

La normalizzazione

Forme normali

- Una **forma normale** è una proprietà di una base di dati relazionale che ne testimonia alcune “qualità” (o, più semplicemente, l'assenza di determinati difetti)

Una relazione con anomalie

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Attività
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Anomalie

- Lo stipendio di ciascun impiegato è ripetuto in tutte le ennuple relative
 - ridondanza
- Se lo stipendio di un impiegato varia, è necessario andarne a modificare il valore in diverse ennuple
 - anomalia di aggiornamento
- Se un impiegato interrompe la partecipazione a tutti i progetti, dobbiamo cancellarlo
 - anomalia di cancellazione
- Un nuovo impiegato senza progetto non può essere inserito
 - anomalia di inserimento

Perché questi fenomeni indesiderabili?

- abbiamo usato un'unica relazione per rappresentare informazioni eterogenee
 - gli impiegati con i relativi stipendi
 - i progetti con i relativi bilanci
 - le partecipazioni degli impiegati ai progetti con i relativi ruoli

**Per studiare in maniera sistematica questi aspetti, è necessario introdurre un vincolo di integrità:
la dipendenza funzionale**

Approfondiamo

- un vincolo di integrità
 - dipendenza funzionale

Esempio:

r su $R(X)$

$Y, Z \subseteq X$

r su $R(I S P B A)$

$\{I\}, \{S\} \subseteq \{I, S, P, B, A\}$

$I, S \subseteq I S P B A$

Dipendenza funzionale (vincolo)

$Y \rightarrow Z$

è soddisfatta

se, per ogni t_1 e t_2 in r con $t_1[Y] = t_2[Y]$,
risulta anche $t_1[Z] = t_2[Z]$

Impiegato $\rightarrow$ **Stipendio**,

per ogni t_1 e t_2 in r con $t_1[I] = t_2[I]$,
risulta anche $t_1[S] = t_2[S]$

Esempio:

r su $R(X)$

$Y, Z \subseteq X$

r su $R(I S P B A)$

$\{I\}, \{S\} \subseteq \{I, S, P, B, A\}$

$I, S \subseteq I S P B A$

Dipendenza funzionale (vincolo)

$Y \rightarrow Z$

è soddisfatta

se, per ogni t_1 e t_2 in r con $t_1[Y] = t_2[Y]$,
risulta anche $t_1[Z] = t_2[Z]$

Impiegato $\rightarrow$ **Stipendio**,

se due ennuple sono uguali su **Impiegato**
allora sono uguali anche su **Stipendio**

Proprietà

- Ogni impiegato ha un solo stipendio (anche se partecipa a più progetti)
- Ogni progetto ha un bilancio
- Ogni impiegato in ciascun progetto ha una sola Attività (anche se può avere attività diverse in progetti diversi)

Dipendenza funzionale

- relazione r su $R(X)$
- due sottoinsiemi non vuoti Y e Z di X
- esiste in r una dipendenza funzionale (FD) da Y a Z se, per ogni coppia di ennuple t_1 e t_2 di r con gli stessi valori su Y , risulta che t_1 e t_2 hanno gli stessi valori anche su Z

Notazione

$Y \rightarrow Z$

- Esempi:

Impiegato $\rightarrow$ Stipendio

Progetto $\rightarrow$ Bilancio

Impiegato Progetto $\rightarrow$ Attività

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Attività
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Impiegato → Stipendio
 Progetto → Bilancio
 Impiegato Progetto → Attività

Quali dipendenze causano problemi?

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Attività
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Impiegato → Stipendio

Progetto → Bilancio

Impiegato → Attività

Quali dipendenze causano problemi?

Impiegato → Stipendio

Progetto → Bilancio

Sì!

