

La progettazione logica

Ristrutturazione



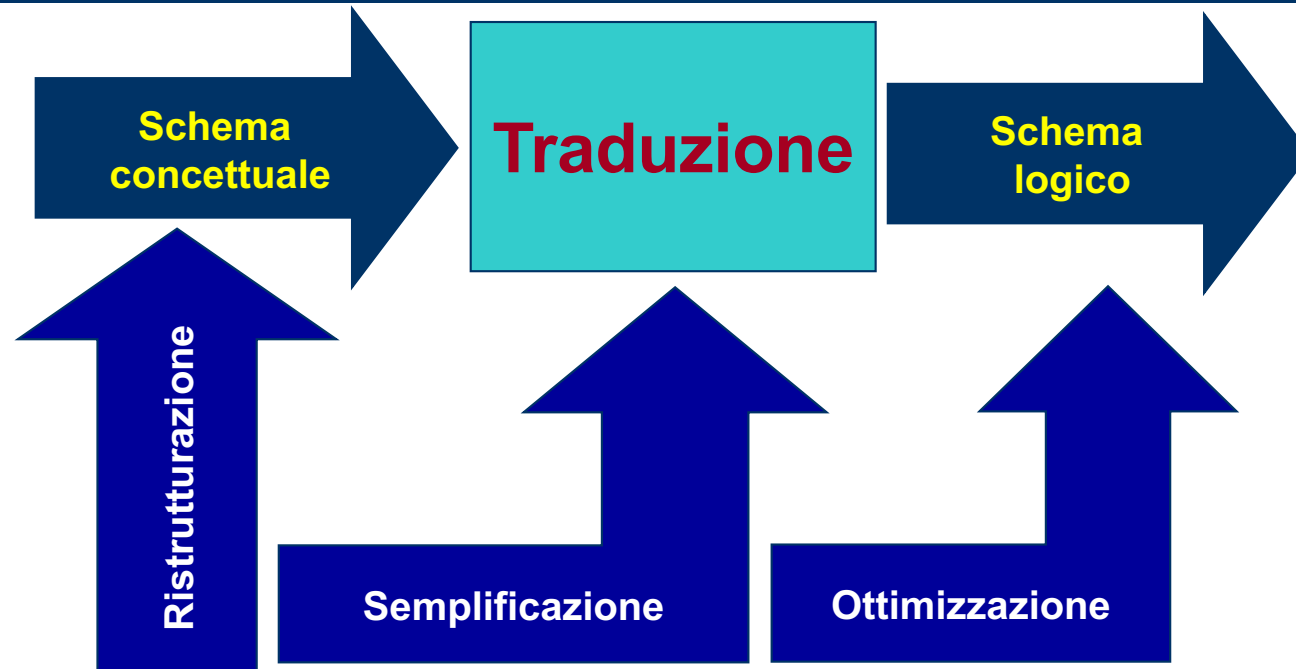
Progettazione Logica

- 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 e operativa, ma indipendente dai dettagli fisici.

La progettazione Logica

- **Scopo:**
 - Costruire uno schema logico in cui siano rappresentate le stesse informazioni presenti nello schema E-R.
- **Non è una semplice traduzione:**
 - Semplificare la rappresentazione per facilitare la traduzione:
 - non tutti i costrutti del modello E-R sono supportati, ad esempio, dal modello relazionale.
 - Ottimizzare il progetto:
 - parametri di prestazioni basati sulle caratteristiche delle operazioni

Il procedimento



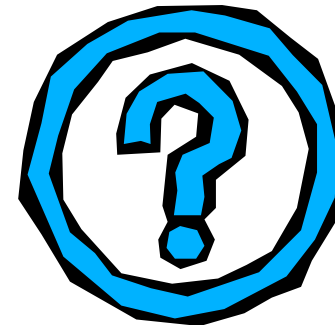
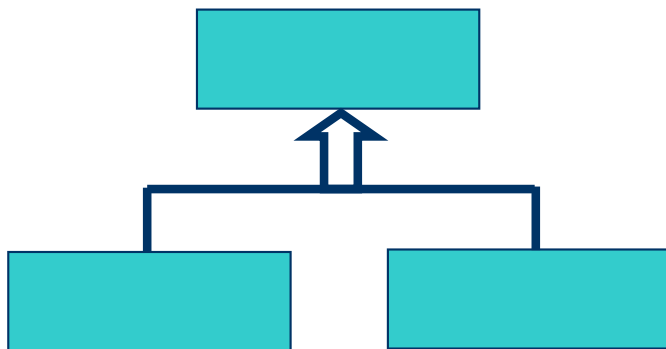
- Prima di operare la traduzione si rende necessario ristrutturare lo schema concettuale:
 - Per semplificare la traduzione
 - Per ottenere uno schema logico ottimizzato

Semplificazione

- La ristrutturazione dello schema concettuale si rende necessaria perché non tutti gli schemi E-R possono essere tradotti nel modello logico (nella fattispecie, è impossibile tradurre attributi composti/multivalore e gerarchie/generalizzazioni)



A	B	C



Il risultato finale

- Al termine delle fasi di ristrutturazione e traduzione si otterrà:
 - Lo schema logico finale nel modello scelto
 - Per il modello relazionale l'insieme di tabelle da istanziare
 - Eventualmente da ottimizzare (normalizzazione)
 - I vincoli di integrità
 - Da apporre alle tabelle
 - Da riportare in appositi programmi

Carico Applicativo

- Serve disporre del carico in termini
 - Di quantità di informazioni
 - Numero e tipo operazioni
- Volume dei dati.
 - Numero di occorrenze di entità ed associazioni
 - Dimensione (media) degli attributi
- Operazioni
 - Caratteristiche
 - Interattive o Batch
 - Frequenza
 - Dati interessati

Alcune notazioni

- Tipo concetto
 - E per entità
 - R per associazioni
- Tipo operazione
 - L per lettura
 - S per scrittura
 - I per interattiva
 - B per batch
- Si considera il seguente costo temporale
 - 1 op di scrittura = 2 op di lettura
$$S = 2L$$

Dimensionamento

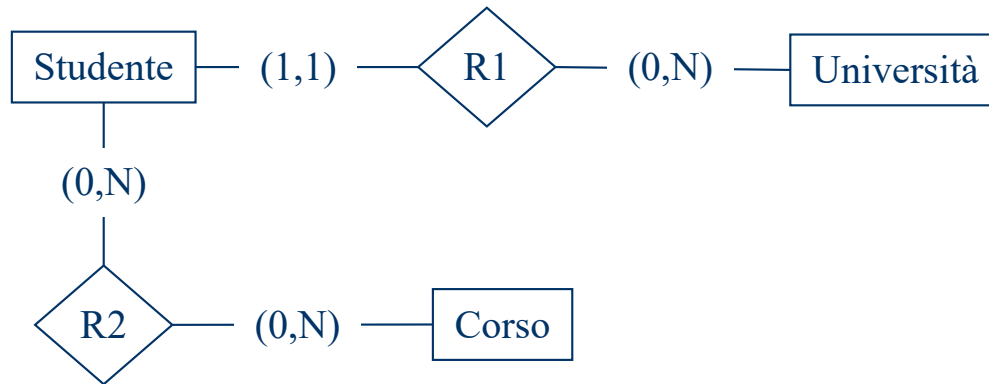


Tavola dei Volumi

Studente	10.000
Università	100
Corso	2.000
R1	10.000
R2	50.000

Si noti che mediamente un corso ha almeno 5 studenti (10.000/2.000)

Carico applicativo

Operazione richiesta

Trova i dati degli studenti afferenti a un corso, I, 3 V/G

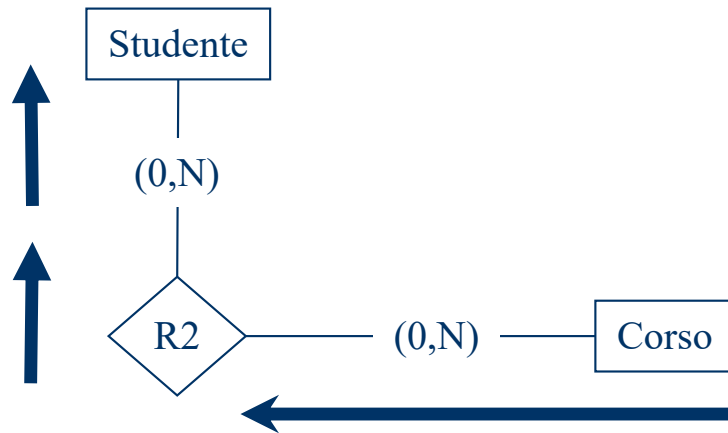
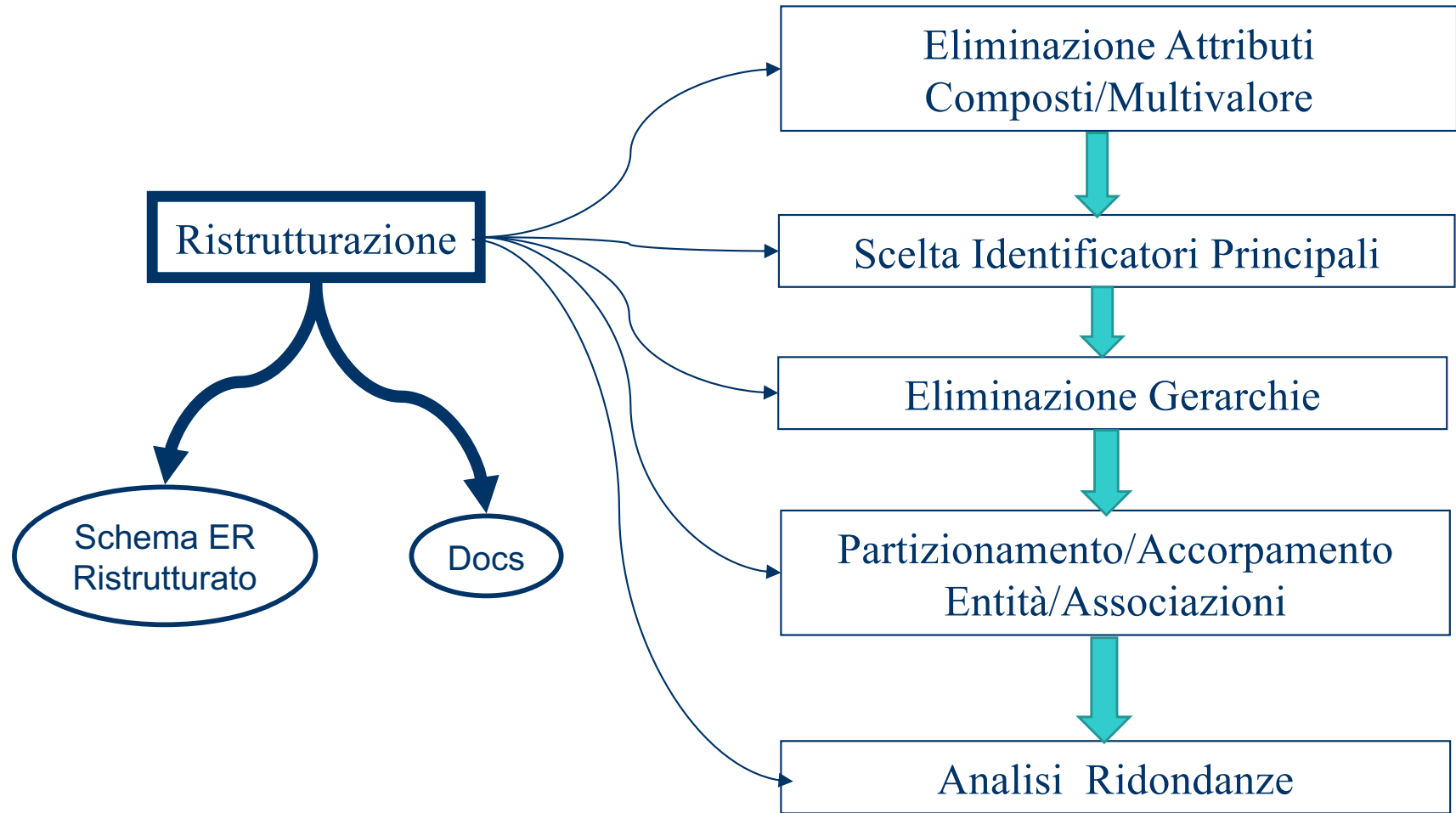


