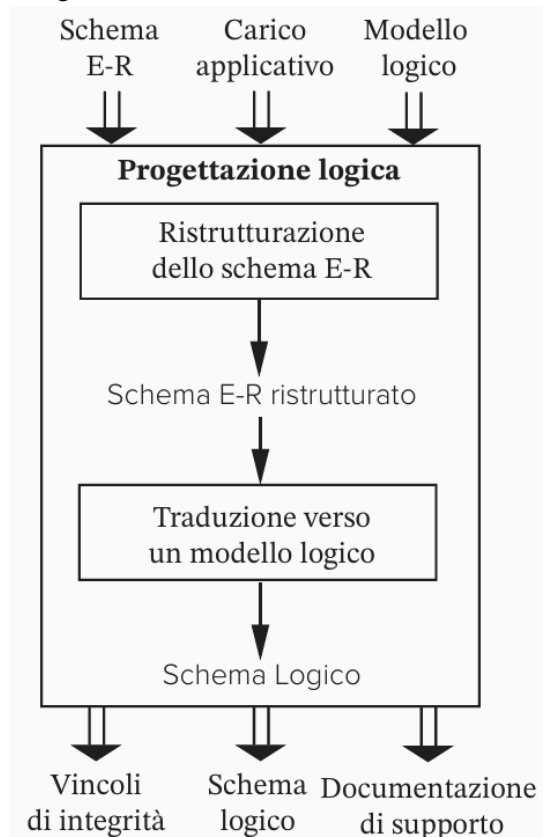


# Logical Design

Le attività principali della progettazione logica sono:

- *ristrutturazione dello schema Entità-Relazione*: si basa su criteri di ottimizzazione dello schema e di semplificazione della fase successiva;
- *traduzione verso il modello logico*: fa riferimento a uno specifico modello logico e può includere un'ulteriore ottimizzazione che si basa sulle caratteristiche del modello logico stesso.

I dati di ingresso sono lo schema concettuale e il carico applicativo previsto, ovvero la dimensione dei dati e le caratteristiche delle operazioni. Il risultato è uno schema E-R ristrutturato. Questo schema e il modello logico scelto sono i dati di ingresso della seconda fase che produce lo schema logico; in questa fase si fanno verifiche della qualità dello schema ed eventuali ulteriori ottimizzazioni. I prodotti finali della progettazione logica sono lo schema logico finale, i vincoli di integrità definiti su di esso e la relativa documentazione.



## Analisi delle prestazioni

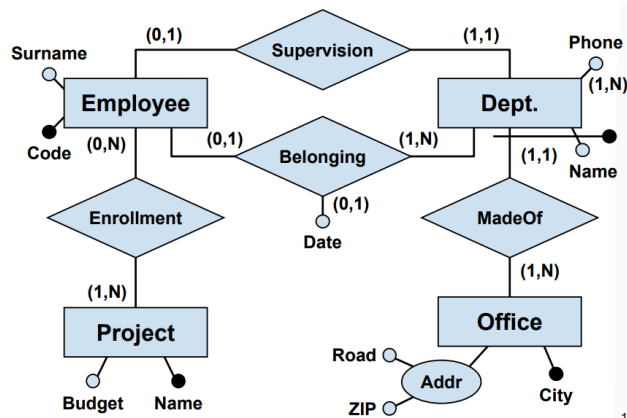
Uno schema E-R può essere modificato per ottimizzare alcuni *indici di prestazione*, chiamati così perché non sono valutabili in maniera precisa in sede di progettazione logica in quanto sono dipendenti anche da parametri fisici, ecc..

- *costo di un'operazione*: valutato in termini di numero di occorrenze di entità e associazioni che mediamente vanno visitate per rispondere a un'operazione sulla base di dati;
- *occupazione di memoria*: valutato in termini dello spazio di memoria necessario per memorizzare i dati descritti dallo schema.

Per lo studio di questi parametri serve conoscere:

- *volume dei dati*:

- numero di occorrenze di ogni entità e associazione dello schema;
- dimensioni di ciascun attributo (di entità o associazione).
- *caratteristiche delle operazioni:*
  - tipo dell'operazione;
  - frequenza (numero medio di esecuzioni in un certo intervallo di tempo);
  - dati coinvolti.