Impiegato Progetto → Attività

No

Le anomalie sono legate ad alcune FD

- gli impiegati hanno un unico stipendio

Impiegato → Stipendio

- i progetti hanno un unico bilancio

Progetto → Bilancio

Non tutte le FD causano anomalie

- In ciascun progetto, un impiegato ha una sola Attività

Impiegato Progetto → Attività

- Il soddisfacimento è più “semplice”, perché Impiegato Progetto è chiave

Nota

- Per ogni chiave si ha una dipendenza funzionale fra la chiave stessa e tutti gli altri attributi (in alcuni casi per "transitività"):

Impiegato Progetto → Stipendio Bilancio Attività

FD e anomalie

- La terza FD corrisponde ad una chiave e non causa anomalie
- Le prime due FD non corrispondono a chiavi e causano anomalie
- La relazione contiene alcune informazioni legate alla chiave e altre ad attributi che non formano una chiave
- Le anomalie sono causate dalla presenza di concetti eterogenei:
 - proprietà degli impiegati (lo stipendio)
 - proprietà di progetti (il bilancio)
 - proprietà della chiave **Impiegato Progetto**

Forma normale di Boyce e Codd (BCNF), intuitivamente

- Una relazione r è in forma normale di Boyce e Codd se, per ogni dipendenza funzionale $X \rightarrow Y$ definita su r , l'insieme X contiene una chiave K di r
- Serve una precisazione

FD banali

- Impiegato Progetto $\rightarrow$ Progetto
 - è sempre soddisfatta (la chiamiamo banale)
- Consideriamo solo FD non banali
 - $Y \rightarrow Z$
 - con Y e Z insiemi disgiunti
- Quelle banali ci sono sempre, ma non causano problemi

Forma normale di Boyce e Codd (BCNF)

- Una relazione r è in forma normale di Boyce e Codd se, per ogni dipendenza funzionale **non banale** $X \rightarrow Y$ definita su r , l'insieme X contiene una chiave K di r
- La forma normale richiede che i concetti in una relazione siano omogenei (solo proprietà direttamente associate alla chiave)

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Attività
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

✗ Impiegato → Stipendio

✗ Progetto → Bilancio

✓ Impiegato Progetto → Attività

Un inciso: chiavi (o superchiavi)

- Come si individuano?
 - verifica
 - ricerca

Verifica

- Dati $R(X)$ e un insieme di FD, l'insieme Y è superchiave?
 - inizializziamo un insieme W con Y
 - consideriamo le FD il cui primo membro è in W e aggiungiamo a W i relativi secondi membri
 - ci fermiamo quando non riusciamo ad aggiungere altro
 - (il valore finale di W è chiamato "chiusura di Y ")
 - se la chiusura di Y è uguale a X , allora Y è superchiave

Esempio

- Nel nostro caso
 - la chiusura di IP è ISPBA e quindi IP è superchiave
 - la chiusura di I è IS e quindi I non è superchiave

Ricerca delle chiavi

- Algoritmi corretti e completi sono abbastanza contorti e non vale la pena approfondire
- È però utile l'intuizione:
 - attributi che compaiono a primo membro e non a secondo
 - "suggerimenti" che vengono da una comprensione del dominio applicativo
- Conviene procedere con l'intuizione e poi la verifica

Che facciamo se una relazione non soddisfa la BCNF?

- La rimpiazziamo con altre relazioni che soddisfano la BCNF

Come?

- Decomponendo sulla base delle dipendenze funzionali, al fine di separare i concetti

Normalizzazione

- Procedura che permette di trasformare schemi che violano la forma normale in schemi che la soddisfano
- La normalizzazione va utilizzata come **tecnica di verifica** dei risultati della progettazione di una base di dati
- **Non costituisce una metodologia di progettazione**

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Attività
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Impiegato → Stipendio
 Progetto → Bilancio
 Impiegato Progetto → Attività

Impiegato	Stipendio
Rossi	20
Verdi	35
Neri	55
Mori	48
Bianchi	48

Impiegato → Stipendio

Impiegato	Progetto	Attività
Rossi	Marte	tecnico
Verdi	Giove	progettista
Verdi	Venere	progettista
Neri	Venere	direttore
Neri	Giove	consulente
Neri	Marte	consulente
Mori	Marte	direttore
Mori	Venere	progettista
Bianchi	Venere	progettista
Bianchi	Giove	direttore

Impiegato → Stipendio

Impiegato → Progetto → Attività

Progetto → Bilancio

Impiegato Progetto → Attività

Progetto	Bilancio
Marte	2
Giove	15
Venere	15

Progetto → Bilancio

Normalizzazione "semplice e intuitiva"

- Una relazione per ogni FD
- Più precisamente
 - per ogni gruppo di FD con lo stesso primo membro
 - definiamo una relazione

