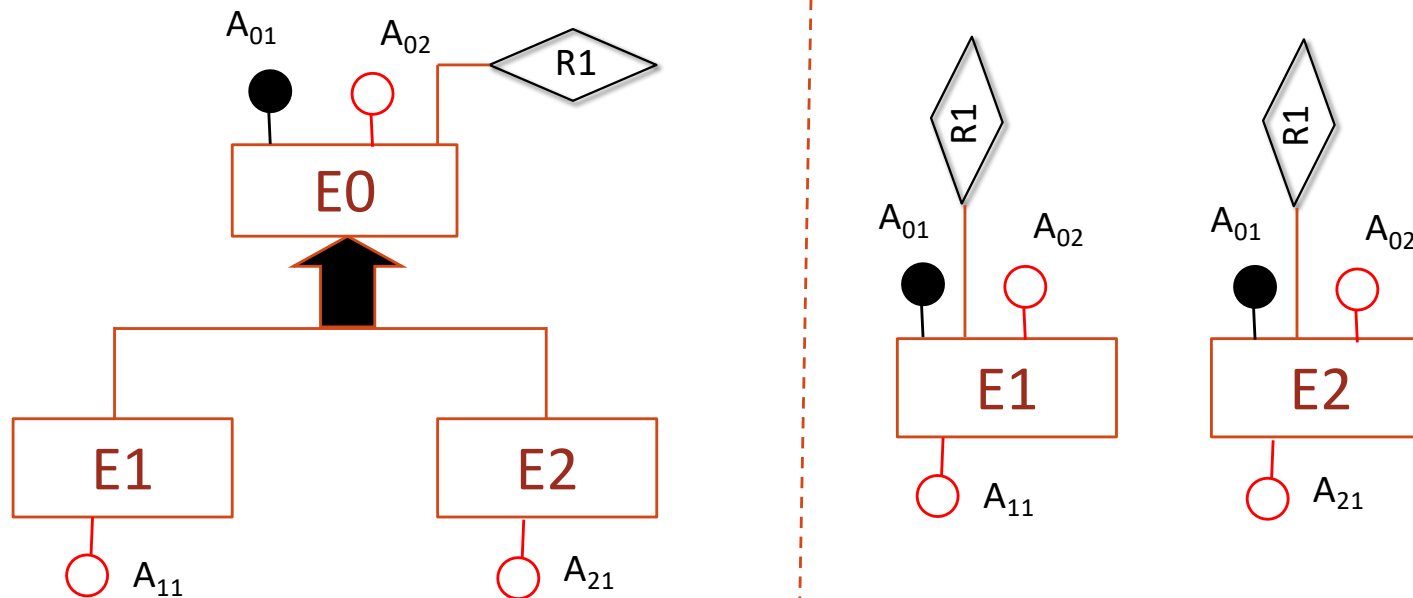
- **F0. Eliminazione delle generalizzazioni**
- **F1.** Eliminazione degli attributi multi-valore
- **F2.** Partizionamento/accorpamento di concetti
- **F3.** Analisi delle ridondanze

Progettazione logica



SOLUZIONE 1 (SOL1): Accorpamento delle entità figlie nell'entità genitore (con accorpamento dei relativi attributi e delle relazioni)...

Progettazione logica



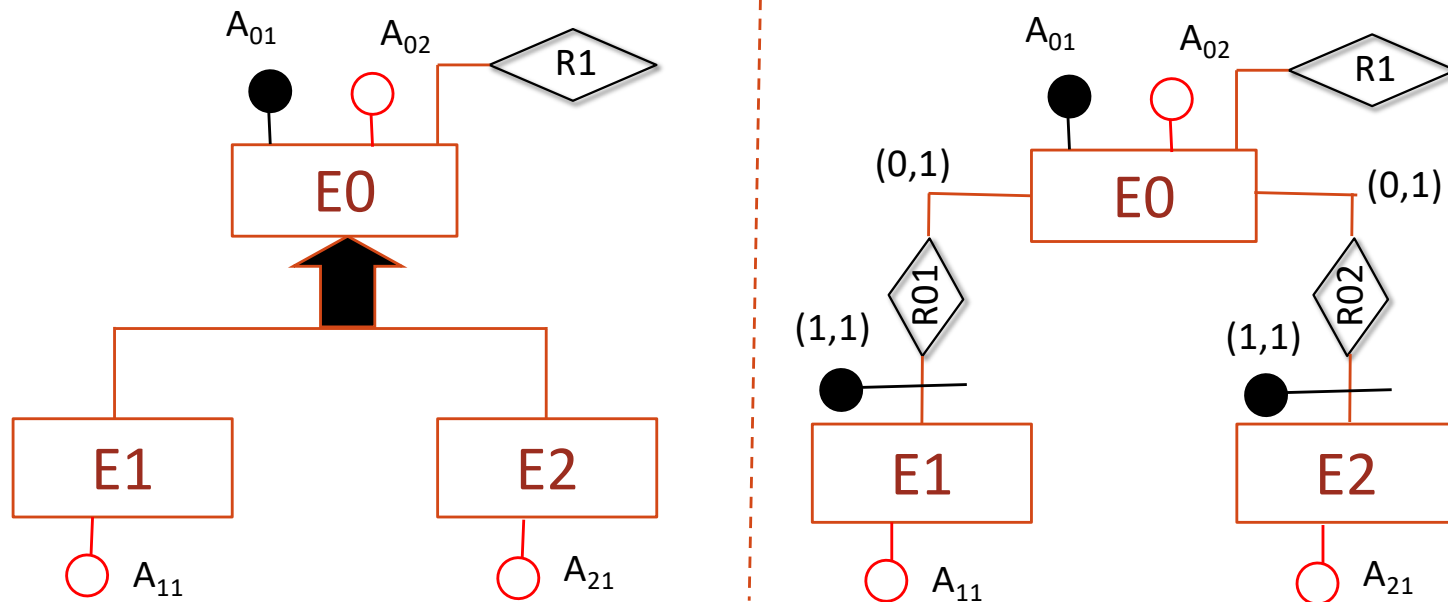
SOLUZIONE 2 (SOL2): Accorpamento delle entità genitore nelle entità figlie (con accorpamento dei relativi attributi e delle relazioni)...

Progettazione logica

Quale traduzione utilizzare?

