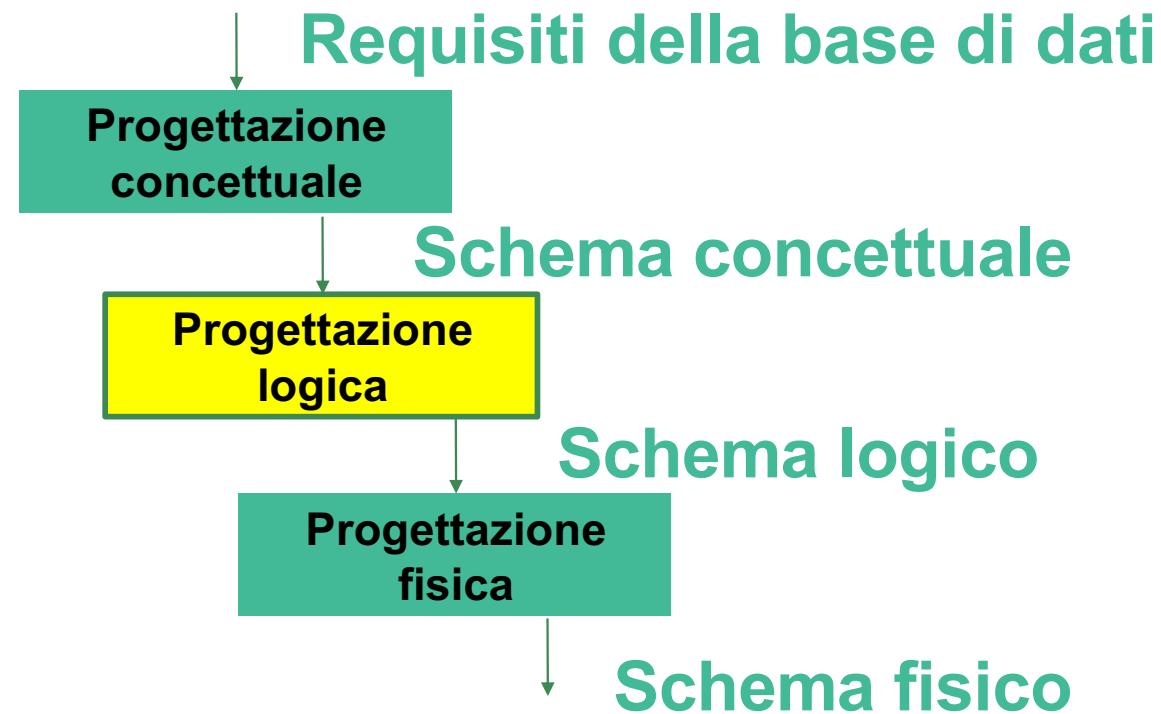


La progettazione logica

PROF. DIOMAIUTA CRESCENZO

UNIVERSITÀ DEGLI STUDI DELLA CAMPANIA «*LUIGI VANVITELLI*»

Progettazione concettuale



Progettazione logica

Permette la traduzione dello schema concettuale in un modello di rappresentazione dei dati.

Il prodotto di questa fase è detto schema logico della base di dati e fa riferimento ad un modello logico dei dati.

Tale modello descrive i dati secondo una rappresentazione concreta, indipendente dai dettagli fisici.

L
• Come memorizzare tabella poco mi interessa

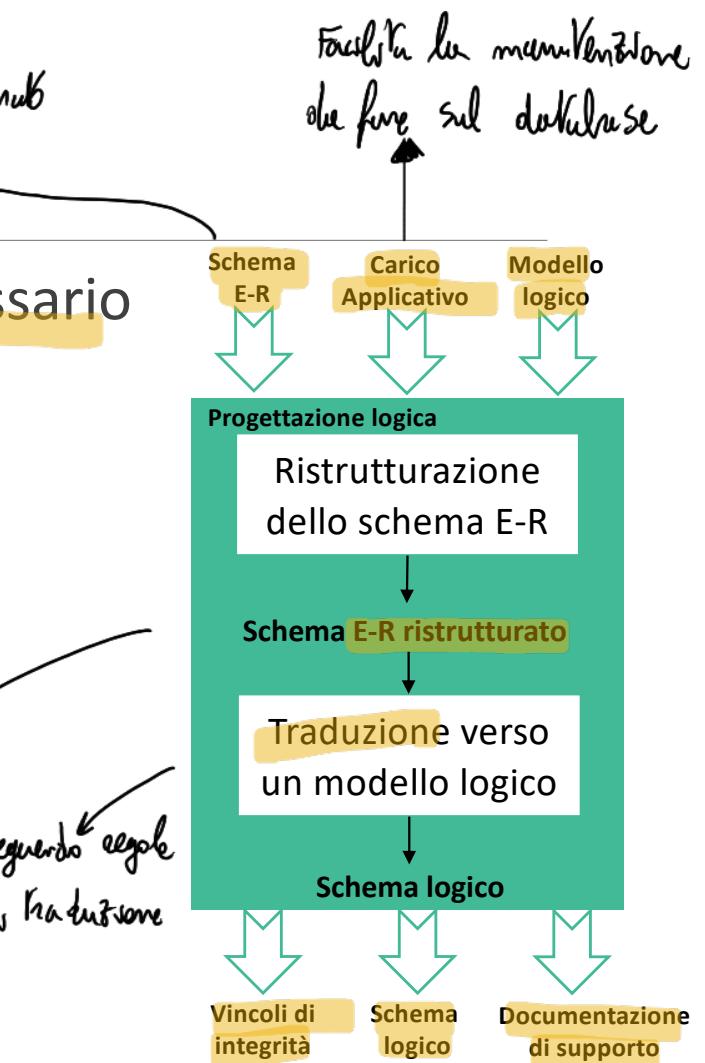
Progettazione logica

Prima di operare la traduzione si rende necessario ristrutturare lo schema concettuale:

Per semplificare la traduzione

Per ottenere uno schema logico ottimizzato

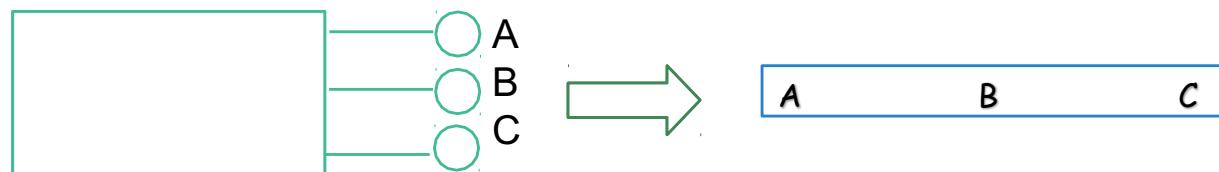
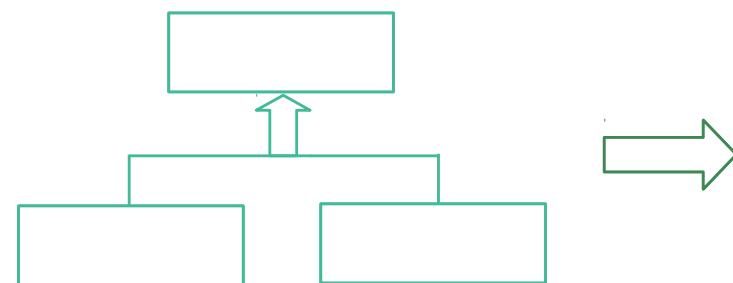
Ecco alcuni pesanti, interruttivi
dove ecc.



Ristrutturazione schema E-R

- La **ristrutturazione** dello **schema concettuale** si rende **necessaria** perché non tutti gli schemi E-R possono essere tradotti nel **modello logico**

- Es: è **impossibile** tradurre
 - Attributi composti/multivalore
 - Gerarchie/generalizzazioni



Ristrutturazione schema E-R

Al termine delle fasi di ristrutturazione e traduzione si otterrà:

- Lo schema logico finale nel modello scelto

- Per il modello relazionale l'insieme di tavole da istanziare
 - Ottimizzare lo schema (normalizzazione)

Non è detto che una volta trovato lo schema che questo sia ottimale. Ve quindi normalizzate.

- I vincoli di integrità

- Da apporre alle tavole
- Da riportare in appositi programmi

Analisi delle prestazioni su schema E-R

- ❑ Per confrontare tra loro diverse alternative di traduzione bisogna conoscere oltre allo schema anche, il cosiddetto “**carico di lavoro**” del DB, seppur in maniera approssimativa:
 - ❑ **Volume dei dati**
 - ❑ numero di occorrenze di ogni entità e associazione
 - ❑ dimensione di ciascun attributo
 - ❑ **Caratteristiche delle operazioni:**
 - ❑ Tipo di operazione (interattiva o batch)
 - ❑ Frequenza
 - ❑ Dati coinvolti (Entità e/o associazioni)
- ❑ Gli indicatori di costo che si utilizzano considerano quindi **due aspetti**
 - ❑ **spazio**: numero di istanze previste nel DB (spazio di memoria occupato)
 - ❑ **tempo**: numero di istanze (di entità e associazioni) visitate durante un'operazione sulla Base Dati

Schema di riferimento

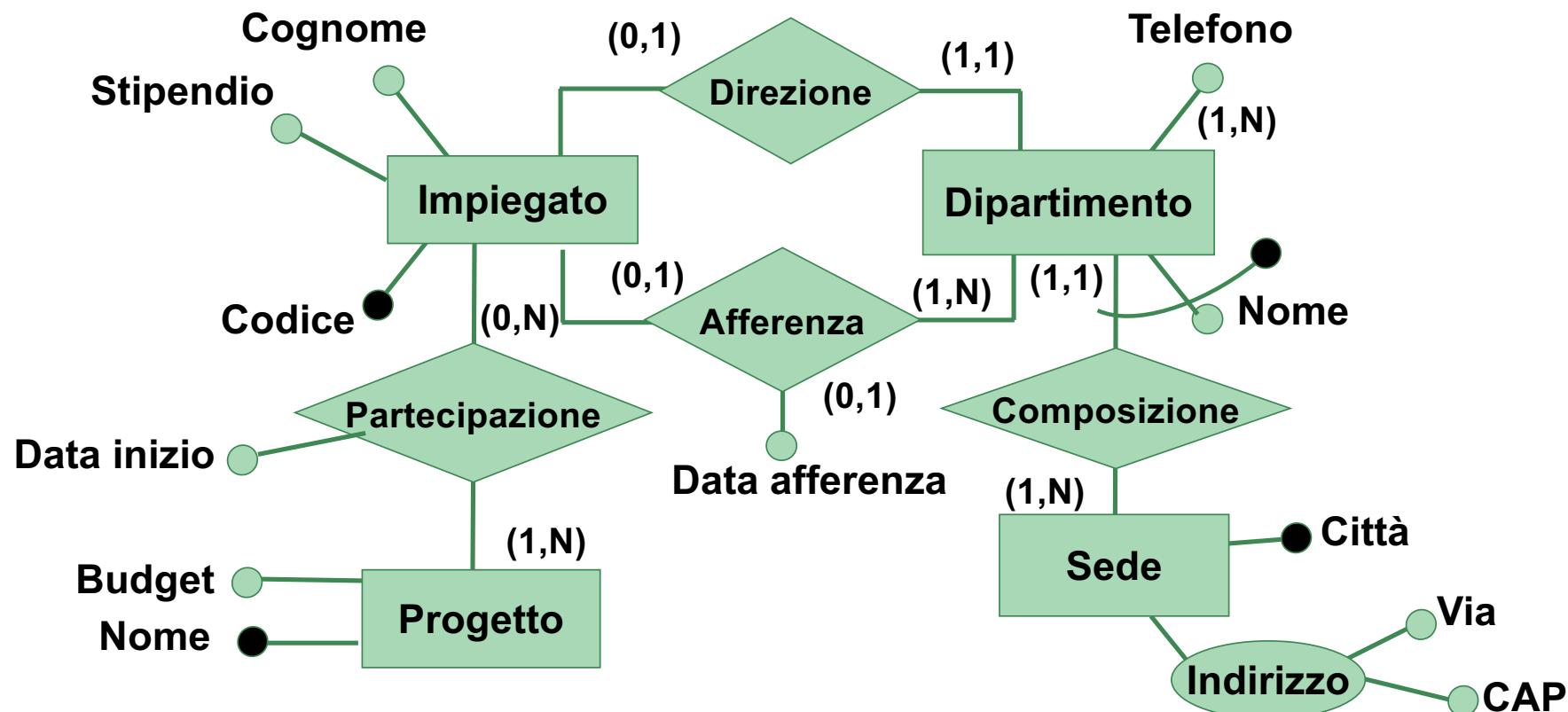


Tavola dei volumi

- Specifica il numero stimato di istanze per ogni entità (E) e associazione (R) dello schema
- I valori sono necessariamente approssimati, ma indicativi
 - I valori indicati con * sono derivati utilizzando i vincoli di cardinalità

Concetto	Tipo	Volume
Sede	Entità	10
Dipartimento	Entità	80
Impiegato	Entità	2000
Progetto	Entità	500
Composizione	Relazione	80
Afferenza	Relazione	1900
Direzione	Relazione	80
Partecipazione	Relazione	6000

arbithanzo
 ho cardinalità
 * 1 a 1 nell'
 ↗ olontan su sede
 ↗ 1 a 1 nel dipartimento

Es.: il # di occorrenze di **Composizione** è pari al # dei Dipartimenti, poiché dalla cardinalità si ricava che un dipartimento appartiene ad una sola sede

Es.: il # di occorrenze di **Afferenza** è poco inferiore al # di Impiegati, poiché dalla cardinalità si ricava che ci sono impiegati che non afferiscono a nessun dipartimento

Es.: Un Impiegato partecipa in media a 3 Progetti, si ha $2000 \times 3 = 6000$ occorrenze per **Partecipazione** e quindi $6000 / 500 = 12$ Impiegati in media su ogni Progetto

Leggermente
 inferiore a
 ampiamente perché
 hanno (0,1).

Tavola delle Operazioni

Operazione 1: Dato il codice di un **impiegato**, trova tutti i dati dell'**impiegato**, del dipartimento nel quale lavora e dei progetti ai quali partecipa

Nome operazione, tipo operazione, frequenza operazione

Operazione	Tipo (I/B)	Frequenza
Operazione 1	I	100 volte al giorno



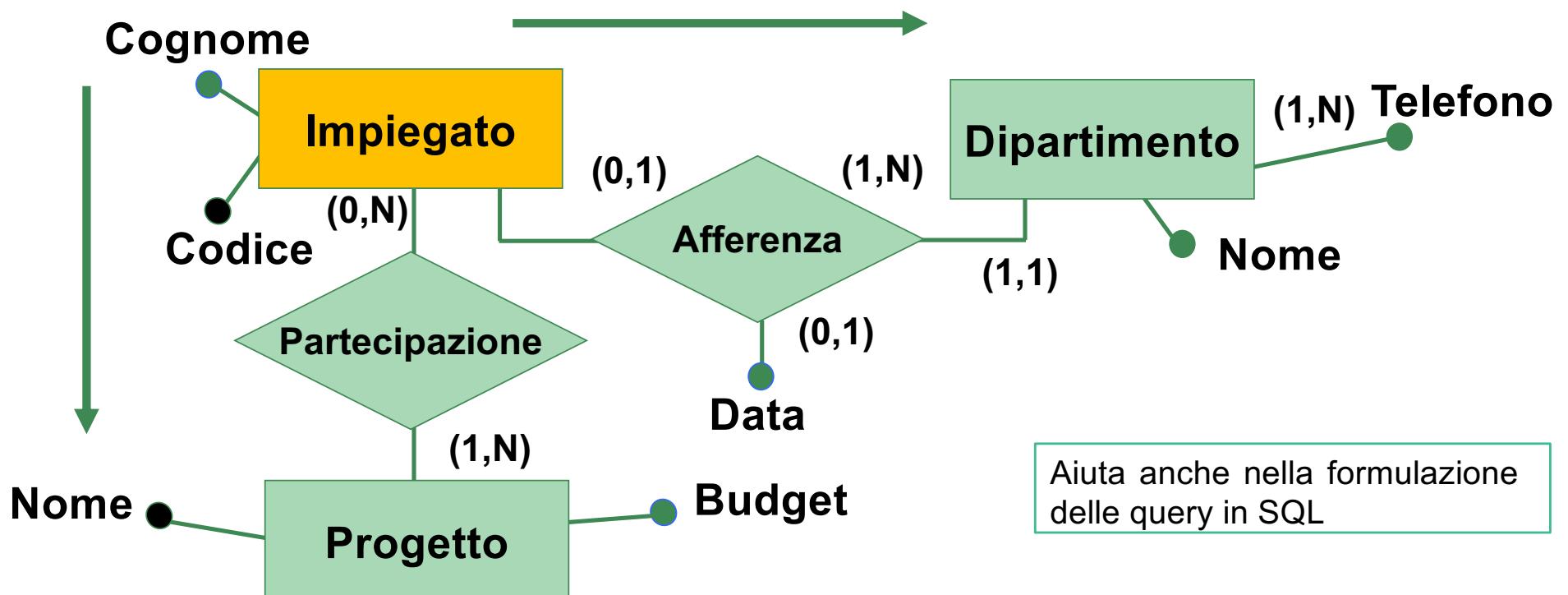
Intervallo / Batch: chiamati dall'utente o eseguita
in automatico

Schema che tiene conto di quale entità deve attraversare per raggiungere quell'operazione

Esempio di valutazione di costo

- Si costruisce una **tavola degli accessi** basata su uno **schema di navigazione**
- Lo **schema di navigazione** è la parte dello schema E/R interessata dall'operazione, estesa con delle frecce per indicare in che modo avviene l'operazione "**naviga**" i dati
- Per individuare le entità/associazioni interessate dall'operazione, si considera:
 - Le entità identificate esternamente inglobano le relative associazioni (che quindi "non contano")
 - Le associazioni inglobano gli identificatori delle entità partecipanti
 - Le entità figlie inglobano l'identificatore dell'entità genitore

Esempio di schema di navigazione



Esempio di tavola degli accessi

- Per ogni entità e associazione interessata dall'operazione, riporta il numero di istanze interessate, e il tipo di accesso (L: lettura; S: scrittura)
- Il numero delle istanze si ricava dalla tavola dei volumi mediante semplici operazioni
- Ad es.: in media ogni impiegato partecipa a $6000/2000 = 3$ progetti) STIMA INIZIALE

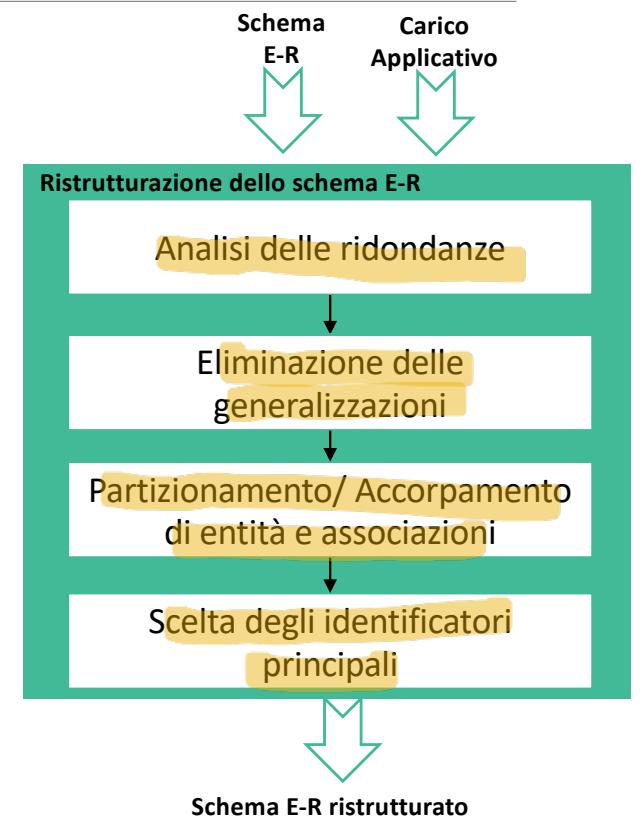
Concetto	Costrutto	Accessi	Tipo
Impiegato	Entità	1	L
Afferenza	Relazione	1	L
Dipartimento	Entità	1	L
Partecipazione	Relazione	3	L
Progetto	Entità	3	L