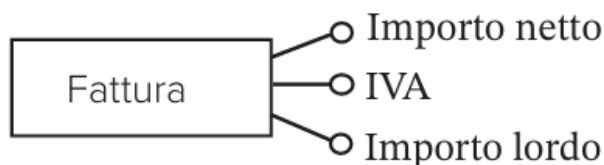


Name	Type	Size
Office	E	10
Dept.	E	80
Employee	E	2'000
Project	E	500
MadeOf	R	80
Belonging	R	1'900
Supervision	R	80
Enrollment	R	6'000

## Ristrutturazione di schemi E-R

Suddivisa in:

- **analisi delle ridondanze:**
  - si decide se eliminare o mantenere eventuali ridondanze (:= presenza di un dato che può essere derivato da altri dati);
  - casi più frequenti:
    - attributi derivabili da altri attributi della stessa entità (o associazione):



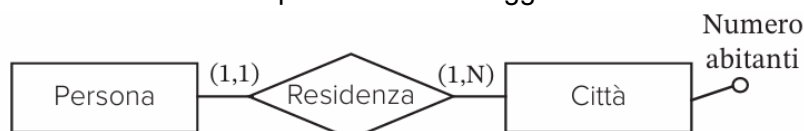
uno degli attributi è deducibile dagli altri attraverso un'operazione di somma o differenza.

- attributi derivabili da attributi di altre entità (o associazioni), di solito attraverso funzioni aggregative:



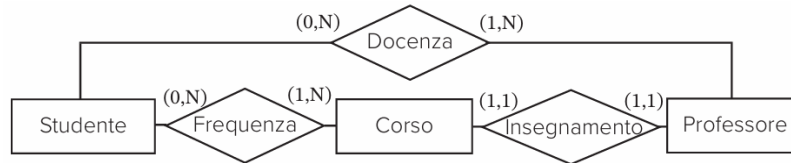
*importo totale* è derivabile attraverso l'associazione *composizione* dall'attributo *prezzo*, sommando i prezzi dei prodotti di un acquisto.

- attributi derivabili da operazioni di conteggio di occorrenze:



*numero abitanti* può essere derivato contando le occorrenze dell'associazione *Residenza* a cui tale città partecipa.

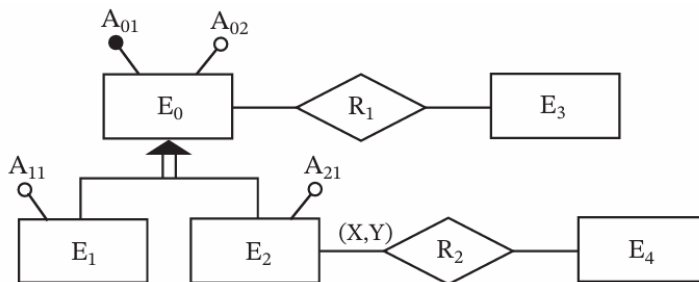
- associazioni derivabili dalla composizione di altre associazioni in presenza di cicli:



l'associazione *Docenza* può essere derivata dalle associazioni *Frequenza* e *Insegnamento*.

- La presenza di un dato derivato presenta:
  - vantaggio: riduzione degli accessi necessari per calcolare il dato derivato;
  - svantaggio: maggiore occupazione di memoria e la necessità di effettuare operazioni aggiuntive per mantenere il dato derivato aggiornato.