- **SOL1** introduce valori nulli ed un attributo aggiuntivo, ma è conveniente quando non ci sono troppe differenze concettuali tra E0, E1 ed E2 ...
- **SOL2** è possibile solo se la generalizzazione è totale, non introduce valori nulli, ma è conveniente quando ci sono operazioni che coinvolgono per lo più E1 ed E2 ma non l'entità genitore E0 ...

Progettazione logica



SOLUZIONE 3 (SOL3): Sostituzione delle generalizzazione con relazioni tra entità genitore ed entità figlie.

Progettazione logica

- **SOL3** non introduce valori nulli, ed è utile quando ci sono operazioni che si riferiscono solo ad istanze di E1, E2 ed E0, ma presenta la necessità di introdurre dei vincoli:
- Un'occorrenza di E0 **non può partecipare in contemporanea** ad R01 ed R02.
- Se la **generalizzazione è totale**, ogni occorrenza di E0 deve appartenere ad R01 o R02 ...

Progettazione logica

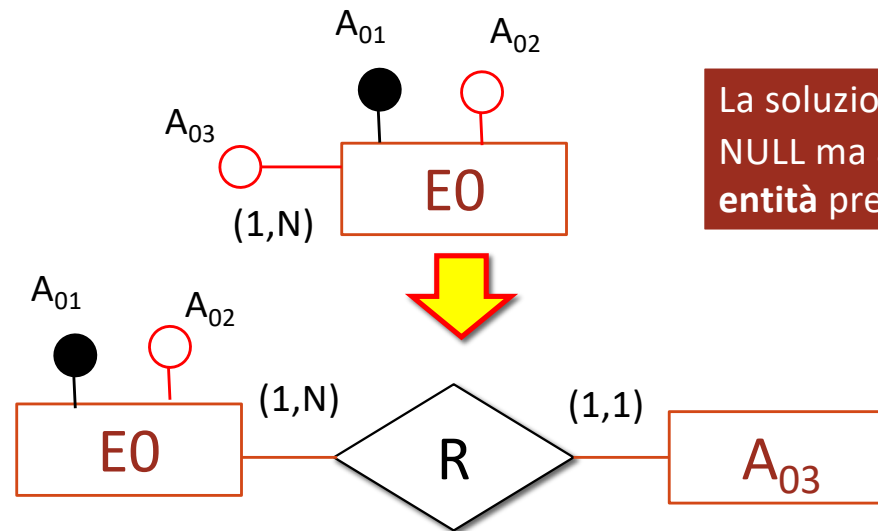
Prima di tradurre il modello E-R, è necessario **ristrutturarlo** per motivi di **correttezza/efficienza**:

FASI (F) PREVISTE (alcune di esse potrebbero non essere necessarie)

- **F0.** Eliminazione delle generalizzazioni
- **F1. Eliminazione degli attributi multi-valore**
- **F2.** Partizionamento/accorpamento di concetti
- **F3.** Analisi delle ridondanze

Progettazione logica

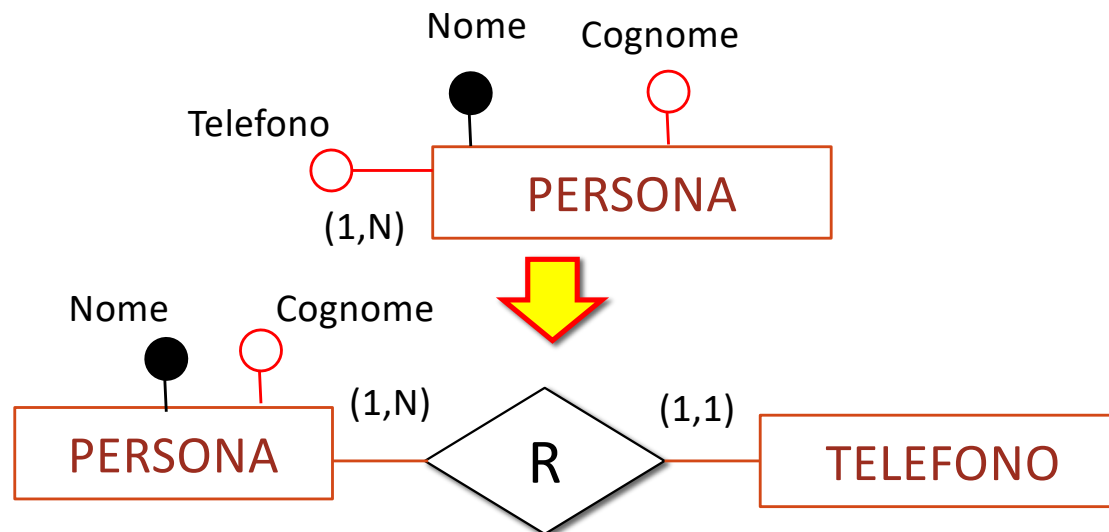
Gli **attributi multivalore** non sono presenti nel modello logico, ma possono essere modellati anche con una **relazione uno-a-molti** ...



La soluzione non introduce valori NULL ma **aumenta il numero di entità** presenti nel sistema ...

Progettazione logica

Gli **attributi multivalore** non sono presenti nel modello logico, ma possono essere modellati anche con una **relazione uno-a-molti** ...



Progettazione logica

Prima di tradurre il modello E-R, è necessario **ristrutturarlo** per motivi di **correttezza/efficienza**:

FASI (F) PREVISTE (alcune di esse potrebbero non essere necessarie)

- **F0.** Eliminazione delle generalizzazioni
- **F1.** Eliminazione degli attributi multi-valore
- **F2. Partizionamento/accorpamento di concetti**
- **F3.** Analisi delle ridondanze