Tavola degli Accessi

Corso	E	1	L
R2	R	5	L
Studente	E	5	L

Costo dell'operazione: $11 * 3 = 33$ L al giorno

Ristrutturazione



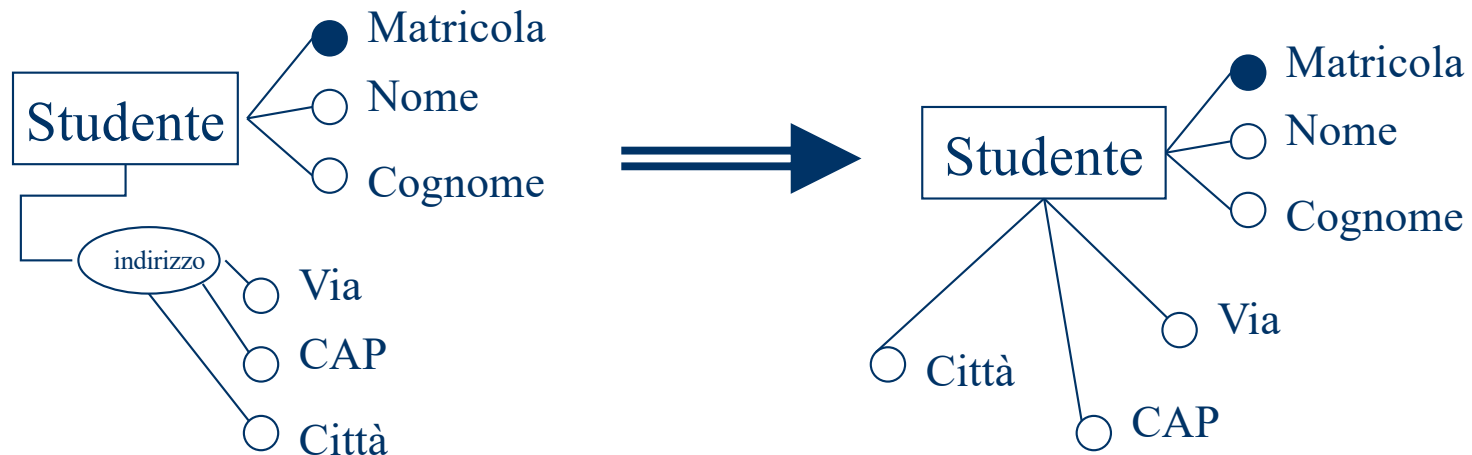
I passi

- Eliminazione di attributi multivalore 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 gerarchie
 - Sulla base dei vincoli di generalizzazioni
- Partizionamento/accorpamento di entità e relazioni
 - Sulla base di valutazioni di convenienza
- Analisi delle ridondanze
 - si decide se mantenere o eliminare le ridondanze (sulla base di analisi delle prestazioni del sistema)