Impiegato	Stipendio
Rossi	20
Verdi	35
Neri	55
Mori	48
Bianchi	48

Impiegato → Stipendio

Impiegato	Progetto	Attività
Rossi	Marte	tecnico
Verdi	Giove	progettista
Verdi	Venere	progettista
Neri	Venere	direttore
Neri	Giove	consulente
Neri	Marte	consulente
Mori	Marte	direttore
Mori	Venere	progettista
Bianchi	Venere	progettista
Bianchi	Giove	direttore

Impiegato Progetto → Attività

Progetto	Bilancio
Marte	2
Giove	15
Venere	15

Progetto → Bilancio

Normalizzazione "semplice e intuitiva"

- per ogni gruppo di FD con lo stesso primo membro
- definiamo una relazione
- **Per definire una procedura efficace conviene però partire dalla relazione data e "decomporre"**

Procedura intuitiva di normalizzazione

- Non valida in generale, ma solo nei "casi semplici"
 - Per ogni dipendenza $X \rightarrow Y$ che viola la BCNF, definire una relazione su XY ed eliminare Y dalla relazione originaria

Procedura intuitiva di normalizzazione, esempio

R(Impiegato Stipendio Progetto Bilancio Attività)

$I \rightarrow S, P \rightarrow B, IP \rightarrow A$

R1(Impiegato Stipendio)

$I \rightarrow S$

OK

R2(Impiegato Progetto Bilancio Attività)

$P \rightarrow B, IP \rightarrow A$

R3(Progetto Bilancio)

$P \rightarrow B$

OK

R4(Impiegato Progetto Attività)

$IP \rightarrow A$

OK

Procedura intuitiva di normalizzazione, attenzione

- Consideriamo un caso più complesso

R (CodCorso, Anno, Titolo, Crediti, Docente, NomeDoc, Semestre, Dipartimento, Indirizzo)

CodCorso → Titolo, Crediti, Dipartimento

CodCorso, Anno → Docente, Semestre

Docente → NomeDoc

Dipartimento → Indirizzo

- Vediamo poi i dettagli, ma se decomponiamo con

CodCorso → Titolo, Crediti, Dipartimento

allora si "perde" il legame fra Dipartimento e Indirizzo

- Perciò la procedura va modificata

Procedura intuitiva di normalizzazione, un po' più generale

- Valida in qualche caso in più
 - Per ogni dipendenza $X \rightarrow Y$ che viola la BCNF, definire una relazione su XZ (dove Z è l'insieme degli attributi che "dipendono" da X , eventualmente per transittività) ed eliminare Z dalla relazione originaria
 - Il concetto di "transittività" andrebbe formalizzato, ma in molti casi reali è intuitivo

Procedura un po' più generale, esempio

- R (CodCorso, Anno, Titolo, Crediti, Docente, NomeDoc, Semestre, Dipartimento, Indirizzo)
 - CodCorso → Titolo, Crediti, Dipartimento
 - CodCorso, Anno → Docente, Semestre
 - Docente → NomeDoc
 - Dipartimento → Indirizzo

Variante nell'ordine delle dipendenze

R (CodCorso, Anno, Titolo, Cred, Doc, NomeDoc, Sem, Dipart, Ind)

Doc → NomeDoc

Dipart → Ind

CodCorso → Titolo, Cred, Dipart

CodCorso, Anno → Doc, Sem

Procedura un po' più generale, esempio

- R (CodCorso, Anno, Titolo, Crediti, Docente, NomeDoc, Semestre, Dipartimento, Indirizzo)
 - CodCorso → Titolo, Crediti, Dipartimento
 - CodCorso, Anno → Docente, Semestre
 - Docente → NomeDoc
 - Dipartimento → Indirizzo
- Chiave?
 - CodCorso, Anno

Verifica della chiave

R (CodCorso, Anno, Titolo, Cred, Doc, NomeDoc, Sem, Dipart, Ind)

CodCorso → Titolo, Cred, Dipart

CodCorso, Anno → Doc, Sem

Doc → NomeDoc

Dipart → Ind

- CodCorso, Anno è chiave?

CodCorso, Anno

CodCorso, Anno, Titolo, Cred, Dipart

CodCorso, Anno, Titolo, Cred, Dipart, Doc, Sem

CodCorso, Anno, Titolo, Cred, Dipart, Doc, Sem, NomeDoc

CodCorso, Anno, Titolo, Cred, Dipart, Doc, Sem, NomeDoc, Ind

Sì!