Si accede ad 1 occorrenza di **Impiegato**, poi si accede ad 1 occorrenza di **Afferenza** (ogni impiegato afferisce al più ad un dipartimento) e tramite quest'ultima ad 1 occorrenza di **Dipartimenti**

Per conoscere i Progetti a cui partecipa si accede a 3 occorrenze di **Partecipazione** e tramite questa a 3 occorrenze di **Progetto**

Ristrutturazione di schemi E-R

- La fase di ristrutturazione di uno schema E-R si può suddividere in una serie di passi da effettuare in sequenza



Ristrutturazione schema E-R: Passi

Eliminazione di attributi multivalue e composti

- si sostituiscono attributi multivalore e composti **con costrutti traducibili in relazionale**

Scelta identificatori primari

- Selezione dell'**identificatore primario per le entità che ne hanno più d'uno**

Eliminazione delle generalizzazioni

- Si sostituiscono le generalizzazioni dello schema **con altri costrutti**

Partizionamento/accorpamento di entità e relazioni

- Sulla base di **valutazioni di convenienza**

Analisi delle ridondanze: Una ridondanza in uno schema E-R **è una informazione significativa ma derivabile da altre**

- in questa fase si decide se **eliminare le ridondanze** eventualmente presenti o **mantenerle (o anche di introdurne di nuove sulla base di analisi delle prestazioni del sistema)**

Analisi delle ridondanze

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

Vantaggi

semplificazione delle interrogazioni

Svantaggi

appesantimento degli aggiornamenti

maggiore occupazione di spazio

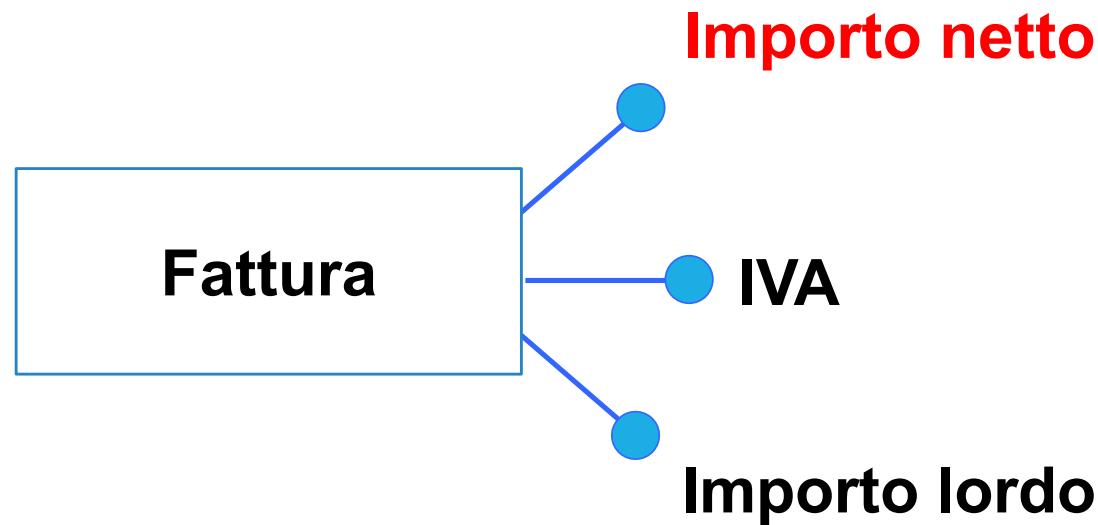
Casi più frequenti di ridondanza

attributi derivabili:

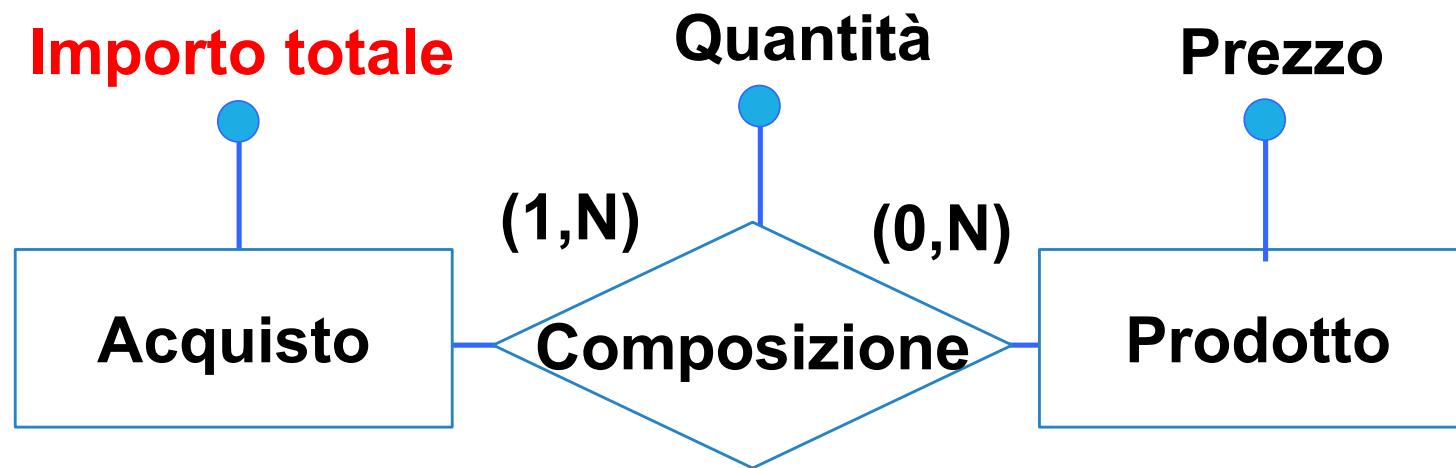
- da altri attributi della stessa entità (o relazioni)
- da attributi di altre entità (o relazioni)

relazioni derivabili dalla composizione di altre (più in generale: cicli di relazioni)

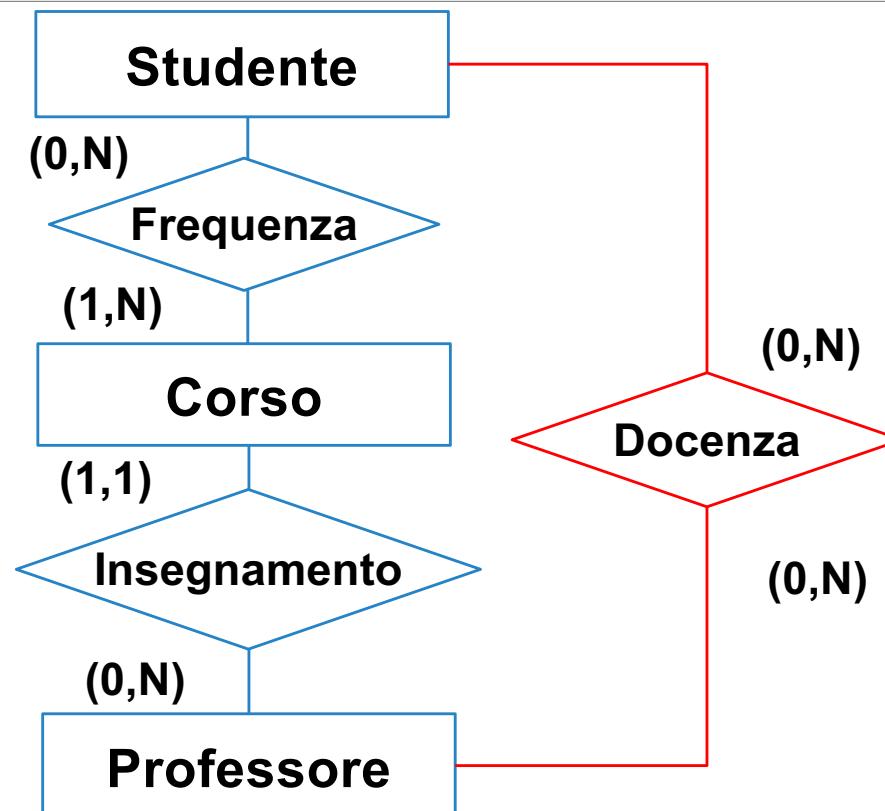
Attributo derivabile



Attributo derivabile da altra entità



Ridondanza dovuta a ciclo



Analisi di una ridondanza

- Si considerano innanzitutto le operazioni influenzate dalla ridondanza, considerando anche le loro frequenze di esecuzione...

Tavola dei volumi

Concetto	Tipo	Volume
Città	E	200
Persona	E	1 000 000
Residenza	R	1 000 000

Operazione 1: memorizza una nuova persona con la relativa città di residenza

Operazione 2: stampa tutti i dati di una città (incluso il numero di abitanti)



Numero abitanti

L>L_m extend until has 4 bytes per 200, val <1 Kb. Poco

Tavola delle operazioni

Operazione	Tipo	Frequenza
Op. 1	I	500 al giorno
Op. 2	I	2 al giorno

Were you surprised by me
when you took me home?

Analisi di una ridondanza: due scenari

Con ridondanza

Concetto	Costrutto	Accessi	Op
Persone	Entità	1	S
Residenza	Relazione	1	S
Città	Entità	1	L
Città	Entità	1	S

Operazione 1

La scrittura ha costo doppio
perché più onerosa

Senza ridondanza

Concetto	Costrutti	Accessi	Op
Persone	Entità	1	S
Residenza	Relazione	1	S

Con ridondanza

Concetto	Costrutto	Accessi	Op
Città	Entità	1	L

Operazione 2

Lettura
di una riga

Senza ridondanza

Concetto	Costrutti	Accessi	Op
Città	Entità	1	L
Residenza	Relazione	5000	L

Deve contenere tutte le righe associate a quella città

Analisi di una ridondanza: due scenari

- È importante considerare la frequenza delle operazioni:
- Con ridondanza:**
 - Operazione 1: 1500 accessi in scrittura (500×3 Scritture) e 500 accessi in lettura al giorno
 - Operazione 2: 2 accessi in lettura al giorno (trascutabile)
- Totale: 3500 accessi al giorno** (*Contiamo doppi gli accessi in scrittura*)

Analisi di una ridondanza: due scenari

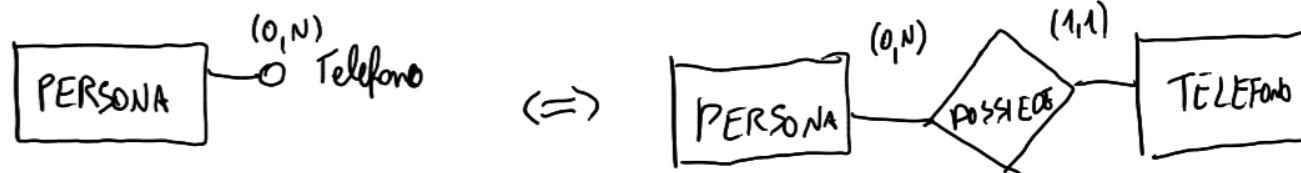
Senza ridondanza:

- Operazione 1: 1000 (500×2 Scritture) accessi in scrittura al giorno
- Operazione 2: 10000 accessi in lettura al giorno
($\text{NumPersone}/\text{NumCittà} * 2$ Frequenza al giorno)
- Totale: 12000 accessi al giorno (Contiamo doppi gli accessi in scrittura)

Si decide pertanto di mantenere la ridondanza

In generale si possono anche fare considerazioni sullo spazio in più richiesto

ES:



Attributi multivalore e composti

- ❑ **eliminazione di attributi multivalore / composti**
- ❑ non direttamente traducibili nello schema logico (violazione prima forma normale)
- ❑ **attributo composto:**
 - ❑ sostituito con tanti attributi semplici quanti sono gli attributi componenti
 - ❑ processo reiterabile
- ❑ **attributo multivalore:**
 - ❑ eliminato introducendo una nuova Entità, legata da una associazione uno a molti all'Entità originale

Scelta identificatori principali

- Per procedere nella traduzione nel modello relazione si deve individuare l'**identificatore principale** di ogni entità
- Perché su di essi andranno realizzati i legami tra entità diverse
 - Ossia le traduzioni delle relazioni
- Perché i sistemi di basi dati definiscono sulle chiavi primarie, a livello fisico, gli indici di accesso alle informazioni
- I due aspetti impongono di scegliere, in presenza di più identificatori, quello da considerare principale

Criteri di scelta

- I criteri di scelta si basano sul rispetto del principio di unicità o su valutazioni dettate dall'efficienza
- Criterio basato sul rispetto del principio di unicità:
 - Non si possono scegliere come principali quegli identificatori basati su attributi che ammettono **valori nulli**
 - Siffatti identificatori non garantiscono l'accesso a tutte le occorrenze dell'entità a cui si riferiscono
 - Non sono chiavi primarie



Criteri di efficienza

- I criteri di scelta basati sull'efficienza sono:
 - L'identificatore composto dal minor numero di attributi è da preferire agli altri
 - Perché il DBMS genera per essi indici più ridotti
 - Che occupano meno spazio
 - Che vengono gestiti più efficientemente
 - Un identificatore usato dal maggior numero di operazioni è da preferire agli altri
 - Perché le operazioni possono trarre vantaggio dal fatto di utilizzare proprio gli indici creati dal DBMS

Gli identificatori da aggiungere

Se nessuno degli identificatori soddisfa i criteri indicati allora è necessario introdurre campi che assumano il ruolo di chiave primaria:

- Un identificatore numerico (id)
 - Generato automaticamente (conteggio degli inserimenti)
- Un codice
 - Generico
 - Numero di serie
- Mnemonico o “parlante”
- Codice fiscale (generato da parti di più campi)

Gli identificatori non principali

Nella documentazione devono essere riportati tutti gli altri identificatori che chiameremo **secondari**

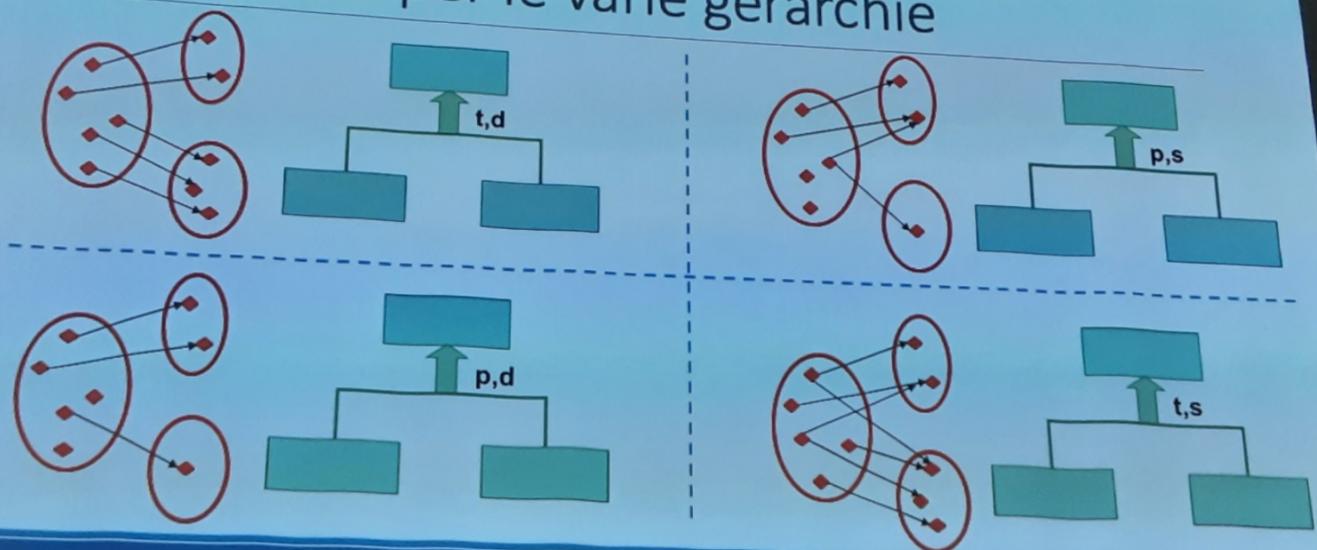
In particolare vanno segnalati quelli sui quali sono definite delle **operazioni**

- In fase di progettazione fisica si potranno attivare ulteriori indici detti anch'essi **secondari**
- Per rendere più efficiente l'accesso ai dati
- Da usare in alternativa agli indici primari

Eliminazione delle gerarchie

- ❑ Il modello relazionale non può rappresentare direttamente le generalizzazioni
- ❑ Si eliminano perciò le gerarchie, sostituendole con entità e associazioni
- ❑ Esistono tre possibili scenari:
 - ❑ Accorpamento delle entità figlie della generalizzazione nel genitore (**collasso verso l'alto**)
 - ❑ La generalizzazione assorbe le specializzazioni
 - ❑ Accorpamento del genitore della generalizzazione nelle entità figlie (**collasso verso il basso**)
 - ❑ Le specializzazioni ereditano la generalizzazione
 - ❑ Sostituzione della generalizzazione con relazioni (**traduzione indipendente**)
 - ❑ La generalizzazione si trasforma in tante relazioni uno a uno che legano l'entità padre con tutte le entità figlie

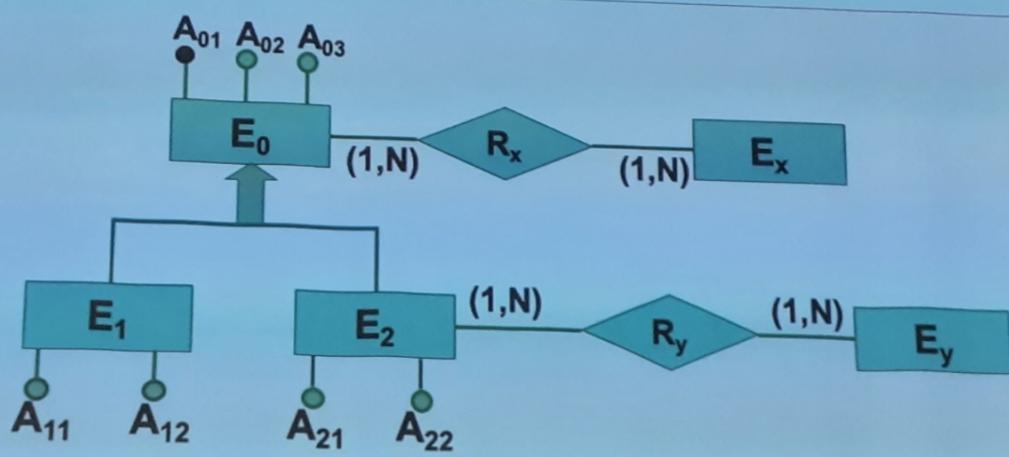
Notazioni per le varie gerarchie



PROF. DIOMAIUTA CRESCENZO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

A seconda delle strutture si usano operazioni diverse

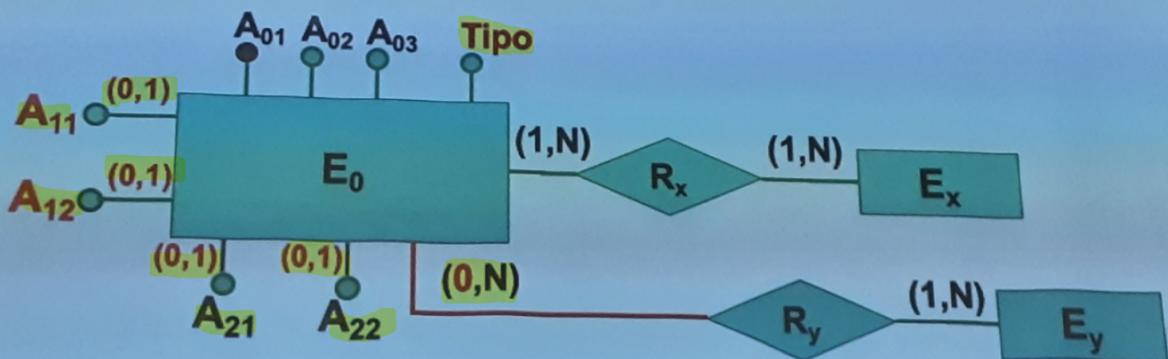
Schema di riferimento



PROF. DIOMAIUTA CRESCENZO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

1. Accoppare nel genitore

□ Accorpamento delle sottoclassi nella superclasse



PROF. DIOMALTA CRESCENTO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

Inghetto i figli nel padre, che prende gli attributi delle entità figlie, ma
è compreso la cardinalità, perché se in E_0 considero occorrenza di E_1 , non
ho attributo di E_2 . Assumeremo valori NULL. Assumeremo anche relazione dell'entità figlia
ma come cardinalità minima ho 0, perché potrei gestire occorrenza di E_1 e non E_2 .
Ho in questo caso un selettore che è attributo tipo.