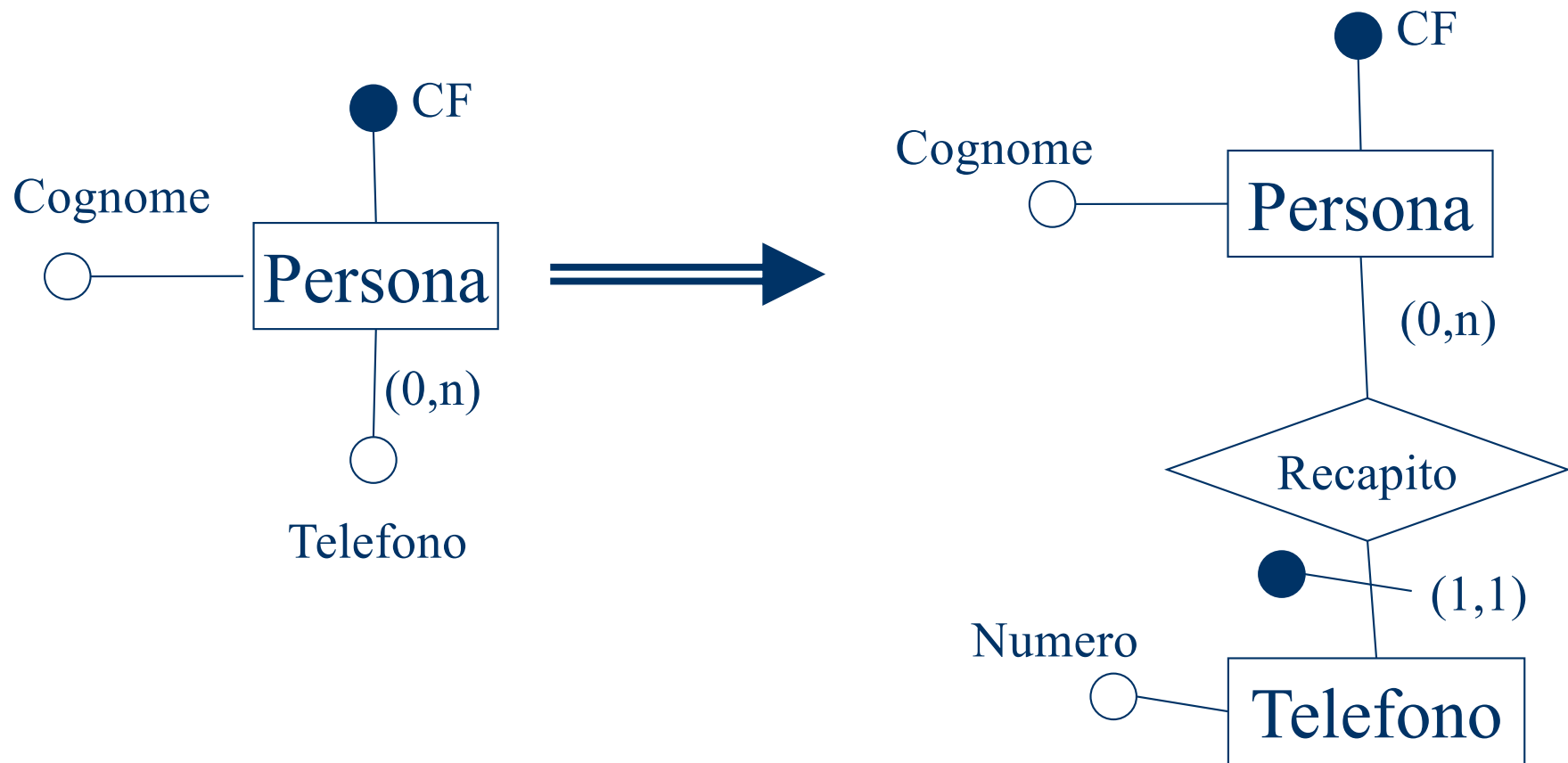
Attributi multivalore / composti

- 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

Eliminazione Attributi Composti



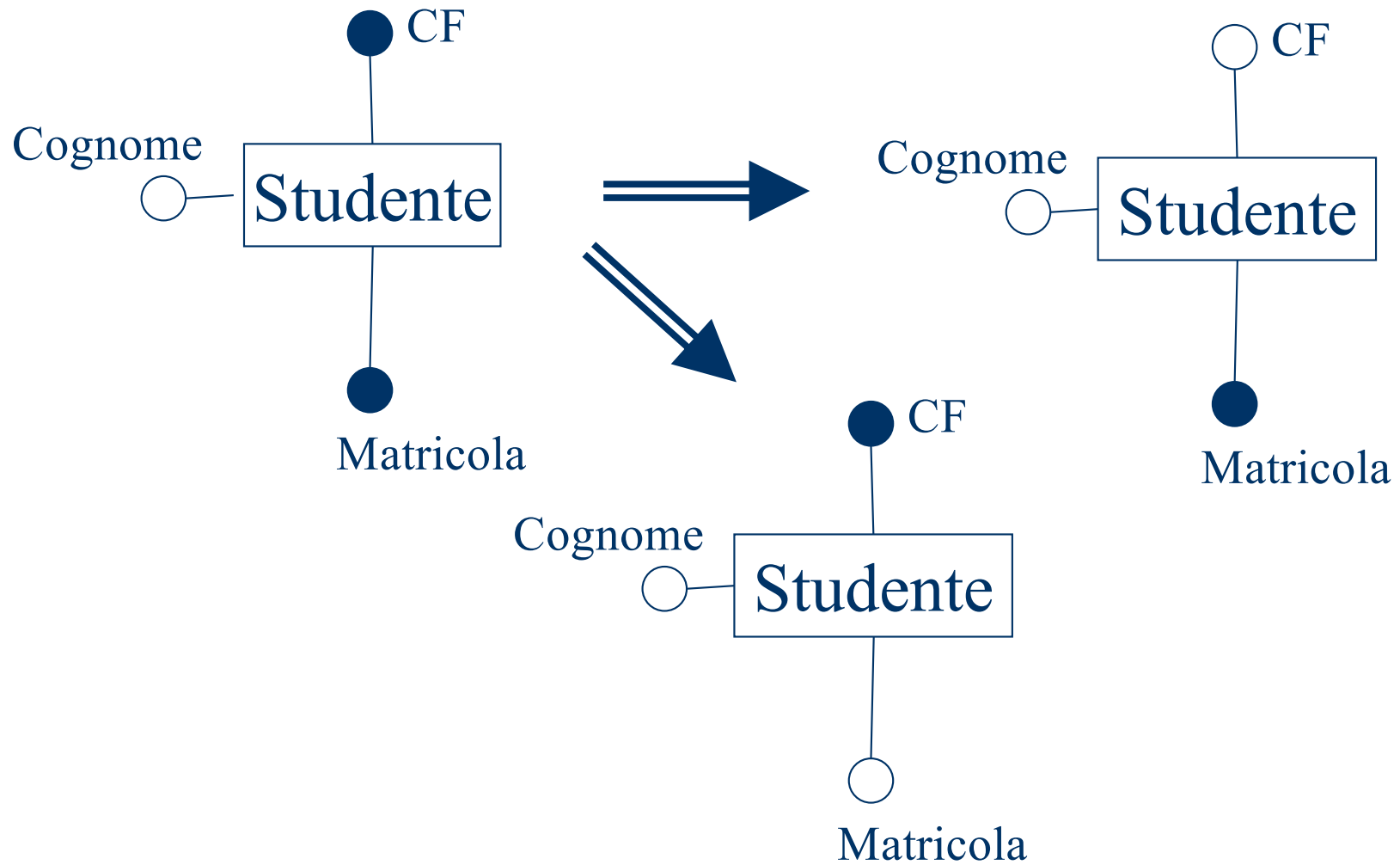
Eliminazione Attributi Multivalore



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

Scelta Identificatori Principali



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

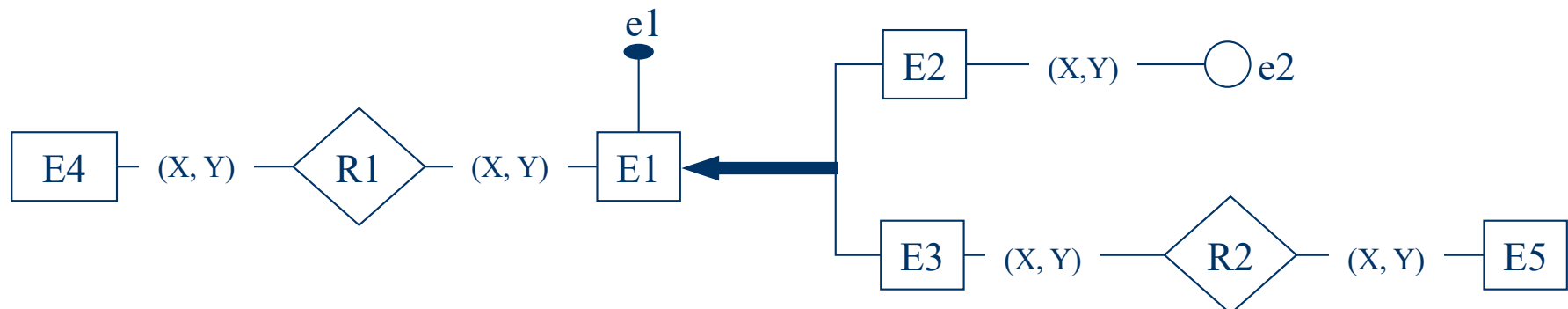
- Nella documentazione devono essere riportati tutti gli altri identificatori che chiameremo secondari
- In particolare vanno segnalati quelli sui quali sono definite delle operazioni
 - In fase di progettazione fisica si potranno attivare ulteriori indici detti anch'essi secondari
 - Per rendere più efficiente l'accesso ai dati
 - Da usare in alternativa agli indici primari

Eliminazione gerarchie

- Le gerarchie di generalizzazione sono costrutti potenti usati nella progettazione concettuale per evidenziare le dipendenze tra le informazioni esistenti nella realtà
- Non trovano, però, nei modelli logici delle basi dati un immediato riscontro
- Le gerarchie di generalizzazione impongono una trasformazione del costrutto usato per rappresentarle in una combinazione di costrutti base (entità e relazioni)

Eliminazione delle gerarchie

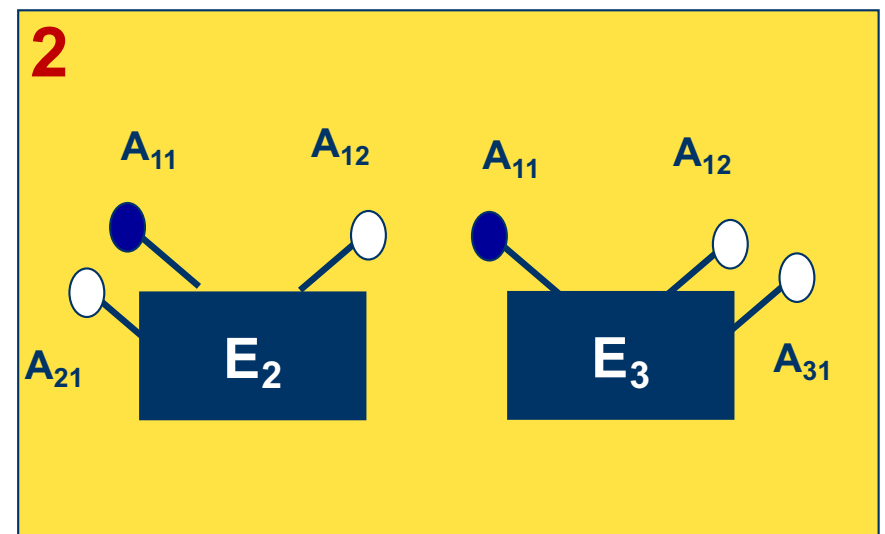
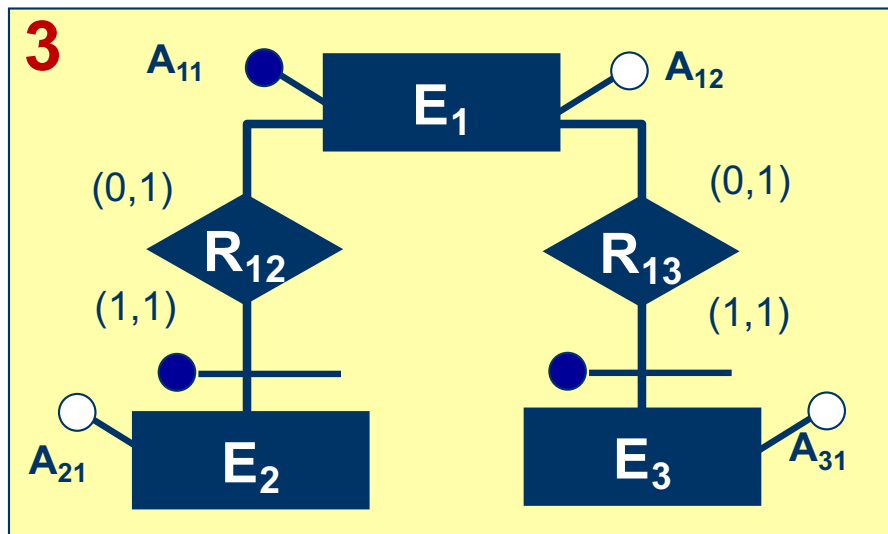
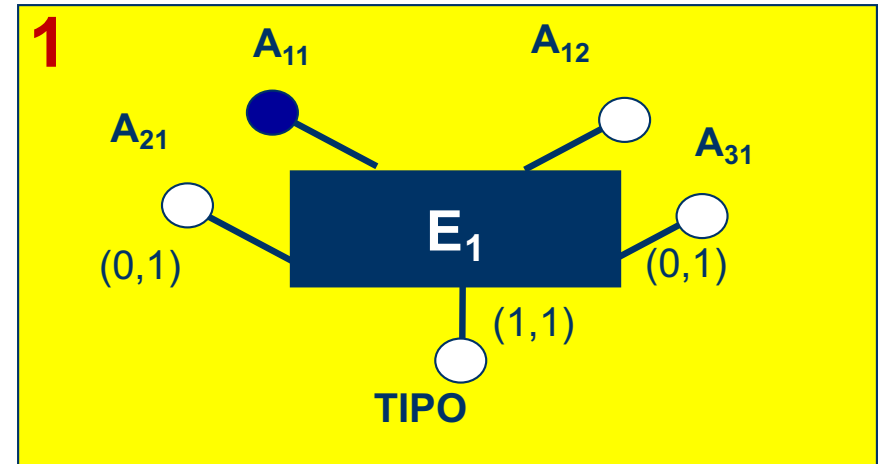
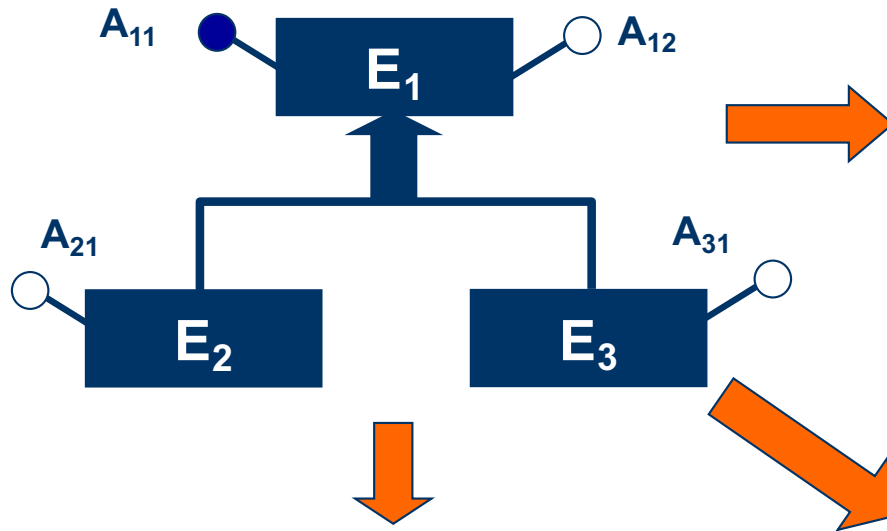
- Tre soluzioni possibili
 - Accorpamento delle entità figlie nell'entità padre
 - Accorpamento dell'entità padre nelle entità figlie
 - Sostituzione della generalizzazione con associazioni



