



Progettazione Logica di Basi di Dati

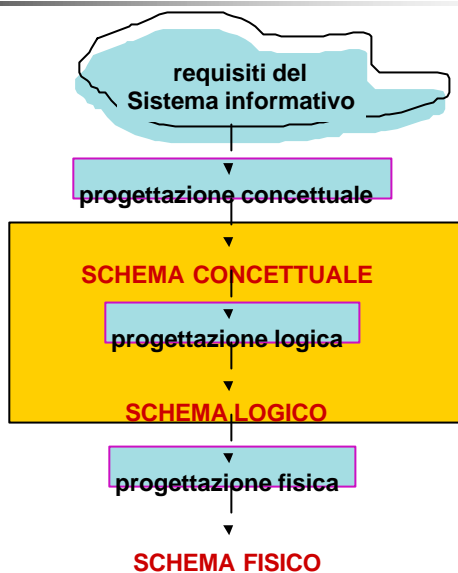
Sistemi Informativi L

Corso di Laurea in Ingegneria dei Processi Gestionali
A.A. 2003/2004

Docente: Prof. Wilma Penzo



Il secondo passo...





Progettazione logica

- Obiettivo della fase di progettazione logica è pervenire, a partire dallo schema concettuale, a uno schema logico che lo rappresenti **in modo fedele** e che sia, al tempo stesso, **“efficiente”**
- L'**efficienza** è legata alle **prestazioni**, ma poiché queste non sono valutabili precisamente a livello concettuale e logico si ricorre a degli **indicatori semplificati**



Fasi della progettazione logica

- La progettazione logica può articolarsi in due fasi principali:
 - **Ristrutturazione**: eliminazione dallo schema E/R di tutti i costrutti che non possono essere direttamente rappresentati nel modello logico target (relazionale nel nostro caso):
 - Eliminazione degli attributi multivalore
 - Eliminazione delle generalizzazioni
 - Partizionamento/accorpamento di entità e associazioni
 - Scelta degli identificatori principali
 - **Traduzione**: si mappano i costrutti residui in elementi del modello relazionale



Fase di ristrutturazione

- Serve a **semplificare la traduzione** e a **“ottimizzare” le prestazioni**
- Per confrontare tra loro diverse alternative bisogna conoscere, almeno in maniera approssimativa, il **“carico di lavoro”**, ovvero:
 - Le principali **operazioni** che la BD dovrà supportare
 - I **“volumi” dei dati** in gioco
- Gli **indicatori** che deriviamo considerano due aspetti
 - **spazio**: numero di istanze previste
 - **tempo**: numero di istanze (di entità e associazioni) visitate durante un'operazione



Schema di riferimento

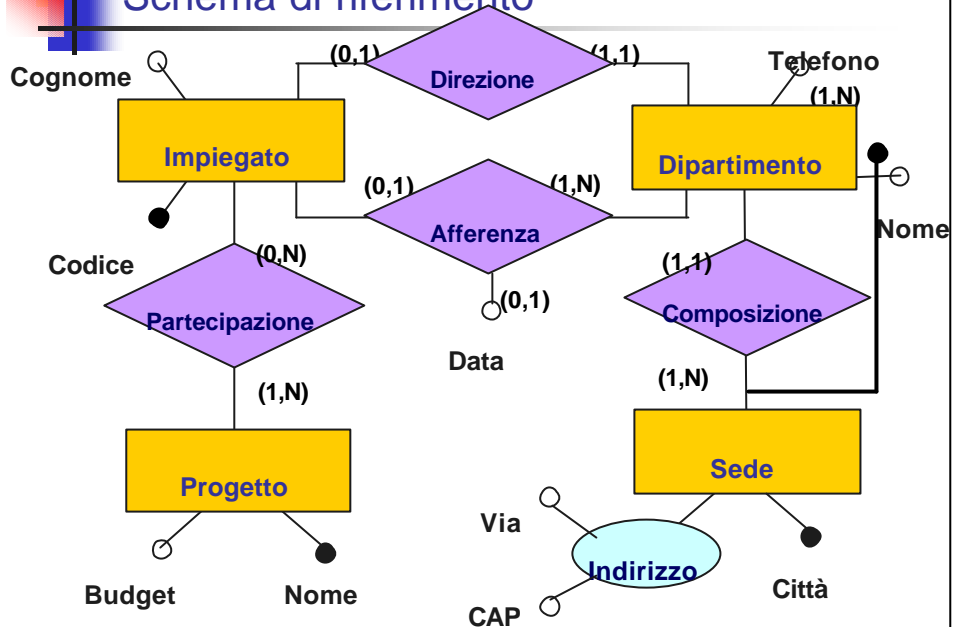




Tavola dei volumi

- Specifica il numero stimato di istanze per ogni entità (E) e associazione (R) dello schema
- I valori sono necessariamente approssimati, ma indicativi

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

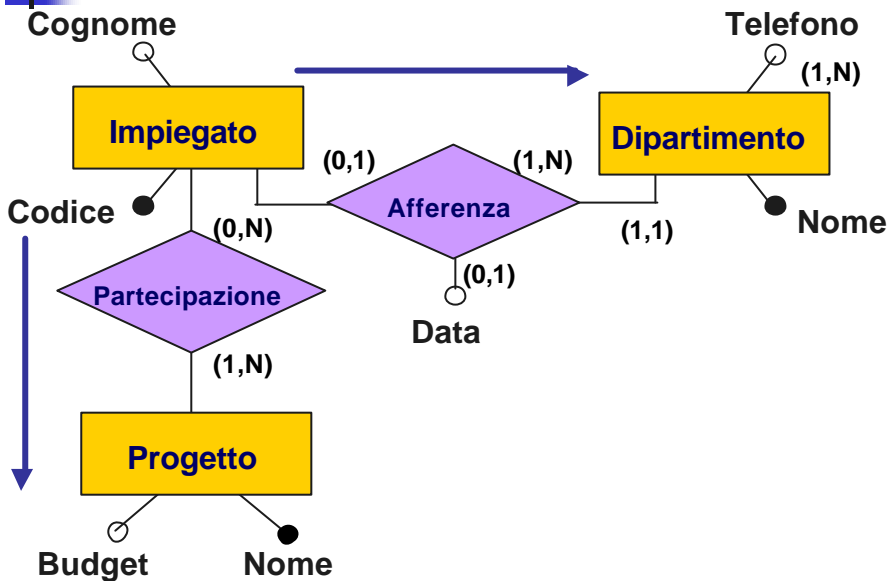


Esempio di valutazione di costo

trova tutti i dati di un impiegato, del dipartimento nel quale lavora e dei progetti ai quali partecipa