Collasso verso l'alto: osservazioni

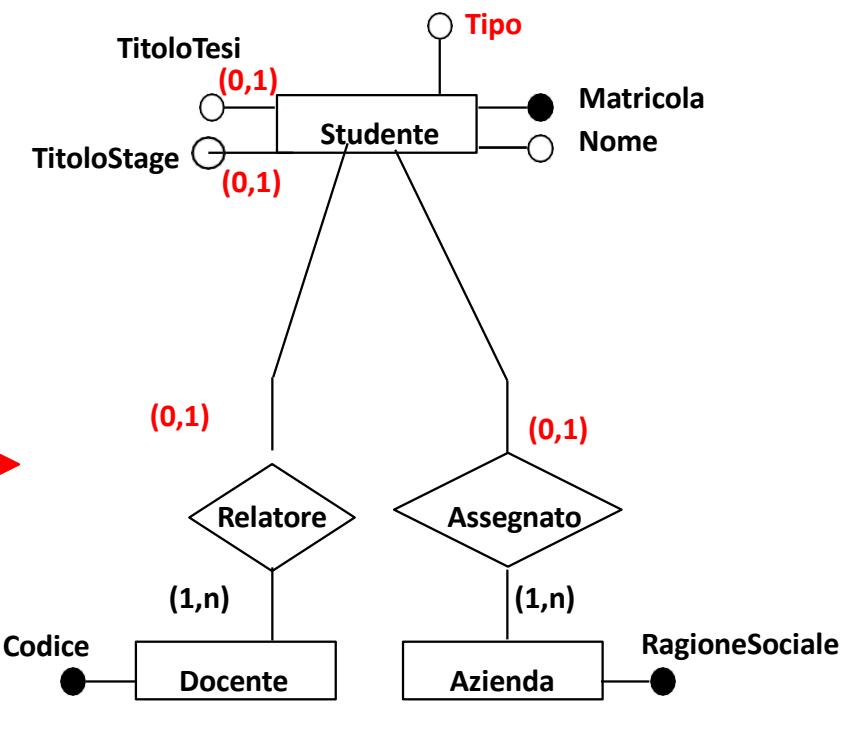
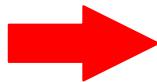
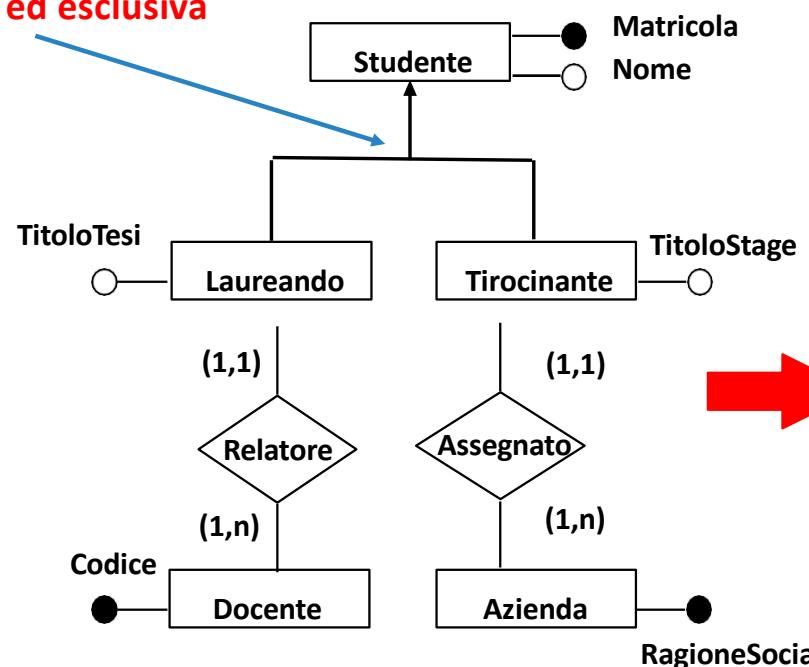
- La soluzione proposta è particolarmente consigliabile quando i vincoli di gerarchia sono di tipo **totale-sovraposta**.
- Si evitano, così, ridondanze dovute a **sovraposizioni**. Questa soluzione può andare bene anche con gerarchie **parziali**.

In genere lei è sempre applicabile, ma non sempre consigliata.

PROF. DIOMALTA CRESCENTO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

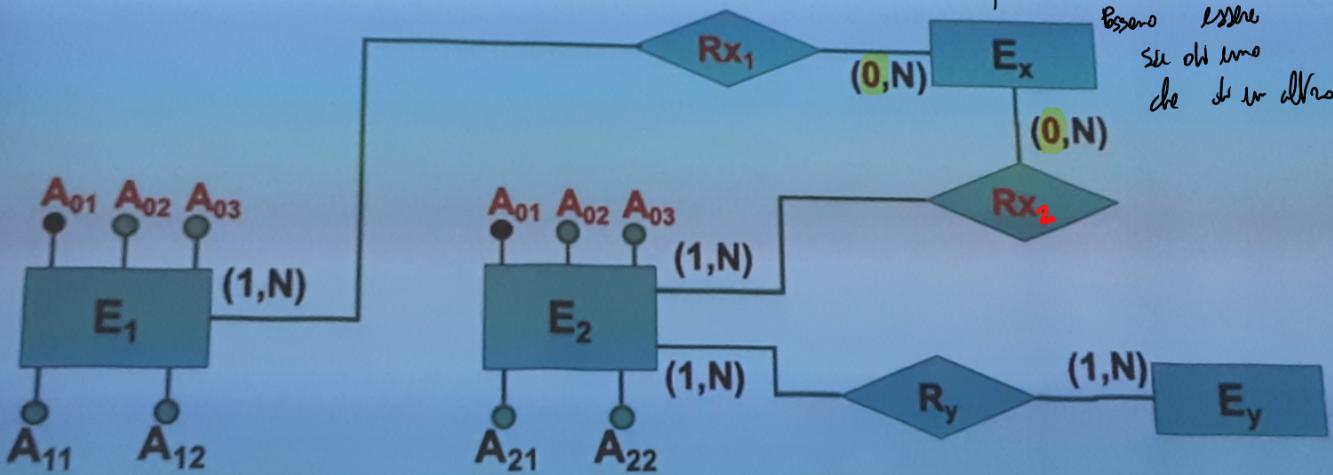
1. Accorpare nel genitore: Esempio

Parziale ed esclusiva



2. Accorpate nelle figlie

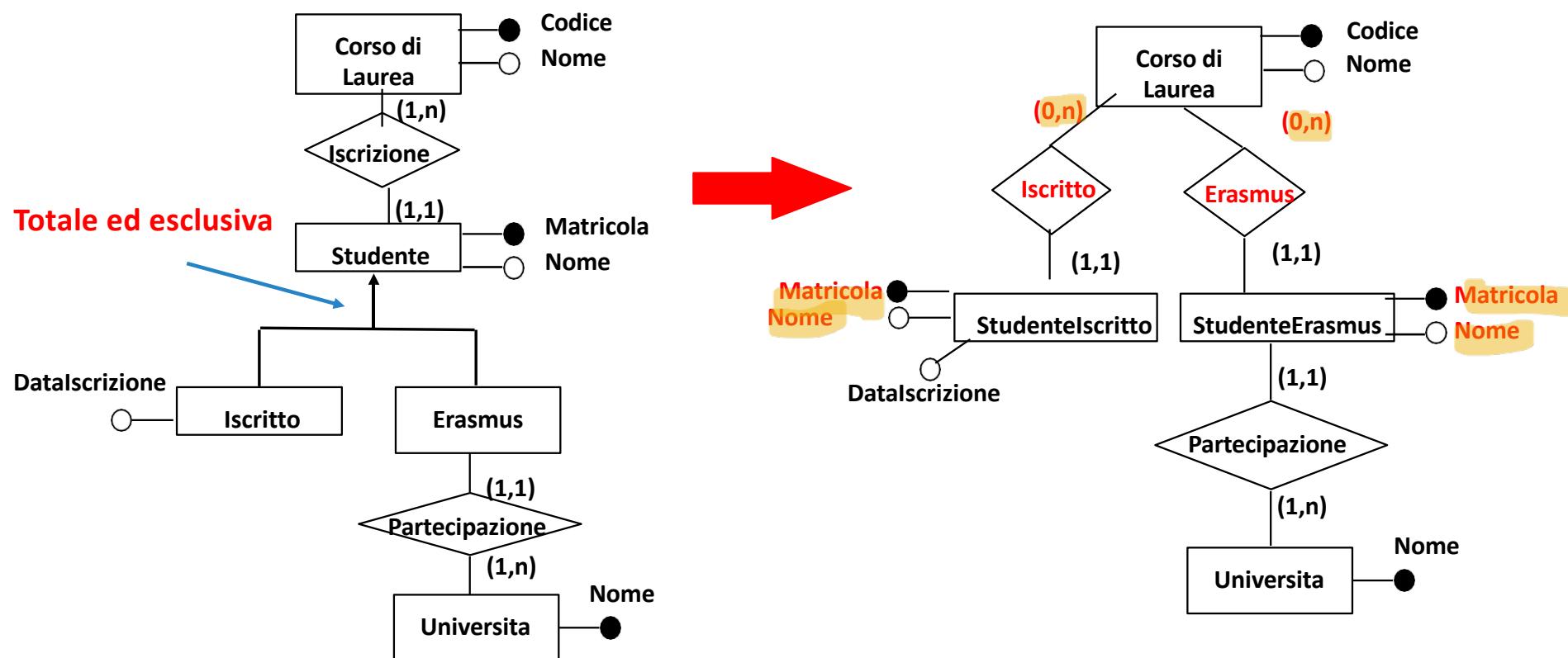
□ Accorpamento della superclasse nelle sottoclassi



Porta attributi del padre sue figlie e apre le relazioni su entrambe

Note: Cambia la cardinalità da punto d'entità connesso perché passa per riferimento solo ad un'entità particolare.

2. Accorpare nelle figlie: Esempio



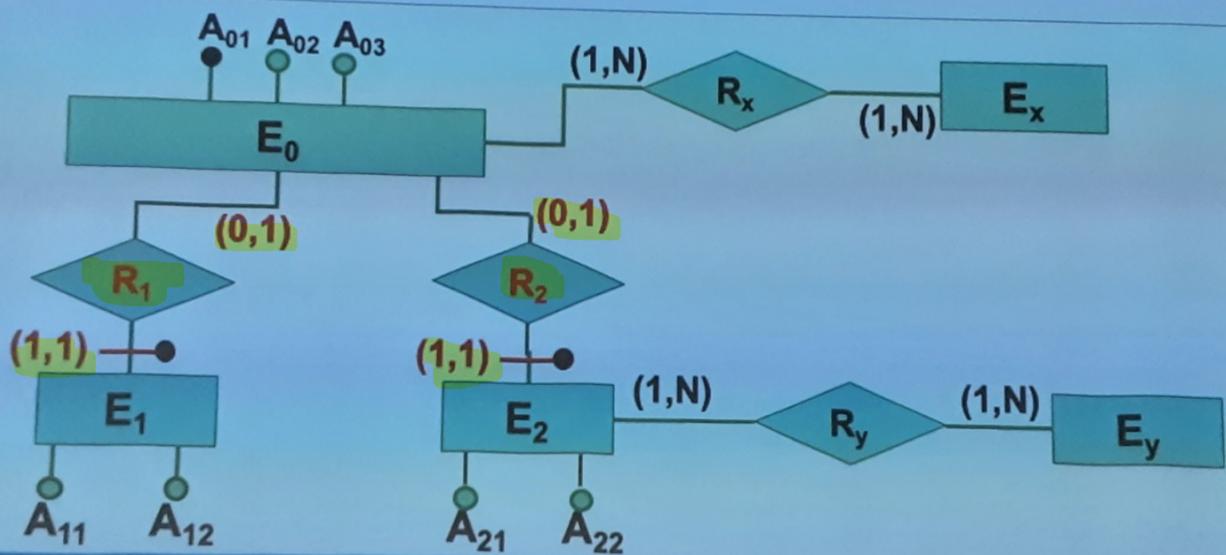
Collasso verso il basso: osservazioni

- Questa soluzione è consigliabile in presenza di generalizzazioni **totali-disgiunte**
- **Se la generalizzazione non è totale non si può applicare**
 - Poiché non si sa dove mettere le istanze di E che non sono né in E1, né in E2
- **Se la generalizzazione non è totale introduce ridondanza**
 - una certa istanza può essere sia in E1 che in E2, e quindi si rappresentano due volte i valori degli attributi ereditati da E
- Per le associazioni che si trasportano sulle entità figlie è necessario garantire che siano ancora verificati i vincoli di cardinalità e/o di foreign key

PROF. DIONISIO CRESCENZO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

Se è possibile significare che c'è solo chi non so dove mettere

3. Sostituzione della generalizzazione con associazioni



PROF. DIONISIO CRESCENZO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

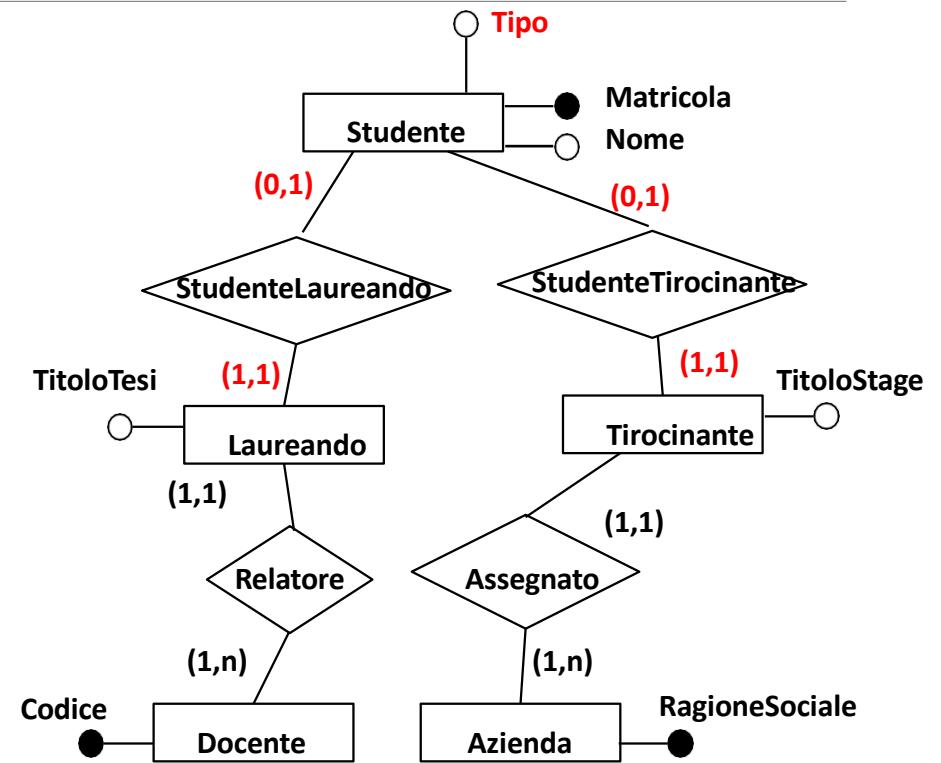
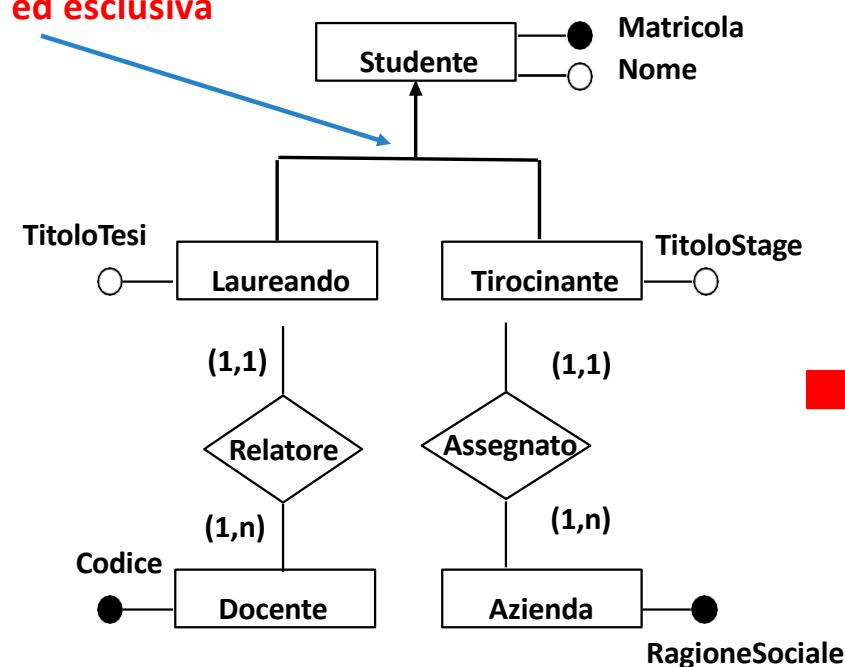
Entità generatore ha la stessa com'è, come le figure. Ma aggiunge relazione/associazione. Cardinalità sempre $(0,1)$ e $(1,1)$. E_1 è sicuramente figlia di E_0 : è sicuramente nel genitore. Lo 0 anche che un padre non è per forza uno dei figli (E_0 può essere E_2 e non E_1). Per identificazione esterna: E_1 è identificabile con identificazione del padre gli attributi del padre entrano anche del figlio. Applicabile sempre.