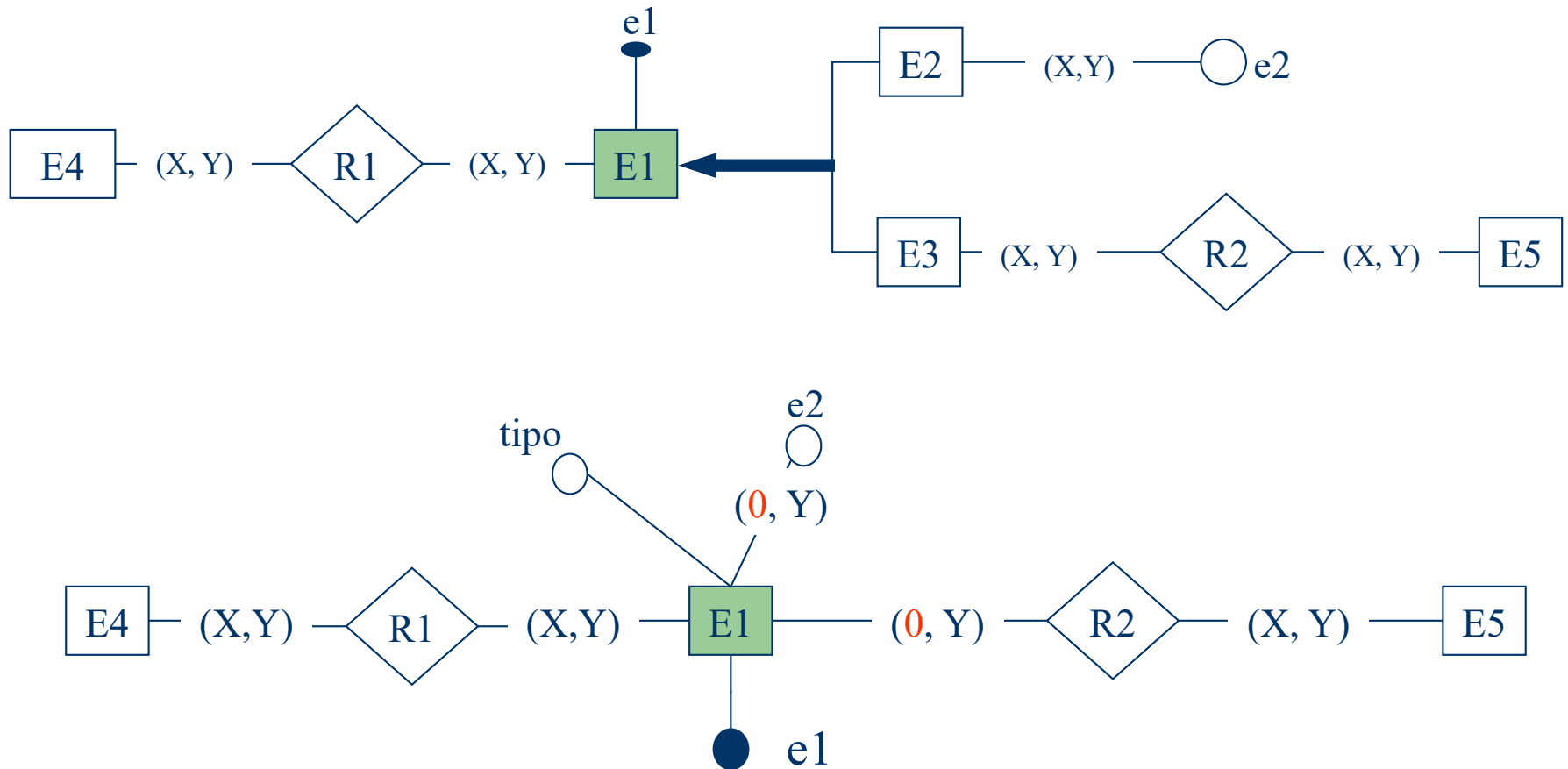
Criteri di scelta

- Tipologia dei vincoli di copertura
 - totale disgiunta
 - parziale disgiunta
 - totale sovrapposta
 - parziale sovrapposta
- Numero e tipo di operazioni su attributi o entità
- Complessità dello schema
 - Numero e tipo di legami con altre entità

Le trasformazioni



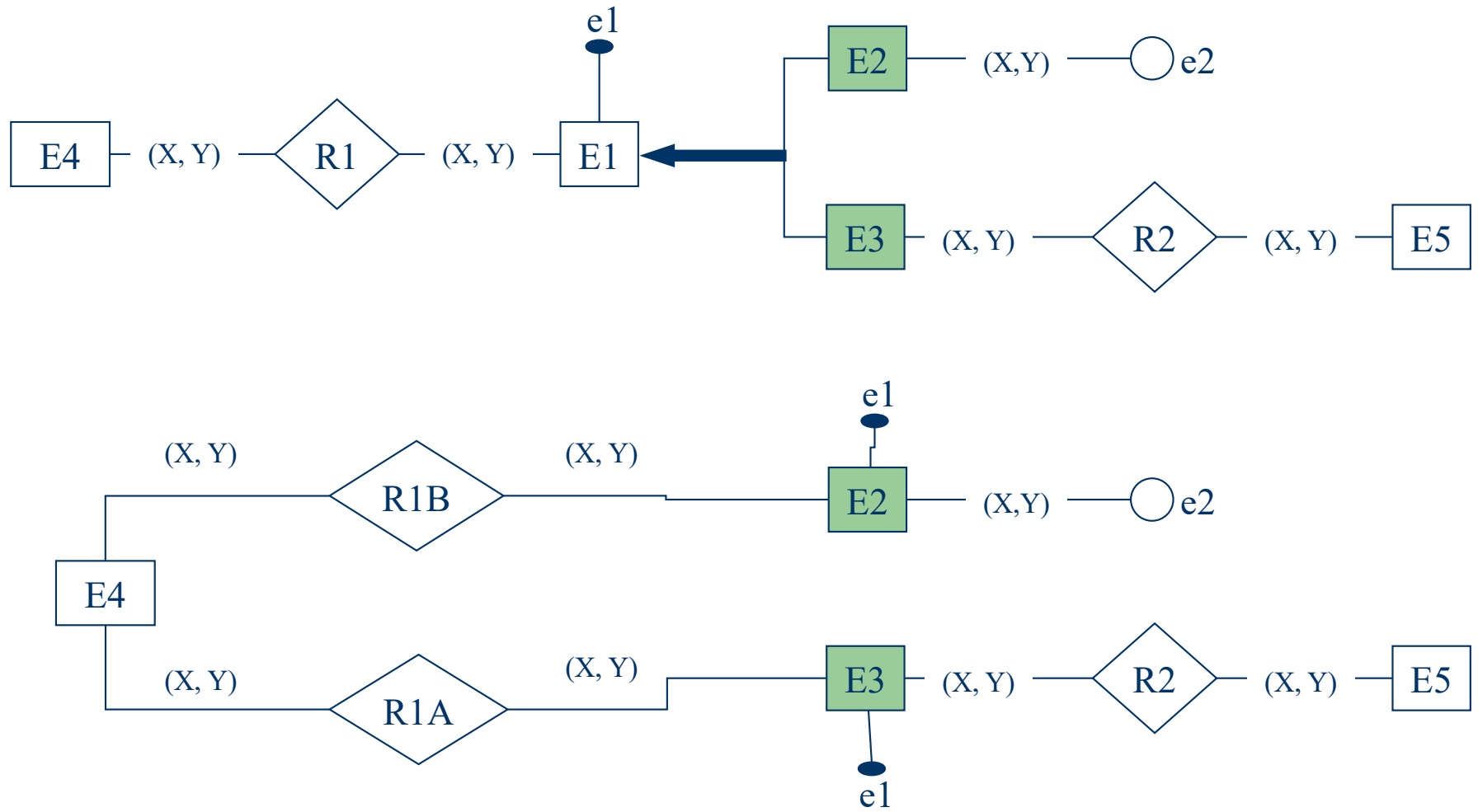
Accorpamento dei figli nel padre



Accorpamento dei figli nel padre

- La trasformazione si applica in presenza di operazioni che non fanno differenza tra le occorrenze e gli attributi di tutte le entità
 - Il numero di accessi diminuisce rispetto alla situazione delle occorrenze e degli attributi distribuiti tra le entità coinvolte
- La soluzione introduce valori nulli per la presenza di occorrenze di una entità figlia non coperte anche dalle altre
 - Occorrenze del padre che non trovano riscontro in alcuna entità figlia (generalizzazione parziale)
 - Occorrenza di una entità figlia che non trova riscontro nell'altre entità figlie (generalizzazione disgiunta)
- Introduzione di un attributo (TIPO) per distinguere le occorrenze delle entità figlie diverse
- Questa prima trasformazione comporta una maggiore occupazione di memoria
 - campi con valori nulli
 - campo TIPO introdotto per distinguere

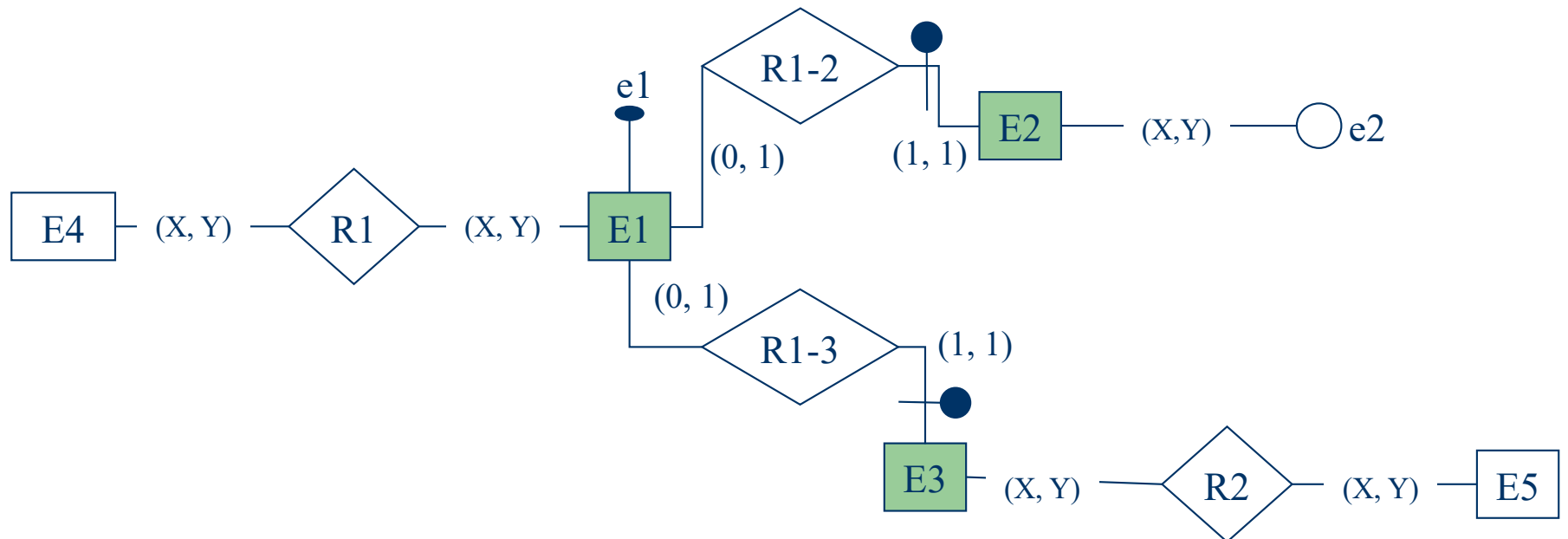
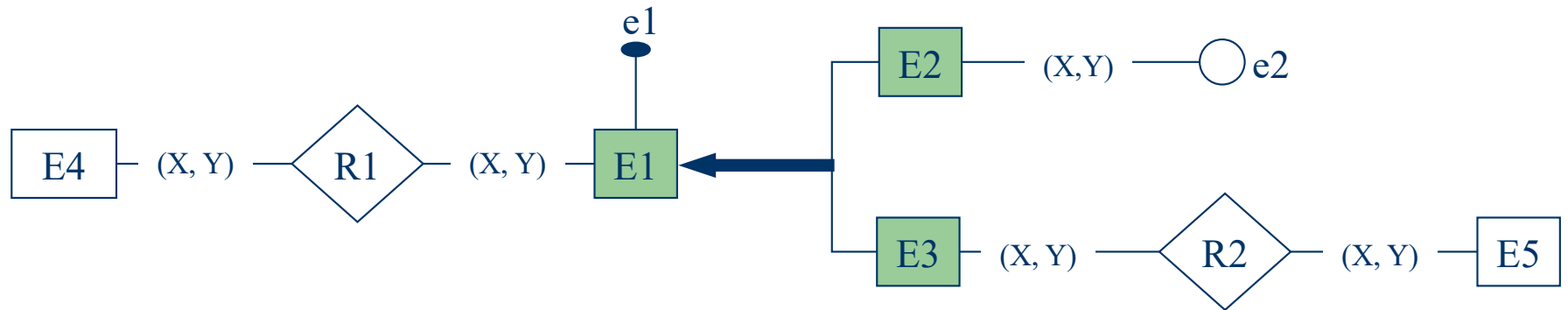
Accorpamento del padre nei figli