- 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 che indicano in che modo l'operazione "naviga" i dati

Esempio di schema di navigazione



9

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)

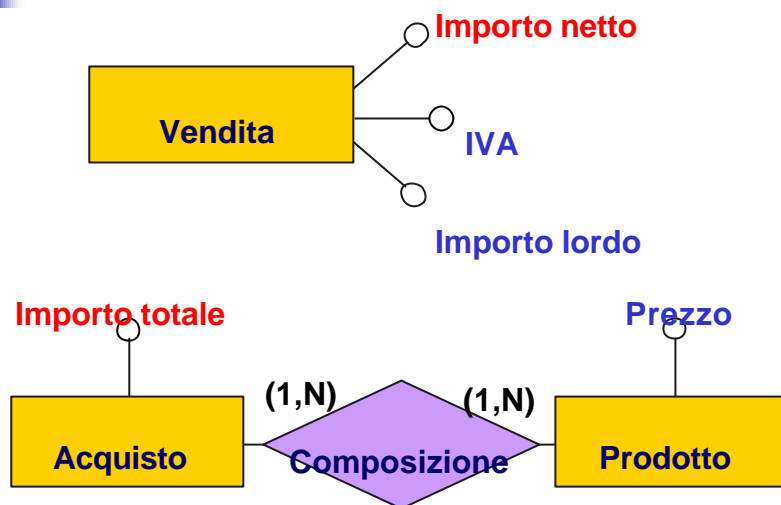
Concetto	Costrutto	Accessi	Tipo
Impiegato	Entità	1	L
Afferenza	Associazione	1	L
Dipartimento	Entità	1	L
Partecipazione	Associazione	3	L
Progetto	Entità	3	L

10

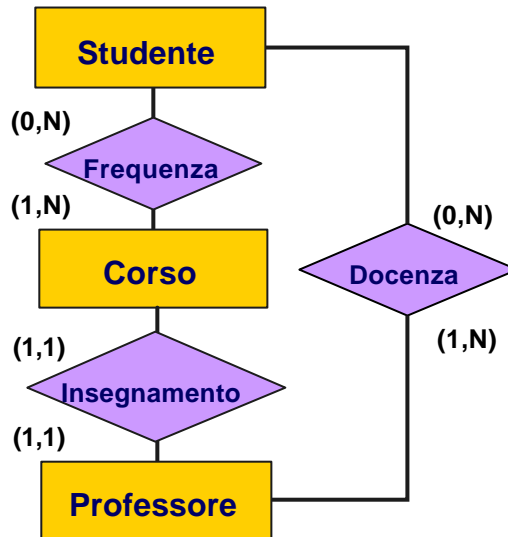
Analisi delle ridondanze

- 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 (**è quindi comunque importante averle individuate in fase di progettazione concettuale!**)
- Se si mantiene una ridondanza
 - si **semplificano alcune interrogazioni**, ma
 - si **appesantiscono gli aggiornamenti**
 - si **occupa maggior spazio**
- Le possibili ridondanze riguardano
 - **Attributi derivabili da altri attributi**
 - **Associazioni derivabili dalla composizione di altre associazioni** (**presenza di cicli**)

Attributi derivabili



Associazioni ridondanti



Esempio di analisi di una ridondanza

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

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



- Operazione 1: memorizza una nuova persona con la relativa città di residenza (500 volte al giorno)
- Operazione 2: stampa tutti i dati di una città (incluso il numero di abitanti) (2 volte al giorno)



In presenza di ridondanza...

Operazione 1

Concetto	Costrutto	Accessi	Tipo
Persona	Entità	1	S
Residenza	Associazione	1	S
Città	Entità	1	L
Città	Entità	1	S

Operazione 2

Concetto	Costrutto	Accessi	Tipo
Città	Entità	1	L



In assenza di ridondanza...

Operazione 1

Concetto	Costrutto	Accessi	Tipo
Persona	Entità	1	S
Residenza	Associazione	1	S

Operazione 2

Concetto	Costrutto	Accessi	Tipo
Città	Entità	1	L
Residenza	Associazione	5000	L



Mantenere o no la ridondanza?

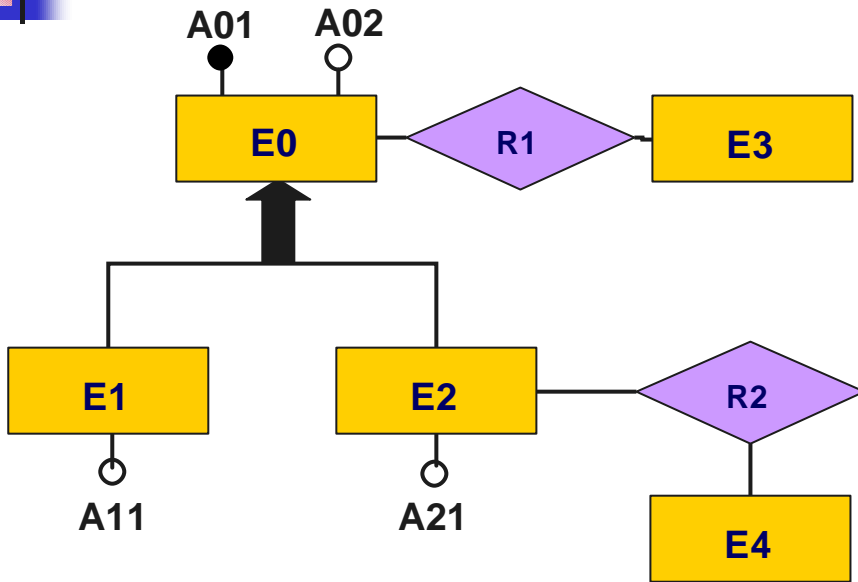
- È importante considerare la frequenza delle operazioni:
- **Con ridondanza:**
 - **Operazione 1:** 1500 accessi in scrittura e 500 accessi in lettura al giorno
 - **Operazione 2:** 2 accessi in lettura al giorno
 - **Totale:** 2002 accessi al giorno
- **Senza ridondanza:**
 - **Operazione 1:** 1000 accessi in scrittura al giorno
 - **Operazione 2:** 10002 accessi in lettura al giorno
 - **Totale:** 11002 accessi al giorno
- Si decide pertanto di mantenere la ridondanza
- In generale si possono anche fare considerazioni sullo spazio in più richiesto...