Ricordiamo la definizione di chiave

- Insieme di attributi che identifica univocamente le ennuple ed è minimale rispetto a questa proprietà
- Per verificare che **CodCorso Anno** è chiave debbo verificare che
 - **CodCorso** da solo non è chiave
 - **Anno** da solo non è chiave

Un insieme che non è chiave

R (CodCorso, Anno, Titolo, Cred, Doc, NomeDoc, Sem, Dipart, Ind)

CodCorso → Titolo, Cred, Dipart

CodCorso, Anno → Doc, Sem

Doc → NomeDoc

Dipart → Ind

- CodCorso è chiave?

CodCorso

CodCorso, Titolo, Cred, Dipart

CodCorso, Titolo, Cred, Dipart, Ind

No!

Attenzione

- La verifica della chiave è semplice
- La ricerca non è semplice:
 - si può procedere per tentativi
 - conviene "intuire" e verificare

Procedura un po' più generale, esempio

- R (CodCorso, Anno, Titolo, Crediti, Docente, NomeDoc, Semestre, Dipartimento, Indirizzo)

CodCorso → Titolo, Crediti, Dipartimento

CodCorso, Anno → Docente, Semestre

Docente → NomeDoc

Dipartimento → Indirizzo

- R (CC, A, T, Cr, Do, N, S, Di, I)

CC → T, Cr, Di

CC, A → Do, S

Do → N

Di → I

Corso

Edizione del corso

Docente

Dipartimento

Procedura un po' più generale, esempio

R (CC, A, T, Cr, Do, N, S, Di, I)



$CC \rightarrow T, Cr, Di$



$CC, A \rightarrow Do, S$



$Do \rightarrow N$



$Di \rightarrow I$

Corso

Edizione del corso

Docente

Dipartimento

R1 (CC, T, Cr, Di, I)



$CC \rightarrow T, Cr, Di$



$Di \rightarrow I$

R2 (CC, A, Do, N, S)



$CC, A \rightarrow Do, S$



$Do \rightarrow N$

R3 (Di, I)

$Di \rightarrow I$

OK

R4 (CC, T, Cr, Di)

$CC \rightarrow T, Cr, Di$

OK

R5 (Do, N)

$Do \rightarrow N$

OK

R2 (CC, A, Do, S)

$CC, A \rightarrow Do, S$

OK

Non sempre così facile

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

~~×~~ Impiegato → Sede

~~×~~ Progetto → Sede

Non sempre così facile

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

~~×~~ Impiegato → Sede
~~×~~ Progetto → Sede

Applichiamo la procedura

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Neri	Saturno
Neri	Venere

Progetto	Sede
Marte	Roma
Giove	Milano
Saturno	Milano
Venere	Milano

Impiegato → Sede

??

- La procedura non va bene:
 - non "conserva" le dipendenze
- Proviamo l'altra tecnica (una relazione per ogni dipendenza)

Una relazione per ogni dipendenza

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Neri	Milano

Progetto	Sede
Marte	Roma
Giove	Milano
Saturno	Milano
Venere	Milano

Proviamo a ricostruire

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Neri	Milano

Progetto	Sede
Marte	Roma
Giove	Milano
Saturno	Milano
Venere	Milano

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Neri	Saturno	Milano
Neri	Venere	Milano
Verdi	Saturno	Milano
Neri	Giove	Milano

Diversa dalla relazione di partenza!

Decomposizione senza perdita

- Una relazione r si **decompone** su X_1 e X_2 **senza perdita di informazione** se il join delle proiezioni di r su X_1 e X_2 è uguale a r stessa (cioè non contiene ennuple spurie)
- Una buona decomposizione deve essere senza perdita

Purtroppo

- L'esempio appena visto ci costringe a dire che la normalizzazione non sempre funziona:
 - vorremmo avere una decomposizione
 - in BCNF
 - senza perdita
 - che "conservi" le dipendenze
 - Non sempre è possibile:
 - ci sono realtà difficili da modellare
 - Anche per questo, usiamo la normalizzazione come strumento di verifica e non di progettazione

Decomposizione senza perdita

- Approfondiamo

<u>IMP</u>	<u>PROGETTO</u>	<u>SEDE</u>
Rossi	MARTE	ROMA
Verdi	GIOVE	MILANO
Neri	GIOVE	MILANO