Accorpamento del padre nei figli

- La trasformazione **si può applicare solo in presenza di generalizzazioni totali**
 - In caso contrario le occorrenze del padre non appartenenti ad alcuna entità figlia non verrebbero rappresentate
- È conveniente in presenza di operazioni distinte e specializzate sulle entità figlie
- Consente di ottenere un risparmio di memoria perché non genera valori nulli e non abbisogna dell'attributo TIPO

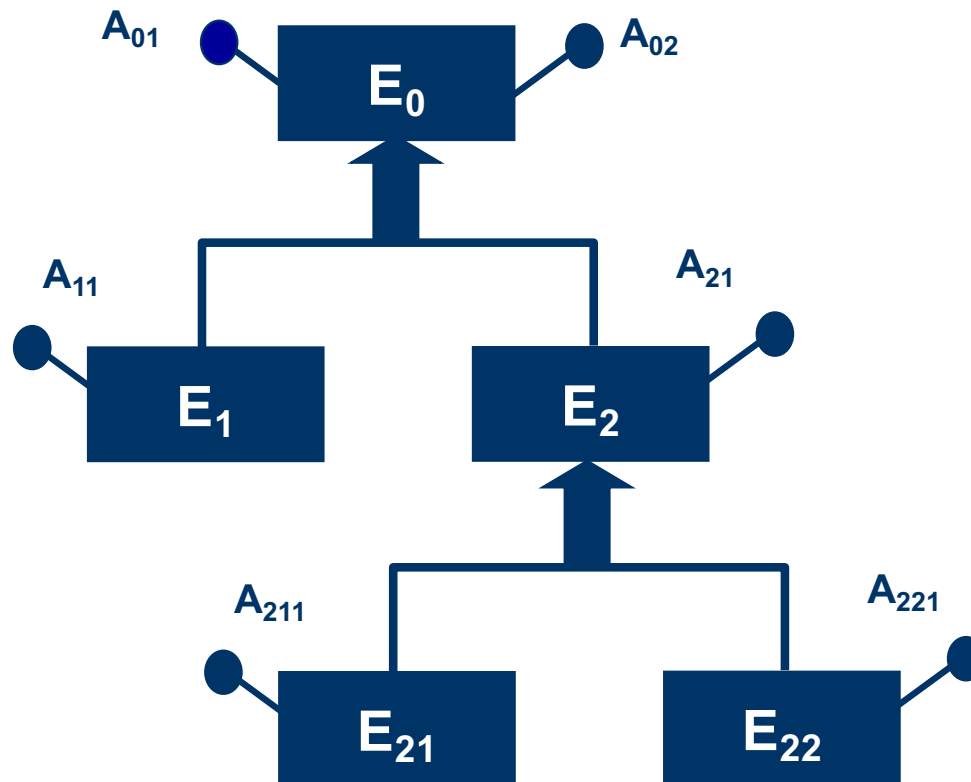
Sostituzione con associazioni



Sostituzione con associazioni

- La trasformazione si può applicare in presenza di generalizzazioni parziali
- È conveniente in presenza di operazioni diverse sia sulla entità padre che sulle entità figlie
- Consente di ottenere un risparmio di memoria perché non genera valori nulli e non abbisogna dell'attributo TIPO
- Prevede un maggior numero di accessi perché si deve accedere all'entità padre per operare sulle entità figlie

Le generalizzazioni a più livelli

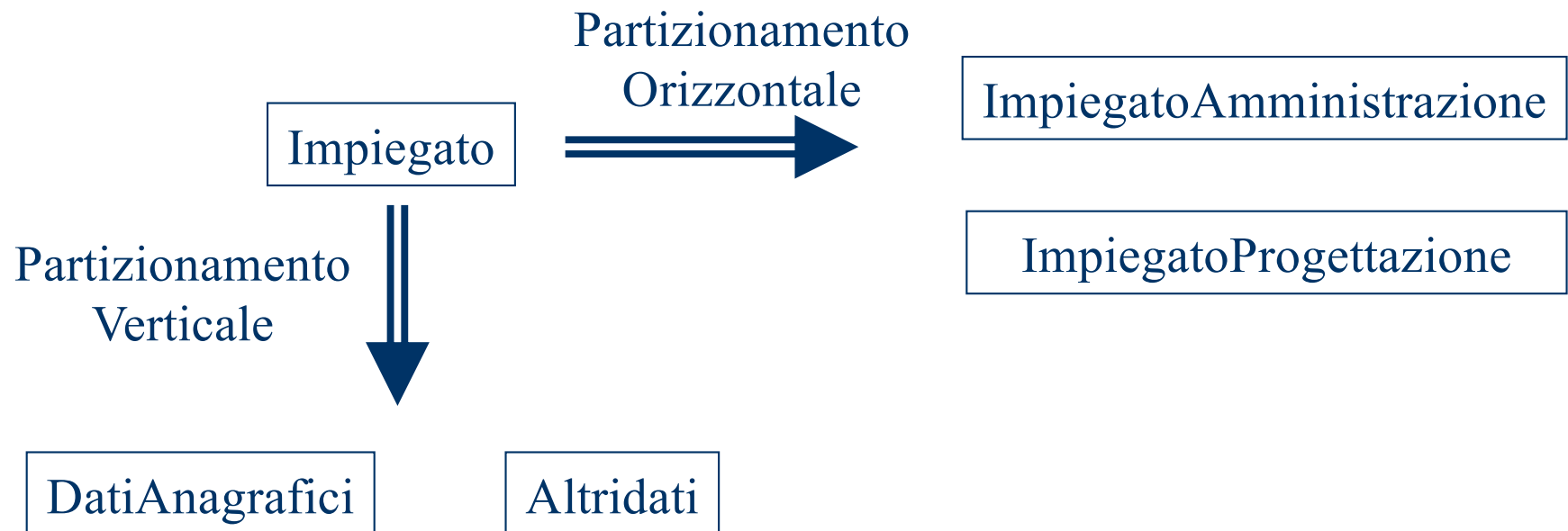


Si procede analizzando una per una le generalizzazioni

La ristrutturazione inizia dai livelli più bassi

Accorpamento/Partizionamento di Entità

- Per migliorare l'efficienza può essere utile partizionare una entità.



- Sulla base degli accessi o della dimensione

Accorpamento/Partizionamento

- In alcuni casi potrebbe essere utile dividere una tabella in due o più tabelle più piccole
 - Nel partizionamento verticale è necessario duplicare l'attributo identificativo
 - Tutte le associazioni che fanno riferimento all'entità vanno replicate
- In alcuni casi potrebbe essere utile accorpare.
 - Le tecniche di valutazione sono simili a quelle usate per le ridondanze
- Divisioni o accorpamenti possono essere applicati anche alle associazioni

Analisi delle Ridondanze

- Ridondanza dei dati
 - Perché ripetuti
 - Perché derivati
 - possono essere ottenuti con un costo computazionale da altri dati
- Pro di una ridondanza:
 - quando serve il dato è già pronto
 - alcune operazioni sono velocizzate
 - riduzione degli accessi necessari per calcolare il dato
- Contro di una ridondanza:
 - il dato ridondante occupa spazio
 - il dato ridondante deve essere tenuto aggiornato

Le ridondanze derivate

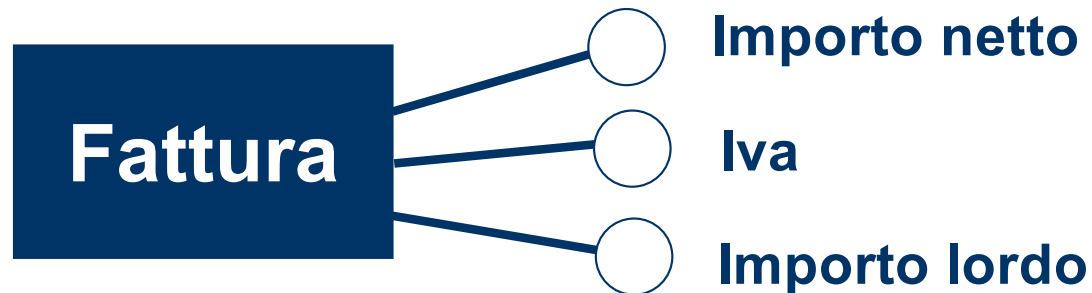
- La presenza di un dato ridondato derivato introduce problemi di consistenza:
 - Il calcolo di un dato costa tanto più quanto più è complessa l'operazione
 - Effettuare l'operazione quando si inseriscono i dati principali implica successivi problemi
 - in modifica
 - in eliminazione
- La decisione di mantenere o eliminare una ridondanza va presa sulla base dei due parametri
 - Costo delle operazioni
 - Occupazione di memoria

I casi di ridondanza derivata

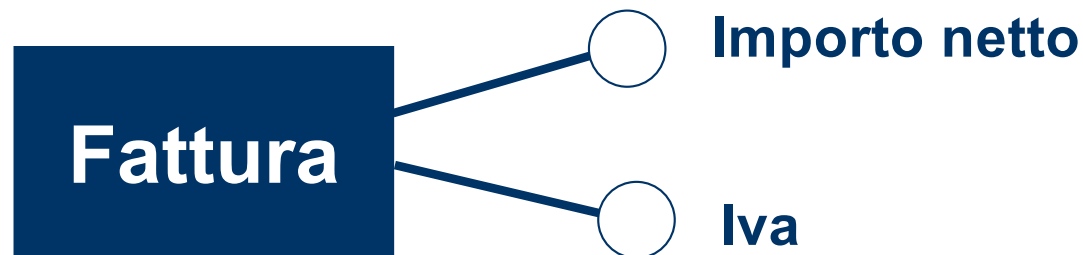
- Esistono quattro possibili casi di ridondanza derivata:
 - Caso 1: un attributo calcolabile da altri attributi dello stesso record della stessa entità o relazione
 - Caso 2: attributi calcolabili da altri attributi appartenenti a entità e relazioni diverse
 - Caso 3: attributi riportanti il conteggio di occorrenze di entità o relazioni
 - Caso 4: attributi derivati da composizioni di relazioni in presenza di cicli