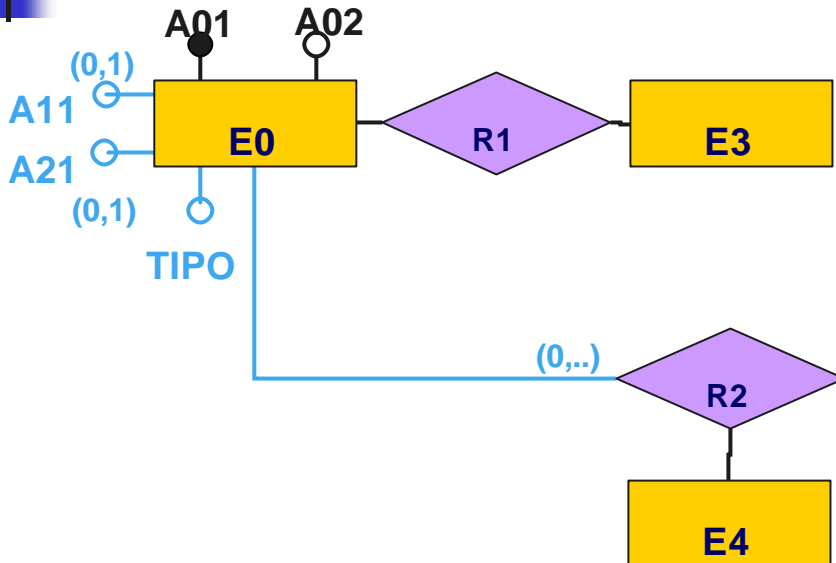
Eliminazione delle gerarchie

- Il modello relazionale non può rappresentare direttamente le generalizzazioni
- Entità e associazioni sono invece direttamente rappresentabili
- **Si eliminano perciò le gerarchie, sostituendole con entità e associazioni**
- Vi sono **3 possibilità** (più altre soluzioni intermedie):
 1. Accorpare le entità figlie nel genitore (collasso verso l'alto)
 2. Accorpare il genitore nelle entità figlie (collasso verso il basso)
 3. Sostituire la generalizzazione con associazioni