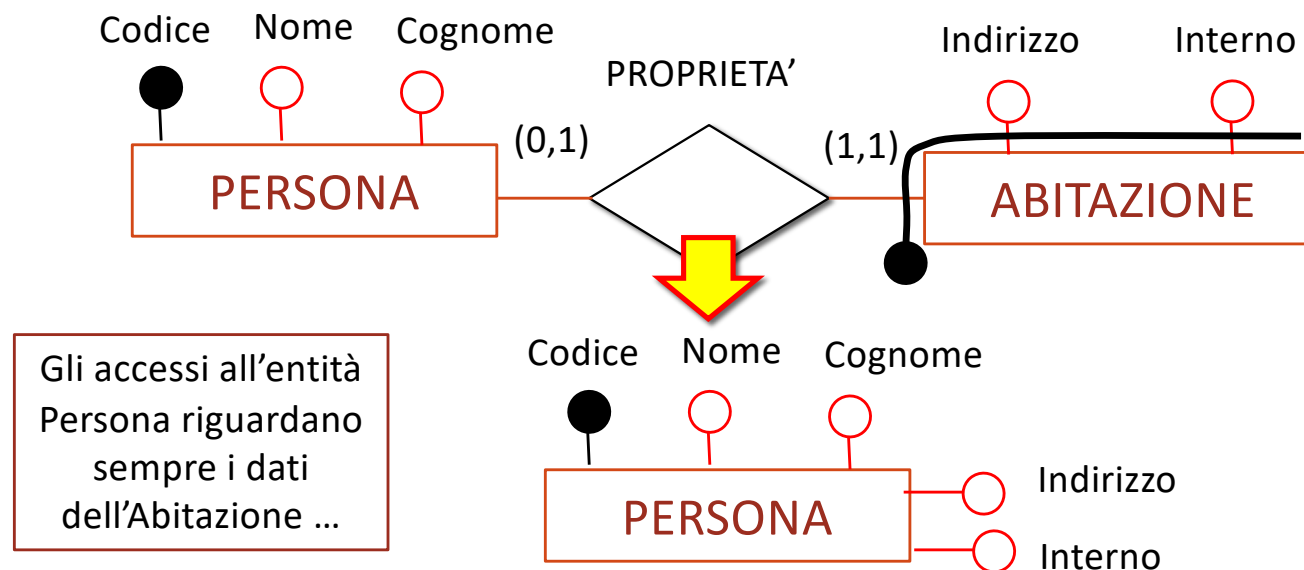
Progettazione logica

Per una dato modello E-R, è possibile **ridurre il numero di accessi**:

- separando attributi che vengono acceduti separatamente → **partizionamenti**
- raggruppando attributi di entità diverse ma acceduti allo stesso tempo → **accorpamenti**
- E' necessario avere **una stima sul volume dei dati** per un'indicazione se/come partizionare/accorpare entità.

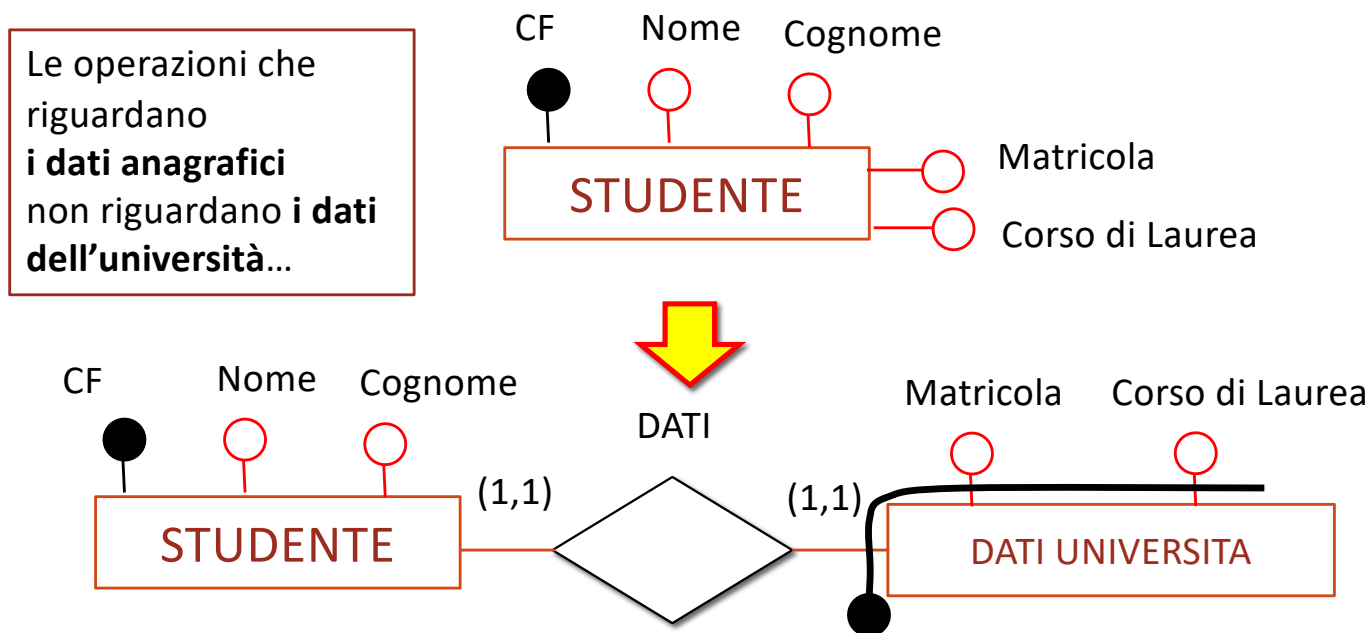
Progettazione logica

Gli accorpamenti di entità riguardano **relazioni uno-ad-uno...**



Progettazione logica

Partizionamento verticale di un'entità sulla base dei suoi attributi.



Progettazione logica

Prima di tradurre il modello E-R, è necessario **ristrutturarlo** per motivi di **correttezza/efficienza**:

FASI (F) PREVISTE (alcune di esse potrebbero non essere necessarie)

- **F0.** Eliminazione delle generalizzazioni
- **F1.** Eliminazione degli attributi multi-valore
- **F2.** Partizionamento/accorpamento di concetti
- **F3.** Analisi delle ridondanze

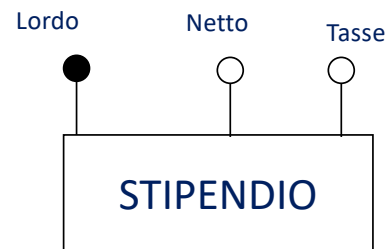
Progettazione logica

Nel modello E-R, potrebbero essere presenti **ridondanze sui dati**, ossia **informazioni significative ma derivabili da altre** già presenti nel modello E-R.