Esempio caso 1

- Un attributo calcolato da altri attributi dello stesso record della stessa entità o associazione



- Ma l'importo lordo potrebbe essere calcolato solo quando serve con risparmio di memoria



Esempio caso 2

- Attributi calcolati da attributi appartenenti a altre entità e relazioni



- Il calcolo dell'importo totale dei prodotti acquistati potrebbe essere calcolato quando serve:
 - Sempre che i prezzi non subiscano modifiche nel tempo

Esempio caso 3

- Calcolo di attributi riportanti il conteggio di occorrenze di entità o relazioni

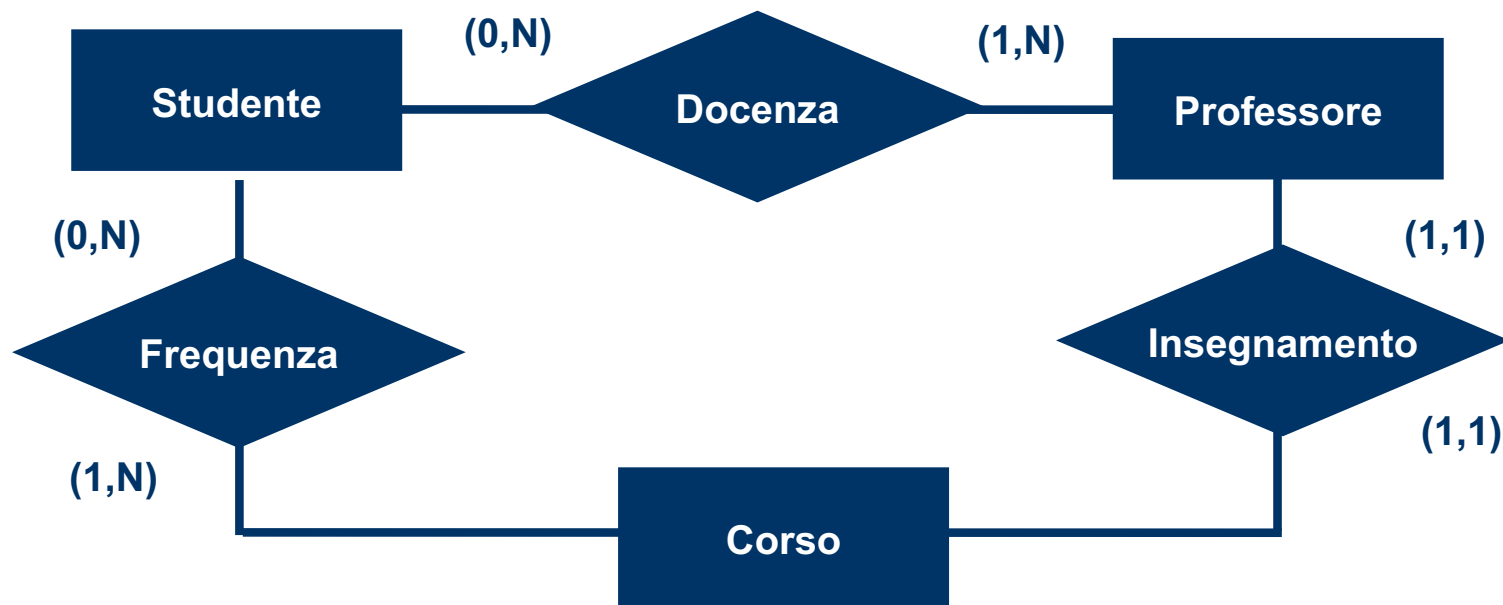
Numero abitanti



- Il numero di abitanti potrebbe essere calcolato quando serve contando le persone che appartengono ad una specifica città
- Diversamente il contatore si deve:
 - Incrementare quando si aggiunge una persona
 - Decrementare quando se ne va

Esempio caso 4

- Attributi derivati da composizioni di relazioni in presenza di cicli



- La relazione Docenza può essere derivata dalle altre due relazioni
- La presenza di cicli non sempre genera ridondanze

Analisi delle Ridondanze

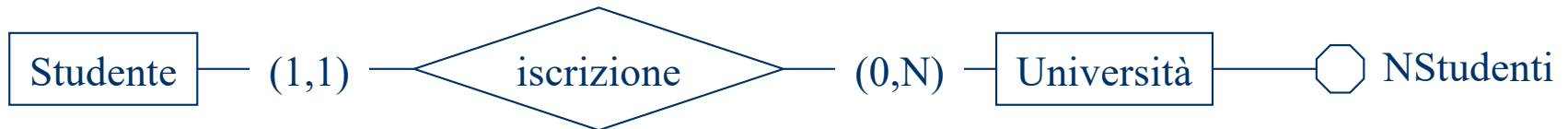


Tavola dei Volumi

Studente	10.000
Università	100 (mediamente 100 studenti per università)
Iscrizione	10.000

● Operazioni

- op1: inserisci nuovo studente, 300 V/G, I
- op2: trova numero studenti Università, 2V/S, I

Con Ridondanza

- Memoria aggiuntiva rispetto all'assenza di ridondanza:
 - 4*100 Byte (int su 4 byte)

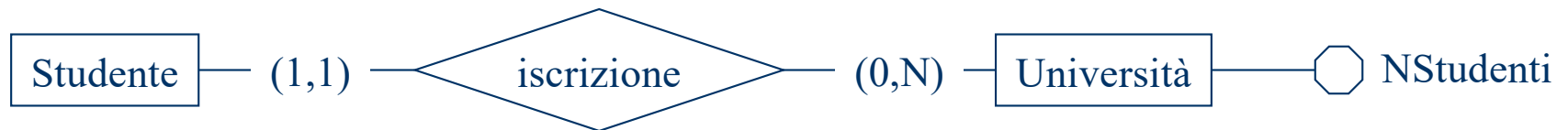


Tavola degli Accessi		op1
Studente	1	S
Iscrizione	1	S
Università	1	L
Università	1	S

$$\begin{aligned} \text{Costo} &= (3S+1L)*300 = 2.100 \text{ L} \\ \text{Costo Giornaliero} &= 2.100 \text{ L} \\ \text{Costo Settimanale (5gg)} &= 10.500 \text{ L} \end{aligned}$$

Tavola degli Accessi		op2
Università	100	L

$$\begin{aligned} \text{Costo} &= 100 \text{ L} \\ \text{Costo Settimanale} &= 200 \text{ L} \end{aligned}$$

Costo Totale = 10.700

Senza Ridondanza



Tavola degli Accessi

Tavola degli Accessi		op1
Studente	1	S
Iscrizione	1	S

Tavola degli Accessi

Tavola degli Accessi		op2
Università	100	L
Iscrizione	10.000	L

$$\text{Costo} = 2S = 4L * 300 = 1.200 \text{ L}$$

$$\text{Costo Giornaliero} = 1.200 \text{ L}$$

$$\text{Costo Settimanale (5gg)} = 6.000 \text{ L}$$

$$\text{Costo} = 10.100 \text{ L}$$

$$\text{Costo Settimanale} = 20.200 \text{ L}$$

Costo Totale = 20.200 L

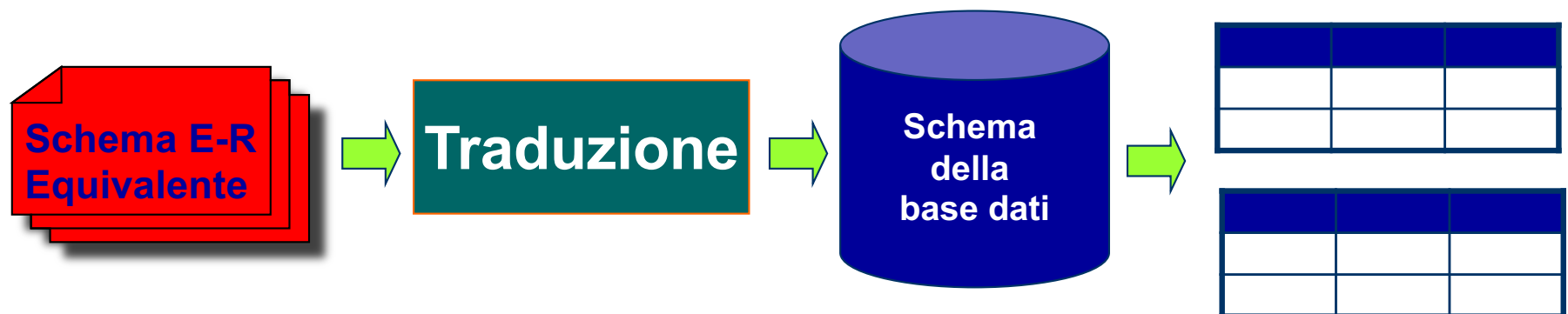
La progettazione logica

Traduzione



Traduzione

- 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



Nota

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

Professore

<u>Nome</u>	Corso	Facoltà



Professore(Nome,Corso,Facoltà)

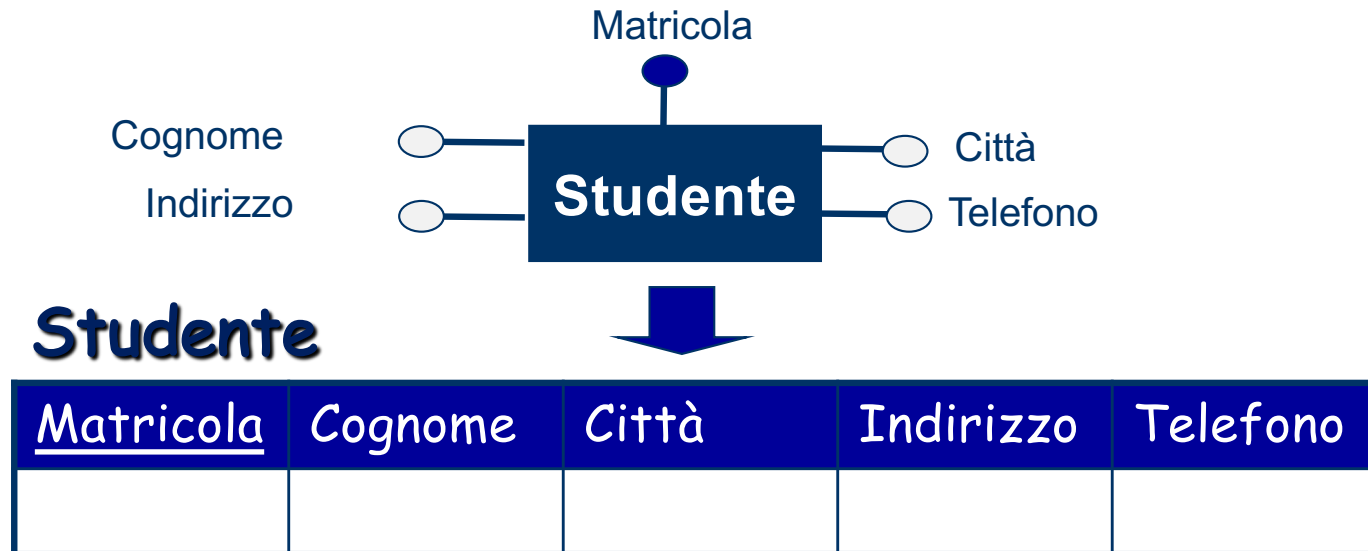
Linee generali

- In generale la traduzione nel modello relazionale trasforma:
 - Le entità in tabelle
 - Le relazioni (associazioni) in tabelle
- Tra le tabelle delle 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 (UaU)
 - Uno a molti (UaM)
 - Molti a molti (MaM)