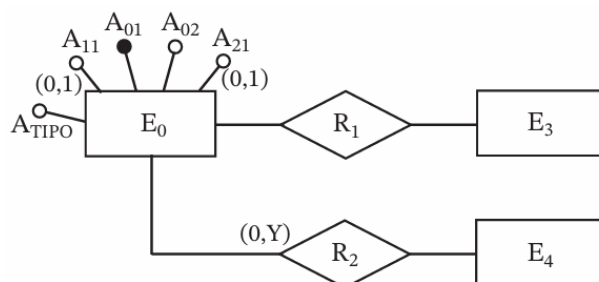
- **eliminazione delle generalizzazioni:**



- i sistemi per la gestione delle basi di dati non consentono di rappresentare direttamente le generalizzazioni, perciò è necessario rappresentarle mediante entità o associazioni, per farlo esistono tre alternative:

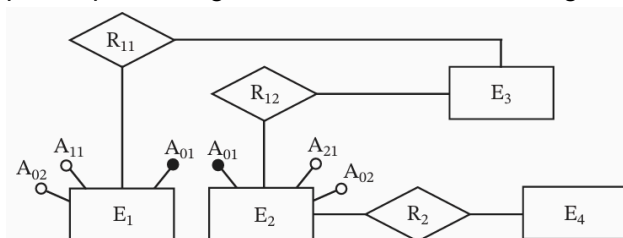
1. *accorpamento delle figlie della generalizzazione nel genitore:*

le entità  $E_1$  ed  $E_2$  vengono eliminate e le loro proprietà vengono aggiunte all'entità genitore  $E_0$ . Inoltre viene aggiunto un attributo che serve a distinguere il "tipo" di un'occorrenza di  $E_0$ , ovvero se apparteneva a  $E_1$  o a  $E_2$  nel caso di generalizzazione totale, o a nessuna di esse nel caso di generalizzazione parziale.



2. *accorpamento del genitore delle generalizzazione nelle figlie:*

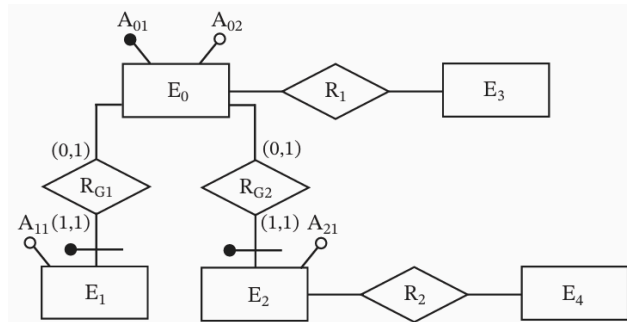
l'entità genitore  $E_0$  viene eliminata, i suoi attributi, il suo identificatore e le relazioni a cui partecipava vengono ereditate dalle entità figlie.



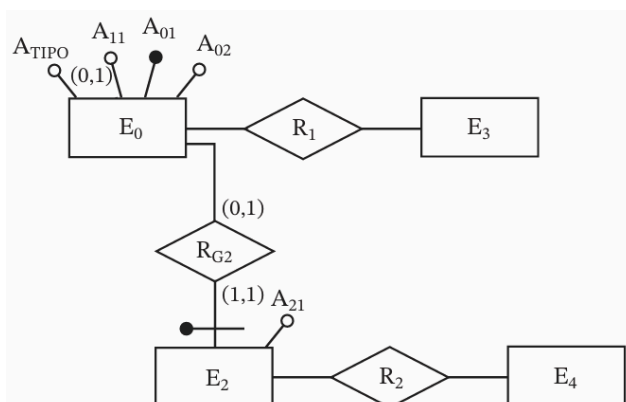
3. *sostituzione della generalizzazione con associazioni:*

la generalizzazione si trasforma in due associazioni uno a uno che legano l'entità genitore con le entità figlie. Non ci sono trasferimenti di attributi o associazioni e le entità figlie sono identificate esternamente dall'entità genitore. Vengono aggiunti dei vincoli:

ogni occorrenza di  $E_0$  non può partecipare contemporaneamente a  $R_{G1}$  e  $R_{G2}$ ; se la generalizzazione è totale ogni occorrenza di  $E_0$  deve partecipare obbligatoriamente a una delle due relazioni.