- **(Eventuali) vantaggi delle ridondanze:**
 - Operazioni sui dati più efficienti
- **Svantaggi delle ridondanze:**
 - Maggiore occupazione di memoria
 - Maggiore complessità degli aggiornamenti

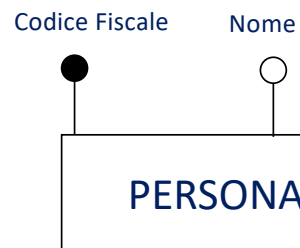
Progettazione logica

- Esempi di **ridondanze concettuali** in un diagramma E-R



ATTRIBUTI DERIVABILI

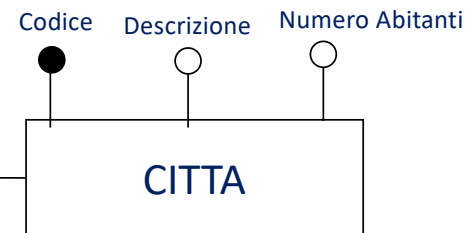
Lo stipendio netto si può ricavare dal lordo e dalle tasse.



(1,1)

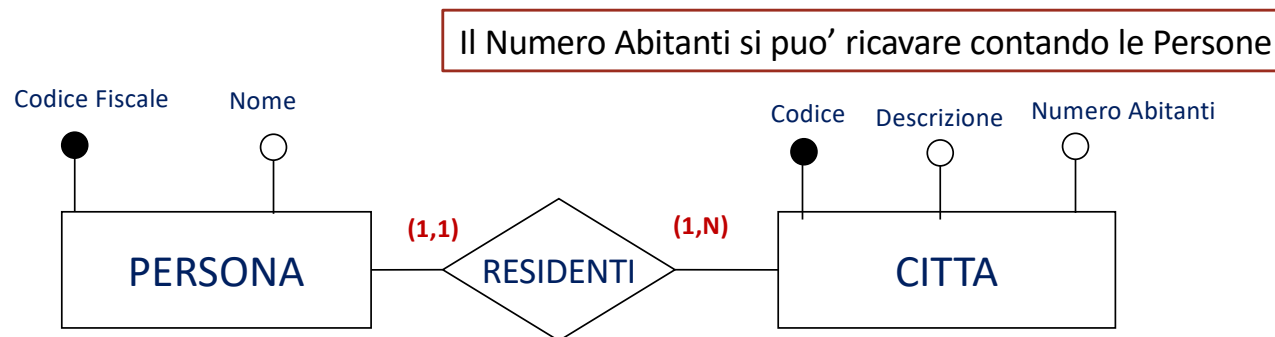
RESIDENTI

(1,N)



Progettazione logica

In questa fase della progettazione logica, è necessario valutare cosa fare delle **ridondanze concettuali**...



- **SOLUZIONE1:** Eliminare l'attributo NumeroAbitanti
- **SOLUZIONE2:** Conservare l'attributo nel diagramma E-R.

Progettazione logica

Per scegliere cosa fare di un attributo ridondante, è possibile utilizzare l'**analisi del modello E-R** che abbiamo visto nella progettazione concettuale.

Sia S lo schema **E-R senza ridondanze**

Sia S_{rid} lo schema **E-R con ridondanze**

1. Si **calcolano il costo e l'occupazione di memoria** di entrambi gli schemi: $\langle c(S), m(S) \rangle$ e $\langle c(S_{rid}), m(S_{rid}) \rangle$
2. Si **confrontano** $c(S)/c(S_{rid})$ e $|m(s) - m(S_{rid})|$
3. Si prende una **decisione** in base al valore delle metriche

Progettazione logica

Per effettuare l'analisi del modello E-R, è necessario disporre delle **tavole dei volumi e delle operazioni**.

- **Operazione 1 (OP1)**: Inserire una nuova persona (200 volte/giorno).
- **Operazione 2 (OP2)**: Visualizzare tutti i dati di una città, incluso il numero di abitanti (5 volte/giorno)

TAVOLA delle OPERAZIONI

Operazione	Tipo	Frequenza
Operazione1	I	200
Operazione2	I	5

Progettazione logica

Per effettuare l'analisi del modello E-R, è necessario disporre delle **tavole dei volumi e delle operazioni**.

TAVOLA dei VOLUMI

Concetto	Tipo	Volume
Città	E	100
Persona	E	500000
Residenti	R	500000

- Assumiamo che le informazioni sui volumi siano fornite dal **documento di specifica**.

Progettazione logica

Analisi dello schema S_{rid} (**caso con ridondanza**):

Operazione 1 (OP1): frequenza 200 volte/giorno

TAVOLA DEGLI ACCESSI

Concetto	Costrutto	Accessi	Tipo
Persona	Entita'	1	W
Residenti	Relazione	1	W
Citta'	Entita'	1	W

$$w_i=1$$
$$\alpha=2$$

$$c(OP1) = 200 * 1 * (3 * 2) = \mathbf{1200}$$

Progettazione logica

Analisi dello schema S_{rid} (**caso con ridondanza**):

Operazione 2 (OP2): frequenza 5 volte/giorno

TAVOLA DEGLI ACCESSI

Concetto	Costrutto	Accessi	Tipo
Citta'	Entita'	1	L

$$w_i=1$$
$$\alpha=2$$

$$c(OP2) = 5 * 1 * (0 * 2 + 1) = 5$$

Progettazione logica

Analisi dello schema S (**caso senza ridondanza**):

Operazione 1 (OP1): frequenza 200 volte/giorno

TAVOLA DEGLI ACCESSI

Concetto	Costrutto	Accessi	Tipo
Persona	Entita'	1	W
Residenti	Relazione	1	W

$$w_i=1$$
$$\alpha=2$$

$$c(OP1) = 200 * 1 * (2 * 2 + 0) = \mathbf{800}$$

Progettazione logica

Analisi dello schema S (**caso senza ridondanza**):

Operazione 2 (OP2): frequenza 5 volte/giorno

TAVOLA DEGLI ACCESSI

Concetto	Costrutto	Accessi	Tipo
Citta'	Entita'	1	L
Residenza	Relazione	5000	L

$$w_i=1$$
$$\alpha=2$$

$$c(Op2) = 5 * 1 * (0 * 2 + 5001) = \mathbf{25005}$$

Progettazione logica

Riassumendo:

➤ $c(S_{rid}) = c(Op1) + c(Op2) = 1200 + 5 \sim 1200$

➤ $c(S) = c(Op1) + c(Op2) = 800 + 25005 \sim 26000$

Vediamo ora l'occupazione di memoria:

➤ $m(S) = X$ (byte)

➤ $m(S_{rid}) = X + 100 * 4 = X + 400$ (byte)

Volume dell'entità Città

Il campo aggiuntivo richiede 4 byte

Progettazione logica

Riassumendo, la **presenza della ridondanza**:

- Introduce un overhead di memoria di **400 byte**
- Migliora lo speedup delle operazioni:
 $26000/1200 \sim \mathbf{20!}$

Risultato dell'analisi delle ridondanze: In questo caso, è conveniente **MANTENERE** (o nel caso introdurre) l'attributo Numero Abitanti!