Sostituzione della generalizzazione con associazioni: Osservazioni

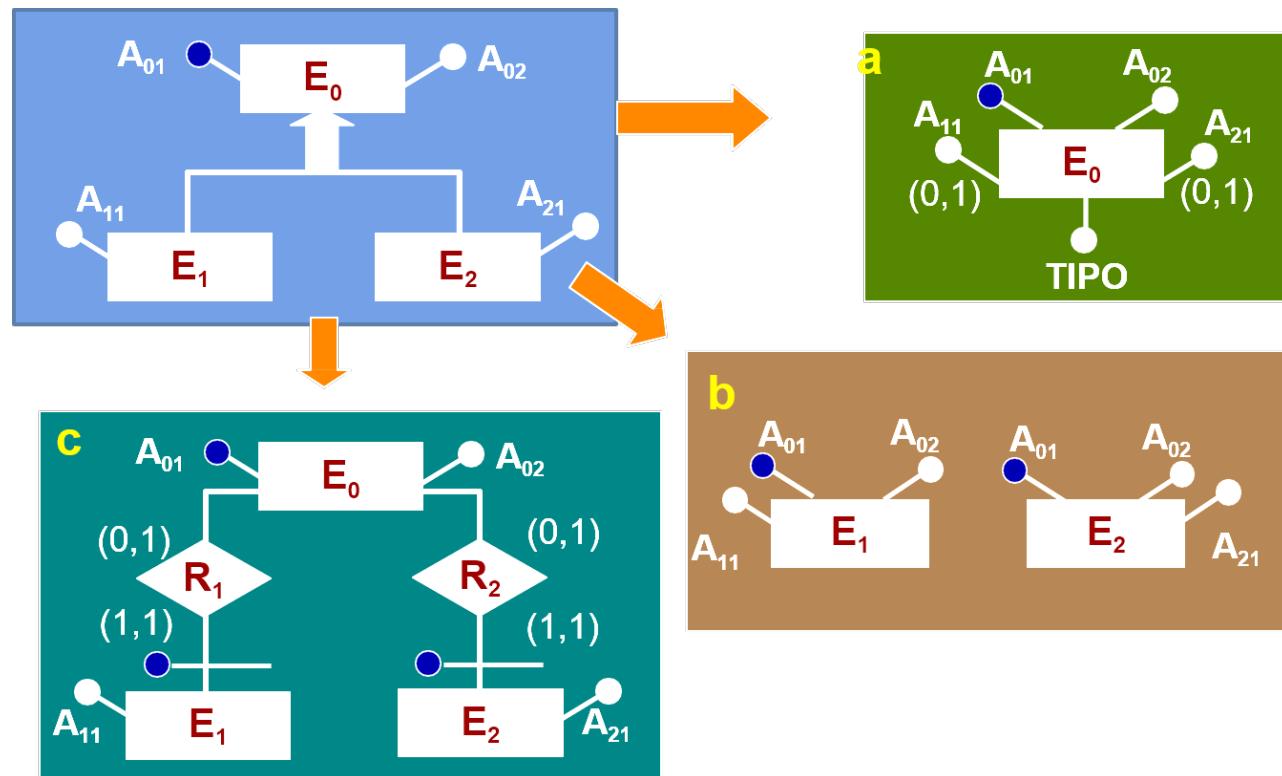
- Questa soluzione è quella più intuitiva e sempre applicabile. Tuttavia è consigliabile dove vi è la necessità di mantenere separate le superclassi e le sottoclassi, ovvero quando i vincoli sono di tipo **parziale-disgiunta**

3. Sostituzione della generalizzazione con relazioni : Esempio

Parziale ed esclusiva



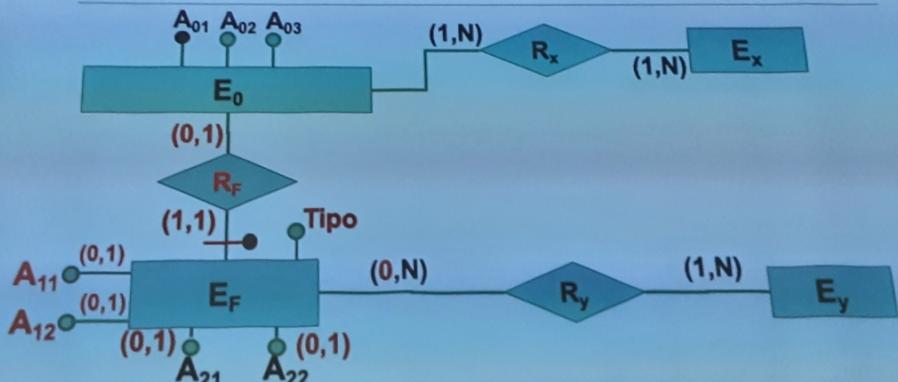
Riepilogo



In sintesi ...

1. L'alternativa (1) è conveniente quando le operazioni non fanno molta distinzione tra le occorrenze e gli attributi di E0, E1 ed E2 (accessi contestuali tra padri e figlie)
 2. L'alternativa (2) è possibile solo se la generalizzazione è totale (gli accessi alle figlie sono distinti)
 3. L'alternativa (3) è conveniente quando la generalizzazione non è totale (gli accessi alle entità figlie sono separati dagli accessi al padre)
- Sono anche possibili soluzioni "ibride", soprattutto in gerarchie a più livelli

4. Ristrutturazione ibrida

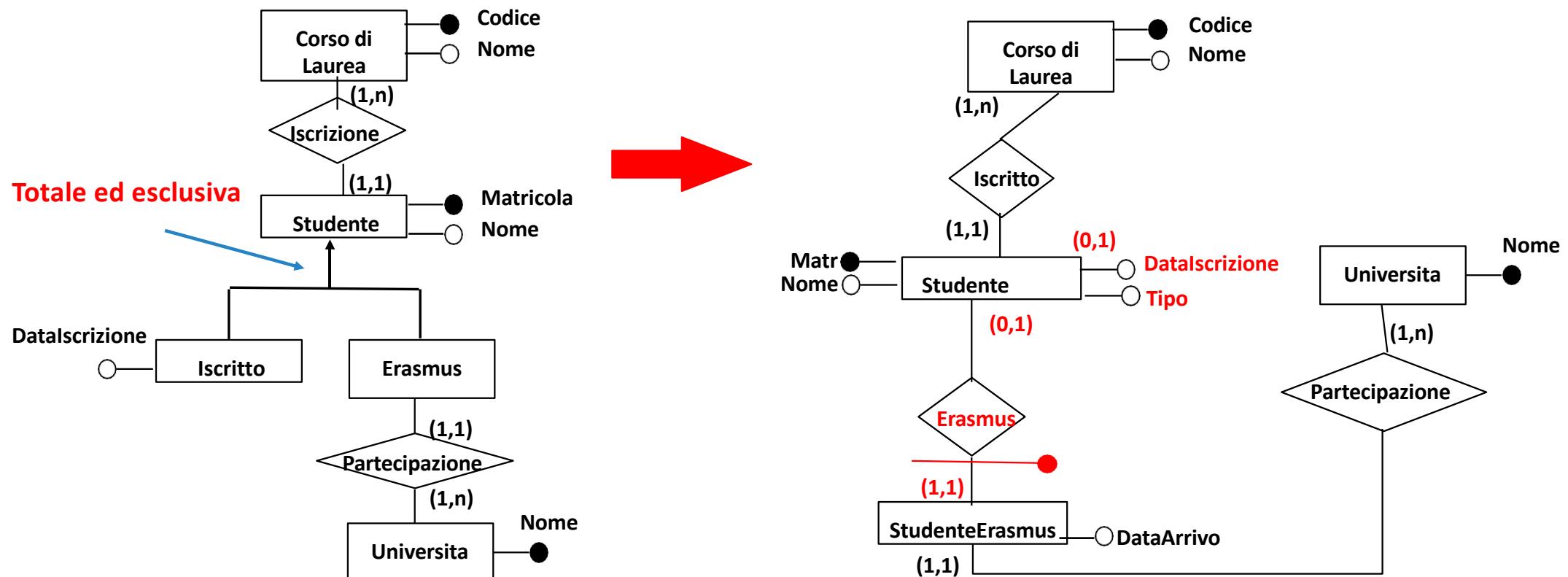


Ibrida: ho molti due entità figlie ma entrambe puote continuare a esistere.

Se ho 1000 righe padre che con 300 e 200 sono figlie, no queste.

500 continuano a esistere come solo padre.

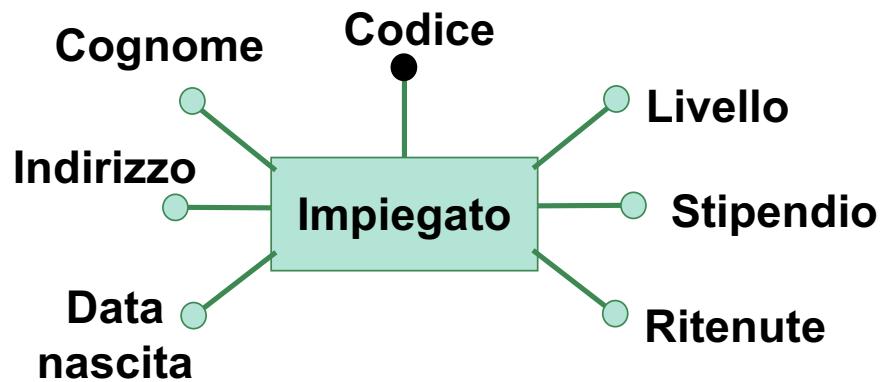
4. Ristrutturazione ibrida: Esempio



Partizionamenti e accorpamenti

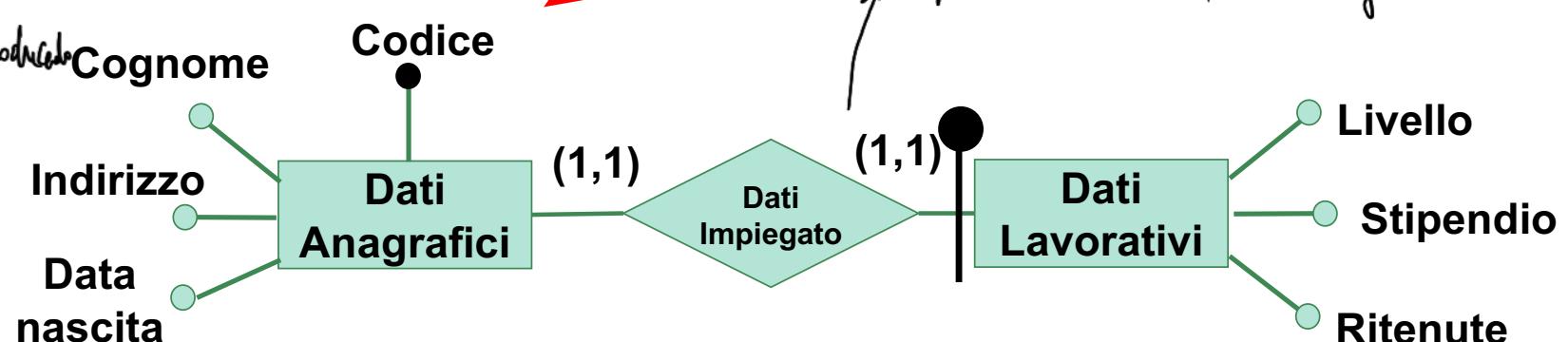
- ❑ È possibile ristrutturare lo schema accorpando o partizionando entità e associazioni
- ❑ Tali ristrutturazioni vengono effettuate per rendere più efficienti le operazioni in base al principio già visto, ovvero:
 - ❑ separando attributi di un concetto che vengono acceduti separatamente
 - ❑ raggruppando attributi di concetti diversi acceduti insieme
- ❑ I casi principali sono:
 - ❑ partizionamento "verticale" di entità: si partizionano gli attributi (*attributi*)
 - ❑ partizionamento "orizzontale" di associazioni: si partizionano le istanze (*associazioni*)
 - ❑ ad es.: storico e corrente (impieghi passati e impiego corrente)
 - ❑ possono cambiare i vincoli di cardinalità
- ❑ accorpamenti di entità e associazioni
 - ❑ per le entità: caso particolare di traduzione di associazioni uno a uno
- ❑ eliminazione di attributi multivalue

Partizionamento verticale di entità

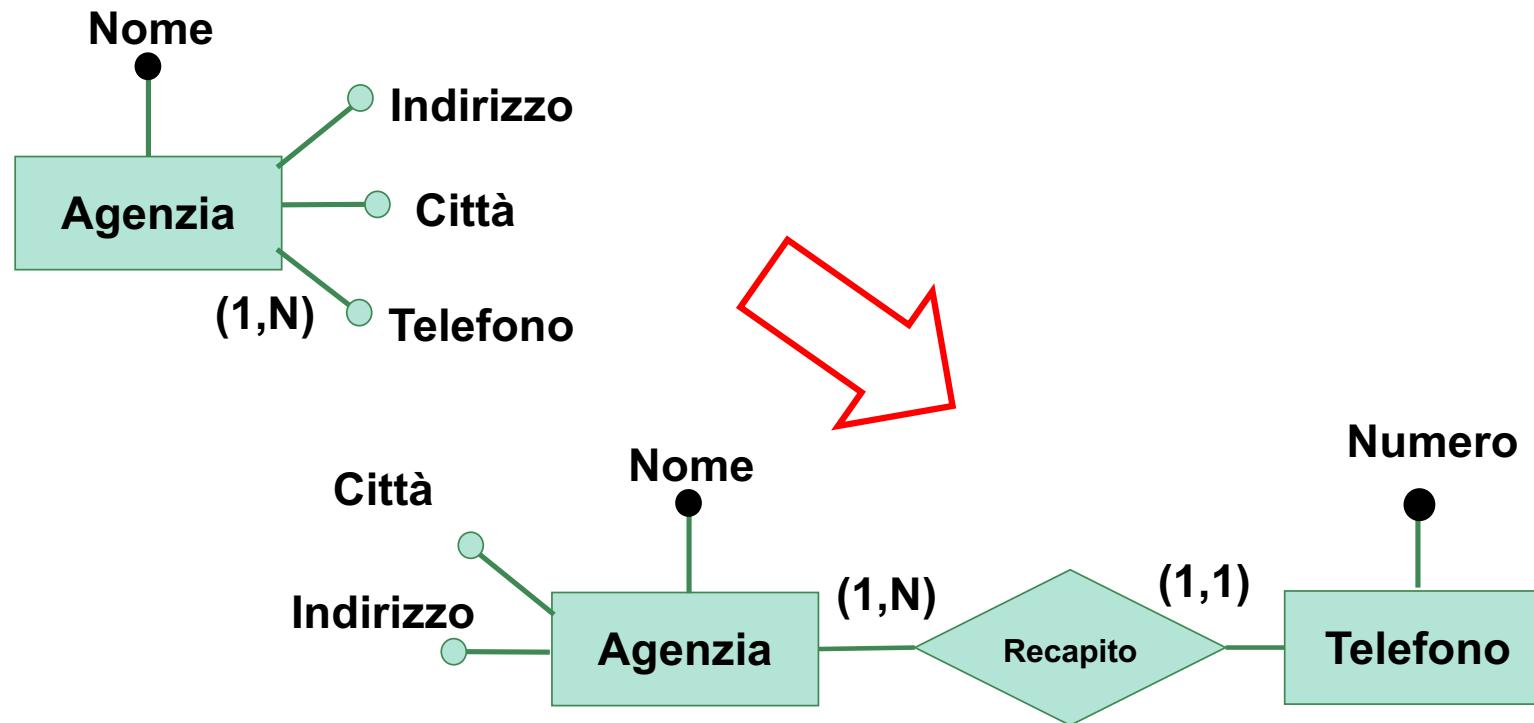


Problema: non un'unica entità può introdurre concetti separati e parlando di concetti separati succede spesso che possono separarsi posso partizionare: prima volta capo-dominio

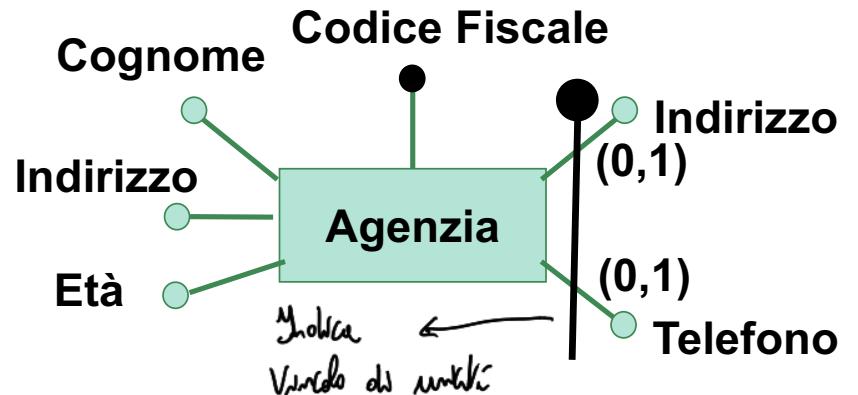
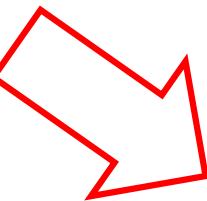
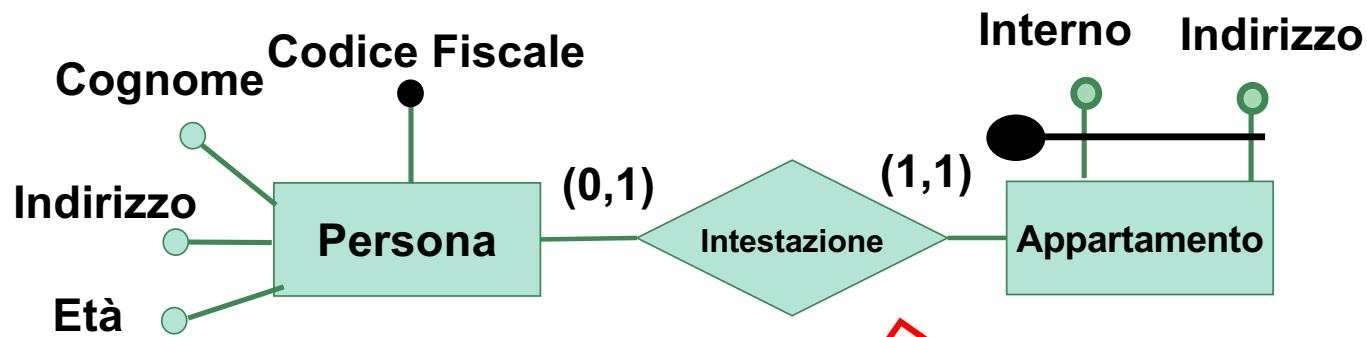
Si separano gli attributi in **gruppi omogenei dal punto di vista dell'uso**



Eliminazione attributo multivalore

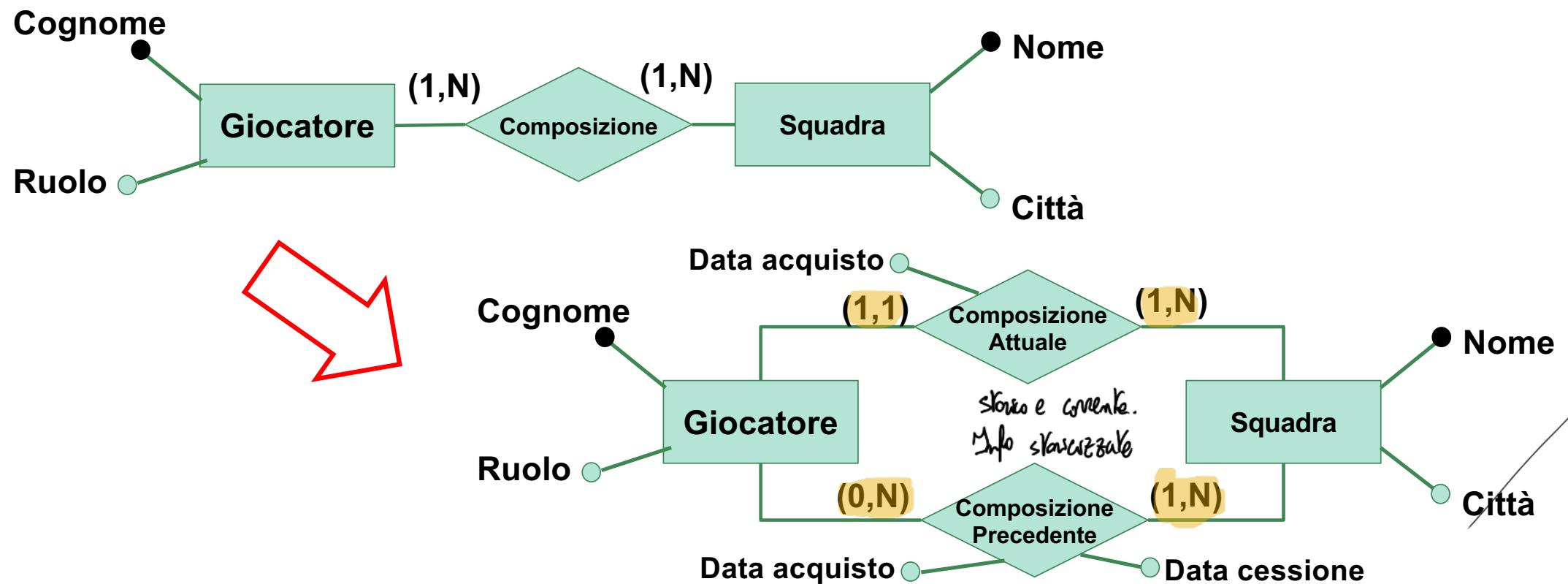


Accorpamento di entità



Comment: ho 2 entità e le accordo perché ho opzionalità.
Questa persona deve aggiungere appartamento.
Concetto separato non ha rapporto.