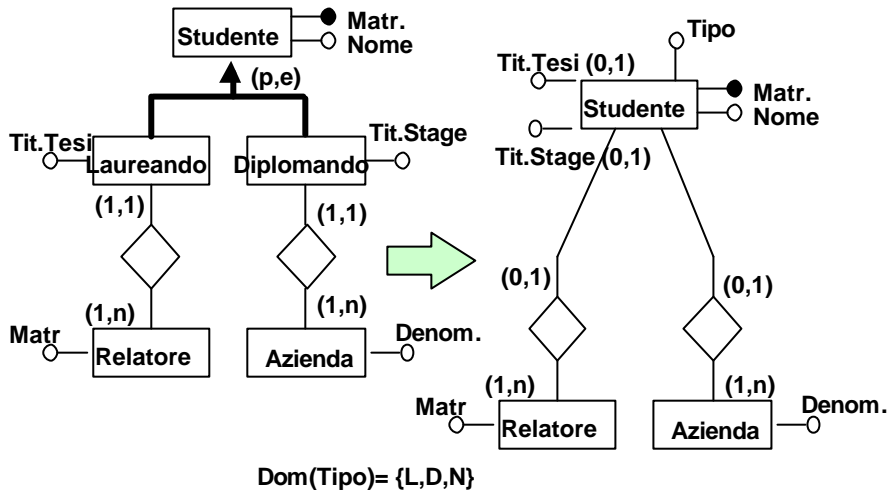
Schema di riferimento



1. Accoppiare nel genitore...



1. Esempio

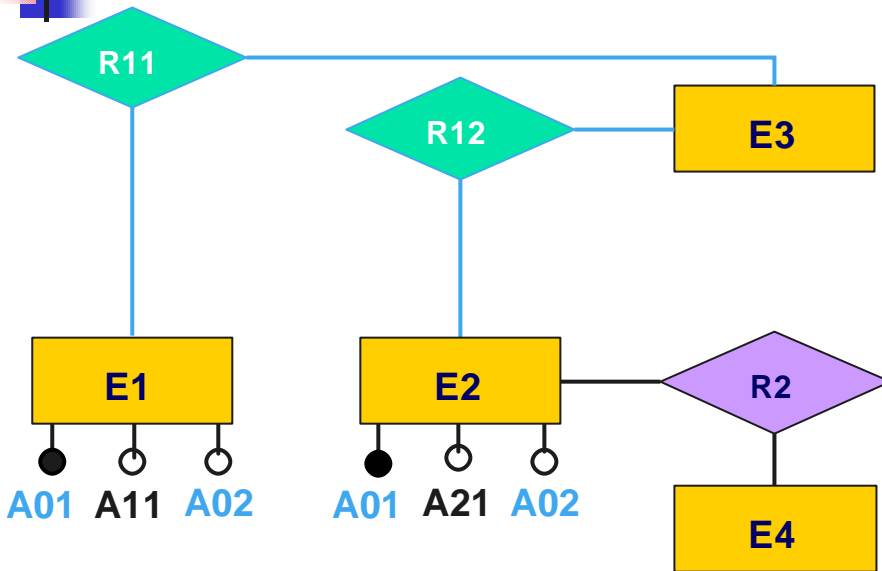


Collasso verso l'alto: osservazioni

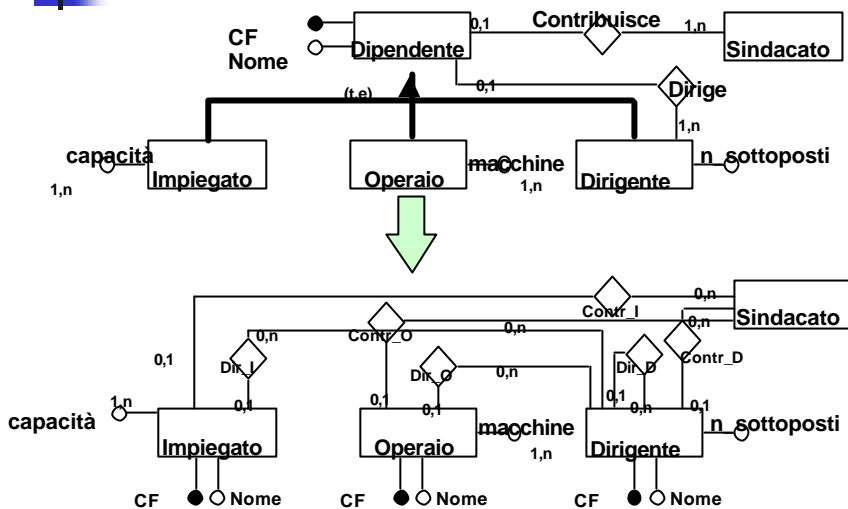
- **“Tipo”** è un attributo **selettore** che specifica se una singola istanza di E appartiene a una delle N sottoentità
- **Copertura**
 - **totale esclusiva**: Tipo ha N valori, quante sono le sottoentità
 - **parziale esclusiva**: Tipo ha N+1 valori; il valore in più serve per le istanze che non appartengono a nessuna sottoentità
 - **sovrapposta**: occorrono tanti selettori quante sono le sottoentità, ciascuno a valore booleano Tipo_i, che è vero per ogni istanza di E che appartiene a E_i; se la copertura è parziale i selettori possono essere tutti falsi, oppure si può aggiungere un selettore
- Le eventuali associazioni connesse alle sottoentità si trasportano su E, le eventuali cardinalità minime diventano 0



2. Accorpare nelle figlie...



2. Esempio

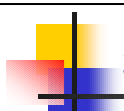
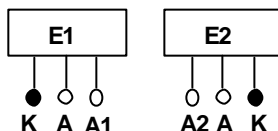




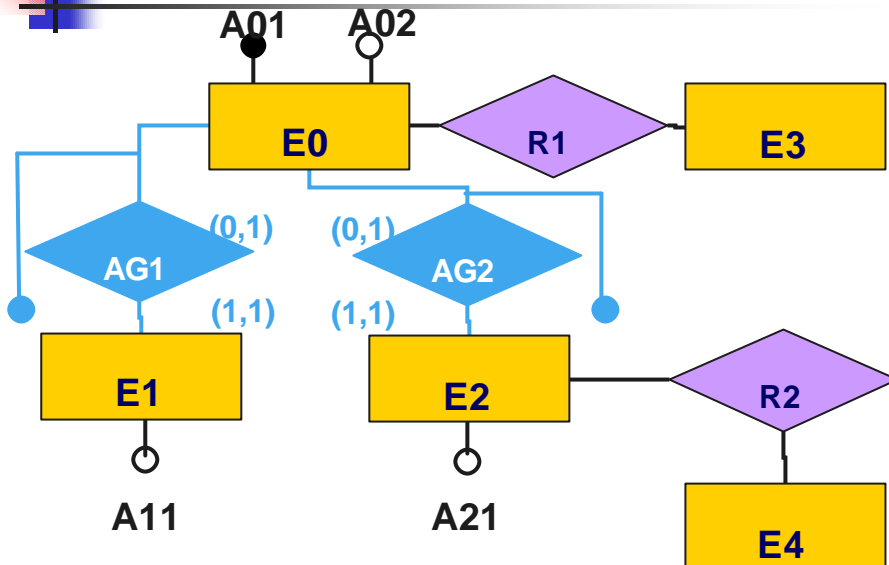
Collasso verso il basso: osservazioni

- Se la copertura non è completa non si può fare
 - non si saprebbe dove mettere le istanze di E che non sono né in E1, né in E2

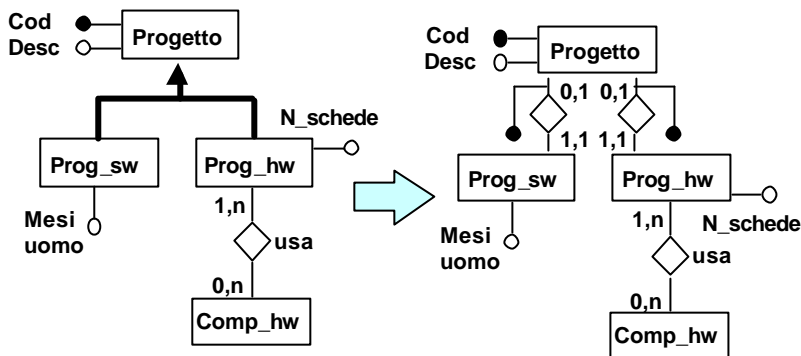
- Se la copertura non è esclusiva introduce ridondanza
 - una certa istanza può essere sia in E1 che in E2, e quindi si rappresentano due volte gli attributi che provengono da E



3. Sostituire con associazioni...



3. Esempio



Cosa conviene fare?

- La scelta fra le alternative si può fare con **metodo simile a quello visto per l'analisi delle ridondanze**, considerando oltre al numero degli accessi anche l'occupazione di spazio
- È possibile seguire alcune **semplici regole generali** (ovvero: **mantieni insieme ciò che viene usato insieme**)