Progettazione logica

Per garantire la qualità dello schema prodotto, la **progettazione logica** tipicamente include **due passaggi**:

- **Ristrutturazione del modello concettuale** → modificare lo schema E-R al fine di abilitare la traduzione nel modello logico e di ottimizzare il progetto nel suo complesso.
- **Traduzione nel modello logico** → traduzione dei costrutti del modello E-R nei costrutti equivalenti del modello relazionale ...

Progettazione logica

La **progettazione logica** deve tradurre i costrutti del modello E-R nei costrutti del modello logico di riferimento (nel nostro caso, **il modello relazionale**), garantendo l'**equivalenza** dei modelli ...

In sintesi:

- Le **entità** diventano **tabelle** sugli stessi attributi.
- Le **relazioni del modello E-R** diventano **tabelle** sugli identificatori delle entità coinvolte (più gli attributi propri),.. ma sono possibili traduzioni differenti sulla base delle cardinalità.

Progettazione logica

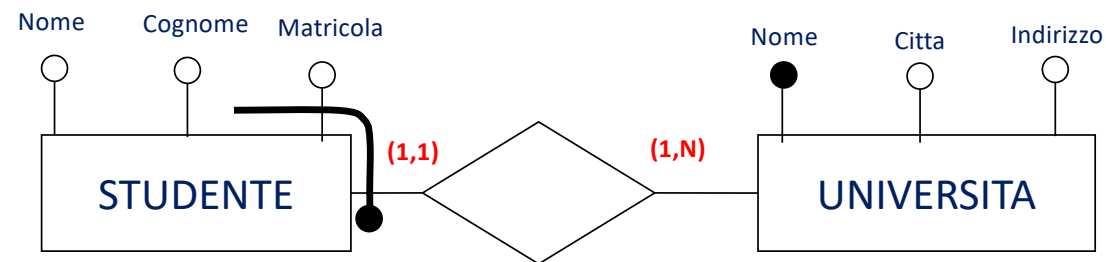
Traduzione di entità con identificatore interno



- Le entità del modello E-R si traducono in tabelle del modello relazionale. L'**identificatore** del modello E-R diventa la **chiave primaria** della tabella.

Progettazione logica

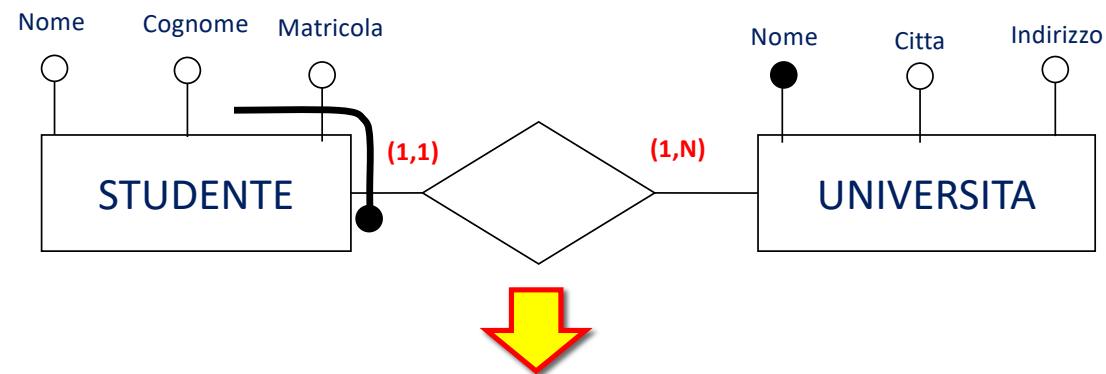
Traduzione di **entità con identificatore esterno**



- Le entità con **identificatore esterno** si traducono in una **tabella** che include tra le chiavi gli identificatori dell'entità esterna.

Progettazione logica

Traduzione di entità con identificatore esterno

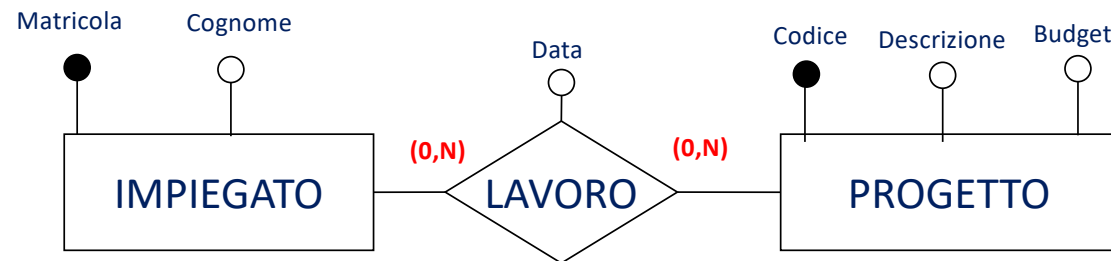


STUDENTE(Matricola, NomeUniversita, Nome, Cognome)

UNIVERSITA(Nome, Città, Indirizzo)

Progettazione logica

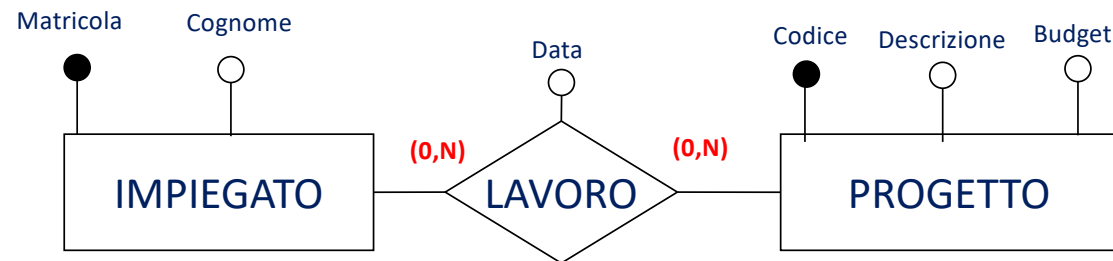
Traduzione di relazioni multi-a-molti



- **Ogni entità diventa una tabella** con lo stesso nome, stessi attributi e per chiave il suo identificatore.
- **Ogni relazione diventa una tabella**, con gli stessi attributi e come chiave gli identificatori delle entità coinvolte.

