
TRASFORMAZIONE DI SCHEMI

Obiettivo della progettazione logica

- Si tratta di "**tradurre**" lo schema concettuale in uno schema logico relazionale che rappresenti gli stessi dati in maniera **corretta** ed **efficiente**.
- Questo richiede una ristrutturazione del modello concettuale
- Osservazione:
 - Non si tratta di una pura e semplice traduzione. Infatti
 - alcuni costrutti dello schema concettuale non sono direttamente rappresentabili
 - Nel modello logico è necessario tenere conto delle prestazioni

Dati di ingresso e uscita

- Ingresso:
 - schema concettuale
 - informazioni sul carico applicativo (dimensioni dei dati e caratteristiche delle operazioni)
 - modello logico
- Uscita:
 - schema logico
 - documentazione associata

Progettazione logica relazionale

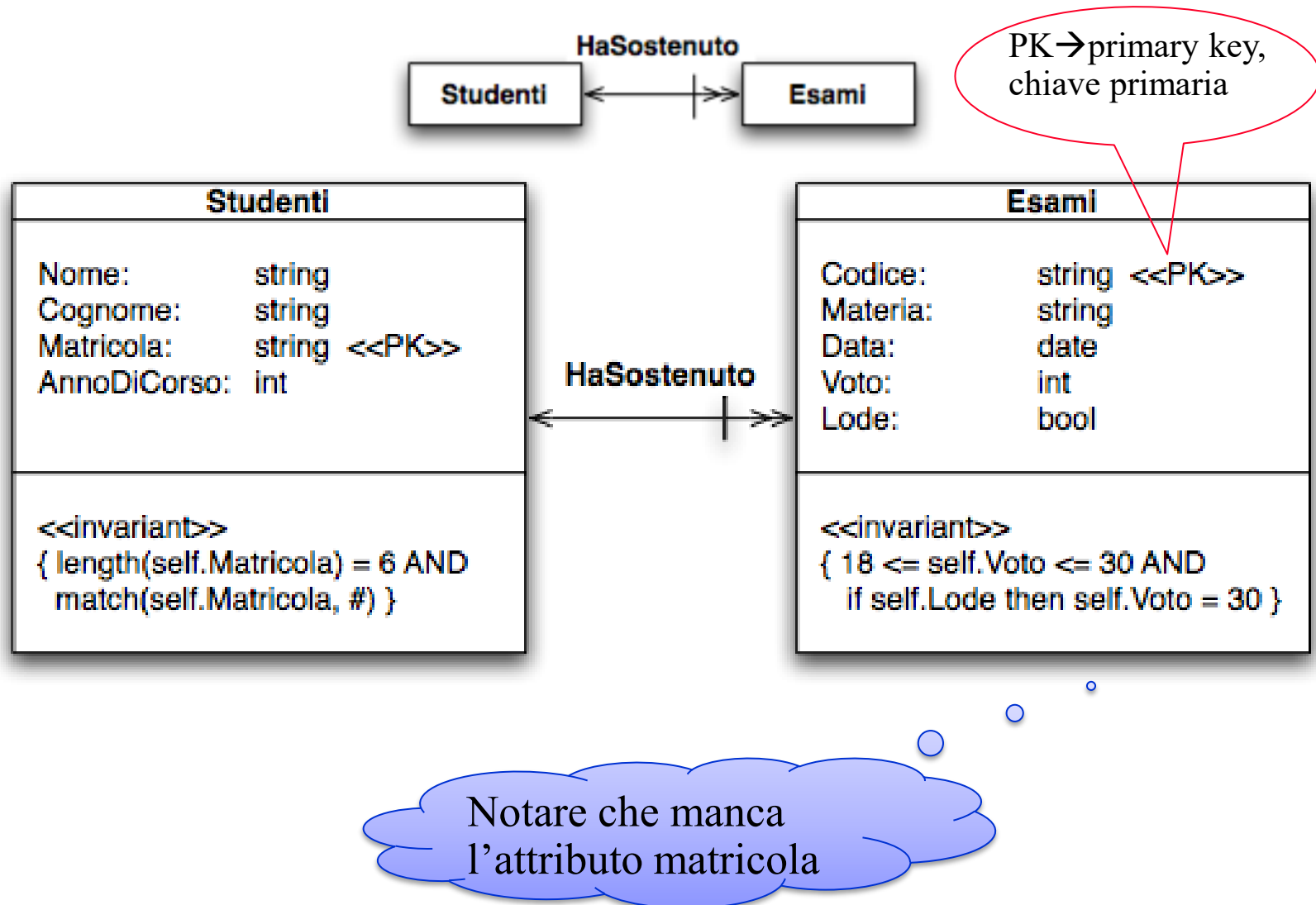
La trasformazione di uno schema ad oggetti in uno schema relazionale avviene eseguendo i seguenti passi:

1. rappresentazione delle associazioni uno ad uno e uno a molti;
2. rappresentazione delle associazioni molti a molti o non binarie;
3. rappresentazione delle gerarchie di inclusione;
4. identificazione delle chiavi primarie;
5. rappresentazione degli attributi multivalore;
6. appiattimento degli attributi composti.

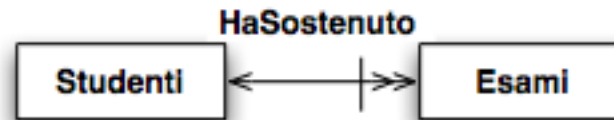
Obiettivo:

- rappresentare le stesse informazioni;
- **minimizzare la ridondanza;**
- produrre uno schema comprensibile, per facilitare la scrittura e manutenzione delle applicazioni.

ESEMPIO - Schema concettuale



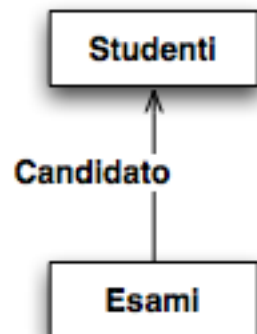
ESEMPIO - Traduzione logica di un'associazione uno Schema:



Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int)

Esami(Materia: string, Candidato*: string, Data: string, Voto: int)

Relazioni:



Studenti

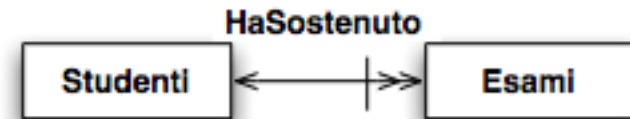
Nome	<u>Matricola</u>	Provincia	AnnoNascita
Isaia	071523	PI	1982
Rossi	067459	LU	1984
Bianchi	079856	LI	1983
Bonini	075649	PI	1984

Chiave
primaria
Esami

<u>Materia</u>	<u>Candidato*</u>	Data	Voto
BD	071523	12/01/06	28
BD	067459	15/09/06	30
FP	079856	25/10/06	30
BD	075649	27/06/06	25
LMM	071523	10/10/06	18

Attributo
aggiunto

ESEMPIO: ALTRE SOLUZIONI???



Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int)

Esami(Numero :int, Materia: string, Candidato*: string, Data: string, Voto: int)

Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int, Esame*:int)

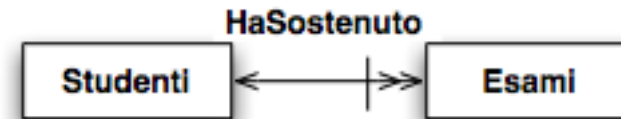
Esami(Numero :int, Materia: string, Data: string, Voto: int)



Esame e Numero corrispondono
ad un ID dell'esame

Altra soluzione?

ESEMPIO: ALTRE SOLUZIONI???



Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int)
Esami(Numero :int, Materia: string, Candidato*: string, Data: string, Voto: int)

Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int,
Esame*:int)
Esami(Numero :int, Materia: string, Data: string, Voto: int)

Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int)
Esami(Numero :int, Materia: string, Data: string, Voto: int)
StudentiEsami(Esame*: int, Candidato*: string)

• Quale preferire?

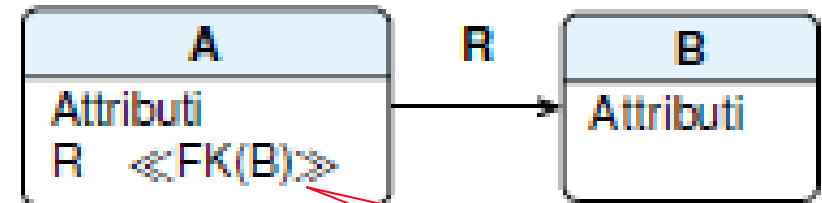
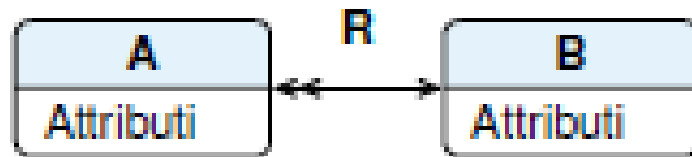
Il modello relazionale

Esame e Numero corrispondono
ad un ID dell'esame (non al
codice della materia)

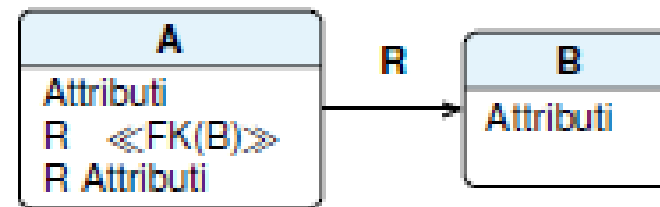
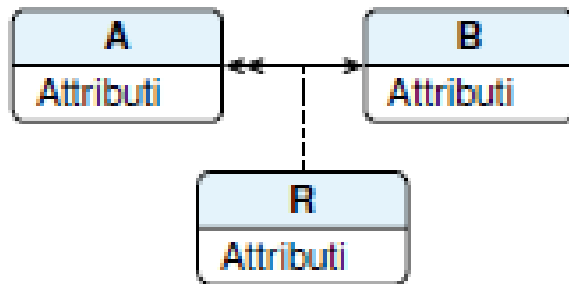
Altra
soluzione?

Rappresentazione delle associazioni uno a molti

- le associazioni uno a molti si rappresentano aggiungendo agli attributi della relazione rispetto a cui l'associazione è univoca una chiave esterna che riferisce l'altra relazione.