La traduzione di entità

La traduzione di una entità produce una tabella che ha:

- Per nome il nome dell'entità
- Come attributi (nomi delle colonne) gli attributi della entità
- Per chiave l'identificatore principale della entità



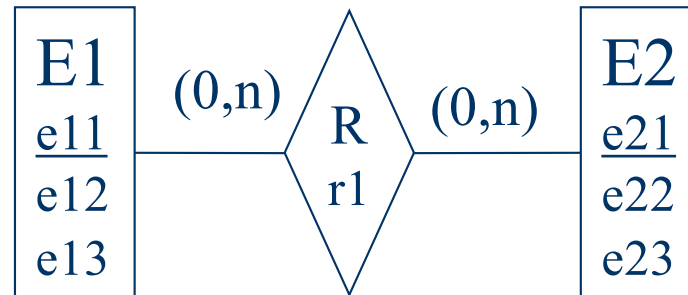
Studente(Matricola, Cognome, Città, Indirizzo, Telefono)

La traduzione di relazioni

La traduzione di una relazione molti a molti produce una tabella che ha:

- Per nome il nome della relazione
- 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 delle chiavi esterne

Schema generale di traduzione



E1 ($e11$, $e12$, $e13$)

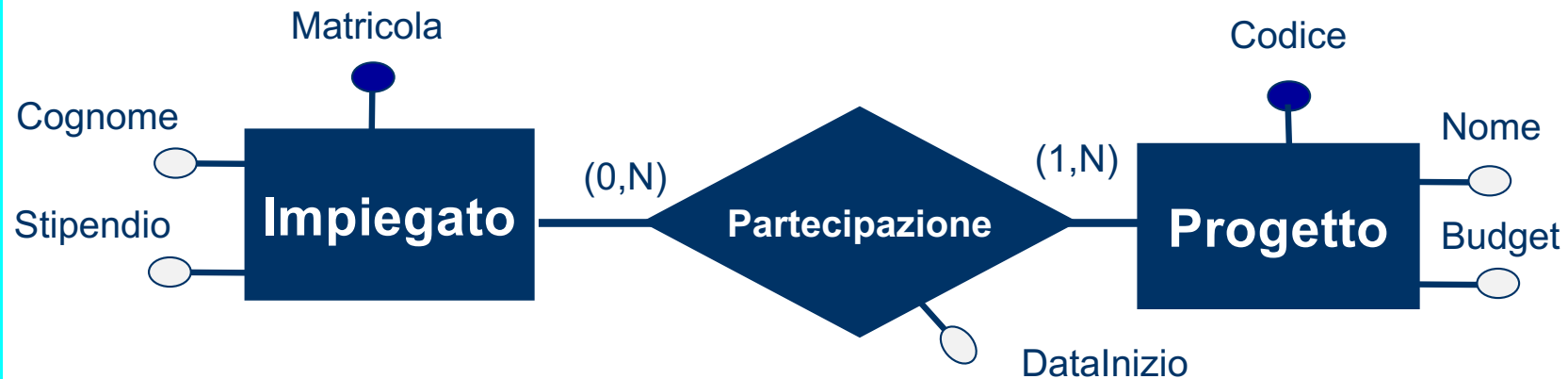
E2 ($e21$, $e22$, $e23$)

R ($e11$, $e21$, $r1$)

Vincoli di integrità referenziale tra

- $e11$ in R ed $e11$ in E1
- $e21$ in R ed $e21$ in E2

Un esempio



Impiegato(Matricola,Cognome,Stipendio)

Progetto(Codice,Nome,Budget)

Partecipazione(Matricola,Codice,DataInizio)

Analisi dell'esempio

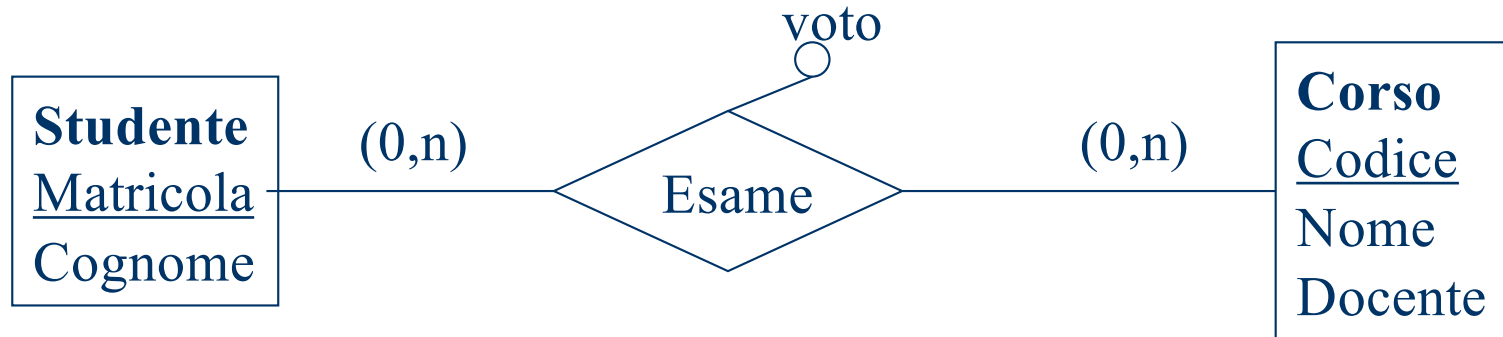
- Per migliorare la comprensione della tabella Partecipazione conviene ridenominare gli attributi presi dalle entità

Partecipazione(Impiegato,Progetto,DataInizio)

- Tra le tabelle si vengono a stabilire i seguenti vincoli di integrità referenziale

Vincoli integrità referenziali			
Tra	Di	E	Di
Impiegato	Partecipazione	Matricola	Impiegato
Progetto	Partecipazione	Codice	Progetto

Traduzione con ridenominazione



Studente (Matricola, Cognome)
Corso (Codice, Nome, Docente)
Esame (Matricola, Codice, voto)

Vincoli di integrità referenziale tra

- Matricola in Esame e Matricola in Studente
- Codice in Esame e Codice in Corso

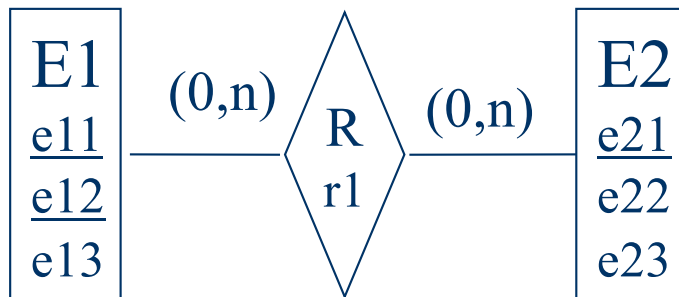
Ridenominazione

Studente (Matricola, Cognome)
Corso (Codice, Nome, Docente)
Esame (Studente, Corso, voto)

Vincoli di integrità referenziale tra

- Studente in Esame e Matricola in Studente
- Corso in Esame e Codice in Corso

Traduzione con chiavi composte



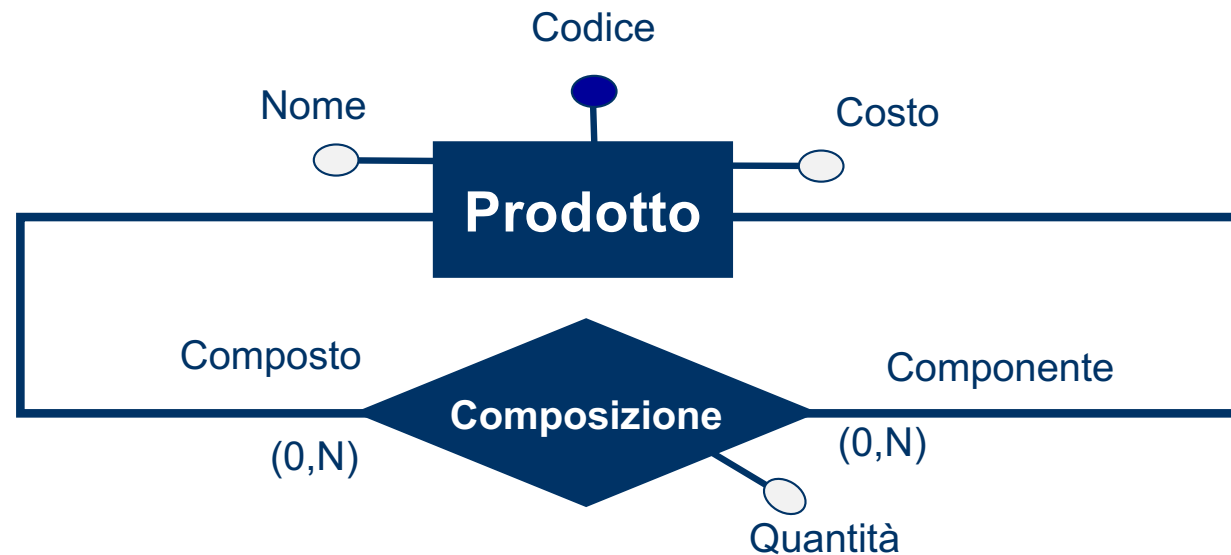
E1 (e11,e12,e13)
E2 (e21,e22,e23)
R (e11,e12,e21,r1)