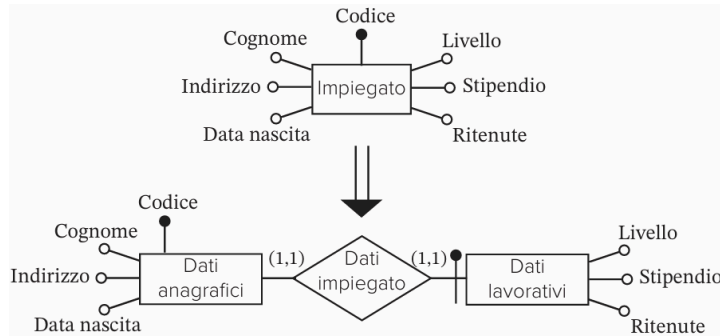


- la scelta tra le varie soluzioni possibili deve essere fatta considerando vantaggi e svantaggi relativamente all'occupazione di memoria e al costo delle operazioni:
  - scelta 1:**
    - conveniente quando le operazioni non fanno distinzione tra le occorrenze e tra gli attributi di  $E_0$ ,  $E_1$  e  $E_2$ ;
    - spreco di memoria per l'eventuale presenza di valori nulli;
    - ci assicura un numero minore di accessi rispetto alle altre soluzioni;
  - scelta 2:**
    - è possibile solo se la generalizzazione è totale, altrimenti le occorrenze dell'entità genitore che non sono occorrenze delle entità figlie non sarebbero rappresentate;
    - E' conveniente quando ci sono operazioni che si riferiscono solo a una delle due entità figlie;
    - risparmio di memoria rispetto alla scelta (1), perché gli attributi non assumono mai valori nulli;
    - riduzione degli accessi rispetto alla scelta (3) perché non si deve visitare l'entità genitore per accedere agli attributi dei figli.
  - scelta 3:**
    - è conveniente quando la generalizzazione non è totale e ci sono operazioni che si riferiscono solo a occorrenze di una delle entità figlie o dell'entità genitore;
    - risparmio di memoria rispetto alla scelta (1), per lo stesso motivo di prima;
    - incremento degli accessi per mantenere la consistenza delle occorrenze rispetto ai vincoli introdotti.
- le alternative viste non sono le uniche ammesse, talvolta è possibile utilizzare una combinazione delle tre soluzioni.

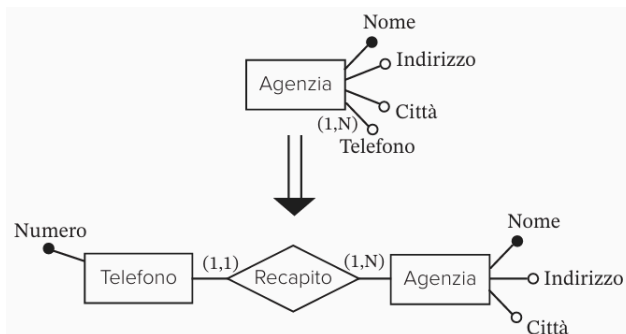


- partizionamento/accorpamento di entità e associazioni:**

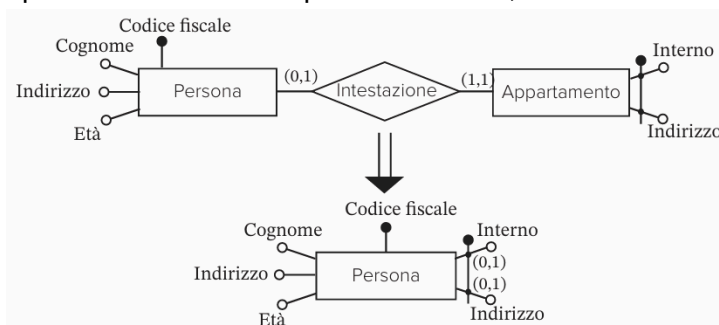
- entità e associazioni possono essere partizionati o accorpati in base a questo principio: gli accessi si riducono
  - separando attributi di uno stesso concetto che vengono acceduti da operazioni diverse;
  - raggruppando attributi di concetti diversi che vengono acceduti dalle stesse operazioni;
- partizionamenti di entità:**
  - decomposizione verticale:** si suddivide il concetto operando sui suoi attributi;
    - generano entità con pochi attributi che possono essere tradotte in strutture logiche sulle quali con un solo accesso è possibile recuperare molti dati.