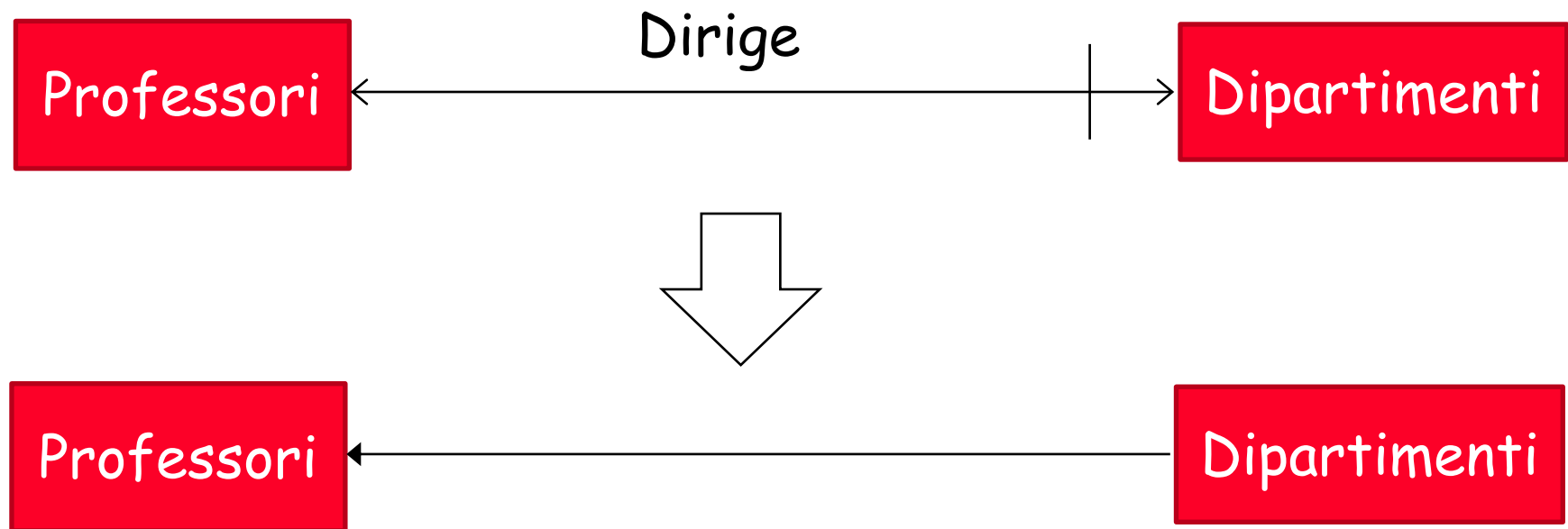


FK → foreign key,
chiave esterna

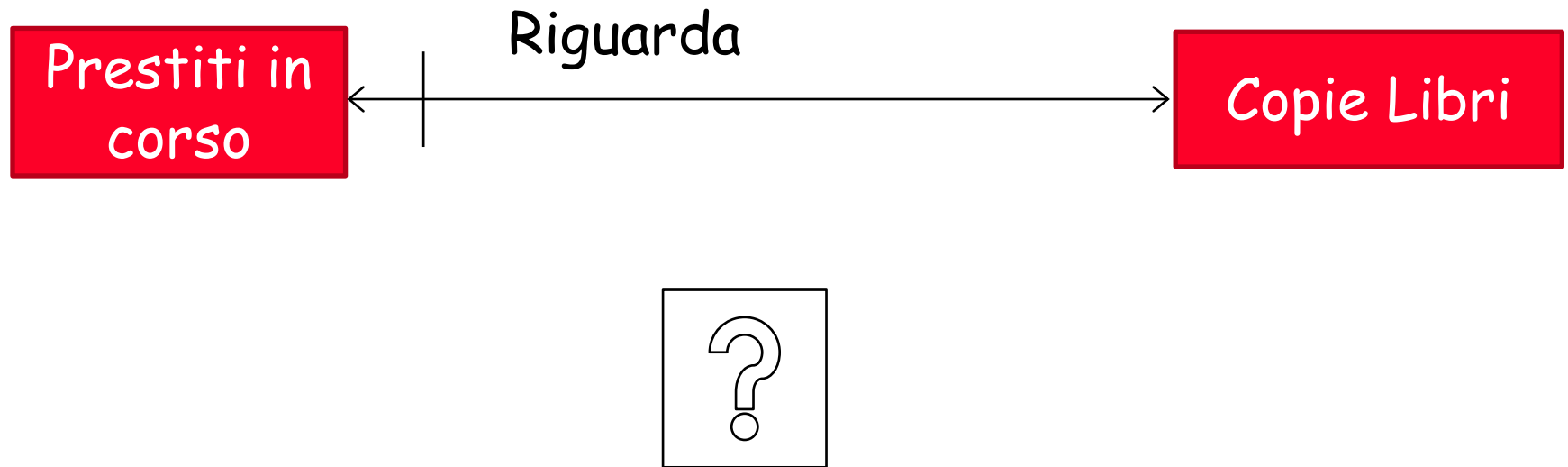


Esempi - Associazione 1 a 1

- Dirige(Professori, Dipartimenti) ha cardinalità (1:1):
 - Un professore può o non può dirigere un solo dipartimento
 - Un dipartimento deve avere un (solo) professore come dirigente

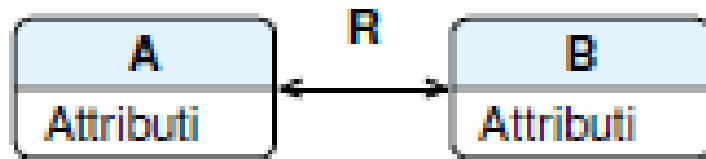


Esempi



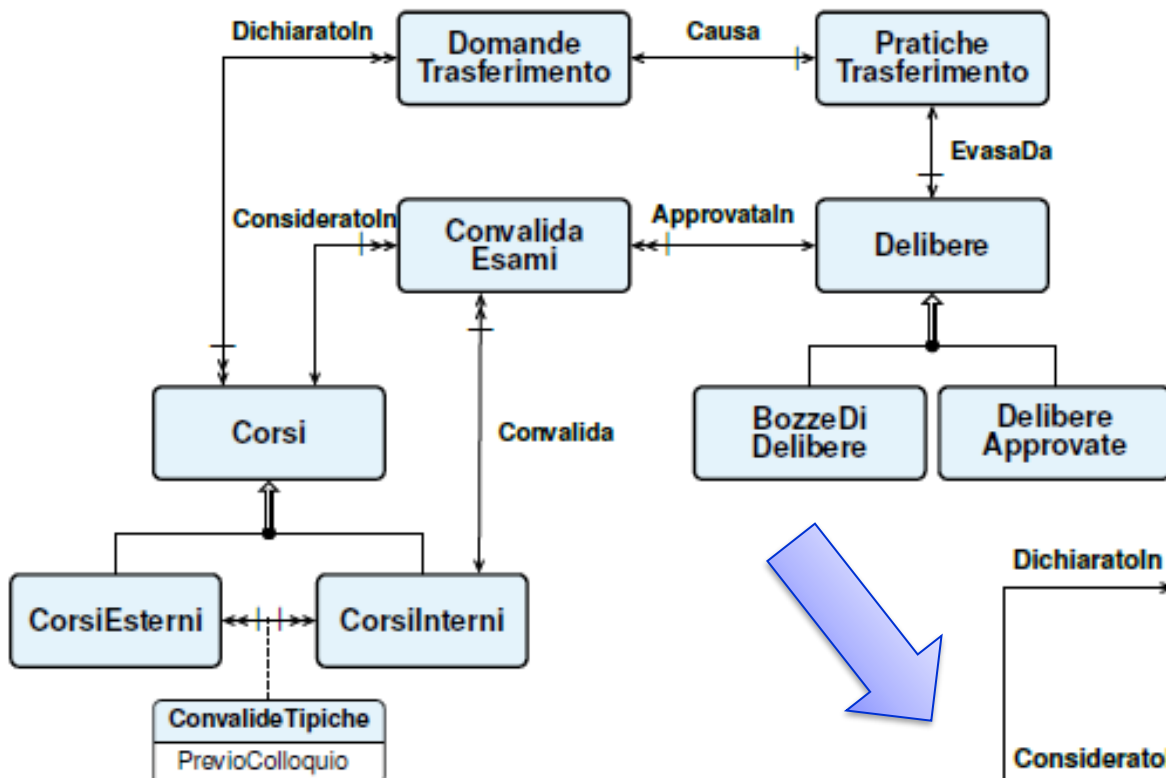
Rappresentazione delle associazioni uno ad uno

- le associazioni uno a uno si rappresentano aggiungendo la chiave esterna ad una qualunque delle due relazioni che riferisce l'altra relazione, preferendo quella rispetto a cui l'associazione è totale, nel caso in cui esista un vincolo di totalità