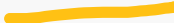
<u>PROG SEDE</u>			<u>IMP SEDE</u>	
MARTE	ROMA	—	Rossi	ROMA
GIOVE	MILANO	—	Verdi	MILANO
		—	Neri	MILANO

condizione sufficiente

- gli attributi comuni sono chiave per almeno una delle proiezioni

Attenzione

- La condizione sufficiente non è necessaria

	<u>IMP</u>	<u>PROG</u>	<u>SEDE</u>
	Rossi	GIOVE	ROMA
	Neri	MARTE	MILANO
	Neri	VENERE	MILANO
	Bruni	MARTE	MILANO
	Bruni	VENERE	MILANO

<u>IMP</u>	<u>SEDE</u>		<u>PROG</u>	<u>SEDE</u>
Rossi	ROMA		GIOVE	ROMA
Neri	MILANO		MARTE	MILANO
Bruni	MILANO		VENERE	MILANO

seconda condizione sufficiente

- le componenti delle eunuple sono indipendenti fra loro

nell'esempio

- ogni sede ha un insieme di impiegati.
- ogni sede ha un insieme di progetti.

• ogni impiegato lavora a tutti i progetti delle proprie sedi

Riassumendo

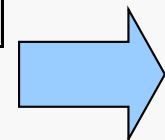
- La decomposizione senza perdita
 - è garantita se gli attributi comuni contengono una **chiave** per almeno una delle relazioni decomposte
 - Più in generale, si ha
 - quando le informazioni in X_1 e X_2 sono fra loro indipendenti

Decomposizione senza perdita

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Verdi	Giove	Roma
Verdi	Venere	Roma
Bruni	Saturno	Milano

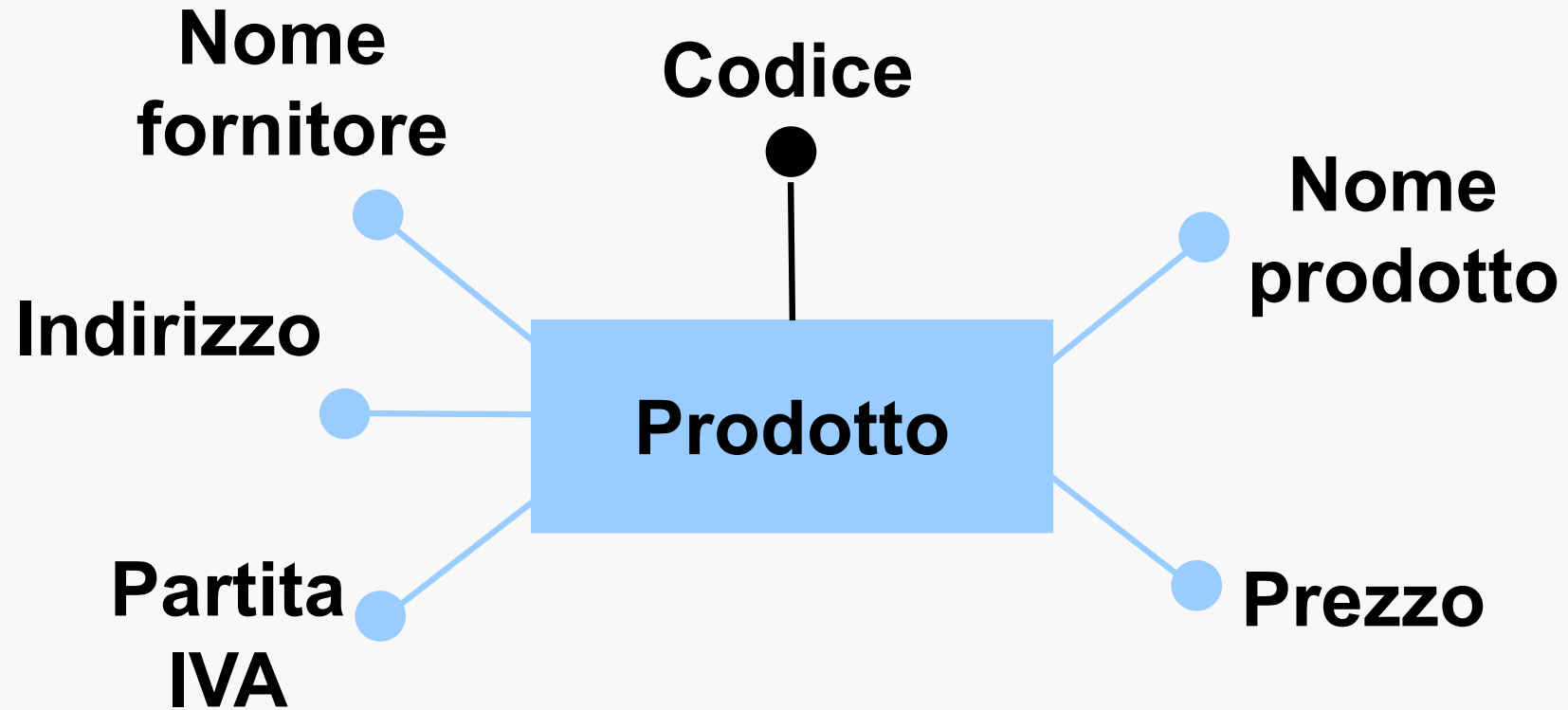
Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Bruni	Saturno