- decomposizione orizzontale:** la suddivisione avviene sulle occorrenze dell'entità, può convenire decomporre l'entità in due entità distinte, corrisponde all'introduzione di una generalizzazione a livello logico.
  - effetto collaterale: dover duplicare le associazioni a cui l'entità originaria partecipava;
- eliminazione di attributi multivalore:**
  - come le generalizzazioni, non sono rappresentabili nel modello relazionale;
  - la ristrutturazione avviene reificando l'attributo multivalore:

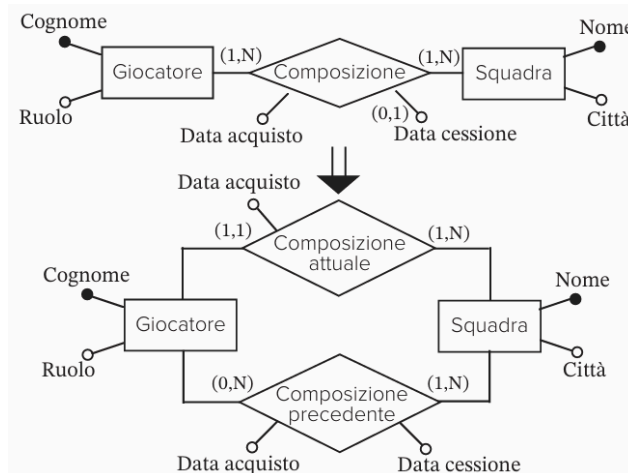


- accorpamento di entità:**
  - operazione inversa del partizionamento;



- effetto collaterale: possibile presenza di valori nulli;
- generalmente di effettuano su associazioni di tipo uno a uno, raramente su associazioni uno a molti, mai su associazioni molti a molti (in quest'ultimo caso generano ridondanze).

- il concetto di partizionamento e accorpamento di entità si può applicare anche sulle associazioni:
  - in alcuni casi può essere utile decomporre un'associazione tra due entità in due (o più) associazioni tra le stesse entità per separare le occorrenze:



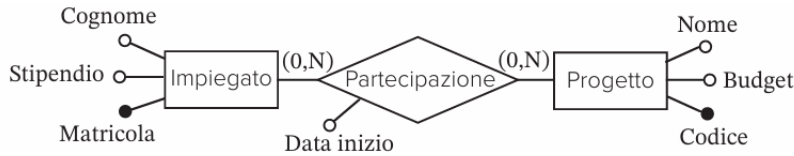
- è possibile accorpare due (o più) associazioni tra le medesime entità (che si riferiscono a due aspetti dello stesso concetto) in un'unica associazione.
- **scelta degli identificatori principali:**
  - essenziale nelle traduzioni verso il modello relazionale:
    - usate per stabilire legami tra dati in relazioni diverse;
    - i sistemi di gestione utilizzano la chiave primaria per la costruzione automatica di *indici*.
  - criteri di decisione:
    - gli attributi con valori nulli non possono essere identificatori principali;
    - un identificatore composto da uno o da pochi attributi è da preferire a quelli costituiti da molti attributi, in quanto:
      - garantisce che gli indici siano di dimensioni ridotte;
      - risparmio di memoria nella realizzazione dei legami logici tra le relazioni;
      - facilita le operazioni di join.
    - un identificatore interno con pochi attributi è preferibile rispetto ad uno esterno, infatti gli identificatori esterni vengono tradotti in chiavi che includono gli identificatori delle entità coinvolte nell'identificazione esterna;
    - un identificatore che viene utilizzato da molte operazioni per accedere alle occorrenze di un'entità è da preferire rispetto agli altri.
  - se nessuno degli identificatori candidati soddisfa i criteri viene introdotto un nuovo attributo *codice* generato appositamente per identificare le occorrenze delle entità.

## Traduzione verso il modello relazionale

Questa seconda fase corrisponde a una traduzione tra modelli di dati diversi: a partire dallo schema E-R ristrutturato (senza generalizzazioni e attributi multivalore, con un solo identificatore) si costruisce uno schema logico *equivalente*, in grado di rappresentare le stesse informazioni.