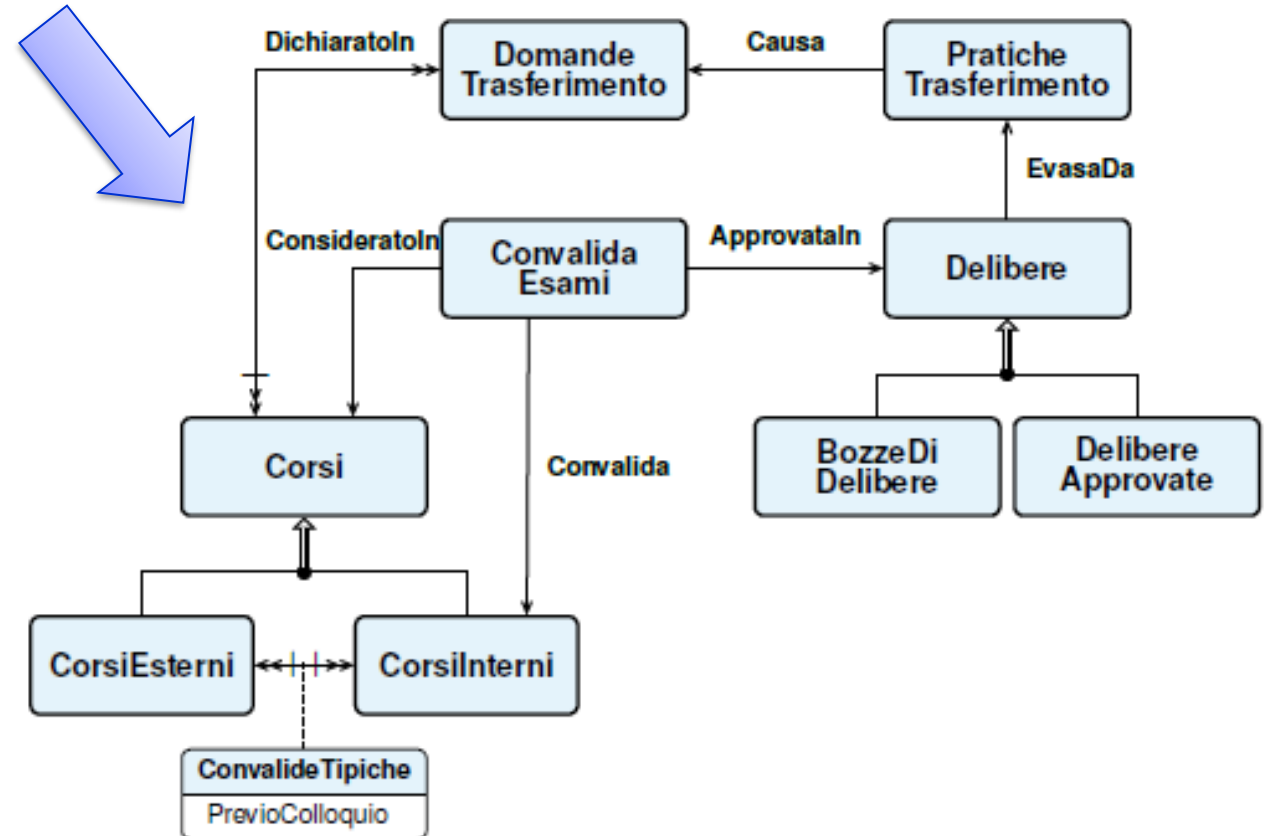


Vincoli sulla cardinalità delle associazioni uno a molti e uno ad uno

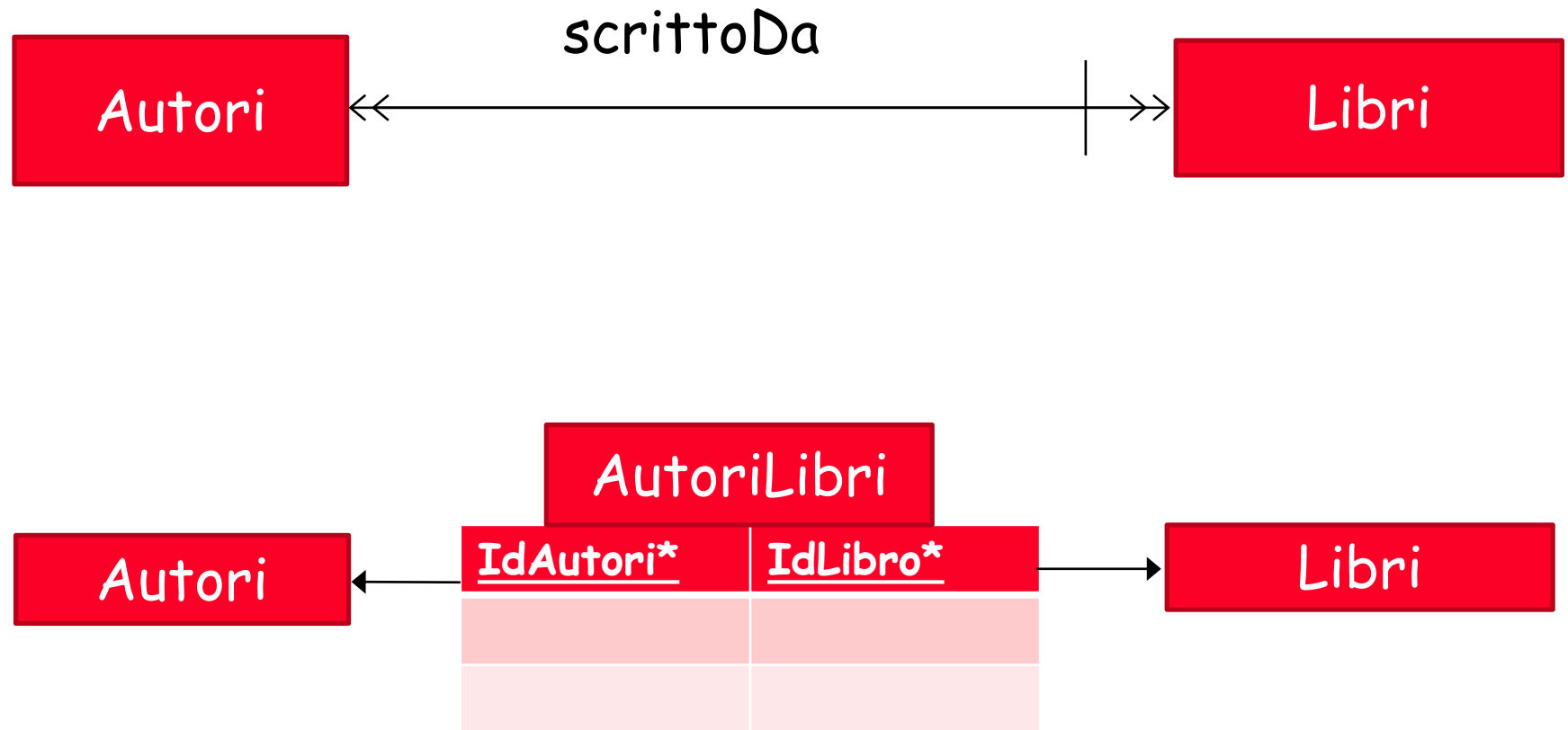
- La direzione dell'associazione rappresentata dalla chiave esterna è detta "**la diretta**" dell'associazione.
- Vincoli sulla cardinalità delle associazioni uno a molti ed uno ad uno:
 - **univocità della diretta**;
 - **totalità della diretta**: si rappresenta imponendo un vincolo **not null** sulla **chiave esterna**;
 - **univocità dell'inversa** e **totalità della diretta**: si rappresenta imponendo un vincolo not null ed un vincolo di chiave sulla chiave esterna.



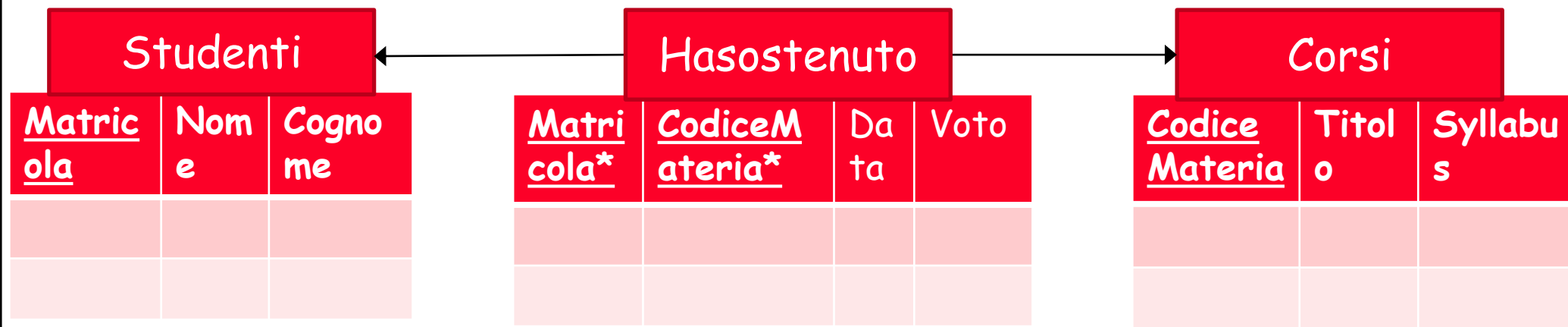
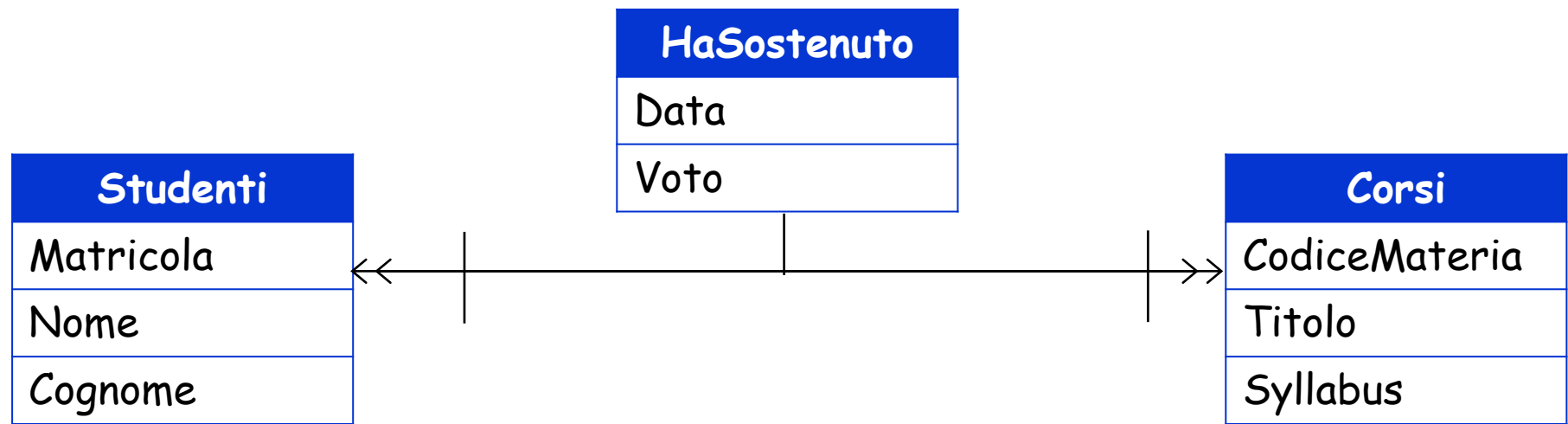
Solo uno a molti e uno ad uno



Esempio - Associazione molti a molti



Esempio - Associazione molti a molti



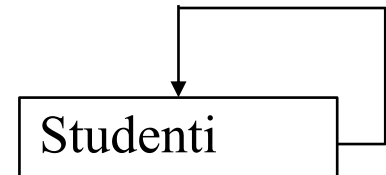
ESEMPIO - Traduzione logico (ricorsione)

Schema:

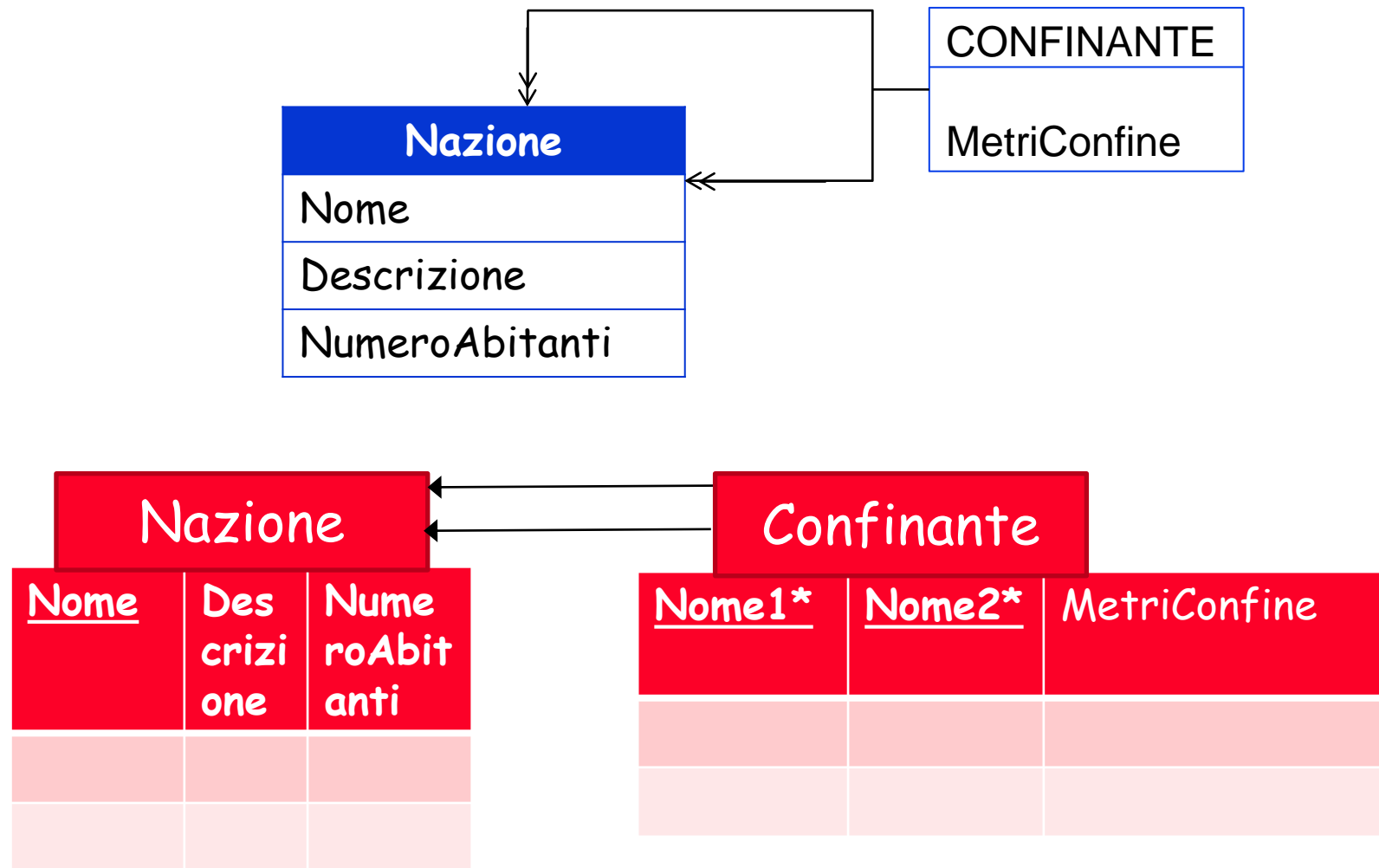
Studenti(Nome: string, Matricola: string, Provincia: string, AnnoNascita:int, ,
TutorStudente*: string)

Studenti

Nome	<u>Matricola</u>	Provincia	AnnoNascita	MatrTutor*
Isaia	071523	PI	1982	067459
Rossi	067459	LU	1984	071523
Bianchi	079856	LI	1983	071523
Bonini	075649	PI	1984	071523

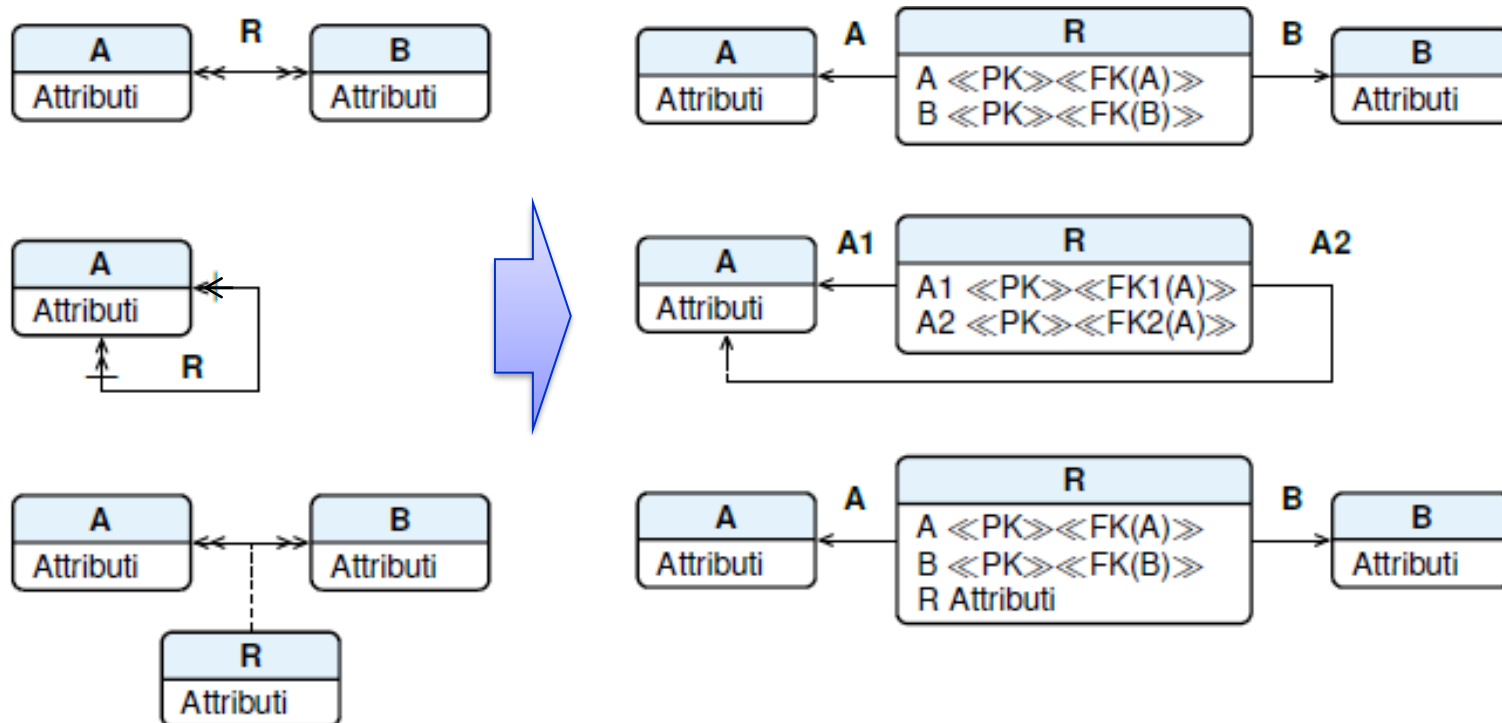


Esempio - Associazione molti a molti (ricorsione)