Partizionamento di associazione (orizzontale)



Traduzione di schemi E/R

- Al termine della ristrutturazione il progettista ha uno schema E-R equivalente
- Ristrutturato sulla base delle esigenze del modello logico
- Può quindi procedere alla traduzione nel modello relazionale e generare lo schema della Base Dati



Traduzione di schemi E/R

- Per non creare confusione tra relazione del modello E-R e relazione del modello relazionale, la traduzione avrà come risultato delle tabelle
- Per comodità di rappresentazione indicheremo le tabelle secondo il loro schema:

Professore

Nome	Corso	Facoltà
------	-------	---------



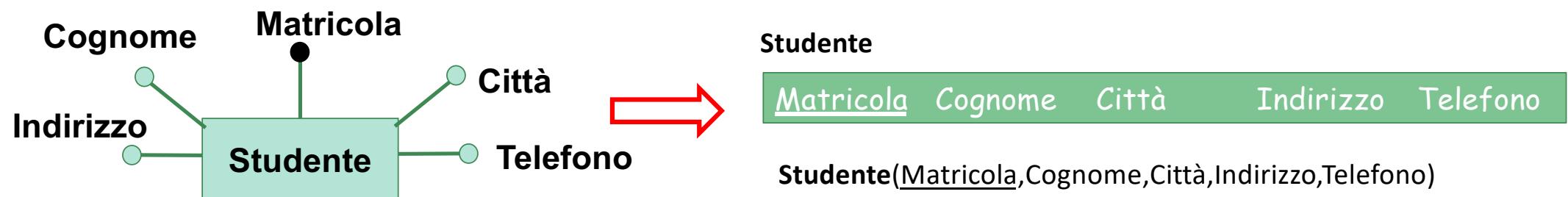
Professore(Nome,Corso,Facoltà)
↳ in genere si usa nome al plurale

Traduzione di schemi E/R

- In generale la traduzione nel modello relazionale trasforma:
 - Le entità in tabelle
 - Le relazioni in tabelle
- Tra le tabelle di relazioni e quelle delle entità collegate si determinano vincoli di integrità referenziali
 - Le tabelle delle relazioni inglobano gli identificatori di tutte le entità collegate
- La traduzione presenta problematiche diverse per le relazioni
 - Uno a uno
 - Uno a molti
 - Molti a molti

Traduzione di entità

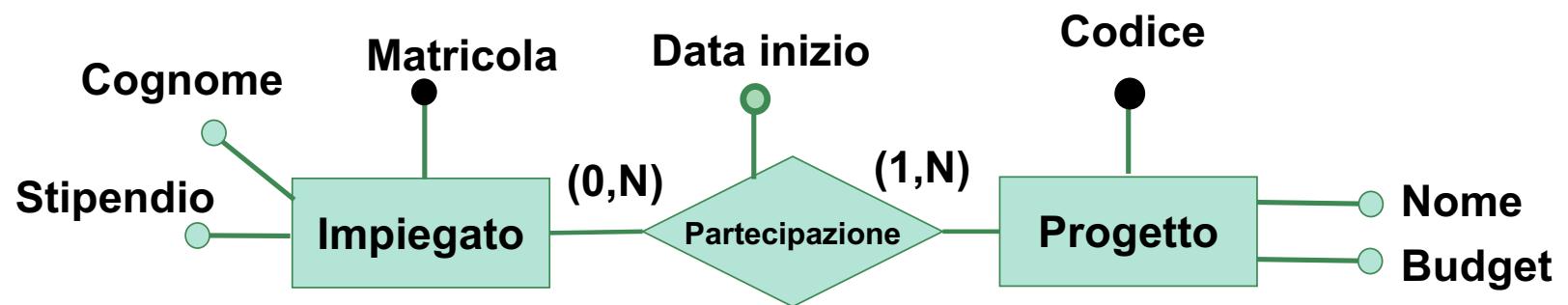
- La traduzione di una entità produce una tabella che ha:
 - Per nome il nome dell'entità (ma al plurale)
 - Come attributi (nomi delle colonne) gli attributi della entità
 - Per chiave l'identificatore principale della entità



La traduzione di relazioni Molti a Molti

- ❑ La traduzione di una relazione molti a molti produce una tabella che ha:
 - ❑ Per nome il nome della relazione (*ma al plurale*)
 - ❑ Come attributi
 - ❑ Gli attributi della relazione (nel caso ne abbia)
 - ❑ Gli identificatori di tutte le entità coinvolte
 - ❑ Tali identificatori formano la chiave della relazione
 - ❑ In alcuni casi formano una superchiave per cui va scelto il sottoinsieme che costituisce la chiave
 - ❑ Per aumentare la leggibilità degli schemi solitamente conviene ridenominare gli attributi degli identificatori

La traduzione di relazioni Molti a Molti

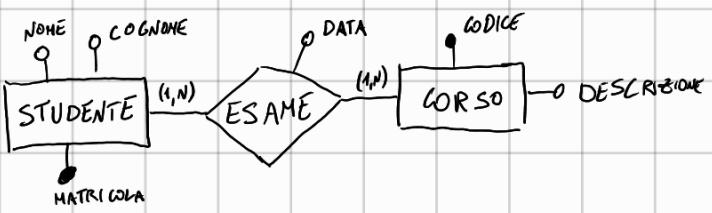


Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

Partecipazione(Matricola, Codice, DataInizio)

↳ doppio vincolo^{1/2} da registrazione referenziale



Students (Matricola, Cognome, Nome)

Corsi (Codice, descrizione)

Esame (Matricola, Codice, Data)

STUDENTI

MAT	COGNOME	NOME
A1022	ROSSI	MARCO
A1023	VERDI	PABLO

CORSI

CODICE	DESCRIZIONE
ING-INF01	SWBD
ING-INF02	ANALISI 1

Come rappresento che Marco Rossi ha sostenuto 2 prove di analisi?

In questo caso esempio di traduzione molti a molti non basta.

Dove aggiungere più attributi / dove rivolgersi all'utente che posso gestire

La traduzione di relazioni Molti a Molti

- Per migliorare la comprensione della tabella Partecipazione conviene ridenominare gli attributi presi dalle entità

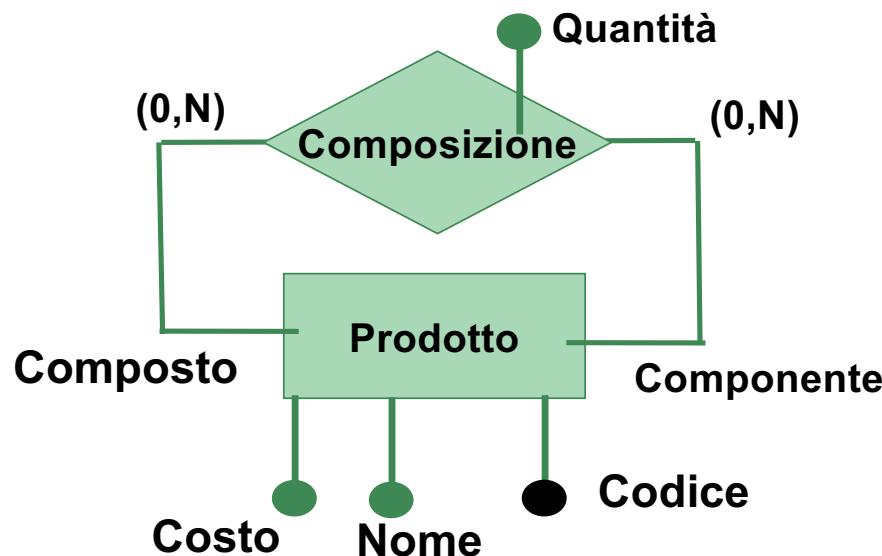
Partecipazione(Impiegato,Progetto,DataInizio)

- Tra le tabelle si vengono a stabilire i seguenti vincoli di integrità referenziale
 - Matricola in Partecipazione e (la chiave di) Impiegato**
 - Codice in Partecipazione e (la chiave di) Progetto**

Sempre cardinalità molti a molti

Relazioni ricorsive

- La ridenominazione degli attributi nella tabella corrispondente alla relazione è obbligatoria nei casi di relazioni ricorsive

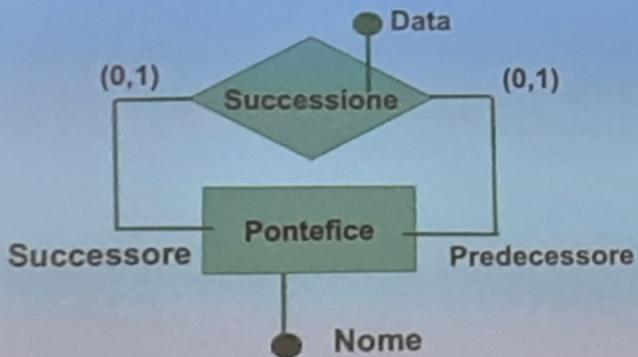


Prodotto(Codice, Nome, Costo)

Composizione(Composto, Componente, Quantità)

Composizione(Codice, Codice, Quantità)

Relazioni ricorsive



Pontefici(Nome)

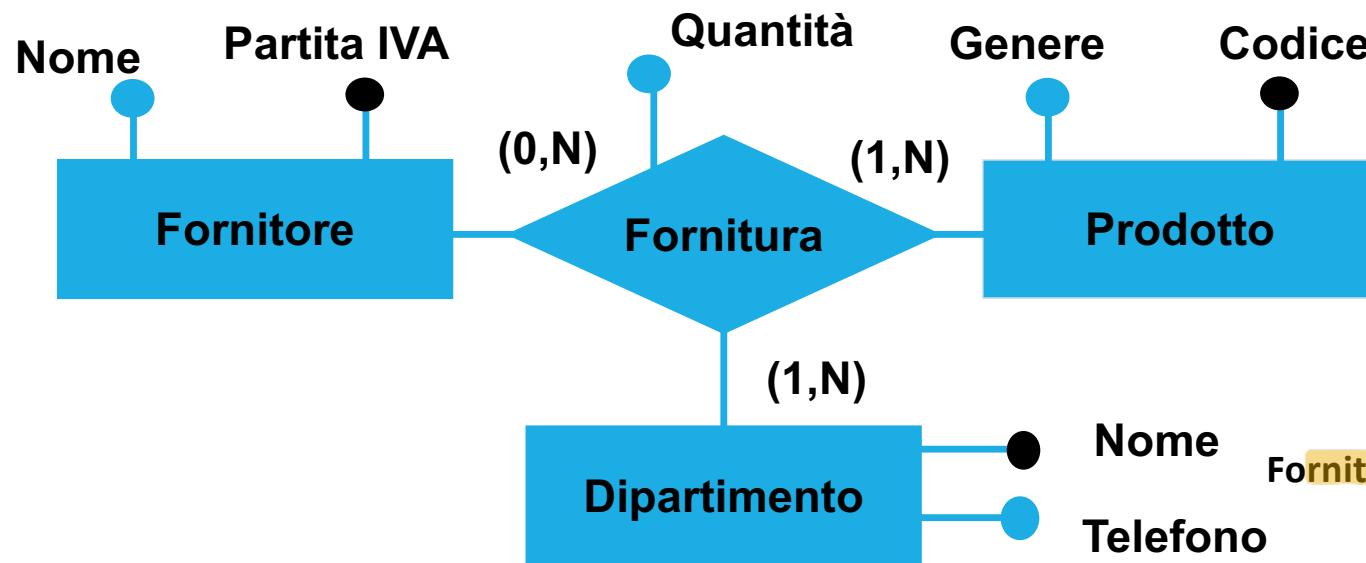
Successioni(Successore, Predecessore, Data)

PROF. GIOVANUTA CRESCENZO "SISTEMI WEB E BASI DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA

Si componibile come traduzione di una mappa a modelli

Relazioni n-arie (*sempre molti a molti*)

- Alle relazioni con più di due entità si applicano le stesse modalità di traduzione



Vedere che la Regola

Fornitore(Partita IVA, Nome)

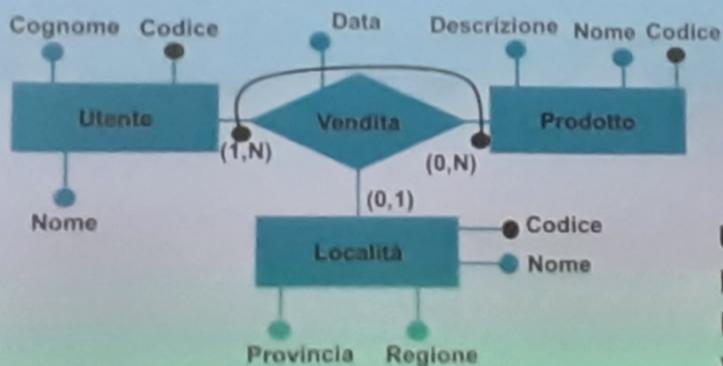
Prodotto(Codice, Genere)

Dipartimento(Nome, Telefono)

Fornitura(Fornitore, Prodotto, Dipartimento, Quantità)

Se queste 3 dicono tutto
non bastano dobbiamo trovare
altri valori altre

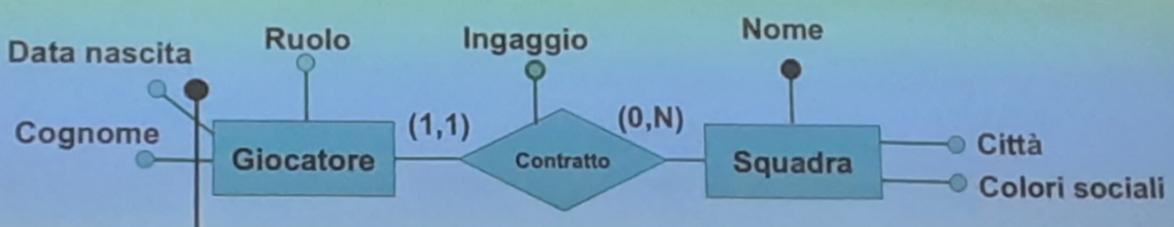
Relazioni n-arie



Utenti(Codice, Nome, Cognome)
Prodotti(Codice, Nome, Descrizione)
Località(Codice, Nome, Provincia, Regione)
Vendite(Utenti, Prodotti, Località, Data)

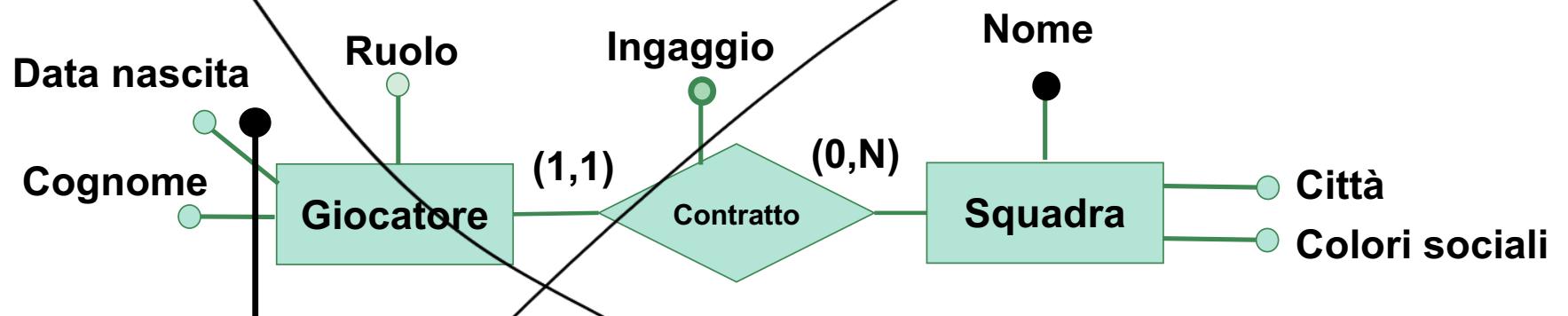
Località ha un ruolo che non influenza le vendite. Siccome utenti e prodotti non sono abbastanza per l'identificazione, ho centrato.