- **entità e associazioni molti a molti**

- **Relazioni binarie:**



la sua traduzione nel modello relazionale prevede:

- per ogni entità:
  - una relazione con lo stesso nome;
  - avente come attributi gli stessi dell'entità;
  - per chiave il suo identificatore.
- per ogni associazione:
  - una relazione con lo stesso nome;
  - avente per attributi gli stessi dell'associazione;
  - gli identificatori delle entità coinvolte che formano la chiave della relazione.

schema relazionale corrispondente:

**Impiegato**(Matricola, Cognome, Stipendio)

**Progetto**(Codice, Nome, Budget)

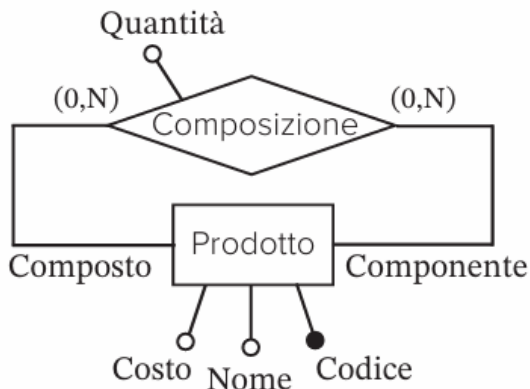
**Partecipazione**(Matricola, Codice, DataInizio)

Per rendere più comprensibile il significato dello schema è conveniente effettuare alcune ridenominazioni: **Partecipazione**(Impiegato, Progetto, DataInizio)

C'è un vincolo di integrità referenziale tra:

- Matricola in **Partecipazione** e la chiave di **Impiegato**;
- Codice in **Partecipazione** e la chiave di **Progetto**;

- **Relazioni ricorsive:**



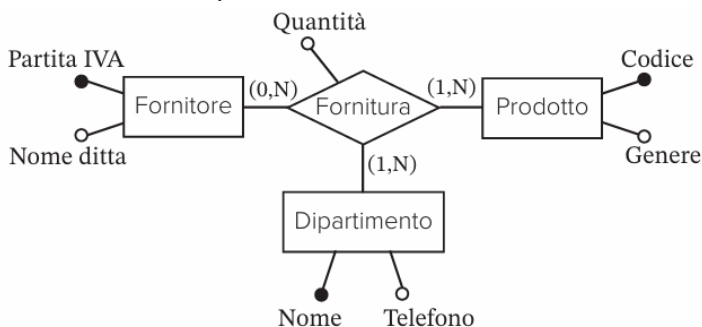
schema relazionale corrispondente:

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

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

gli attributi Composto e Componente contengono codici di prodotti: il primo ha il secondo come componente.

- **Associazioni con più di due entità:**



si traducono in maniera analoga alle associazioni binarie:

**Fornitore**(PartitaIVA, *NomeDitta*)

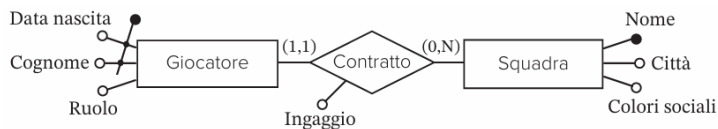
**Prodotto**(Codice, *Genere*)

**Dipartimento**(Nome, *Telefono*)

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

- **associazioni uno a molti**

- **associazione binaria**



schema relazionale corrispondente:

**Giocatore**(Cognome, DataNascita, *Ruolo*)

**Squadra**(Nome, *Città*, *ColoriSociali*)

**Contratto**(Giocatore, DataNascitaGiocatore, *NomeSquadra*, *Ingaggio*)

In **Contratto** la chiave è costituita solo dall'identificatore di **Giocatore** perché la sua cardinalità implica che ogni giocatore ha un contratto con una sola squadra. Dal momento che **Giocatore** e **Contratto** hanno la stessa chiave è possibile fonderle in un'unica relazione:

**Giocatore**(Cognome, DataNascita, *Ruolo*, *NomeSquadra*, *Ingaggio*)

**Squadra**(Nome, *Città*, *ColoriSociali*)

Con questa soluzione:

- abbiamo meno relazioni;
- è possibile avere valori nulli sugli attributi *NomeSquadra* e *Ingaggio*.