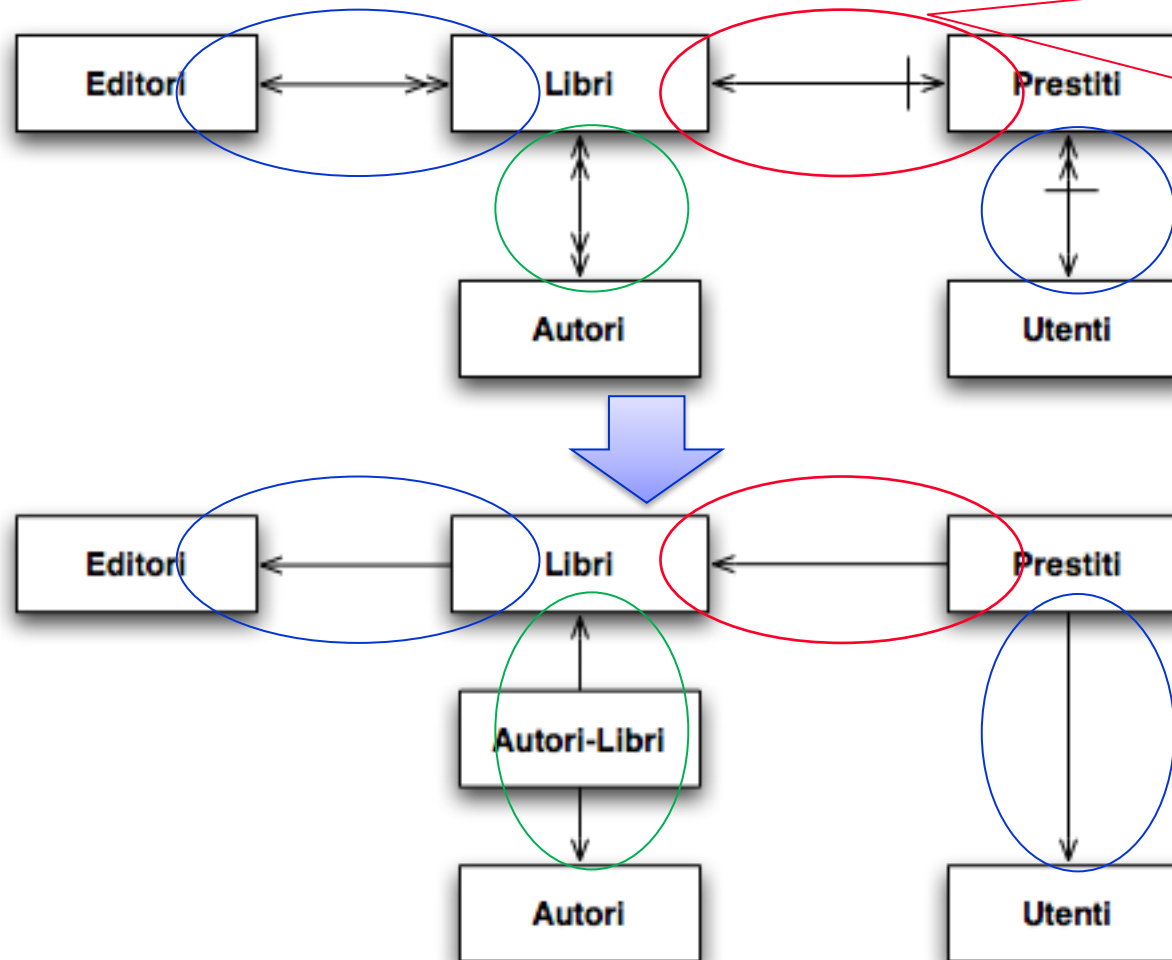


Rappresentazione delle associazioni molti a molti

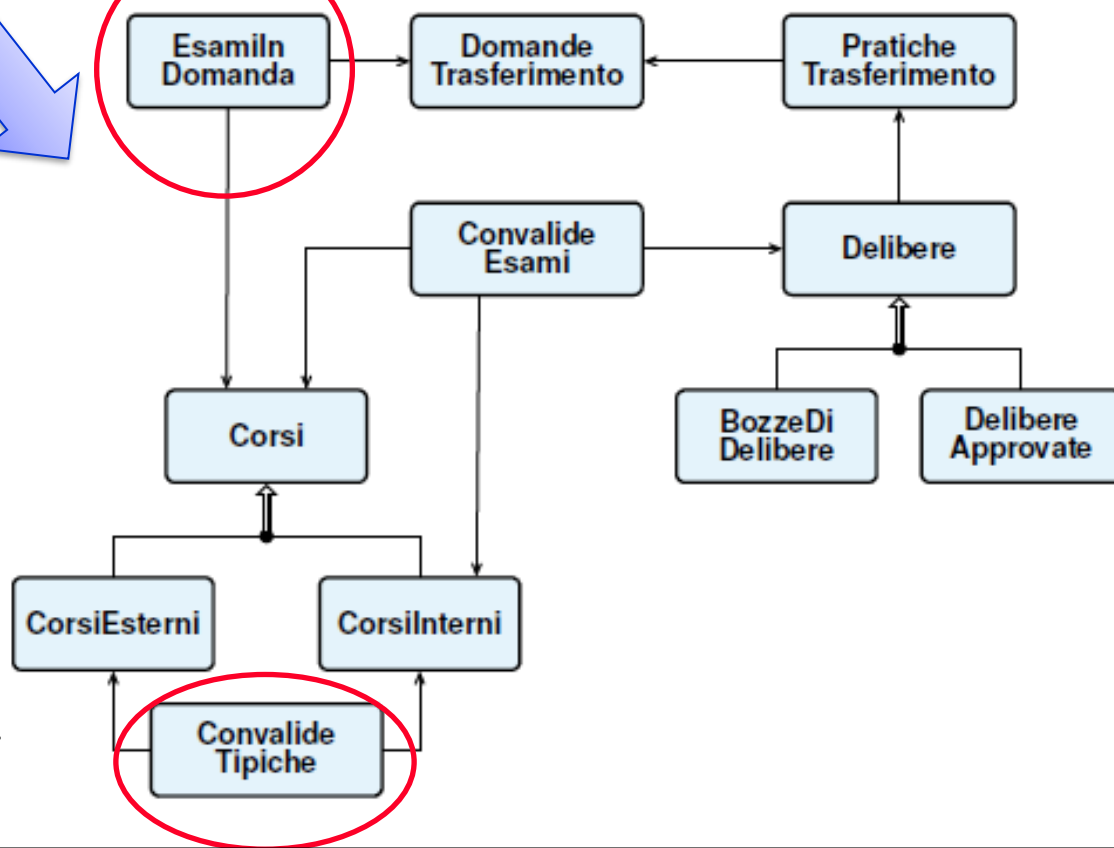
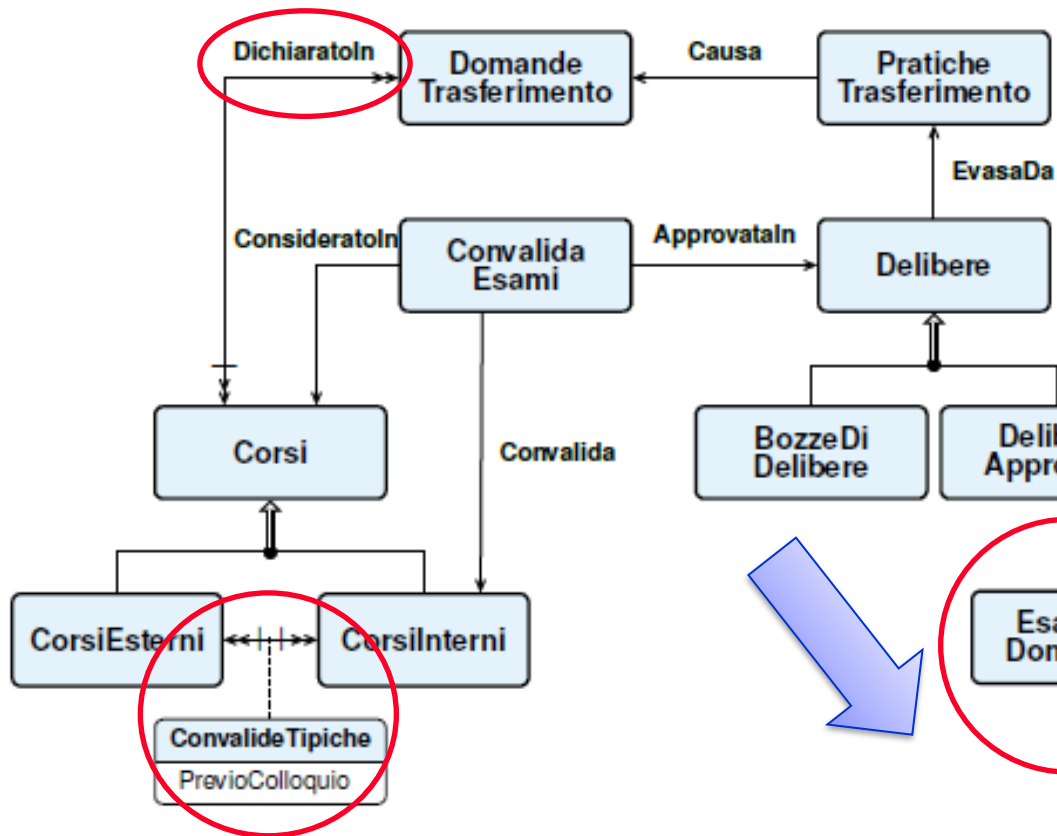
- Un'associazione molti a molti tra due classi si rappresenta aggiungendo allo schema una nuova relazione che contiene due chiavi esterne che riferiscono le due relazioni coinvolte; la chiave primaria di questa relazione è costituita dall'insieme di tutti i suoi attributi.



TRASFORMAZIONE DI SCHEMI A OGGETTI IN RELAZIONALI

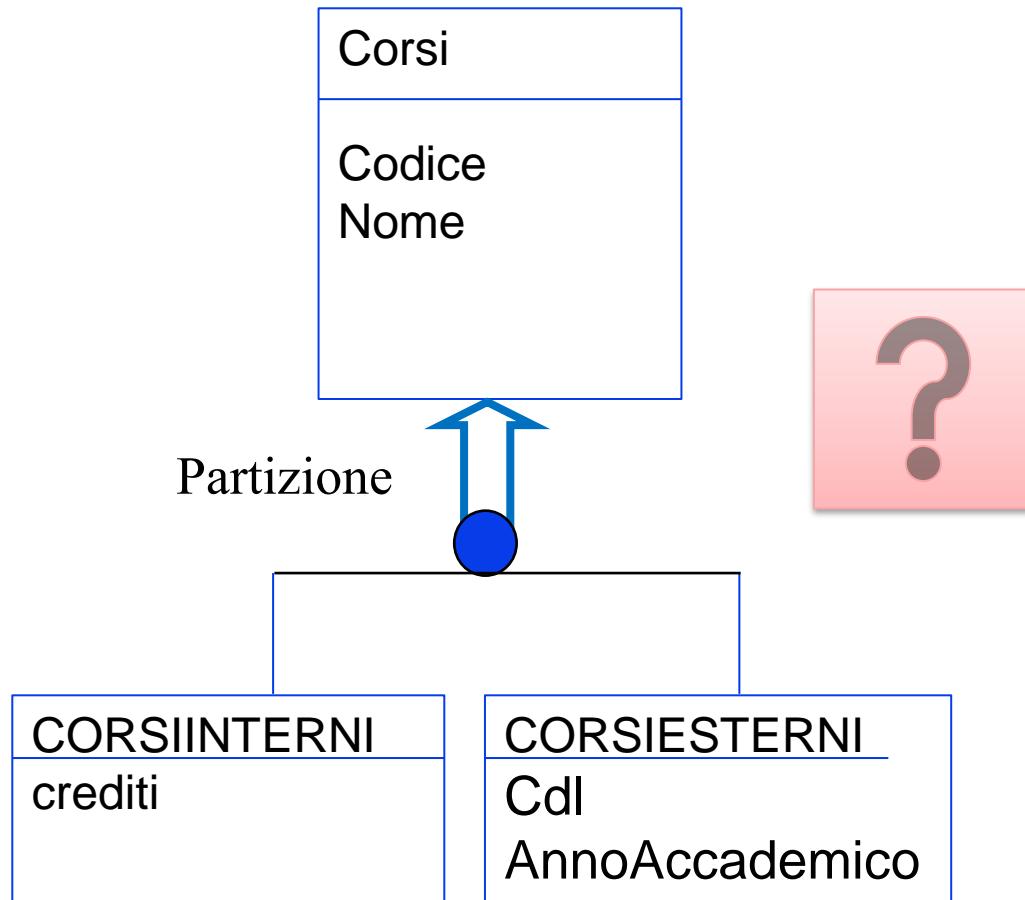


Si preferisce la direzione totale: ogni prestito ha un libro, ma non tutti i libri hanno un prestito



Il modello relazionale

Traduzione delle gerarchie



Rappresentazione delle gerarchie fra classi

- il modello relazionale non può rappresentare direttamente le gerarchie
- Bisogna eliminare le gerarchie, sostituendole con classi e relazioni:
 1. accorpamento delle figlie della gerarchia nel genitore
(relazione unica)
 2. accorpamento del genitore della gerarchia nelle figlie
(partizionamento orizzontale)
 3. sostituzione della gerarchia con relazioni
(partizionamento verticale)

Accorpamento delle figlie della gerarchia nel genitore (relazione unica)

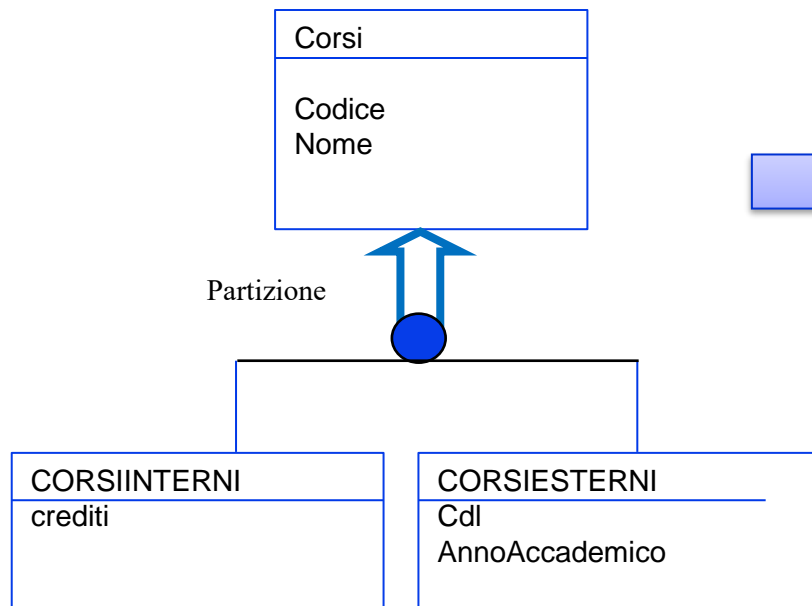
Se A_0 è la classe genitore di A_1 ed A_2 , le classi A_1 e A_2 vengono eliminate ed accorpate ad A_0

Ad A_0 viene aggiunto un **attributo** (**Discriminatore**) che indica da quale delle classi figlie deriva una certa istanza, e gli **attributi di A_1 ed A_2** vengono assorbiti dalla classe genitore, e assumono valore nullo sulle istanze provenienti dall'altra classe.

Infine, una **relazione relativa a solo una delle classi figlie** viene acquisita dalla classe genitore e avrà comunque cardinalità minima uguale a 0, in quanto gli elementi dell'altra classe non contribuiscono alla relazione.