Progettazione logica

Traduzione di relazioni multi-a-molti



IMPIEGATO(Matricola, Cognome)

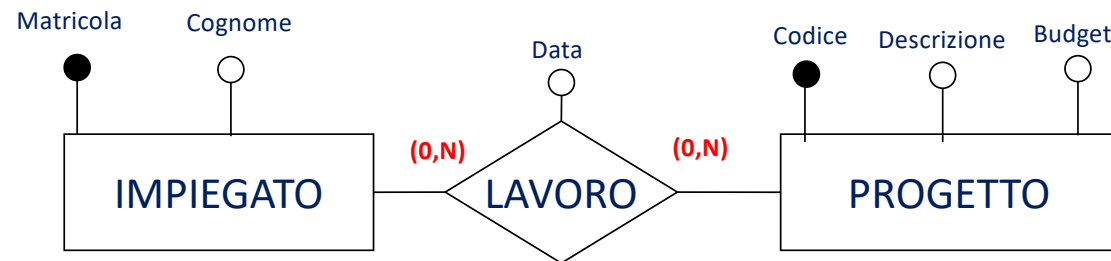
PROGETTO(Codice, Descrizione, Budget)

LAVORO(Matricola, Codice, Data)

Vincoli di integrità
tra gli attributi

Progettazione logica

Traduzione di relazioni multi-a-molti



IMPIEGATO(Matricola, Cognome)

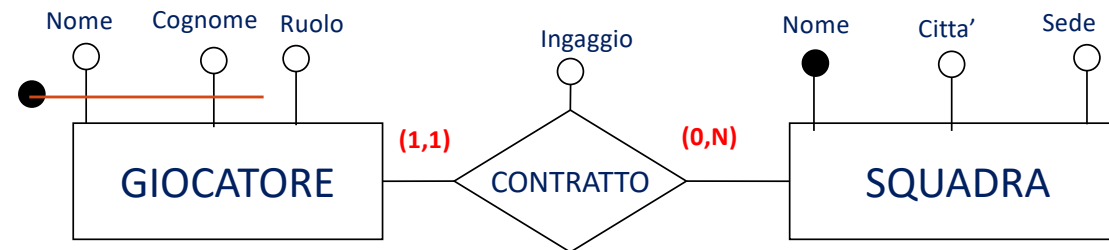
PROGETTO(Codice, Descrizione, Budget)

LAVORO(MatImpiegato, CodProgetto, Data)

E' possibile
ridenominare gli
attributi della
relazione

Progettazione logica

Traduzione di relazioni uno-a-molti

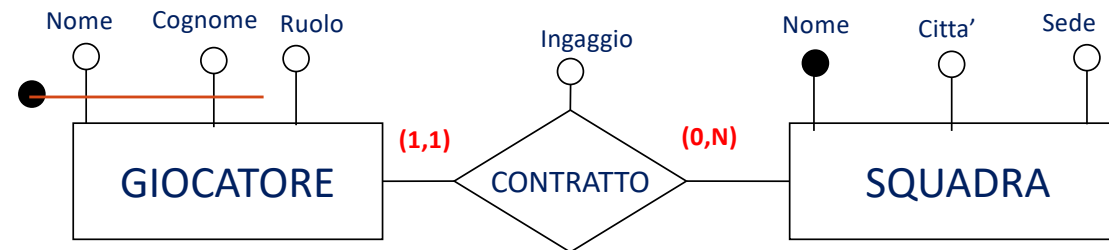


Sono possibili **due traduzioni**:

1. Traducendo la **relazione come una tabella separata** (come nel caso delle relazioni molti-a-molti).
2. **Inglobando la relazione nell'entità** con card. massima 1.

Progettazione logica

Traduzione di relazioni uno-a-molti



GIOCATORE(Nome, Cognome, Ruolo)

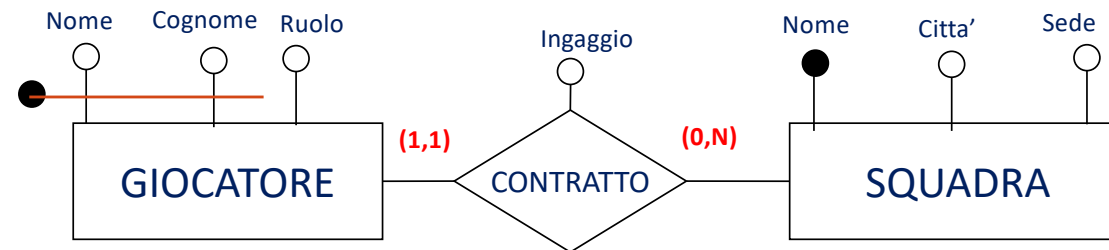
TRADUZIONE 1

SQUADRA(Nome, Citta', Sede)

CONTRATTO(Nome, Cognome, NomeSquadra, Ingaggio)

Progettazione logica

Traduzione di relazioni uno-a-molti



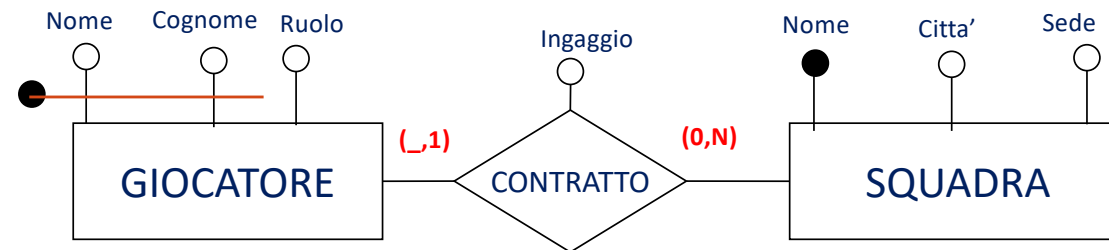
TRADUZIONE 2

GIOCATORE(Nome, Cognome, Ruolo, NomeSquadra, Ingaggio)