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Verdi	Roma
Bruni	Milano

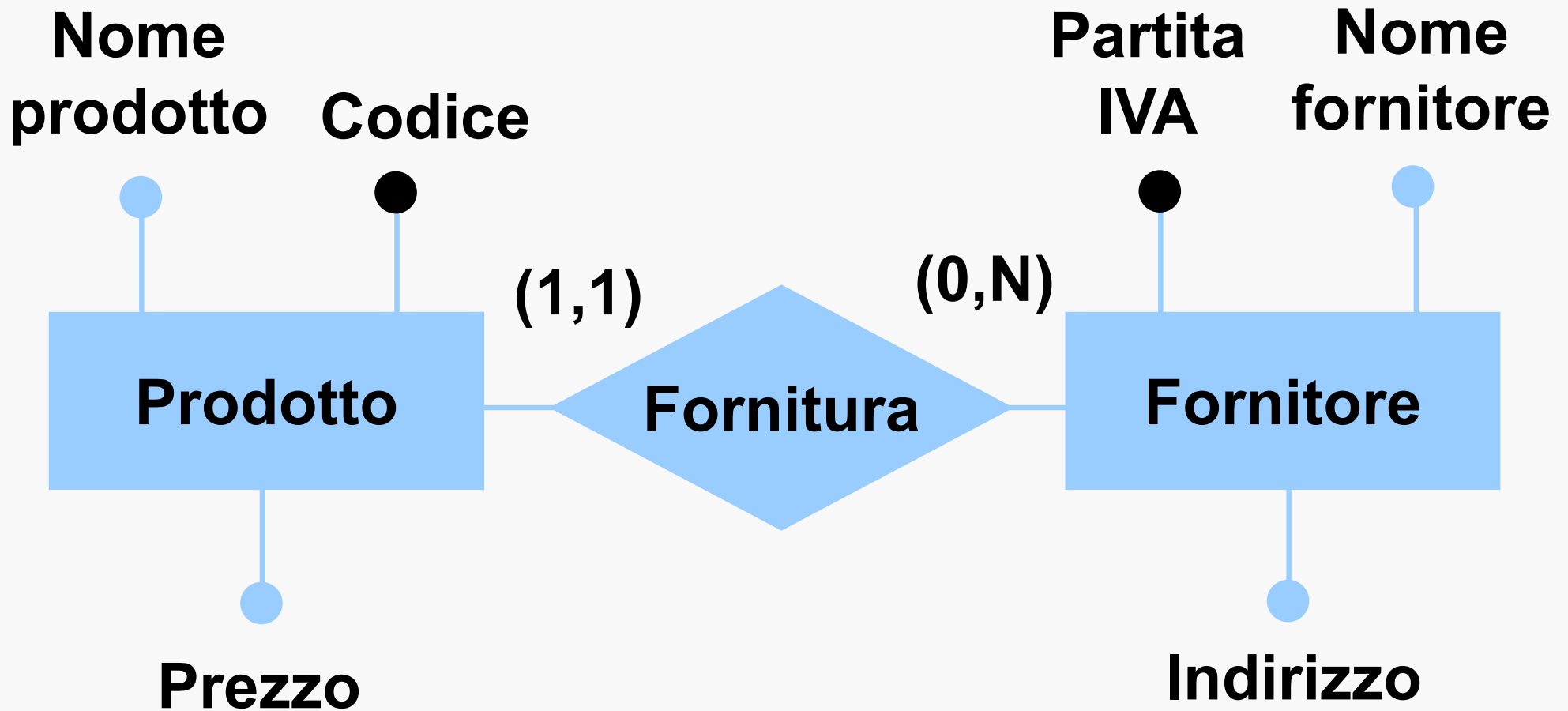


Progettazione e normalizzazione

- La teoria della normalizzazione può essere usata nella progettazione logica per verificare lo schema relazionale finale
- Si può usare anche durante la progettazione concettuale per verificare la qualità dello schema concettuale