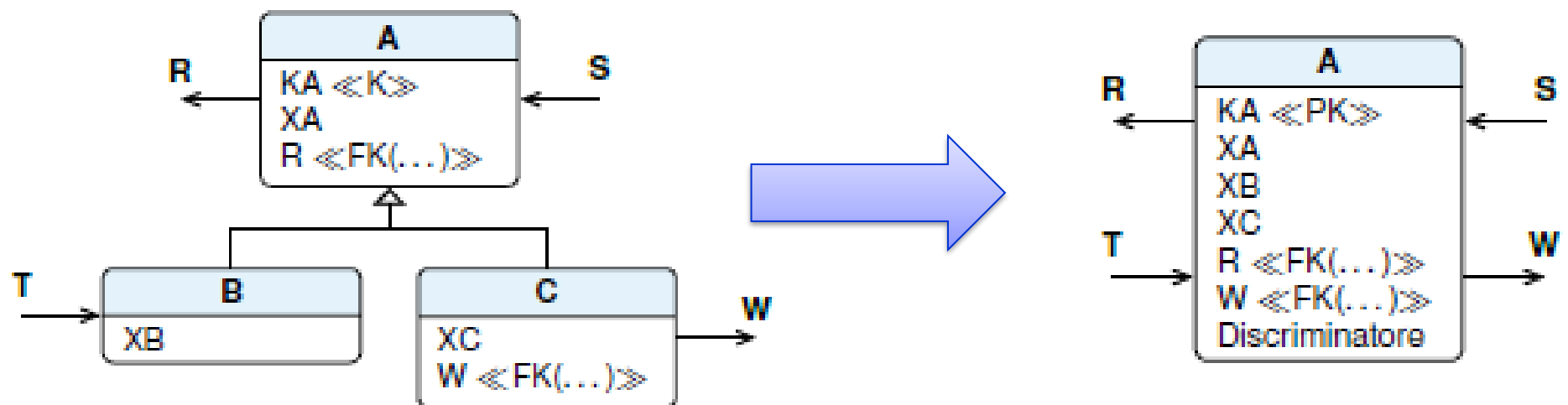
Accorpamento delle figlie della gerarchia nel genitore (relazione unica)

- Classe Corsi con due attributi Codice (la chiave), Nome e con due sottoclassi di tipo partizione:
 - CorsiInterni, con un attributo Crediti,
 - CorsiEsterni, con due attributi CorsoDiLaurea, AnnoAccademico.



Corsi	
Codice	:int <<PK>>
Nome	:string
Crediti	:int
CorsoDiLaurea	:string
AnnoAccademico	:int
InternoOEsterno	:('Interno', 'Esterno')

Accorpamento delle figlie della gerarchia nel genitore (relazione unica)

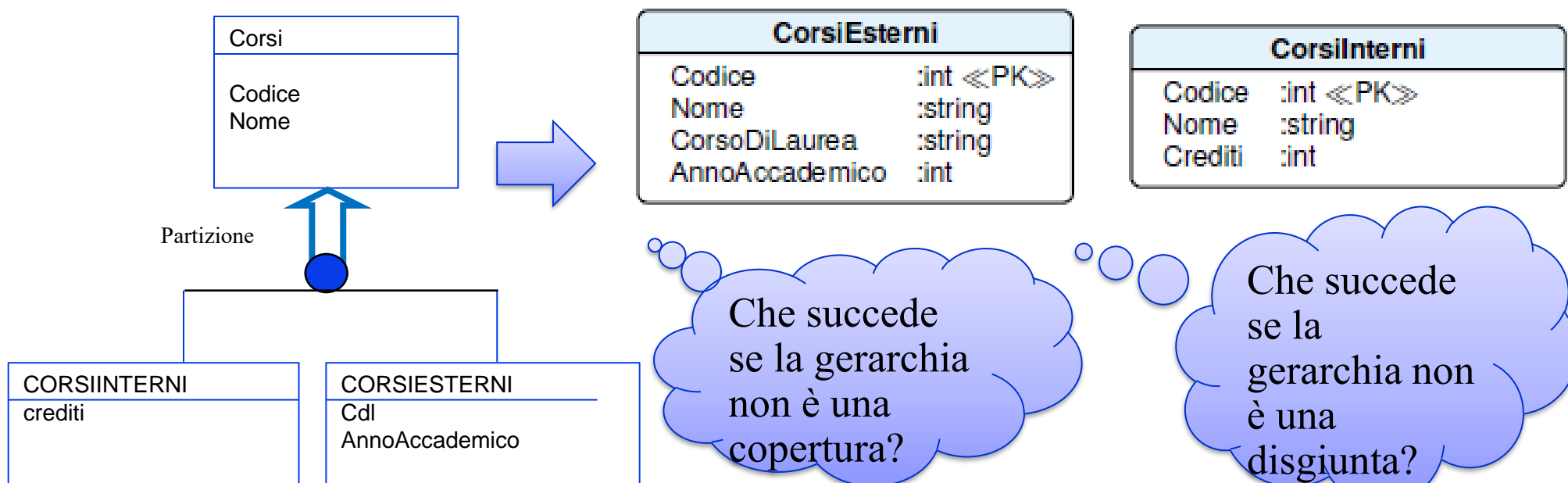


Accorpamento del genitore della gerarchia nelle figlie (partizionamento orizzontale)

- La classe genitore A_0 viene eliminata, e le classi figlie A_1 ed A_2 ereditano le proprietà (attributi, identificatore e relazioni) dell'classe genitore

Accorpamento del genitore della gerarchia nelle figlie (partizionamento orizzontale)

- Classe Corsi con due attributi Codice (la chiave), Nome e con due sottoclassi di tipo partizione:
 - CorsiInterni, con un attributo Crediti,
 - CorsiEsterni, con due attributi CorsoDiLaurea, AnnoAccademico.

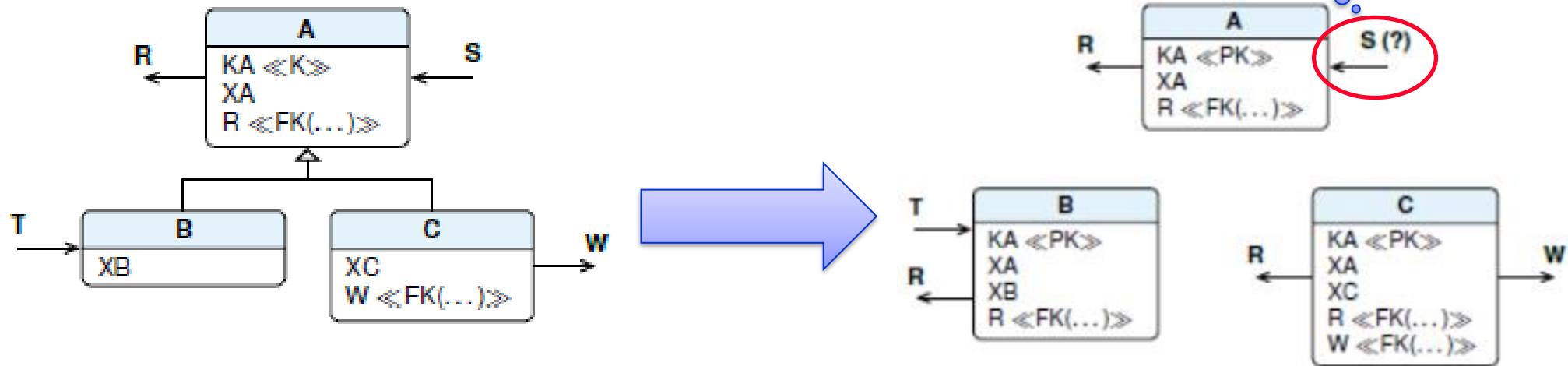


sostituzione della gerarchia con relazioni

Accorpamento del genitore della gerarchia nelle figlie (**partizionamento orizzontale**)

- il partizionamento orizzontale divide gli elementi della superclasse in più relazioni diverse, per cui non è possibile mantenere un vincolo referenziale verso la superclasse stessa; in conclusione, questa tecnica non si usa se nello schema relazionale grafico c'è una freccia che entra nella superclasse;

Che succede se esiste una chiave esterna verso la tabella A?

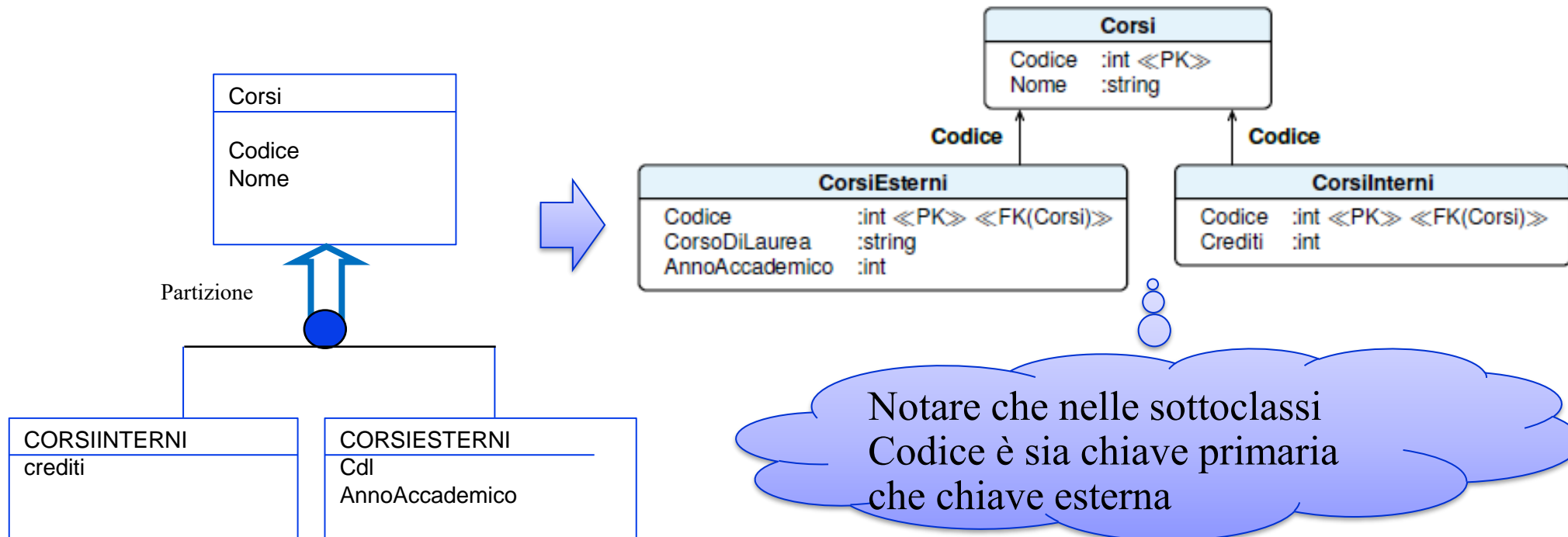


sostituzione della gerarchia con relazioni (partizionamento verticale)