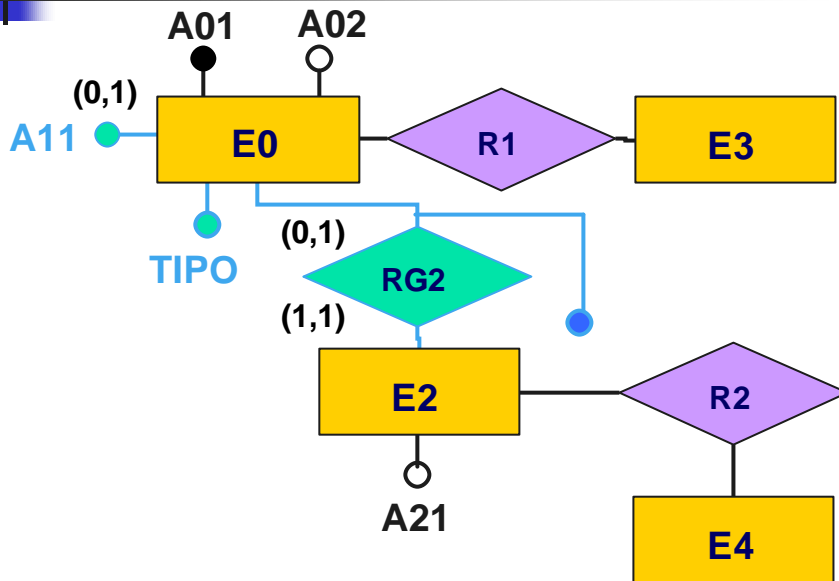
Sol. 1. conviene se gli accessi al genitore e alle figlie sono contestuali

Sol. 2. conviene se gli accessi alle figlie sono distinti (ma è possibile solo con generalizzazioni totali)

Sol. 3. conviene se gli accessi alle entità figlie sono separati dagli accessi al padre

- Sono anche possibili soluzioni "ibride", soprattutto in gerarchie a più livelli

Una soluzione ibrida...

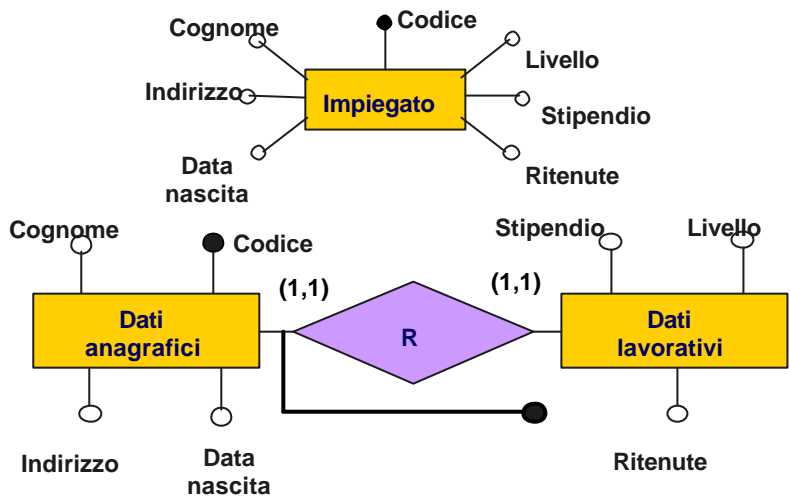


Partizionamenti e accorpamenti

- È possibile ristrutturare lo schema accorpendo o partizionando entità e associazioni
- Tali ristrutturazioni vengono effettuate per rendere più efficienti le operazioni in base al principio già visto, ovvero:
- Gli accessi si riducono:
 - separando attributi di un concetto che vengono acceduti separatamente
 - raggruppando attributi di concetti diversi acceduti insieme
- I casi principali sono:
 - partizionamento "verticale" di entità
 - partizionamento "orizzontale" di associazioni
 - eliminazione di attributi multivalore
 - Accorpamenti di entità e associazioni