Vincolo di integrità referenziale tra:

- *NomeSquadra* di **Giocatore** e *Nome* di **Squadra**.

- **associazioni ternarie**

L'entità che partecipa all'associazione ternaria con cardinalità massima uguale a 1, viene tradotta in una relazione che contiene anche gli identificatori delle altre entità coinvolte nell'associazione. Se l'entità **Prodotto** nell'esempio di associazione ternaria precedente avesse cardinalità (1,1), quindi per ogni prodotto esiste un solo fornitore e un solo dipartimento al quale viene fornito, allora:

**Fornitore**(PartitaIVA, *NomeDitta*)

**Dipartimento**(Nome, *Telefono*)

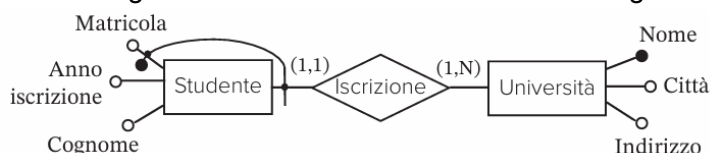
**Prodotto**(Codice, *Genere*, *Fornitore*, *Dipartimento*, *Quantità*)

Vincoli di integrità referenziale:

- tra l'attributo *Fornitore* della relazione **Prodotto** e l'attributo PartitaIVA di **Fornitore**;
- tra l'attributo *Dipartimento* della relazione **Prodotto** e l'attributo *Nome* della relazione **Dipartimento**.

- **entità con identificatore esterno:**

danno luogo a relazioni con chiavi che includono gli identificatori delle entità "identificanti".



schema relazionale:

**Studente**(Matricola, *NomeUniversità*, *Cognome*, *AnnoIscrizione*)

**Università**(Nome, *Città*, *Indirizzo*)

vincolo di integrità referenziale:

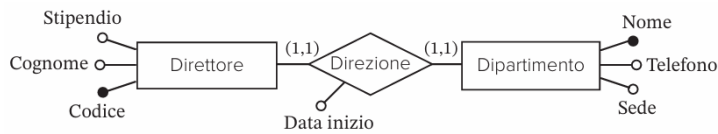


- tra l'attributo NomeUniversità della relazione **Studente** e l'attributo Nome di **Università**  
Rappresentando l'identificatore esterno si rappresenta direttamente anche l'associazione, infatti le entità identificate esternamente partecipano all'associazione sempre con una cardinalità minima e massima pari a 1. Inoltre, questo tipo di traduzione è valido indipendentemente dalla cardinalità con cui l'altra entità partecipa all'associazione.

- associazioni uno a uno**

diverse possibilità di traduzione:

- partecipazioni obbligatorie per entrambe le entità*



due possibilità:

- Direttore**(Codice, *Cognome*, *Stipendio*, *DipartimentoDiretto*, *InizioDirezione*)  
**Dipartimento**(Nome, *Telefono*, *Sede*)
- Direttore**(Codice, *Cognome*, *Stipendio*)  
**Dipartimento**(Nome, *Telefono*, *Sede*, *Direttore*, *InizioDirezione*)

E' possibile rappresentare l'associazione in una qualunque delle relazioni che rappresentano le due entità.

- partecipazione opzionale per una sola entità*



una sola soluzione è preferibile rispetto alle altre:

**Impiegato**(Codice, *Cognome*, *Stipendio*)  
**Dipartimento**(Nome, *Telefono*, *Sede*, *Direttore*, *InizioDirezione*)

in quanto in questo modo non è possibile avere valori nulli.

- partecipazione opzionale*

**Impiegato**(Codice, *Cognome*, *Stipendio*)  
**Dipartimento**(Nome, *Telefono*, *Sede*)  
**Direzione**(Direttore, *Dipartimento*, *DataInizioDirezione*)

- non presenta mai valori nulli sugli attributi dell'associazione;
- abbiamo una relazione in più  $\Rightarrow$  rende più complessa la base di dati.

Questa soluzione è da prendere in considerazione solo se il numero di occorrenze dell'associazione è molto basso rispetto alle occorrenze delle entità che partecipano all'associazione.