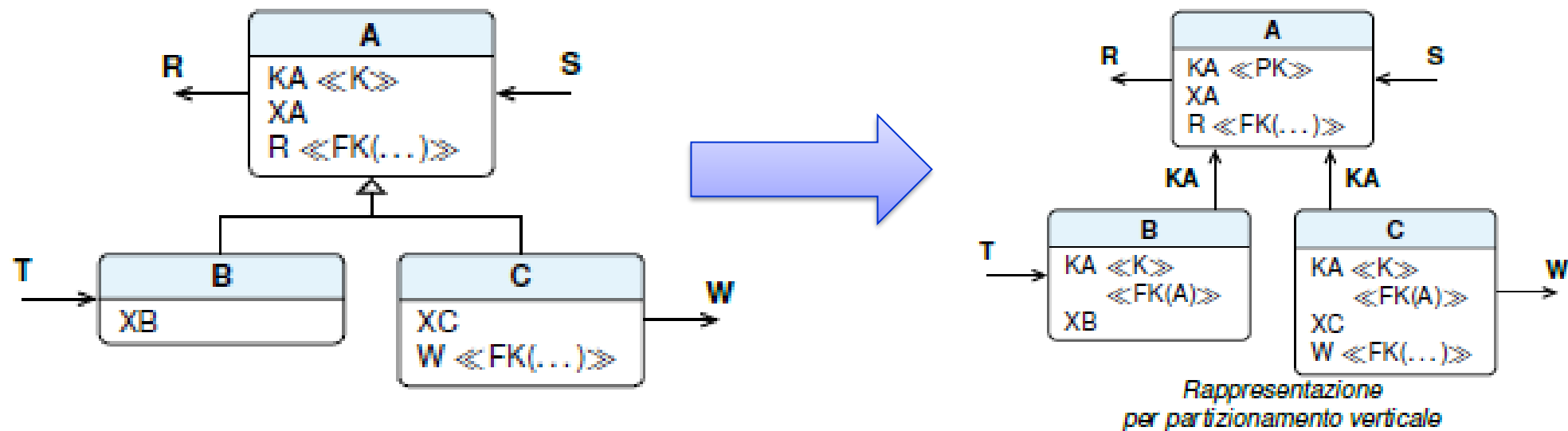
- La gerarchia si trasforma in due associazioni uno a uno che legano rispettivamente la classe genitore con le classi figlie.
- In questo caso non c'è un trasferimento di attributi o di associazioni e le classi figlie A_1 ed A_2 sono identificate esternamente dalla classe genitore A_0 .
- Nello schema ottenuto vanno aggiunti dei vincoli: ogni occorrenza di A_0 non può partecipare contemporaneamente alle due associazioni, e se la gerarchia è totale, deve partecipare ad almeno una delle due

sostituzione della gerarchia con relazioni (partizionamento verticale)

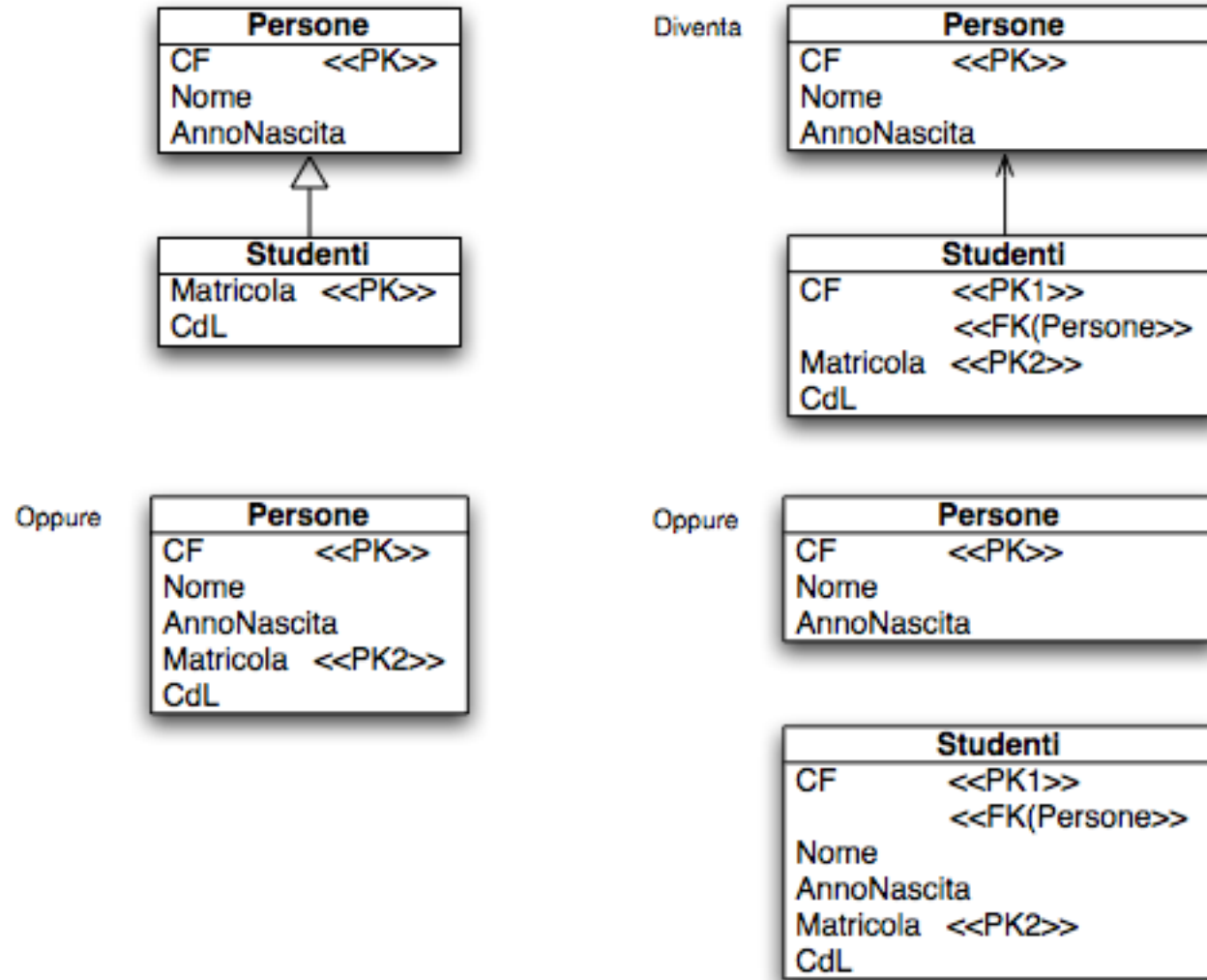
- Classe Corsi con due attributi Codice (la chiave), Nome e con due sottoclassi di tipo partizione:
 - CorsiInterni, con un attributo Crediti,
 - CorsiEsterni, con due attributi CorsoDiLaurea, AnnoAccademico.



sostituzione della gerarchia con relazioni (partizionamento verticale)



Riepilogo: LE SOTTOCLASSI



Esempio - Campo-multivalore

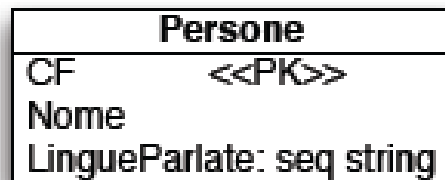
- Gestione persone con «più» indirizzi email

CodiceFiscale	Email_1	Email_2	Email_3	...
RSSMRT	RSS@	MRT@	RMT@	
BNCGNN	BNC@			



CodiceFiscale	IndirizziEmail
RSSMRT	RSS@
RSSMRT	MRT@
RSSMRT	RMT@
BNCGNN	BNC@

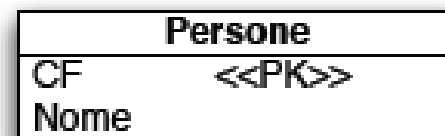
GLI ATTRIBUTI MULTIVALORE



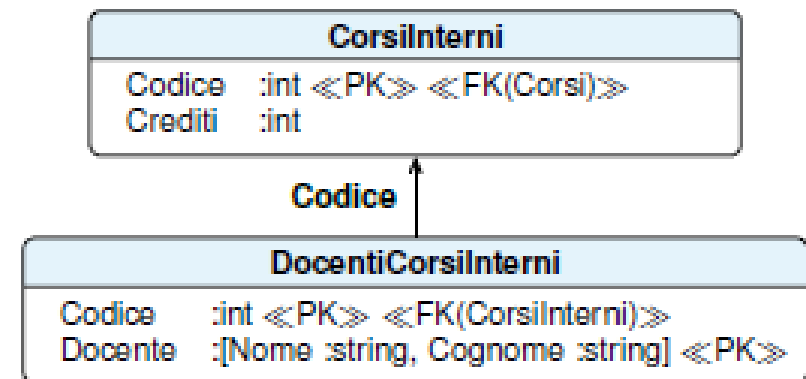
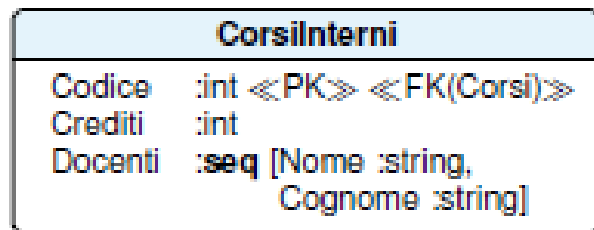
Diventa