SQUADRA(Nome, Citta', Sede)

Progettazione logica

Traduzione di relazioni uno-a-molti

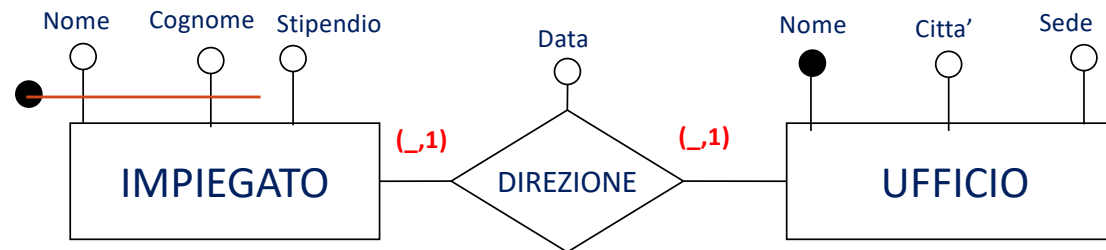


Cosa accade se vario la cardinalità min. di GIOCATORE?

- $\text{cardMin}=0 \rightarrow$ **Soluzione 1** preferibile
- $\text{cardMin}=1 \rightarrow$ **Soluzione 2** preferibile

Progettazione logica

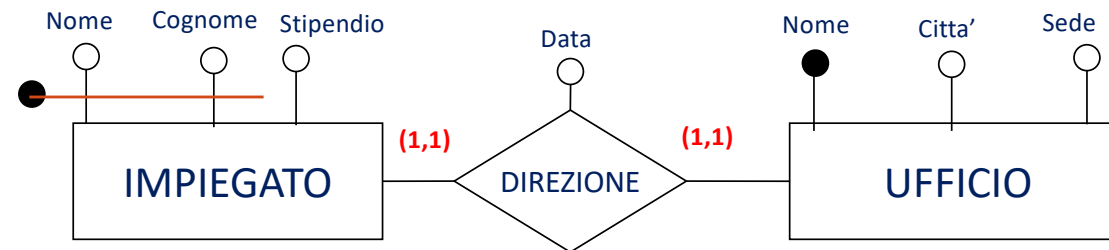
Traduzione di relazioni uno-a-uno



- Sono possibili **3 diverse alternative**, in base alla **cardinalità minima** delle due entità in gioco ...

Progettazione logica

Traduzione di relazioni uno-a-uno

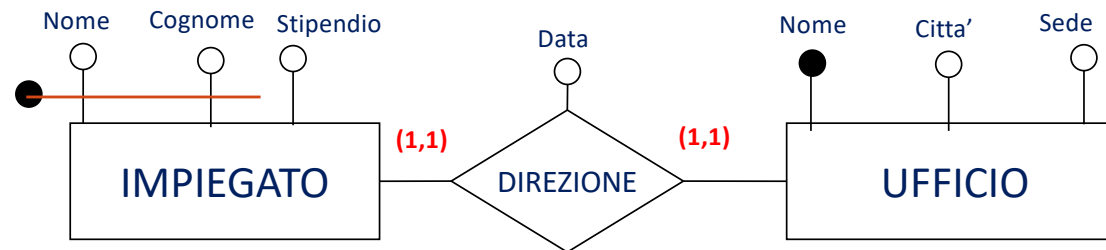


- **Caso 1:** Cardinalità obbligatorie per entrambe le entità (cardMin pari ad 1 per entrambe).

Si traduce il modello **inglobando la relazione in una delle due entità** (traduzioni simmetriche).

Progettazione logica

Traduzione di relazioni uno-a-uno



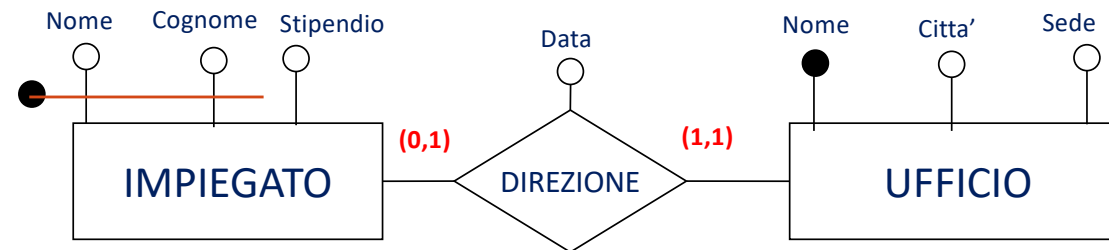
IMPIEGATO(Nome, Cognome, Stipendio, Data, NomeUfficio)

UFFICIO(Nome, Città', Sede)

In alternativa, è possibile inglobare la relazione DIREZIONE nell'entità UFFICIO ...

Progettazione logica

Traduzione di relazioni uno-a-uno

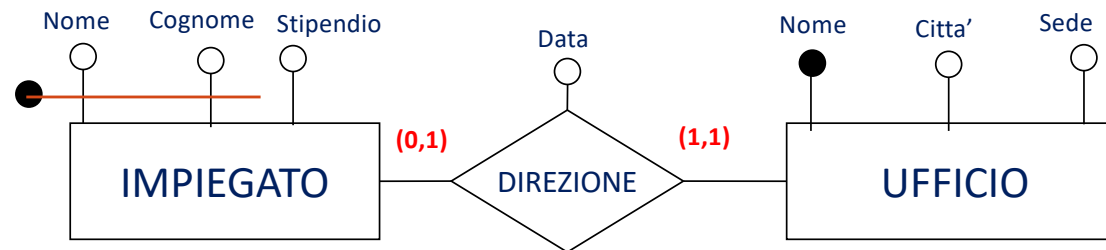


- **Caso 2:** Partecipazione obbligatoria per una delle entità (cardMax=1 per una delle due).