Vincoli di integrità referenziale tra

- (e11,e12) in R ed (e11,e12) in E1
- e21 in R ed e21 in E2

Necessità della ridenominazione

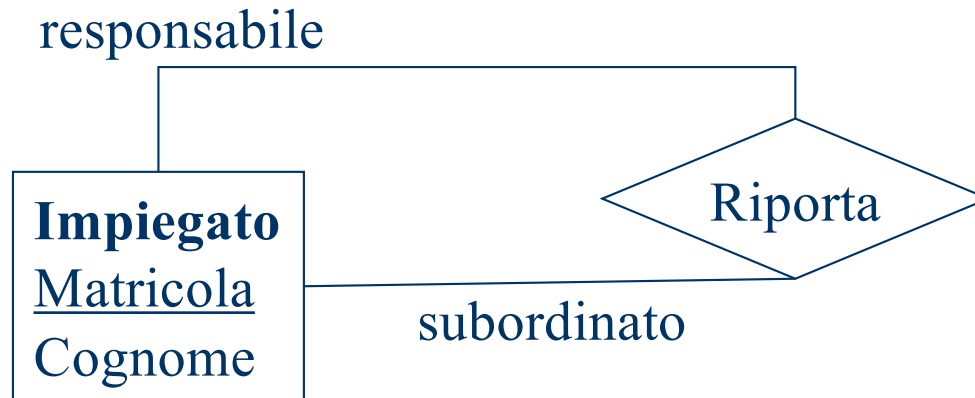
- La ridenominazione degli attributi nella tabella corrispondente alla relazione è obbligatoria nei casi di relazioni ricorsive



Prodotto(Codice, Nome, Costo)

Composizione(Composto, Componente, Quantità)

Traduzione con ridenominazione



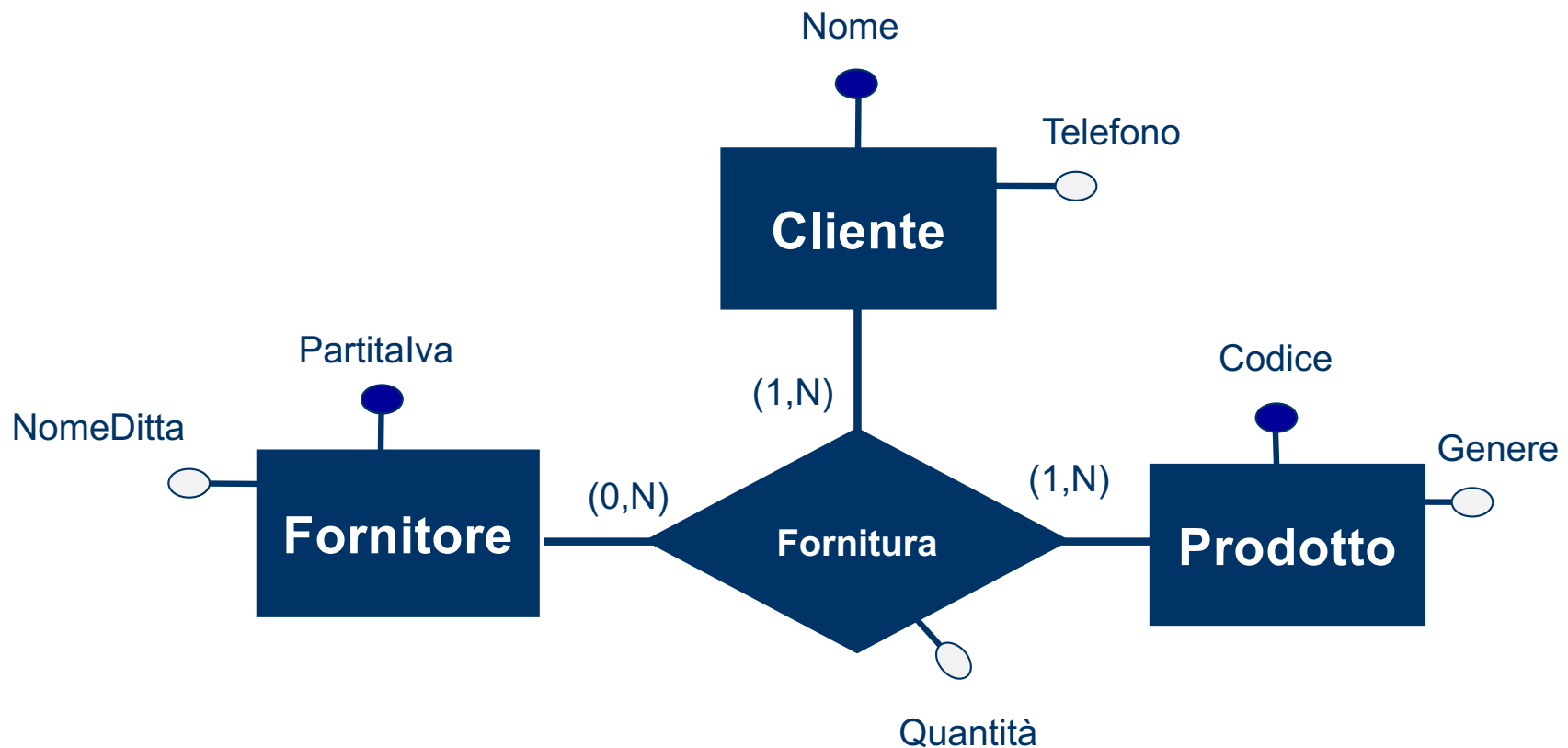
Impiegato (Matricola, Cognome)
Riporta (Subordinato, Responsabile)

Vincoli di integrità referenziale tra

- Subordinato in Riporta e Matricola in Impiegato
- Responsabile in Riporta e Matricola in Impiegato

Esempio

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



La traduzione

Fornitore(PartitaIva, NomeDitta)

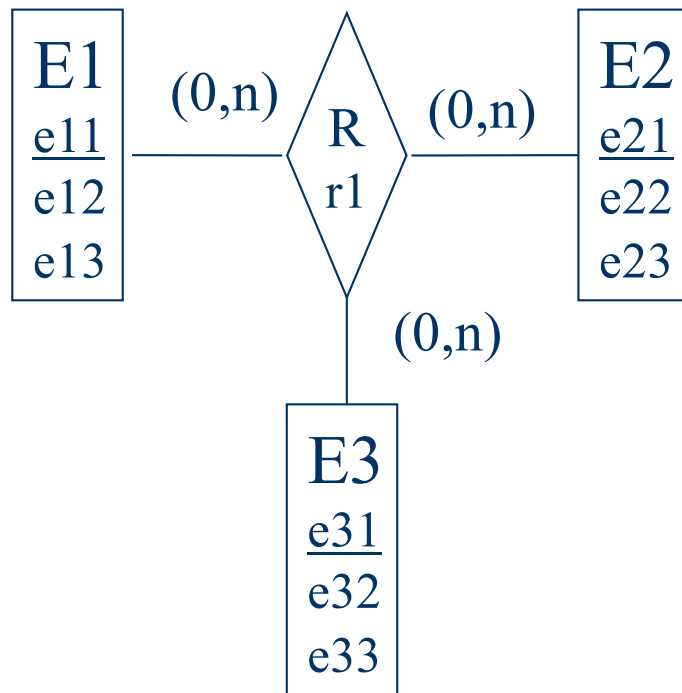
Prodotto(Codice, Genere)

Cliente(Nome, Telefono)

Fornitura(Fornitore, Cliente, Prodotto, Quantità)

Vincoli integrità referenziali			
Tra	Di	E	Di
Fornitore	Fornitura	PartitaIva	Fornitore
Cliente	Fornitura	Nome	Cliente
Prodotto	Fornitura	Codice	Prodotto

Traduzione di associazione a tre



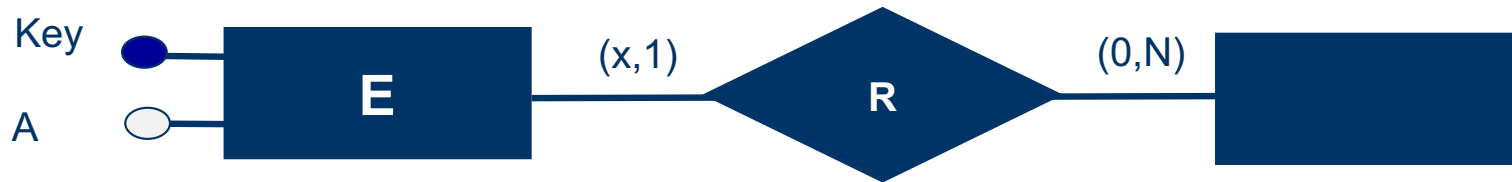
E1 (e11,e12,e13)
E2 (e21,e22,e23)
E3 (e31,e32,e33)
R (e11,e21,e31,r1)

Vincoli di integrità referenziale tra

- e11 in R ed e11 in E1
- e21 in R ed e21 in E2
- e31 in R ed e31 in E3

La traduzione di relazioni UaM

Nelle relazioni uno a molti



- si ha che 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 **Key** è chiave sia per le occorrenze di **E** che per quelle di **R**

Un esempio di traduzione con (1,1)



Giocatore(CodiceFiscale,Nome,Ruolo)

Squadra(Nome,Città)

Contratto(Giocatore,Squadra,Ingaggio)

Vincoli integrità referenziali

Tra	Di	E	Di
Giocatore	Contratto	CodiceFiscale	Giocatore
Squadra	Contratto	Nome	Squadra

Una ottimizzazione

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

Giocatore(CodiceFiscale,Nome,Ruolo,Squadra**,Ingaggio)**

Squadra(Nome,Città)

Vincoli integrità referenziali			
Tra	Di	E	Di
Giocatore	Squadra	Nome	Squadra

- Si noti che nel caso di cardinalità (0,1) questa soluzione presenterà dei valori nulli per gli attributi Squadra e Ingaggio della tabella Giocatore

Identificatori esterni

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



Studente(Matricola, NomeUniversità, Nome, Corso, Anno)

Università(Nome, Città)

Vincoli integrità referenziali

Tra	Di	E	Di
NomeUniversità	Studente	Nome	Università

La traduzione di relazioni UaU

- Alla traduzione delle relazioni uno a uno si estendono le considerazioni fatte per il caso UaM
- 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

Esempio



La prima soluzione:

Dipendente(Codice, Nome, Stipendio, Dipartimento, Anno)

Dipartimento(Nome, Sede)

La seconda soluzione (simmetrica alla prima):

Dipendente(Codice, Nome, Stipendio)

Dipartimento(Nome, Sede, Direttore, Anno)

Le altre soluzioni

- La terza soluzione:

Dipendente(Codice,Nome,Stipendio)

Dipartimento(Nome,Sede)

Direzione(Direttore,Dipartimento,Anno)

- La quarta soluzione (simmetrica alla terza):

Dipendente(Codice,Nome,Stipendio)

Dipartimento(Nome,Sede)

Direzione(Direttore,Dipartimento,Anno)

Il caso di partecipazioni opzionali

- 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
- 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

Esempio



In questo caso si deve tradurre in:

Dipendente(Codice, Nome, Stipendio)

Dipartimento(Nome, Sede, **Direttore**, Anno)