Oppure

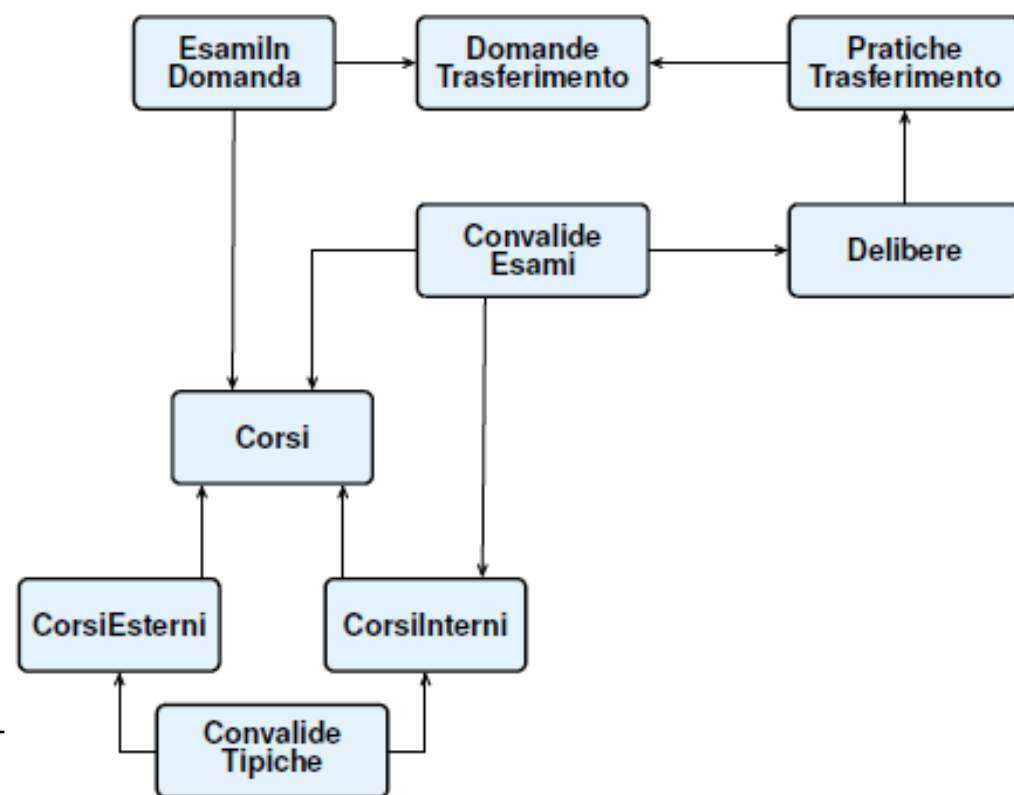
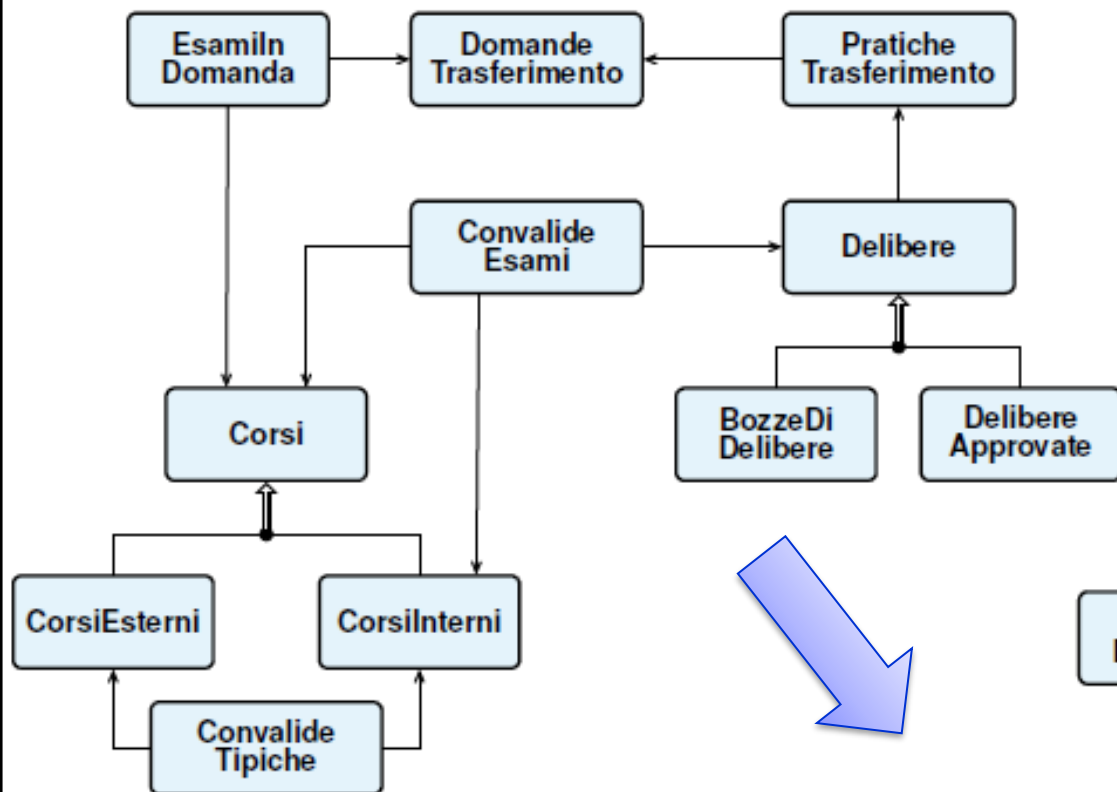


Rappresentazione delle proprietà multivalore

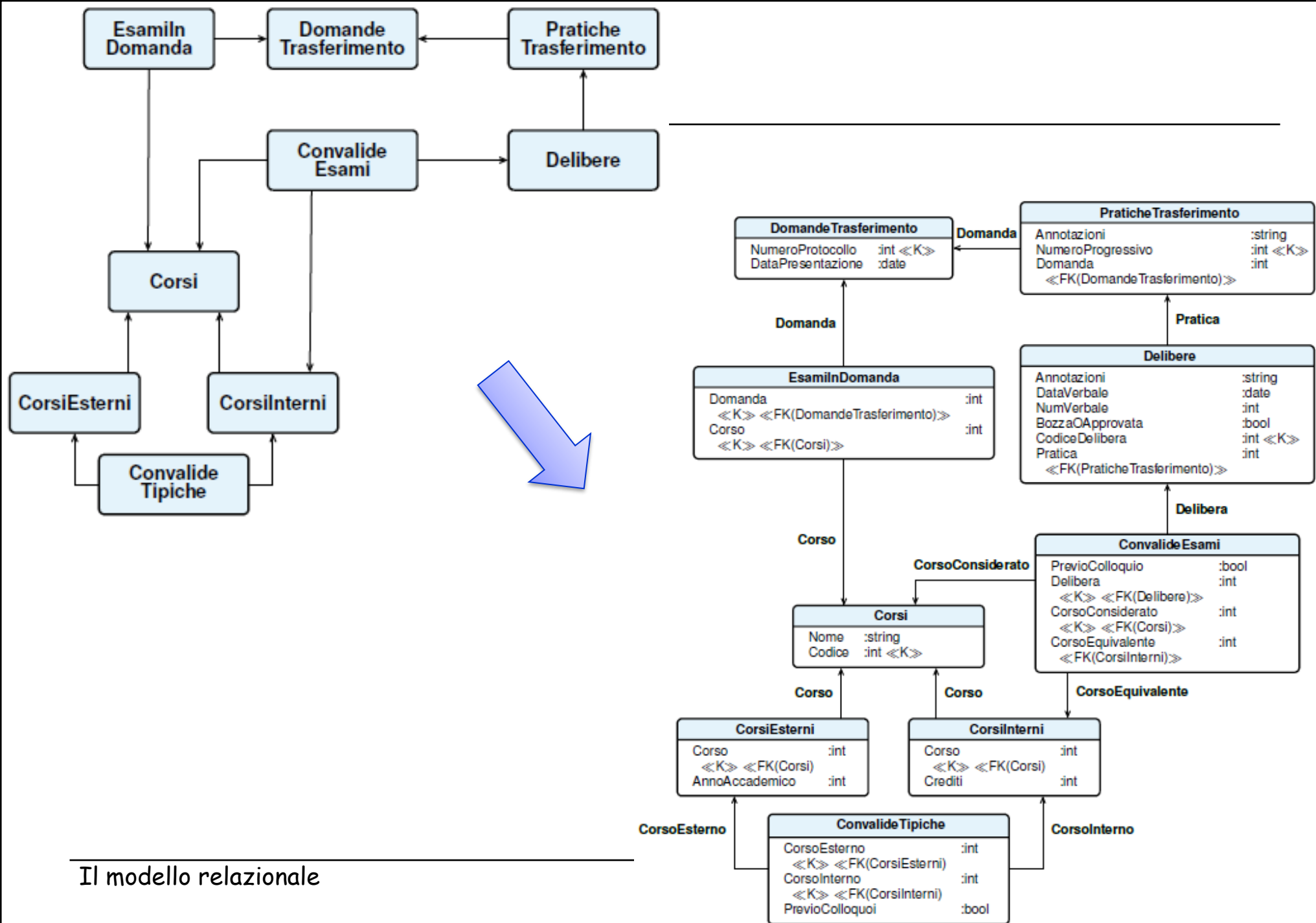


Appiattimento degli attributi composti

DocentiCorsiInterni	
Codice	:int «PK» «FK(CorsiInterni)»
Nome	:string «PK»
Cognome	:string «PK»



Il modello relazionale

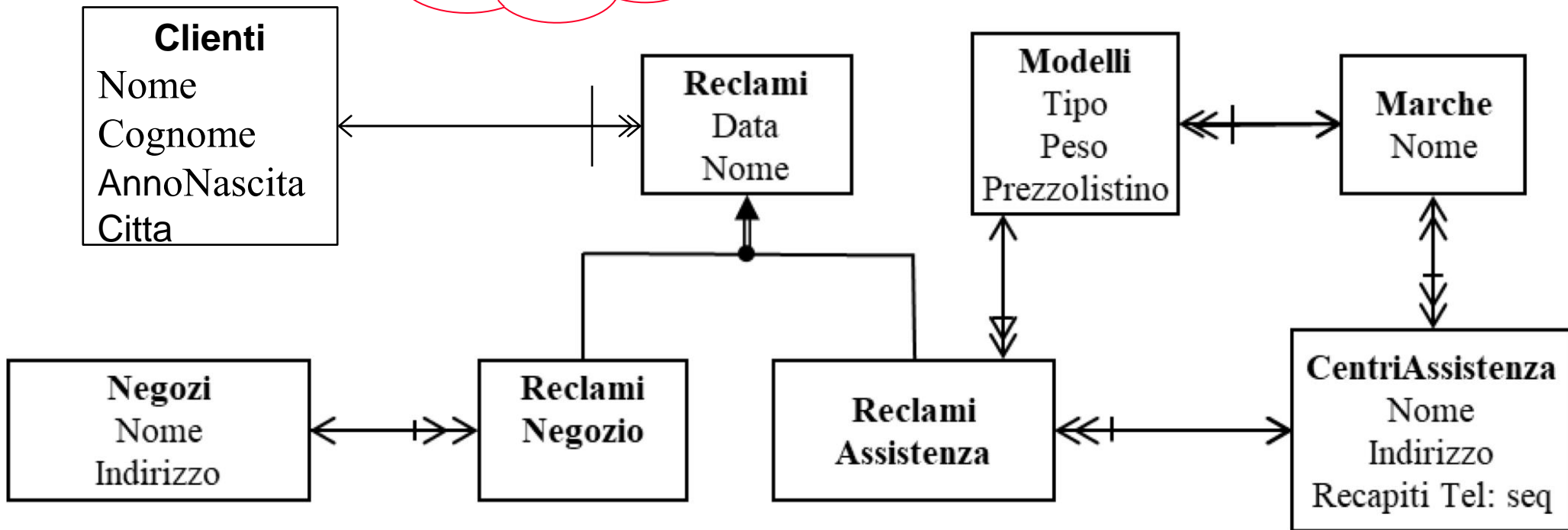


Il modello relazionale

Primo compito di Basi di Dati - 3/4/2019 - VARIANTE

- Variante: i reclami possono essere redatti da clienti registrati nella base di dati di cui interessano nome, cognome, anno nascita, città

E lo schema logico?



Schema logico - Primo compito di Basi di Dati - 3/4/2019