Si traduce il modello inglobando la relazione nell'entità che ha partecipazione obbligatoria ...

Progettazione logica

Traduzione di relazioni uno-a-uno

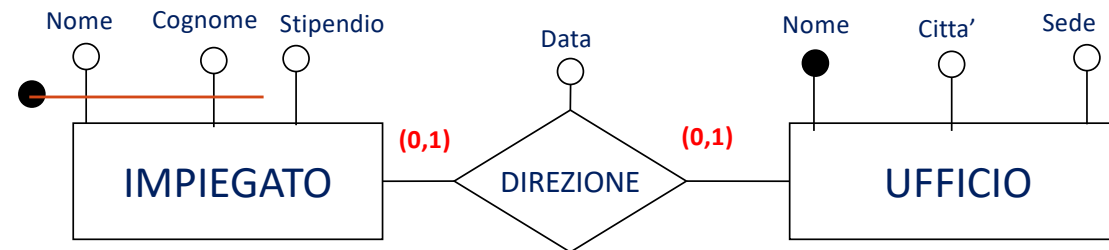


IMPIEGATO(Nome, Cognome, Stipendio)

UFFICIO(Nome, Città, Sede, Data, NomeDirettore, CognomeDirettore)

Progettazione logica

Traduzione di relazioni uno-a-uno

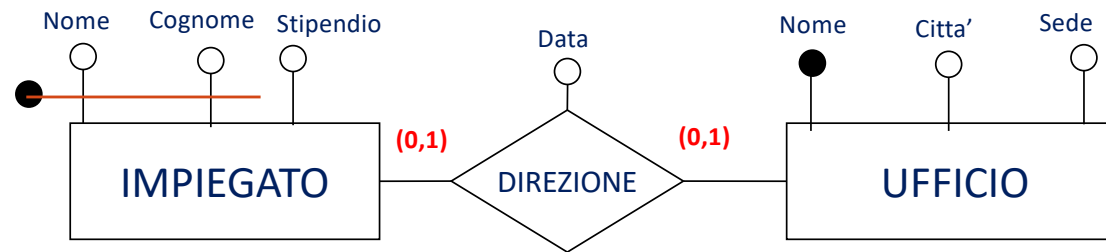


- **Caso 3: Partecipazione facoltativa per entrambe le entità** (cardMin pari a 0 per entrambe).

Si traduce il modello **traducendo la relazione come una tabella a sè stante** (analogo del caso uno-a-molti).

Progettazione logica

Traduzione di relazioni uno-a-uno



IMPIEGATO(Nome, Cognome, Stipendio)

UFFICIO(Nome, Citta', Sede)

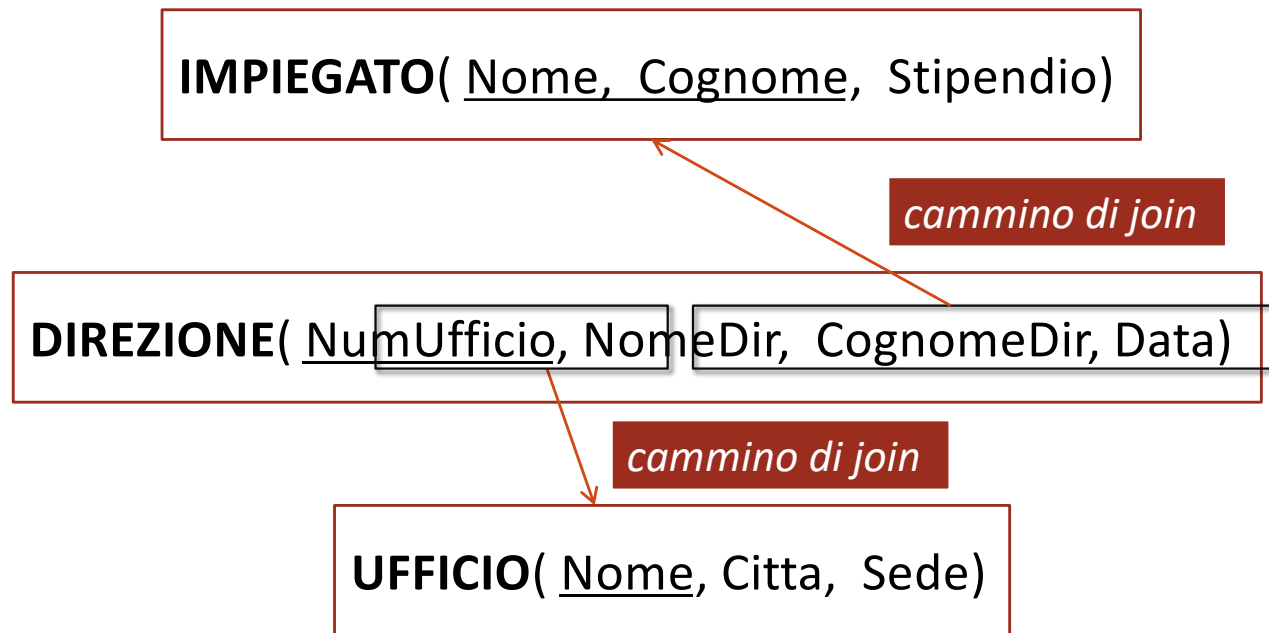
DIREZIONE(NomeUfficio, NomeDirettore, CognomeDirettore, Data)

Progettazione logica

Come per la fase di progettazione concettuale, è necessario **corredare lo schema logico di opportuna documentazione** perchè **non tutti i vincoli sono esprimibili nello schema logico:**

- Tabella delle **business rules** (vista in precedenza)
- **Insieme dei vincoli di integrità** referenziali
 - Rappresentati **attraverso tabella**
 - Rappresentati **in maniera grafica** (diagramma logico).

Progettazione logica



Esempio di **diagramma logico**, con **vincoli di integrità** ...

Progettazione logica

Ricapitolando:

- **STEP 2: Progettazione Logica**
 - STEP 2.1: **Analisi delle ridondanze**
 - STEP 2.2: Eliminazione delle **generalizzazioni** e degli **attributi multi-valore**
 - STEP 2.3: **Accorpamenti/partizionamenti** di concetti
 - STEP 2.4: Traduzione nel **modello logico**