La traduzione di relazioni Uno a Molti



Un'associazione uno a molti dello schema concettuale si traduce in una relazione dello schema logico avendo lo stesso nome (ma al plurale) dell'associazione e per attributi gli attributi dell'associazione e gli identificatori delle entità coinvolte. L'identificatore dell'entità dal lato uno costituisce la chiave primaria della relazione, ed ogni identificatore ha un vincolo di integrità referenziale con il corrispondente attributo dell'entità da cui discende

La traduzione di relazioni Uno a Molti

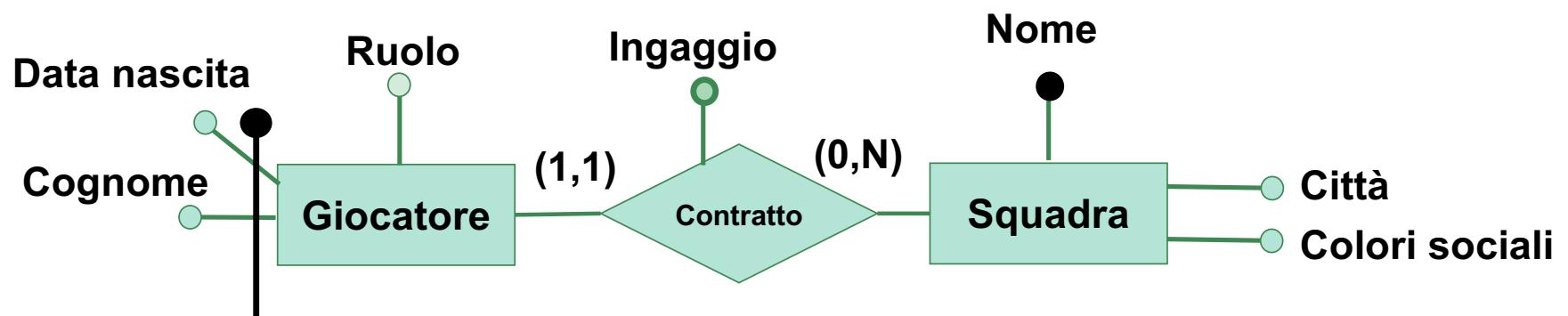


Le occorrenze della relazione R sono:

- Tante quante quelle della entità E se $x=1$
- Al massimo quelle della entità E se $x=0$

In entrambi i casi la chiave è chiave sia per le occorrenze di E che per quelle di R

La traduzione di relazioni Uno a Molti



Giocatore(Cognome, DataNascita, Ruolo)
Squadra(Nome, Città, ColoriSociali)
Contratto(Giocatore, DataNascitaGiocatore, NomeSquadra, Ingaggio)

Qui non ho scelto nome Squadra come identificatore univoco. L'ho come chiave primaria nel campo dell'entità a 1. Gli altri non sono necessari.
^{L>Cognome}

La traduzione di relazioni Uno a Molti

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

Squadra(Nome, Città, ColoriSociali)

Soluzione più compatta

- con vincolo di integrità referenziale fra Squadra in Giocatore e la chiave di Squadra
- Si noti che nel caso di cardinalità (0,1) la seconda soluzione presenterà dei valori nulli per gli attributi Squadra e Ingaggio della tabella Giocatore

Poiché le tabelle **Giocatore** e **Contratto** hanno la stessa chiave e lo stesso numero di occorrenze, si può procedere ad una loro fusione

La traduzione di relazioni Uno a Molti

- con vincolo di integrità referenziale fra Squadra in Giocatore e la chiave di Squadra
- Si noti che nel caso di cardinalità (0,1) la seconda soluzione presenterà dei valori nulli per gli attributi Squadra e Ingaggio della tabella Giocatore

Poiché le tabelle Giocatore e Contratto hanno la stessa chiave e lo stesso numero di occorrenze, si può procedere ad una loro fusione con la seguente regola:

"Una associazione uno a molti si traduce aggiungendo alla relazione che traduce l'entità dal lato uno gli attributi dell'associazione e l'identificatore dell'entità dal lato molti, eventualmente ridenominato"

Giocatore(Cognome, DataNascita, Ruolo, **Squadra**, Ingaggio)
Squadra(Nome, Città, ColoriSociali)

Soluzione più compatta

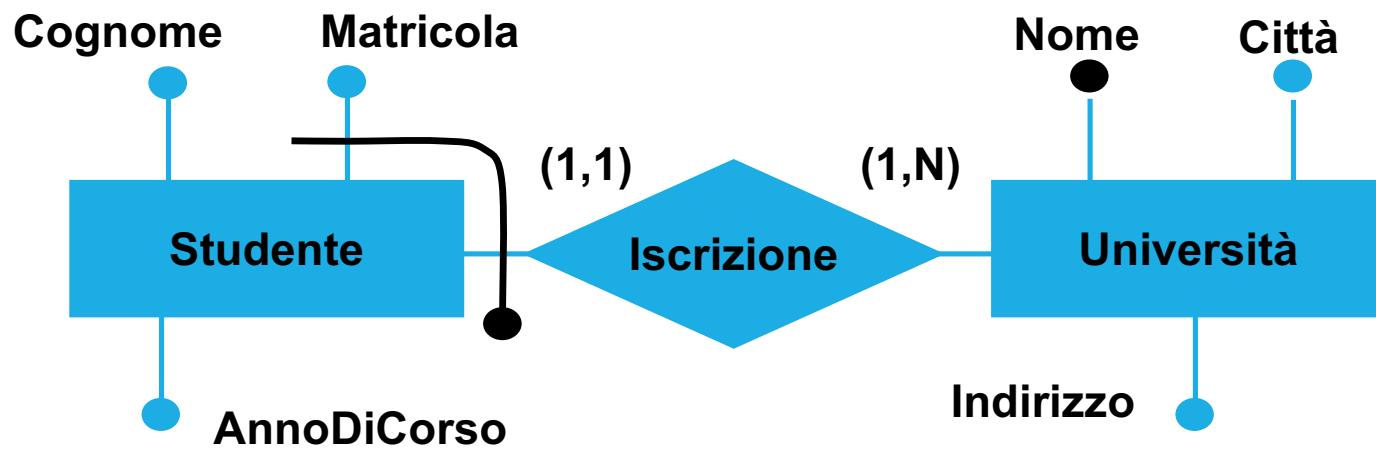
PREG. TOMMASO CRISCIANO "INSEMI WEB E BASE DI DATI" - UNIVERSITÀ DEGLI STUDI DELLA
CAMPANIA "L. VANVITELLI"

Ora vengo comunque del legame tramite concetto di chiave esterna.

Prendo chiave primaria dell'entità a N e devo metterla come chiave esterna
dell'entità a 1 più altri. obbl. relazione

NOTA: gli attributi della chiave esterna sono gli identificatori principali dell'entità a N.

Entità con identificazione esterna



La presenza degli identificatori esterni non altera il tipo di traduzione anche perché la chiave già contiene gli identificatori delle entità identificanti

Studente(Matricola, Università, Cognome, AnnoDiCorso)

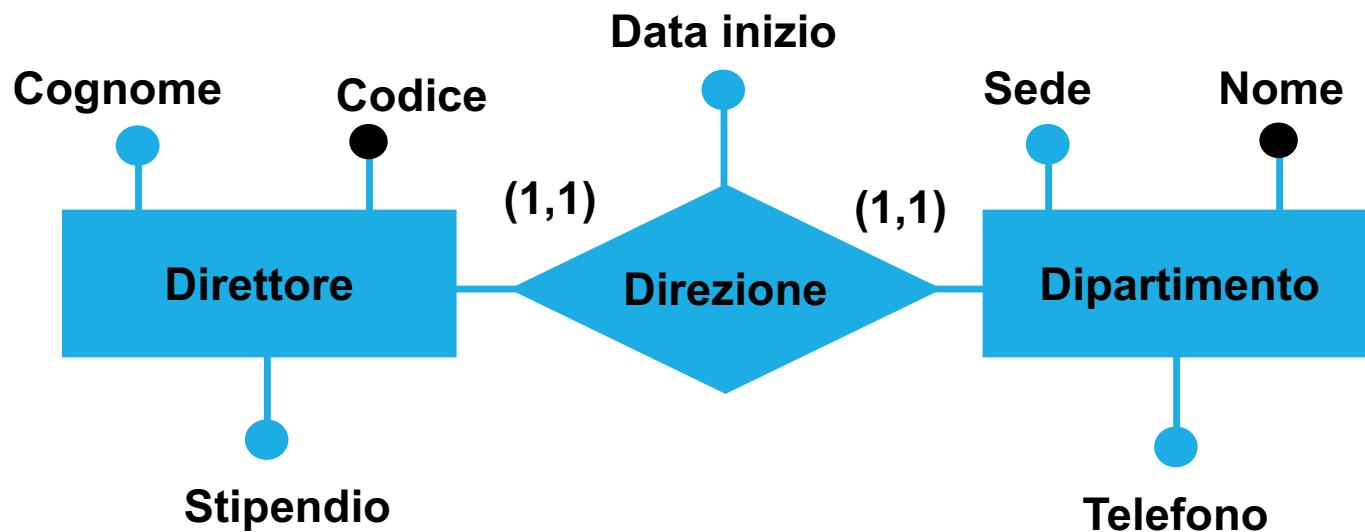
Università(Nome, Città, Indirizzo)

Gli identificatori esterni apprezzano gli identificatori
della relazione a N.

La traduzione di relazioni Uno a Uno

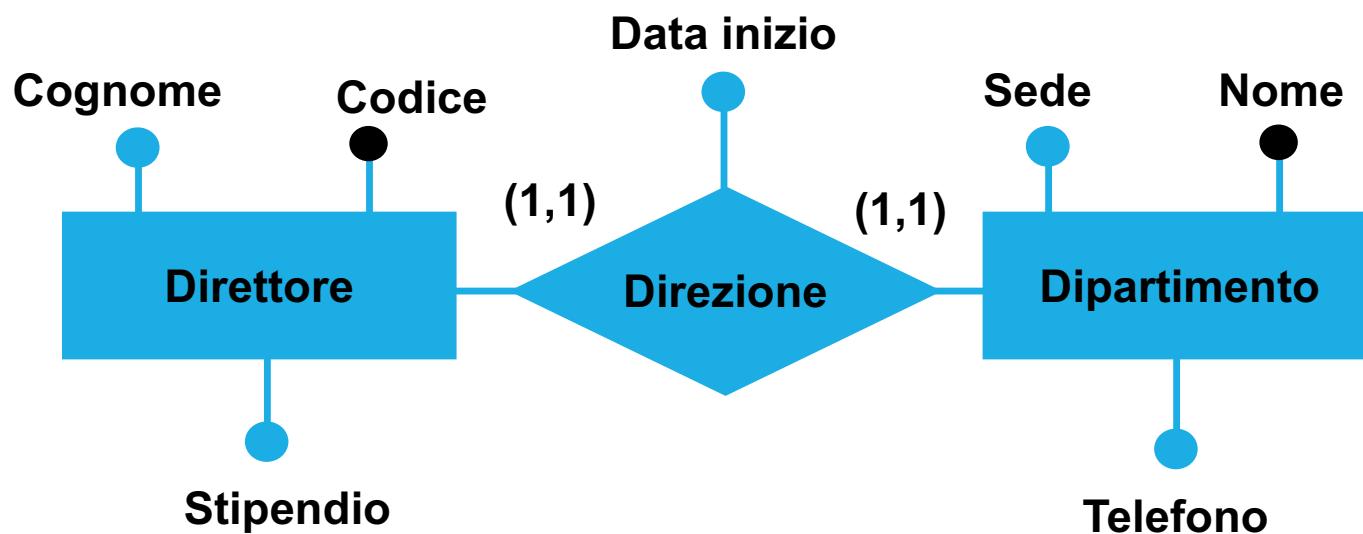
- Alla traduzione delle relazioni uno a uno si estendono le considerazioni fatte per il caso Uno a Molti
- Le soluzioni diventano però diverse, perché si può scegliere indifferentemente su quale delle due entità sviluppare la traduzione
- Nel caso di partecipazioni opzionali delle entità vanno distinti i casi di:
 - Partecipazione opzionale di una sola entità
 - Partecipazione opzionale di entrambe le entità

La traduzione di relazioni Uno a Uno



- fondere da una parte o dall'altra
- fondere tutto? In genere è da evitare: porta possibili anomalie in fase di normalizzazione

La traduzione di relazioni Uno a Uno

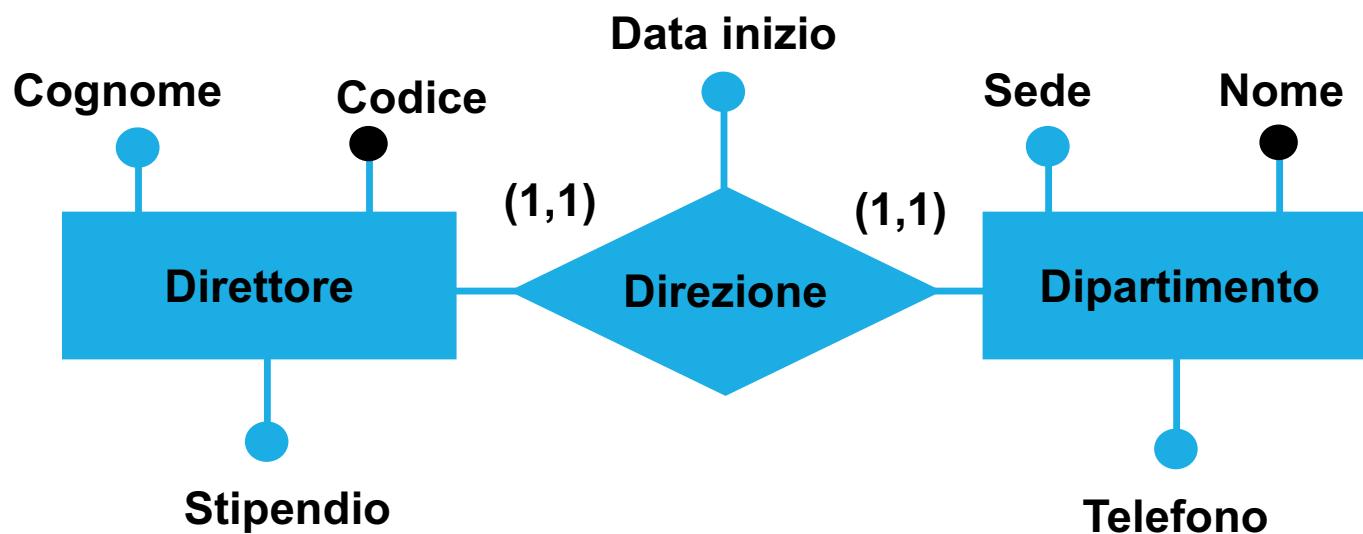


- La prima soluzione:

Direttore(Codice,Cognome,Stipendio,DipartimentoDiritto,InizioDirezione)

Dipartimento(Nome,Sede, Telefono)

La traduzione di relazioni Uno a Uno

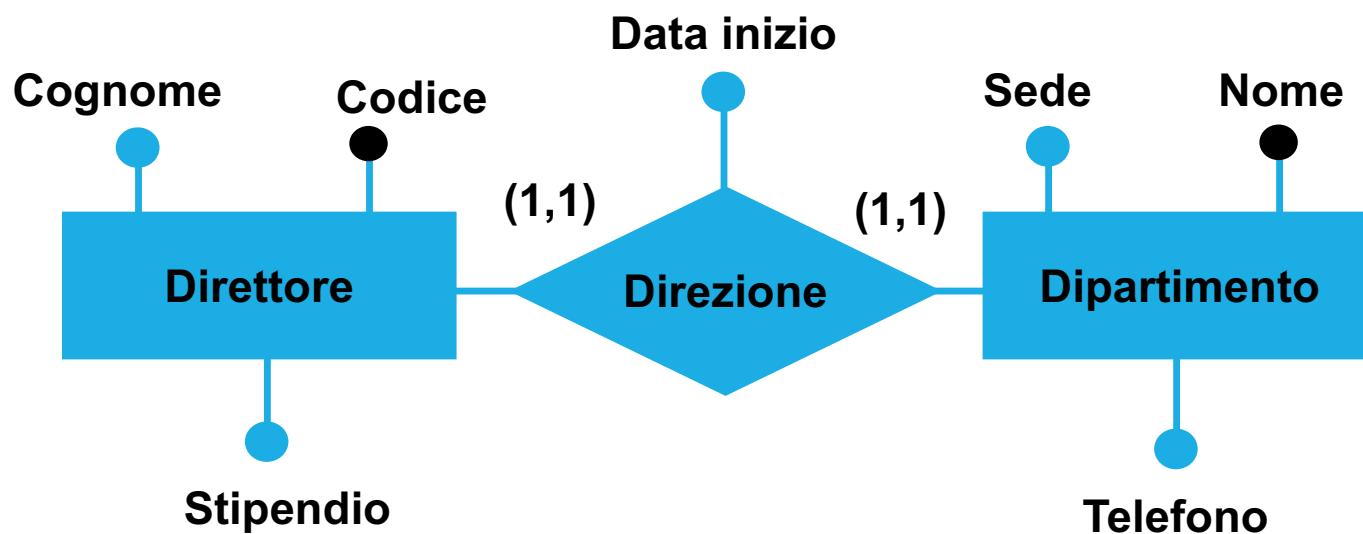


- La seconda soluzione:

Direttore(Codice,Cognome,Stipendio)

Dipartimento(Nome,Sede, Telefono, Direttore, InizioDirezione)

La traduzione di relazioni Uno a Uno



- La terza soluzione:

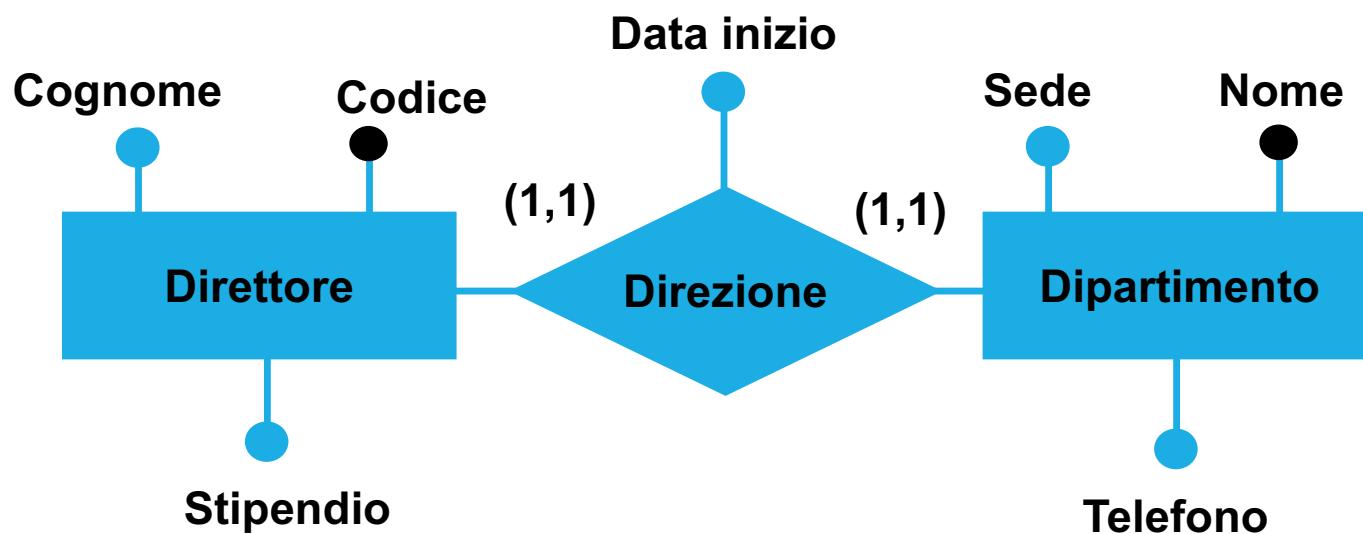
Direttore(Codice, Cognome, Stipendio)

Dipartimento(Nome, Sede, Telefono)

Direzione(Direttore, Dipartimento, Anno)

Direzione traducibile come tabella a parte, ma solo una delle due è
solam. riproducibile.