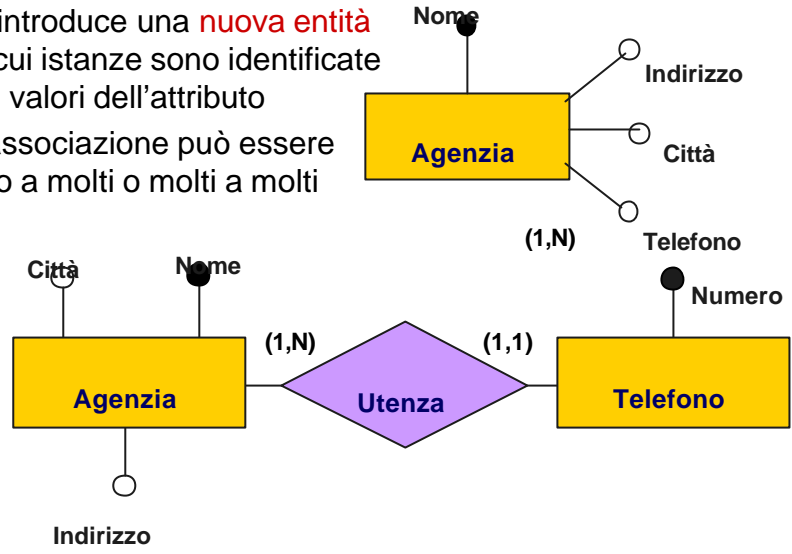
Partizionamento verticale di entità

- Si separano gli attributi in gruppi omogenei

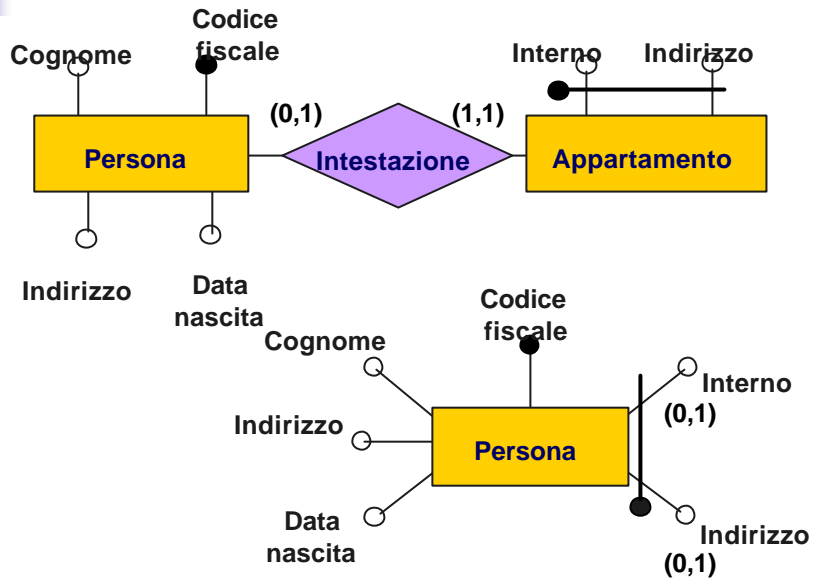


Eliminazione di attributi multivalore

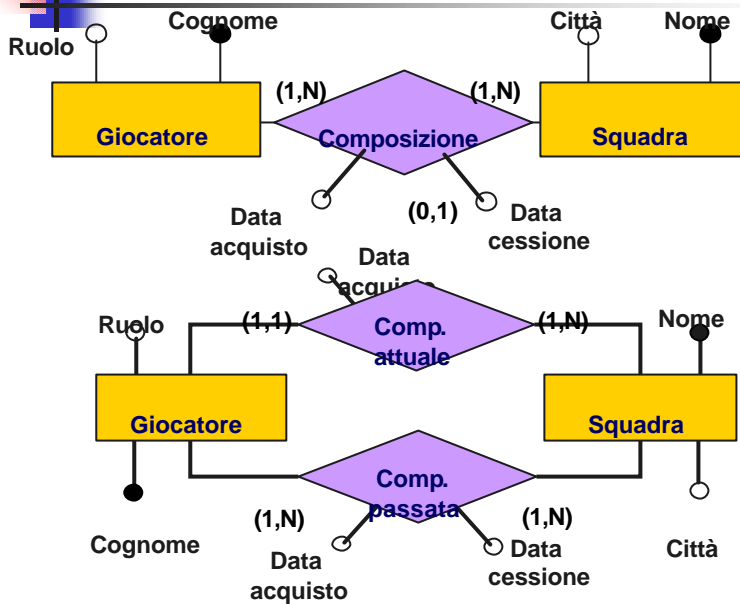
- Si introduce una **nuova entità** le cui istanze sono identificate dai valori dell'attributo
- L'associazione può essere uno a molti o molti a molti



Accorpamento di entità



Partizionamento orizzontale di associazioni





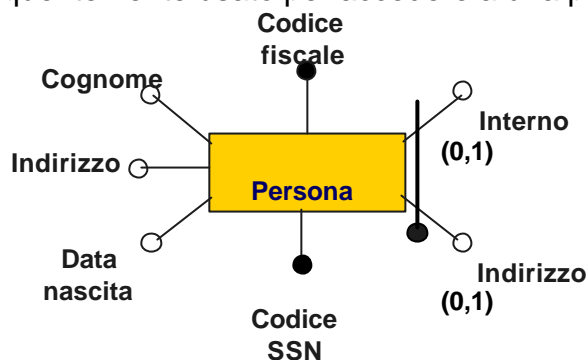
Scelta degli identificatori principali

- È un'operazione indispensabile per la traduzione nel modello relazionale, che corrisponde alla scelta della **chiave primaria**
- I criteri da adottare sono:
 - assenza di opzionalità (valori NULL)
 - semplicità
 - utilizzo nelle operazioni più frequenti o importanti
- Se nessuno degli identificatori soddisfa i requisiti si introducono dei nuovi attributi (dei "codici") allo scopo



Identificatori principali: esempio

- L'identificatore {Interno, Indirizzo} è opzionale, quindi non può essere scelto
- Tra Codice fiscale e Codice SSN la scelta dipende da quale è più frequentemente usato per accedere a una persona

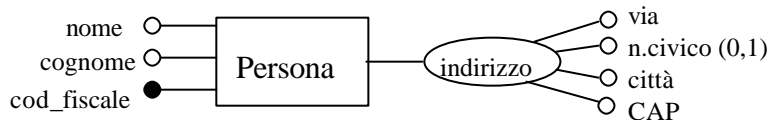




Traduzione delle entità

Idea di base:

- Ogni entità è tradotta con una relazione con gli stessi attributi
 - La **chiave primaria** coincide con l'**identificatore principale** dell'entità
 - Gli **attributi composti** vengono ricorsivamente suddivisi nelle loro componenti, oppure si mappano in un singolo attributo della relazione, il cui dominio va opportunamente definito
 - Per brevità, usiamo l'asterisco (*) per indicare la possibilità di **valori nulli**



Persona(CF, Cognome, Nome, Via, NCivico*, Città, CAP)

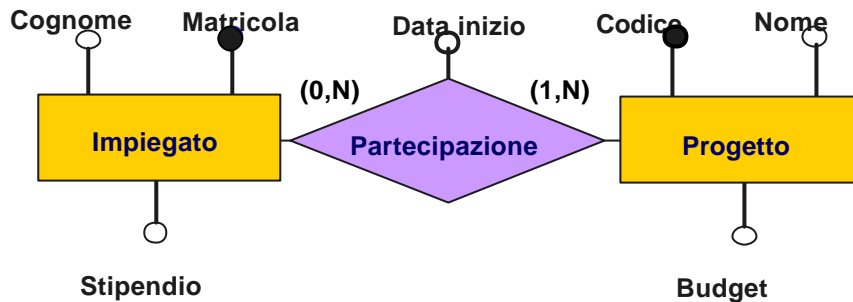


Traduzione delle associazioni

Idea di base:

- Ogni associazione è tradotta con una relazione con gli stessi attributi, cui si aggiungono gli identificatori di tutte le entità che essa collega
 - gli **identificatori delle entità** collegate costituiscono una **superchiave**
 - la **chiave** dipende dalle **cardinalità massime** delle entità nell'associazione
 - Le cardinalità minime determinano, a seconda del tipo di traduzione effettuata, la presenza o meno di valori nulli (e quindi incidono su vincoli e occupazione inutile di memoria)

Entità e associazione molti a molti



Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

Partecipazione(Matricola, Codice, DataInizio)

FK: Matricola REFERENCES Impiegato

FK: Codice REFERENCES Progetto

Nomi delle foreign key

- Non è ovviamente necessario mantenere per gli attributi chiave della relazione che traduce l'associazione gli stessi nomi delle primary key referenziate, ma conviene usare nomi più espressivi

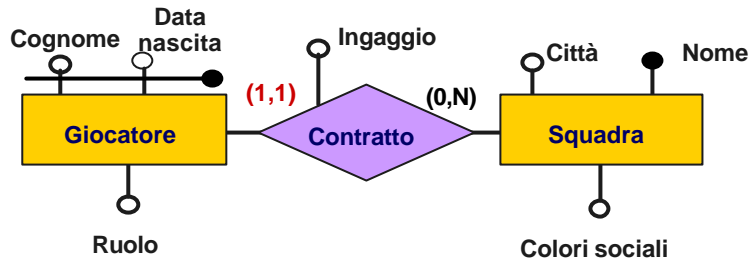
Partecipazione(Impiegato, CodProgetto, DataInizio)

FK: Impiegato REFERENCES Impiegato

FK: CodProgetto REFERENCES Progetto

- Ovviamente se le entità collegate hanno un identificatore con lo stesso nome la ridenominazione è obbligatoria!