La traduzione di relazioni Uno a Uno



- La quarta soluzione (simmetrica alla terza):
Direttore(Codice,Cognome,Stipendio)
Dipartimento(Nome,Sede, Telefono)
Direzione(Direttore,Dipartimento, Anno)

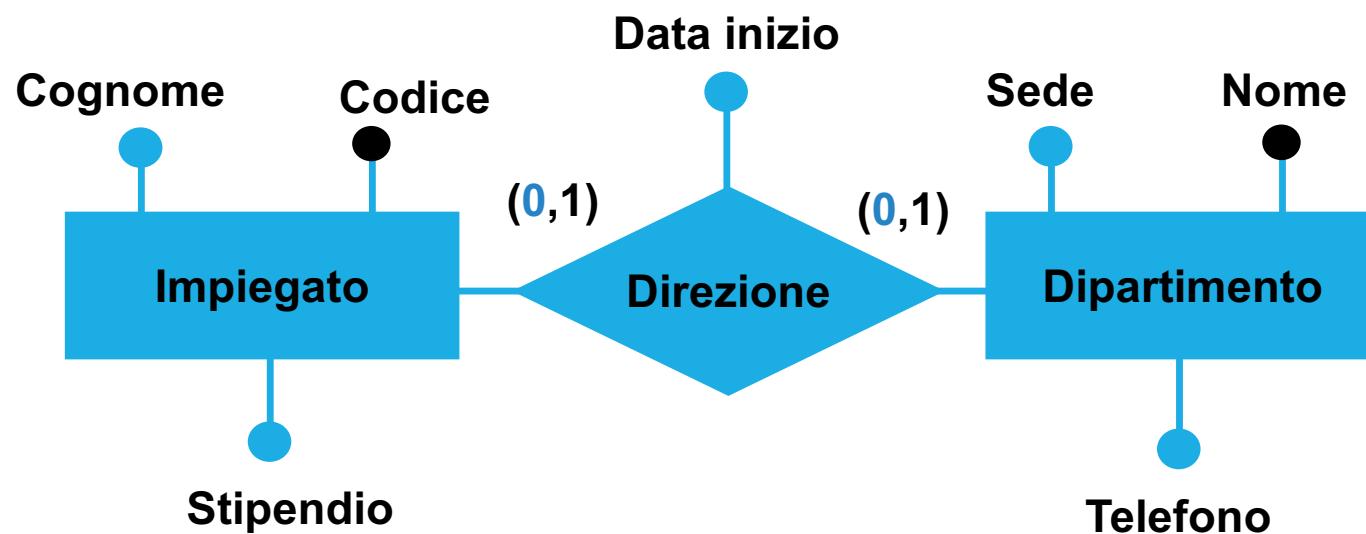
NOTA: qui devo proporre una tabella in più. Ne devo tenere conto.

La traduzione di relazioni Uno a Uno: con partecipazione opzionale

- Nel caso di partecipazioni opzionali (0,1) di entrambe le entità, sono da preferire la terza o la quarta soluzione per evitare la gestione di valori nulli. Metto tutto in quella tabella.
- Nel caso di partecipazione opzionale di una sola delle entità, è da preferire, tra la prima e la seconda soluzione, quella che vede inglobata la tabella della relazione nella tabella dell'altra entità.

Es: dipartimento senza direzione nella prima soluzione o secondo soluzione due valori nulli sulle chiavi esterne.

La traduzione di relazioni Uno a Uno: con partecipazione opzionale



In questo caso si deve tradurre in:

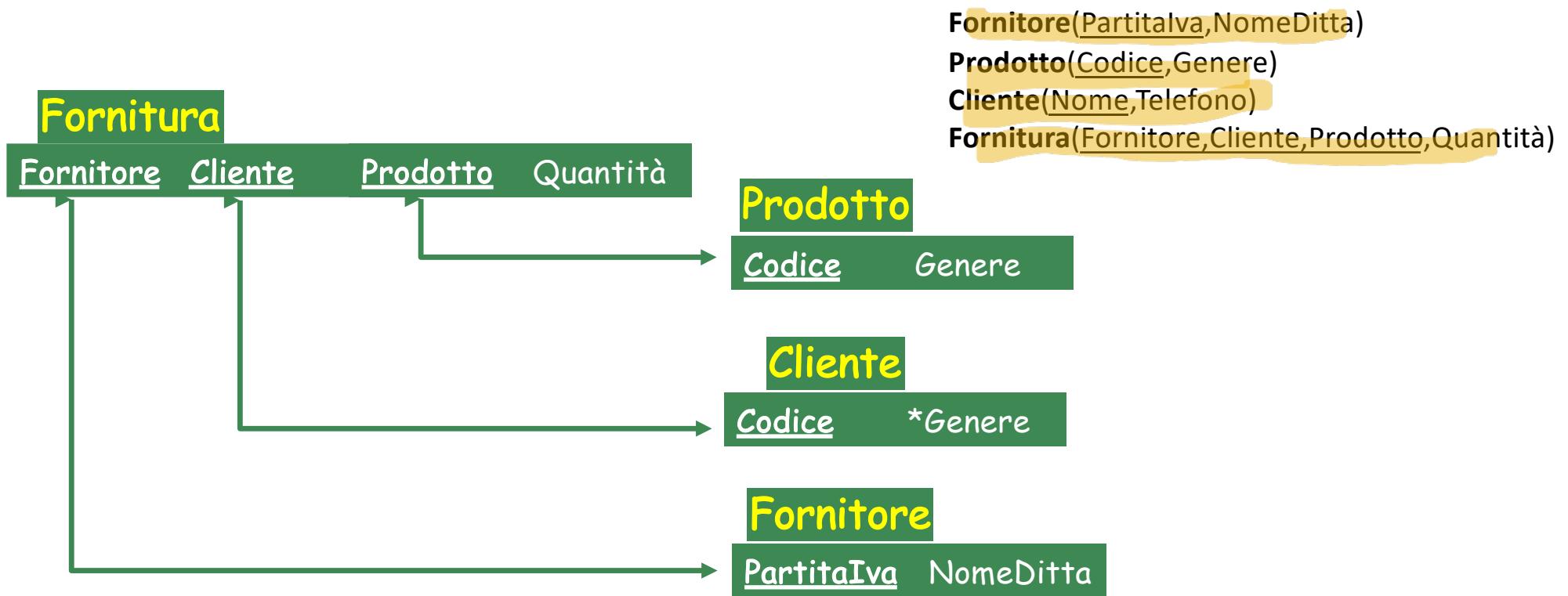
Impiegato(Codice,Cognome,Stipendio)
Dipartimento(Nome,Sede,Direttore,Anno)

• Pdh' essere null

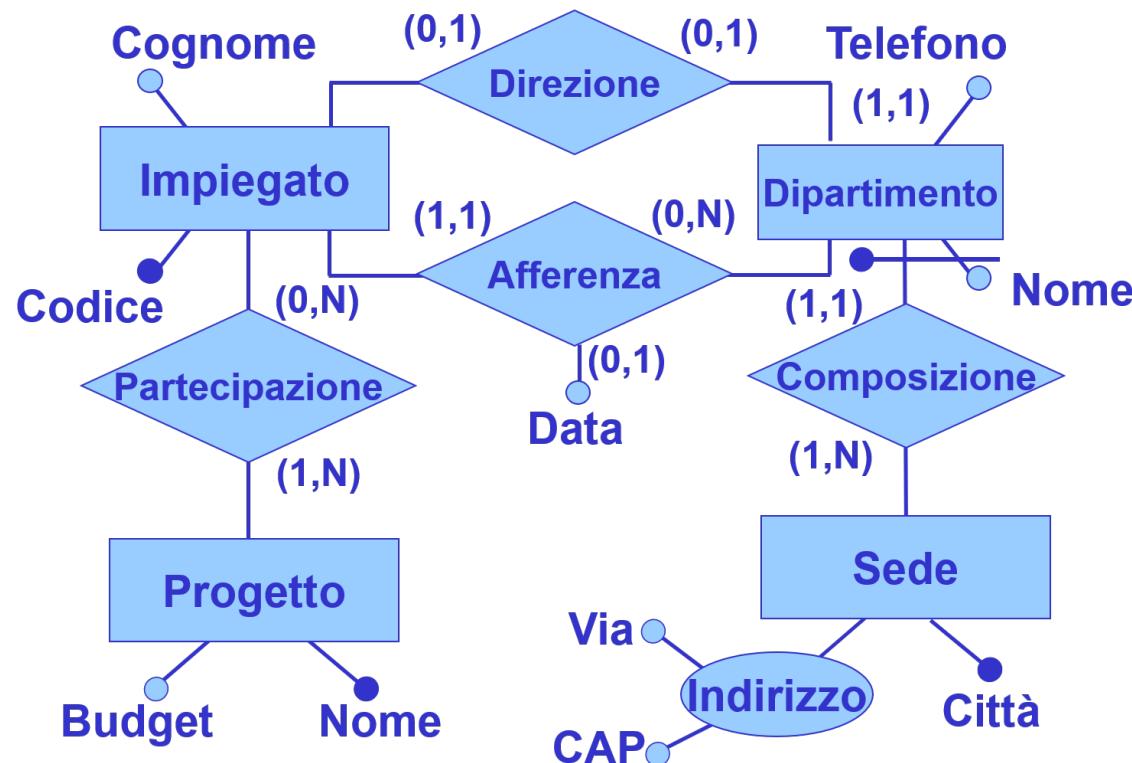
Direttore

surelle Impiegato.

Esempio



Esempio



Esempio

- Per le entità E che partecipano ad associazioni sempre con $\text{max-card}(E,R) = n$ la traduzione è immediata:

Sede(Città, Via, CAP)

Progetto(Nome, Budget)

Se ho conoscenza N sì sarebbe non
dovranno chiedere esterne.

- Anche l'associazione Partecipazione si traduce immediatamente:

Partecipazione(Impiegato, Progetto)

- L'entità Dipartimento si traduce importando l'identificatore di Sede e inglobando l'associazione Direzione

Dipartimento(Nome, Città, Telefono, Direttore)

- Per tradurre l'associazione Afferenza, assumendo che siano pochi gli impiegati che non afferiscono a nessun dipartimento, si opta per una rappresentazione compatta

Impiegato(Codice, Cognome, CittàDip*, NomeDip*, Data)

Osservazioni finali

- ❑ La progettazione logica, pur potendosi avvalere di strumenti software (quali DB-Main), non va svolta "alla cieca"; in presenza di diverse alternative occorre valutare:
 - ❑ La presenza o meno di valori nulli, e la loro incidenza, che dipende dal volume dei dati
 - ❑ Il modo con cui le operazioni navigano lo schema E/R, che poi si traduce in operazioni di join tra le relazioni che vengono create
- ❑ I casi visti non esauriscono l'argomento e lasciano sempre lo spazio per soluzioni specifiche "ad hoc"
- ❑ Ad esempio, associazioni uno a molti con $\text{max-card}(E2,R) = K$, con K "piccolo", possono al limite essere tradotte con 1 sola relazione, prevedendo K repliche degli attributi di $E2$ (es. tipico: numeri di telefono)

In sintesi:

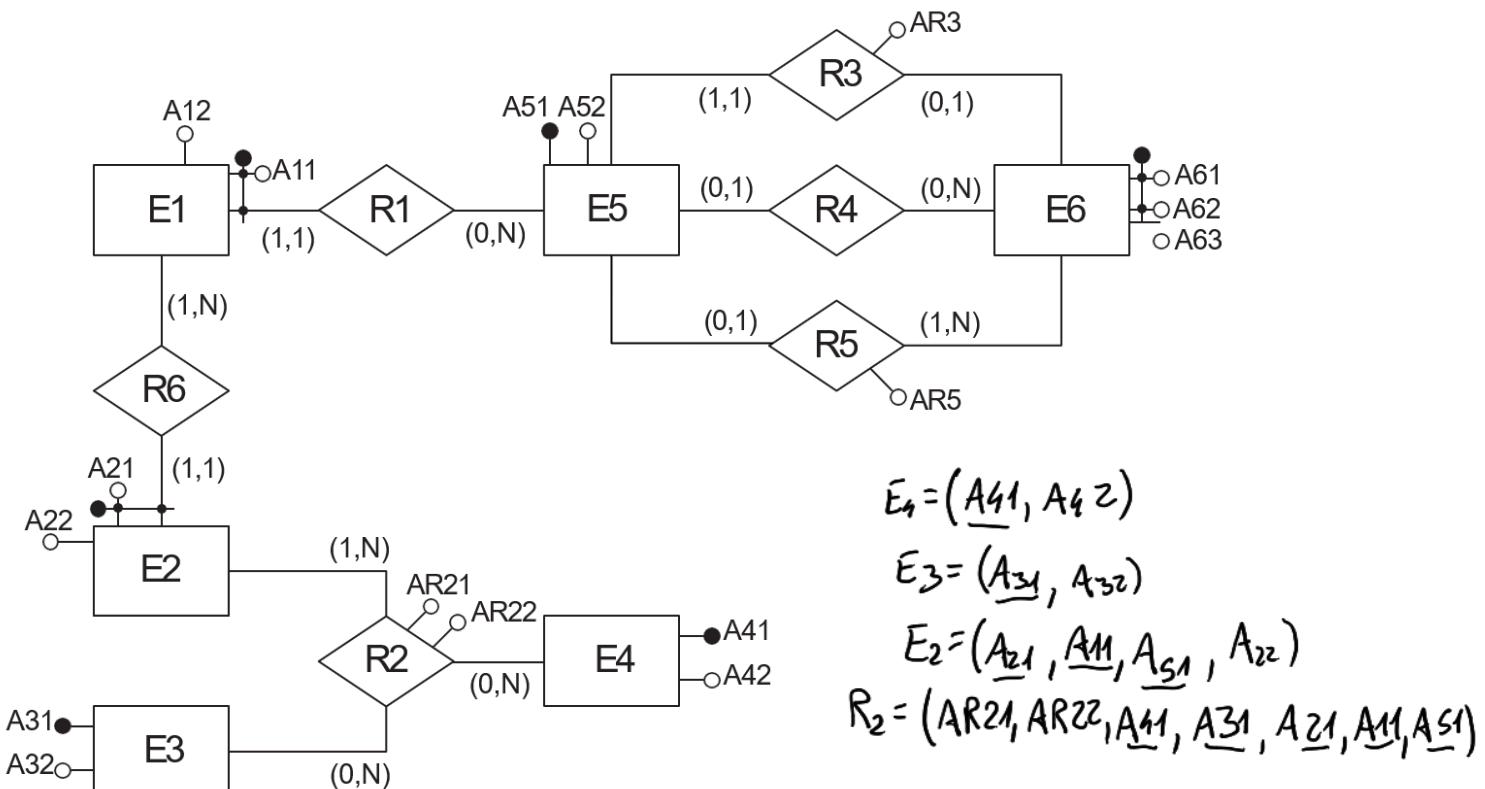
- La fase di progettazione logica ha lo scopo di derivare uno schema logico che sia il più fedele possibile rispetto allo schema E/R di partenza, e che sia al tempo spesso "efficiente"
- I confronti tra le diverse alternative vengono eseguiti considerando le principali operazioni interessate e i volumi dei dati in gioco
- La fase di ristrutturazione elimina dallo schema E/R tutti i costrutti che non possono essere direttamente rappresentati nel modello logico, e apporta modifiche strutturali sulla base di considerazioni di efficienza
- La fase di traduzione opera traducendo entità e associazioni.
- Le diverse alternative che si hanno a disposizione per tradurre le associazioni dipendono dalle cardinalità massime in gioco, le quali determinano anche le chiavi delle relazioni che si ottengono.
- Le cardinalità minime possono portare, in funzione della traduzione scelta, ad avere valori nulli

$$E_1 = (\underline{A_{11}}, \underline{A_{51}}, A^{12})$$

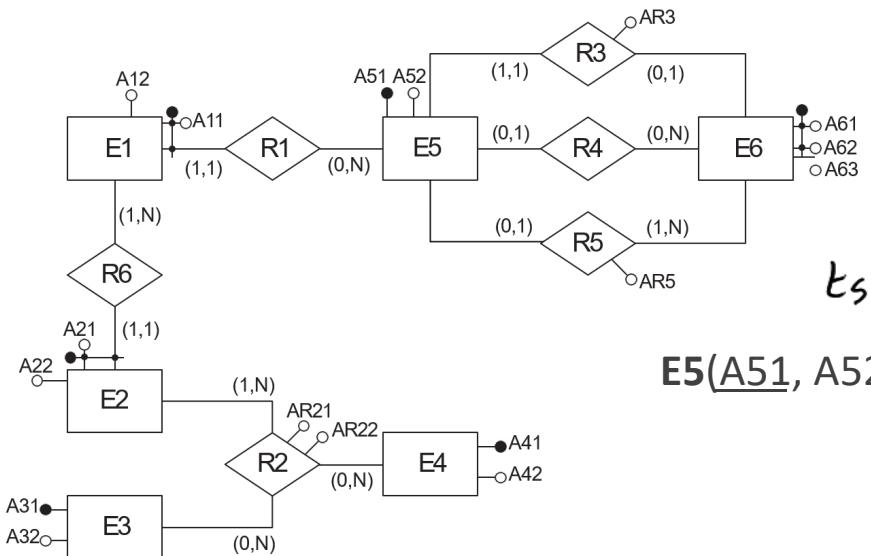
$$E_5 = (\underline{A_{51}}, \underline{A_{61}}, \underline{A_{62}}, AR_3, AR_5^*)$$

$$E_6 = (\underline{A_{61}}, \underline{A_{62}}, A_{63})$$

Traduzione di schemi complessi



Traduzione di schemi complessi: Risultato



$$E_S = (\underline{A_{51}}, \underline{A_{61}}, \underline{A_{62}}, AR_3, AR_5^*)$$

E5(A51, A52, A61R3, A62R3, AR3, A61R4, A62R4, A61R5, A62R5, AR5)

E1(A11, A51, A12)

E2(A21, A11, A51, A22)

Percorso chiuso **E3(A31, A32)**

Percorso di E1
è A11 e A51 **E4(A41, A42)**

E6(A61, A62, A63)

R2(A21, A11, A51, A31, A41, AR21, AR22)

Considerare puntate da chi non ha chiavi esterne

Tabelle riassuntive

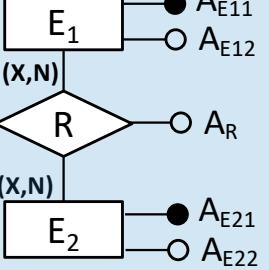
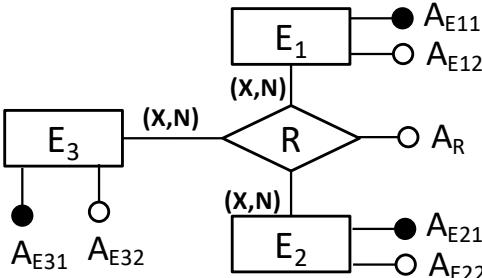
Tipologia	Concetto iniziale	Risultati possibili
Associazione binaria molti a molti	 <pre> classDiagram class E1 { A_E11 A_E12 } class E2 { A_E21 A_E22 } E1 "2" --> "2" E2 : R R "2" --> "2" E1 </pre>	$E_1(A_{E11}, A_{E12})$ $E_2(A_{E21}, A_{E22})$ $R(A_{E11}, A_{E21}, A_R)$
Associazione ternaria molti a molti	 <pre> classDiagram class E1 { A_E11 A_E12 } class E2 { A_E21 A_E22 } class E3 { A_E31 A_E32 } E1 "2" --> "2" E2 : R E1 "2" --> "2" E3 : R E2 "2" --> "2" E3 : R R "2" --> "2" E1 </pre>	$E_1(A_{E11}, A_{E12})$ $E_2(A_{E21}, A_{E22})$ $E_3(A_{E31}, A_{E32})$ $R(A_{E11}, A_{E21}, A_{E31}, A_R)$

Tabelle riassuntive

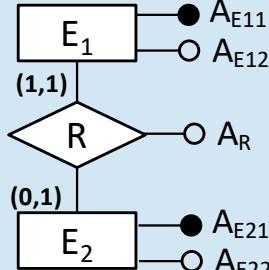
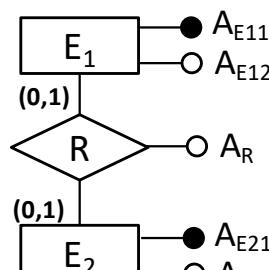
Tipologia	Concetto iniziale	Risultati possibili
Associazione uno a molti con partecipazione obbligatoria	<pre> graph TD E1[Entity E1] --> A_E11 E1 --> A_E12 E1 --- R{Relationship R} R --> A_R R --> E2[Entity E2] E2 --> A_E21 E2 --> A_E22 R -- (1,1) --> E1 R -- (X,N) --> E2 </pre>	$E_1(A_{E11}, A_{E12}, A_{E21}, A_R)$ $E_2(A_{E21}, A_{E22})$
Associazione uno a molti con partecipazione opzionale	<pre> graph TD E1[Entity E1] --> A_E11 E1 --> A_E12 E1 --- R{Relationship R} R --> A_R R --> E2[Entity E2] E2 --> A_E21 E2 --> A_E22 R -- (0,1) --> E1 R -- (X,N) --> E2 </pre>	$E_1(A_{E11}, A_{E12})$ $E_2(A_{E21}, A_{E22})$ $R(A_{E11}, A_{E21}, A_R)$ Oppure: $E_1(A_{E11}, A_{E12}, A_{E21}^*, A_R^*)$ $E_2(A_{E21}, A_{E22})$ <p style="margin-left: 200px;">Applicabile solo a molti</p> <p>*: Valore nullo sull'attributo</p> <p style="margin-left: 200px;">Compatti</p>