Associazioni uno a molti (1)



Giocatore(Cognome, DataNascita, Ruolo)

Squadra(Nome, Città, ColoriSociali)

Contratto(CognGiocatore, DataNascG, **Squadra**, Ingaggio)

FK: (CognGiocatore, DataNascG) REFERENCES Giocatore

FK: Squadra REFERENCES Squadra

Il Nome della Squadra non fa parte della chiave di Contratto (perché?)

Associazioni uno a molti (2)

- Poiché un giocatore ha un contratto con una sola squadra, nella relazione Contratto un giocatore non può apparire in più tuple
- Si può pertanto pensare anche ad una **soluzione più compatta, facente uso di 2 sole relazioni**

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

FK: Squadra REFERENCES Squadra

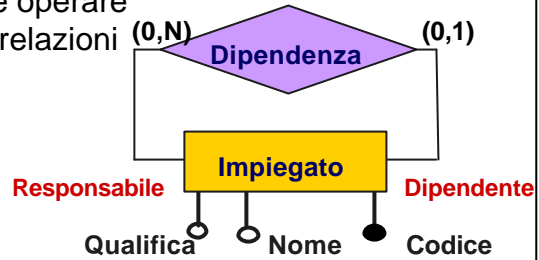
Squadra(Nome, Città, ColoriSociali)

che corrisponde a tradurre l'associazione insieme a Giocatore (ovvero all'entità che partecipa con cardinalità massima 1)

- Se fosse $\text{min-card}(\text{Giocatore}, \text{Contratto}) = 0$, allora gli attributi **Squadra** e **Ingaggio** devono entrambi ammettere valore nullo (e per un giocatore o lo sono entrambi o non lo è nessuno dei due)

Associazioni ad anello uno a molti

- In questo caso è possibile operare una traduzione con 1 o 2 relazioni



1 relazione:

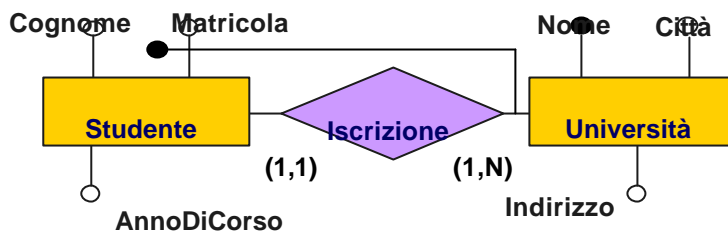
Impiegato(Codice, Nome, Qualifica, Responsabile*)
 FK: Responsabile REFERENCES Impiegato

2 relazioni:

Impiegato(Codice, Nome, Qualifica)
 Dipendenza(Dipendente, Responsabile)
 FK: Dipendente REFERENCES Impiegato
 FK: Responsabile REFERENCES Impiegato

Entità con identificazione esterna

- Nel caso di entità identificata esternamente, si "importa" l'identificatore della/e entità identificante/i.
- L'associazione relativa risulta automaticamente tradotta



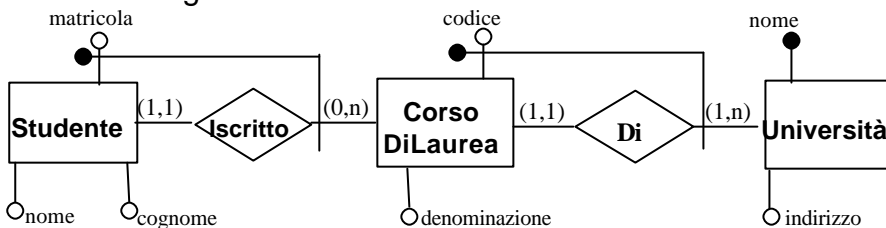
Studente(Matricola, Università, Cognome, AnnoDiCorso)

FK: Università REFERENCES Università

Università(Nome, Città, Indirizzo)

Identificazioni esterne: una precisazione

- Nel caso generale, si possono avere **identificazioni esterne in cascata**
- Per operare correttamente occorre **partire dalle entità non identificate esternamente** e propagare gli identificatori che così si ottengono



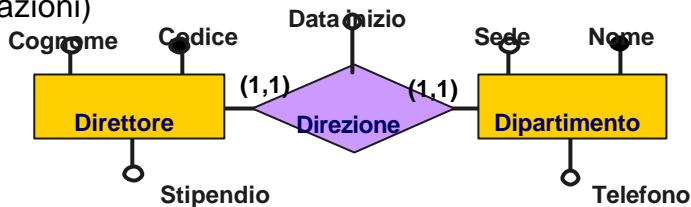
Università(Nome, Indirizzo)

CorsoDiLaurea(Università, Codice, Denominazione)

Studente(Università, CodiceCdL, Matricola, Cognome, Nome)

Associazioni uno a uno (1)

- Si hanno a disposizione varie possibilità (traduzione con 1, 2 o 3 relazioni)



3 relazioni:

Direttore(Codice, Cognome, Stipendio)

Dipartimento(Nome, Sede, Telefono)

Direzio(Direttore, Dipartimento, DataInizio)

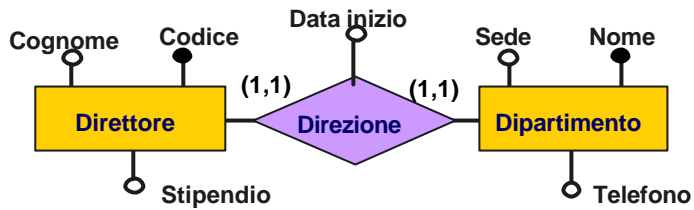
FK:...