Tabelle riassuntive

Tipologia	Concetto iniziale	Risultati possibili
Associazione con identificatore esterno	<pre> classDiagram class E1 { A_E11 A_E12 A_R } class E2 { A_E21 A_E22 } E1 "1..1" *-- "1..1" E2 : R { X, Y } </pre>	$E_1(A_{E12}, A_{E21}, A_{E11}, A_R)$ $E_2(A_{E21}, A_{E22})$
Associazione uno a uno con partecipazione obbligatoria per entrambe le entità	<pre> classDiagram class E1 { A_E11 A_E12 } class E2 { A_E21 A_E22 } E1 "1..1" *-- "1..1" E2 : R { X, Y } </pre>	$E_1(A_{E11}, A_{E12}, A_{E21}, A_R)$ $E_2(A_{E21}, A_{E22})$ Oppure: $E_2(A_{E21}, A_{E22}, A_{E11}, A_R)$ $E_1(A_{E11}, A_{E12})$

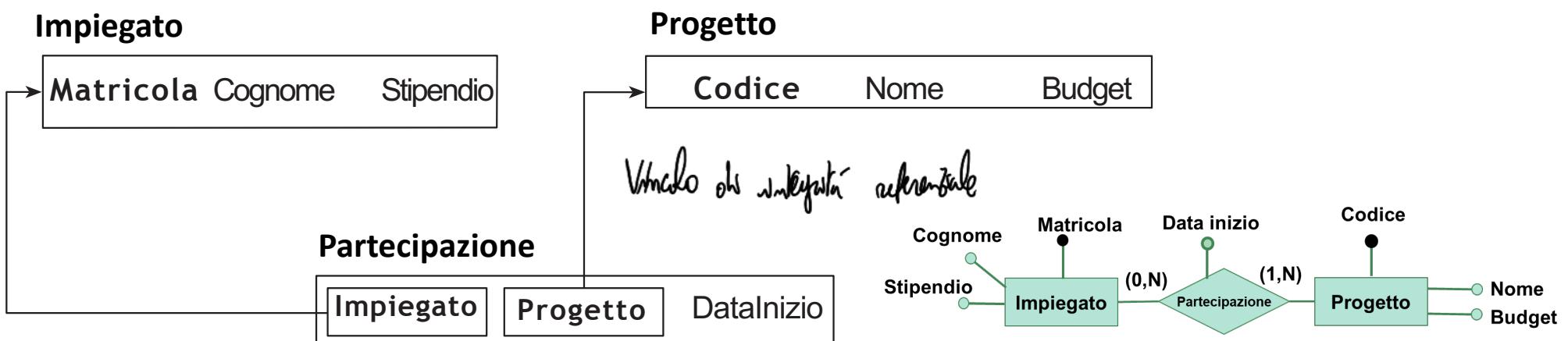
----: Chiave alternativa a quella indicata con —

Tabelle riassuntive

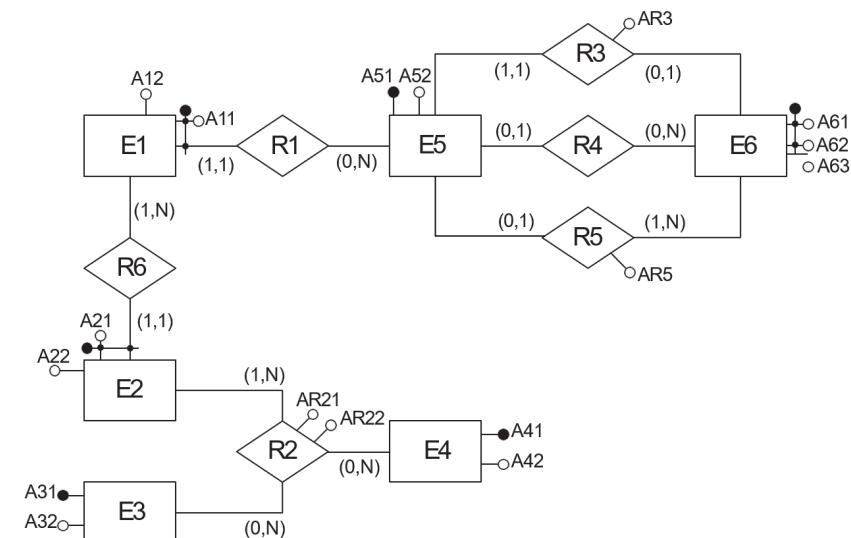
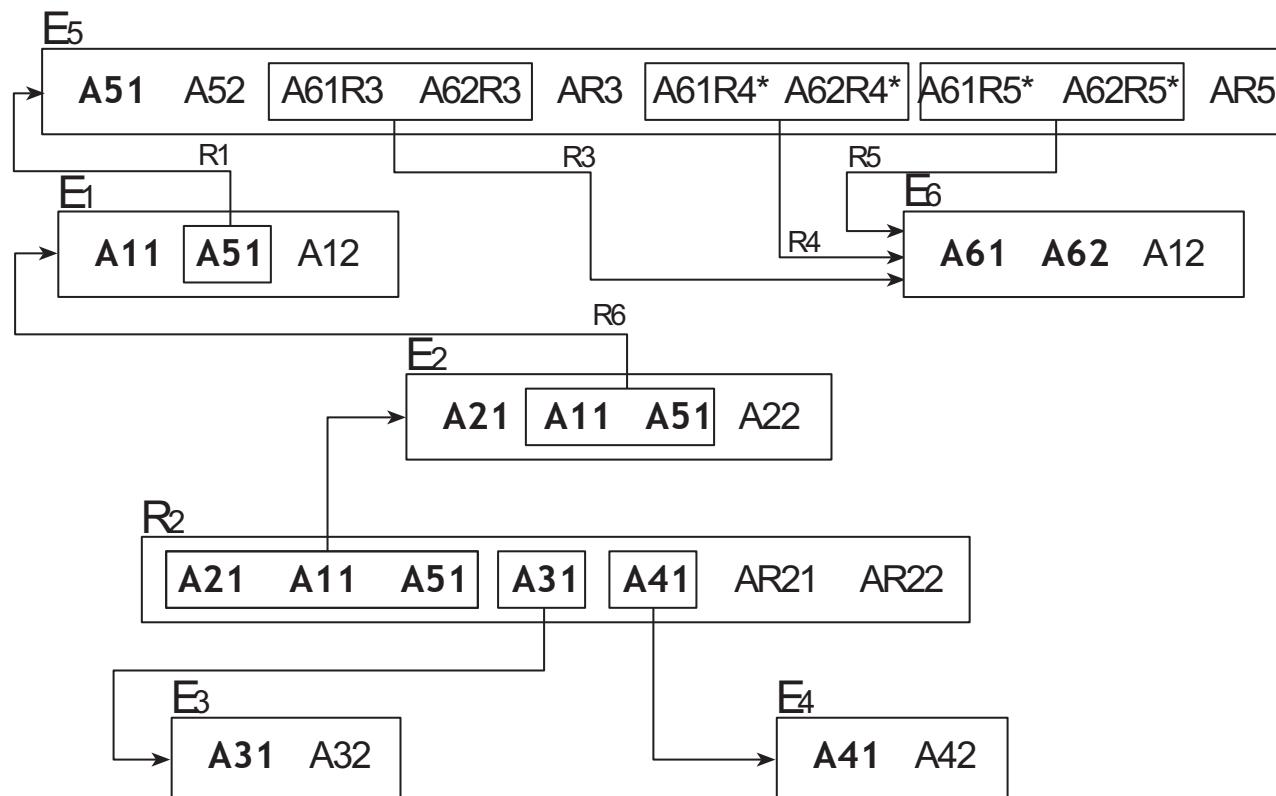
Tipologia	Concetto iniziale	Risultati possibili
<p>Associazione uno a uno con partecipazione opzionale per una entità</p>	 <pre> classDiagram class E1 { A_E11 A_E12 } class E2 { A_E21 A_E22 } E1 "1..1" *--o R --> E2 </pre>	<p>come 1a N</p> <p>$E_1(A_{E11}, A_{E12}, A_{E21}, A_R)$</p> <p>$E_2(A_{E21}, A_{E22})$</p> <p>chiavi esterne</p>
<p>Associazione uno a uno con partecipazione opzionale per entrambe le entità</p>	 <pre> classDiagram class E1 { A_E11 A_E12 } class E2 { A_E21 A_E22 } E1 "0..1" *--o R --> E2 </pre>	<p>$E_1(A_{E11}, A_{E12})$</p> <p>$E_2(A_{E21}, A_{E22}, A_{E11}^*, A_R^*)$</p> <p>Oppure:</p> <p>$E_1(A_{E11}, A_{E12}, A_{E21}^*, A_R^*)$</p> <p>$E_2(A_{E21}, A_{E22})$</p> <p>Oppure:</p> <p>$E_1(A_{E11}, A_{E12})$</p> <p>$E_2(A_{E21}, A_{E22})$</p> <p>$R(A_{E11}, A_{E21}, A_R)$</p> <p>Non ha supposto come accadrà di R se E1 non è legato a E2 per es.</p> <p>allora</p>

Documentazione di schemi logici

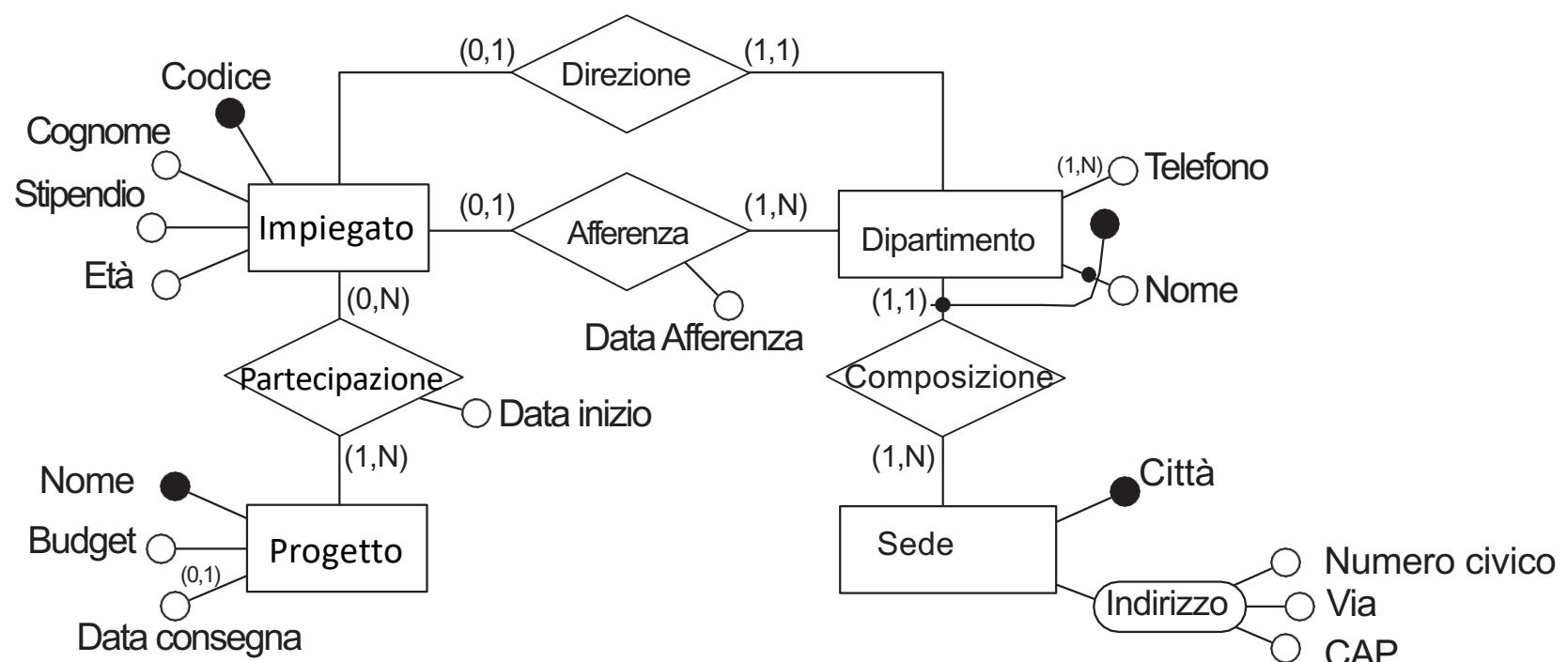
- Come nel caso della progettazione concettuale, il risultato della progettazione logica non è costituito solo da un semplice schema di una base di dati, ma anche da una **documentazione associata**
- Esiste un formalismo grafico che permette di rappresentare sia le relazioni con i relativi attributi sia i vincoli di integrità referenziale



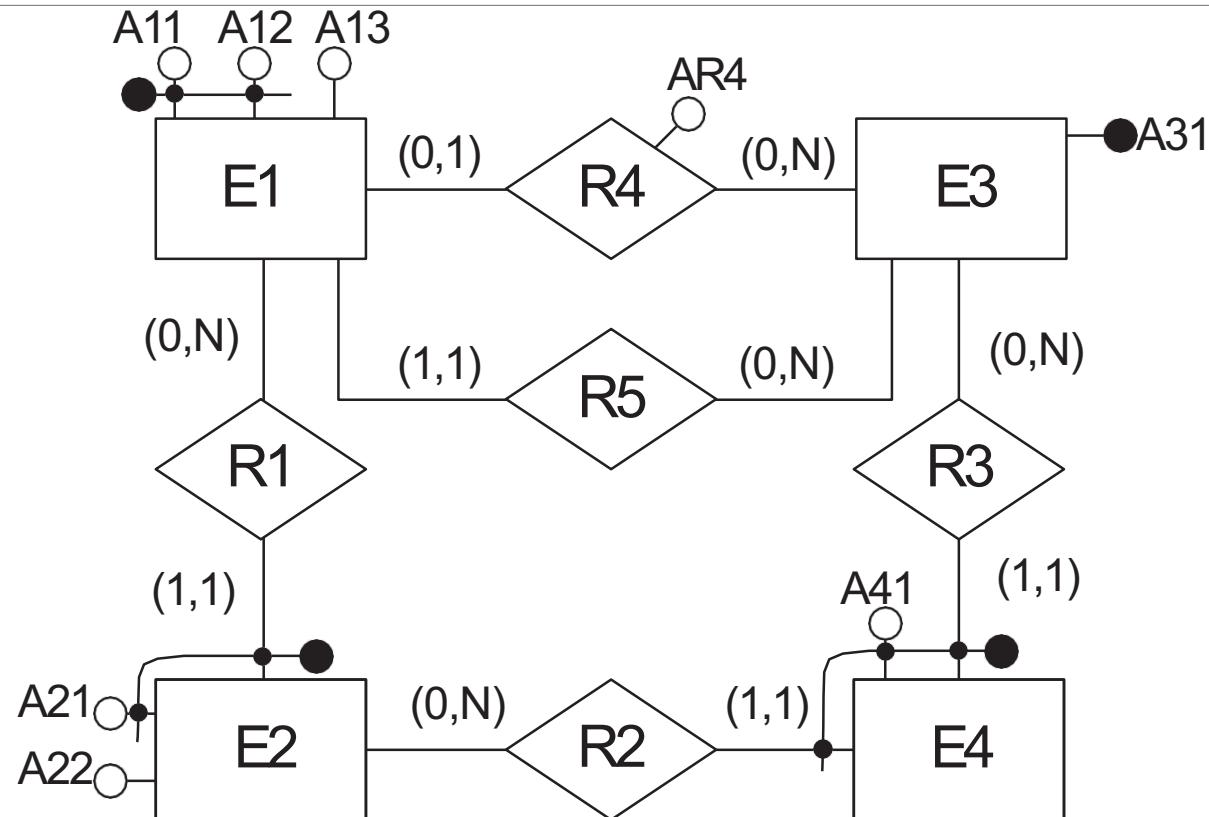
Documentazione di schemi logici



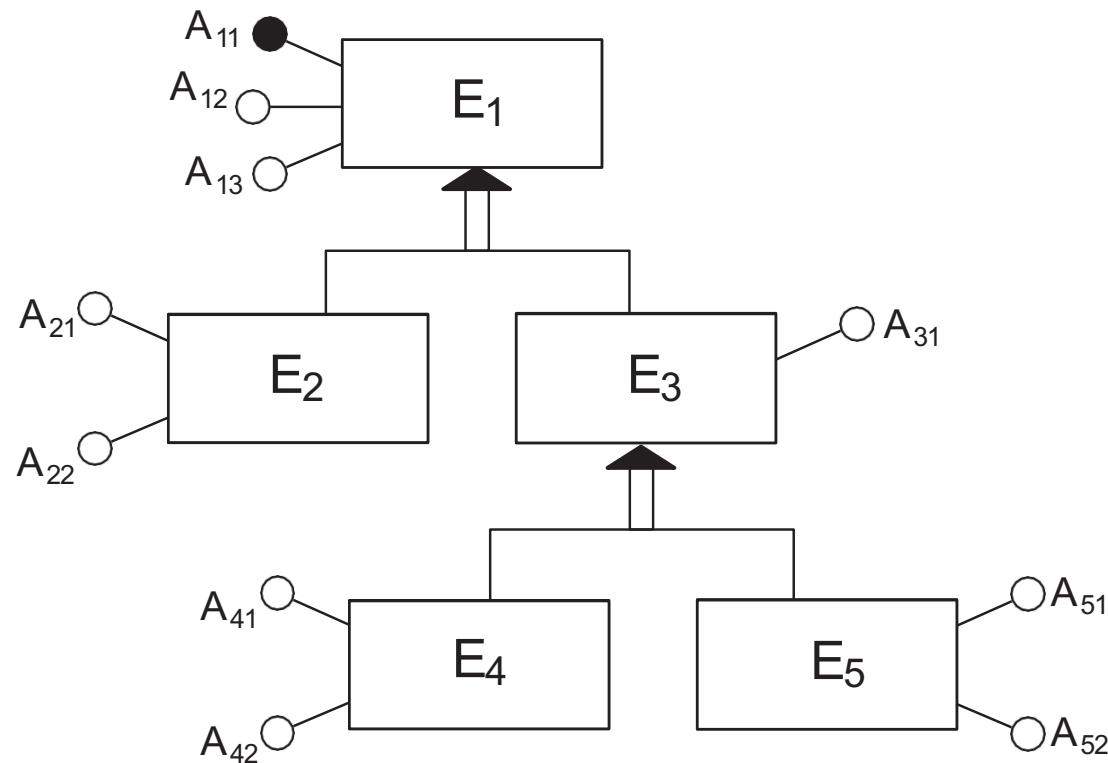
Esercizio 1: Traduci nel modello logico



Esercizio 2: Traduci nel modello logico



Esercizio 3: Traduci nel modello logico



Esercizio 4: Traduci nel modello logico