Unique(Dipartimento)

L'identificatore di una delle 2 entità viene scelto come chiave primaria, l'altro dà origine a una chiave alternativa

La scelta dipende dall'importanza relativa delle chiavi

Associazioni uno a uno (2)



2 relazioni:

Direttore(Codice, Cognome, Stipendio, Dipartimento, DataInizio)

FK: Dipartimento REFERENCES Dipartimento

Unique(Dipartimento)

Dipartimento(Nome, Sede, Telefono)

oppure

Direttore(Codice, Cognome, Stipendio)

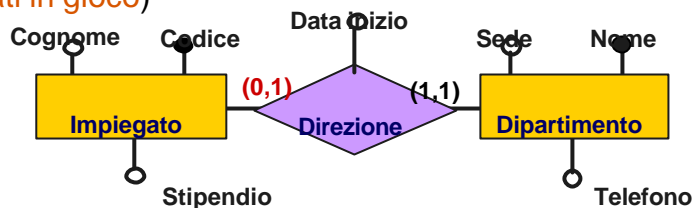
Dipartimento(Nome, Sede, Telefono, Direttore, DataInizio)

FK: Direttore REFERENCES Direttore

Unique(Direttore)

Associazioni uno a uno (3)

- Se $\text{min-card}(E,R) = 0$, tradurre l'associazione R inglobandola in E non è in generale una buona scelta (dipende dai volumi dei dati in gioco)



Impiegato(Codice, Cognome, Stipendio, Dipartimento*, DataInizio*)

FK: Dipartimento REFERENCES Dipartimento

Unique(Dipartimento)

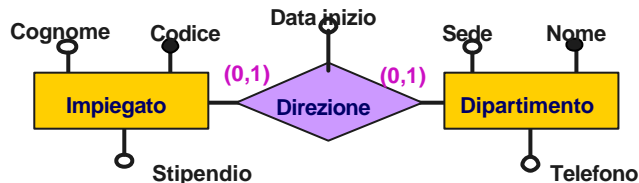
CHECK (((Dipartimento IS NOT NULL) AND (DataInizio IS NOT NULL)) OR
((Dipartimento IS NULL) AND (DataInizio IS NULL)))

Dipartimento(Nome, Sede, Telefono) **TROPPI VALORI NULLII!**

Associazione uno a uno (4)

- In linea di principio la traduzione con una sola relazione non andrebbe qui considerata, in quanto corrisponde a un accorpamento di entità, oggetto della fase di ristrutturazione. In ogni caso:

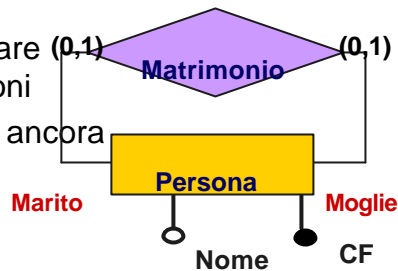
- Se $\text{min-card}(E1,R) = \text{min-card}(E2,R) = 1$ si avranno **due chiavi, entrambe senza valori nulli** (la chiave primaria è "la più importante")
- Se $\text{min-card}(E1,R) = 0$ e $\text{min-card}(E2,R) = 1$ la chiave derivante da E1 ammetterà valori nulli, e **la chiave primaria si ottiene da E2**
- Se $\text{min-card}(E1,R) = \text{min-card}(E2,R) = 0$ entrambe le chiavi hanno valori nulli, quindi si rende necessario **introdurre un codice**



ImpDip(CodiceImpDip, CodicImp*, ..., Dipartimento*, ..., DataInizio*) 51

Associazioni ad anello uno a uno

- In questo caso è possibile operare una traduzione con 1 o 2 relazioni
- La traduzione con 1 relazione è ancora problematica se entrambe le partecipazioni sono opzionali



1 relazione:

Persona(Codice, CFUomo*, NomeUomo*, CFDonna*, NomeDonna*)

2 relazioni:

Persona(CF, Nome)

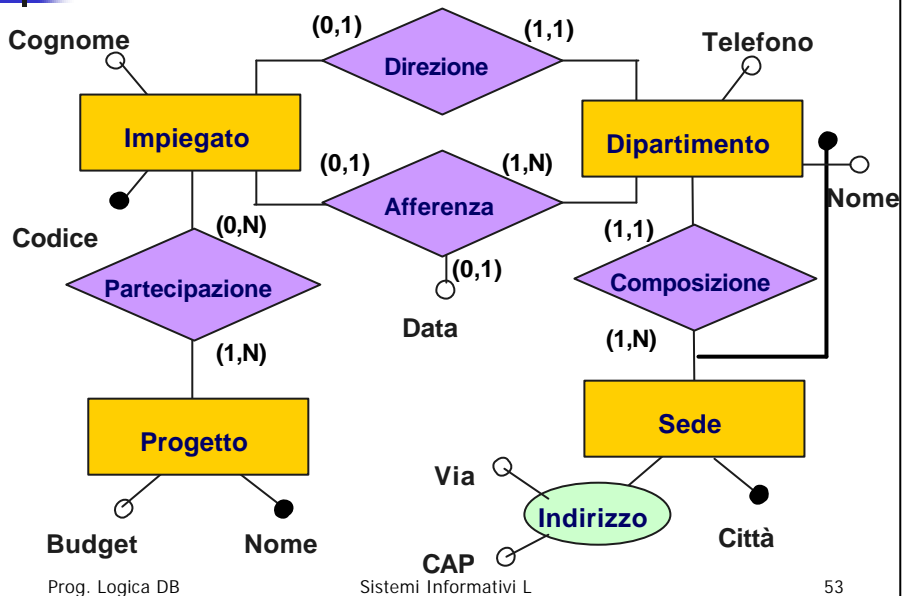
Matrimonio(Marito, Moglie)

FK: Marito REFERENCES Persona

FK: Moglie REFERENCES Persona

Unique (Moglie)

Esempio di riferimento



Schema logico relazionale

- Per le entità E che partecipano ad associazioni sempre con $\max\text{-card}(E,R) = n$ la traduzione è immediata:
 - Sede(Città, Via, CAP)**
 - Progetto(Nome, Budget)**
- 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 CASE, 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 $\max\text{-card}(E2,R) = K$, con K “piccolo”, possono al limite essere tradotte con 1 sola relazione, prevedendo K repliche degli attributi di E2 (es. tipico: numeri di telefono)



Riassumiamo:

- 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