PartitaIVA → NomeFornitore Indirizzo



PRODOTTI

<u>CODICE</u>	NOME P	PREZZO	P.IVA	NOME F	IND
---------------	--------	--------	-------	--------	-----

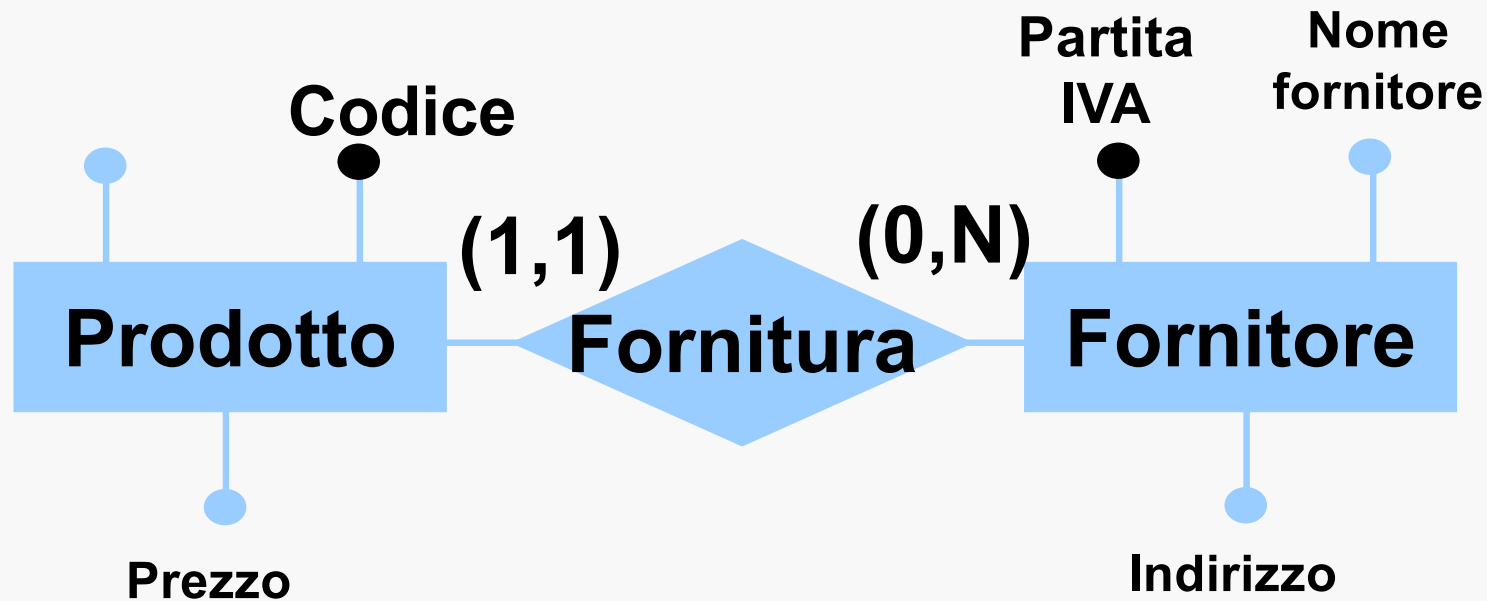
X P.IVA → NOME F, IND

PRODOTTI

<u>CODICE</u>	NOME P	PREZZO	P.IVA
---------------	--------	--------	-------

FORNITORE

<u>P.IVA</u>	NOME F	IND
--------------	--------	-----

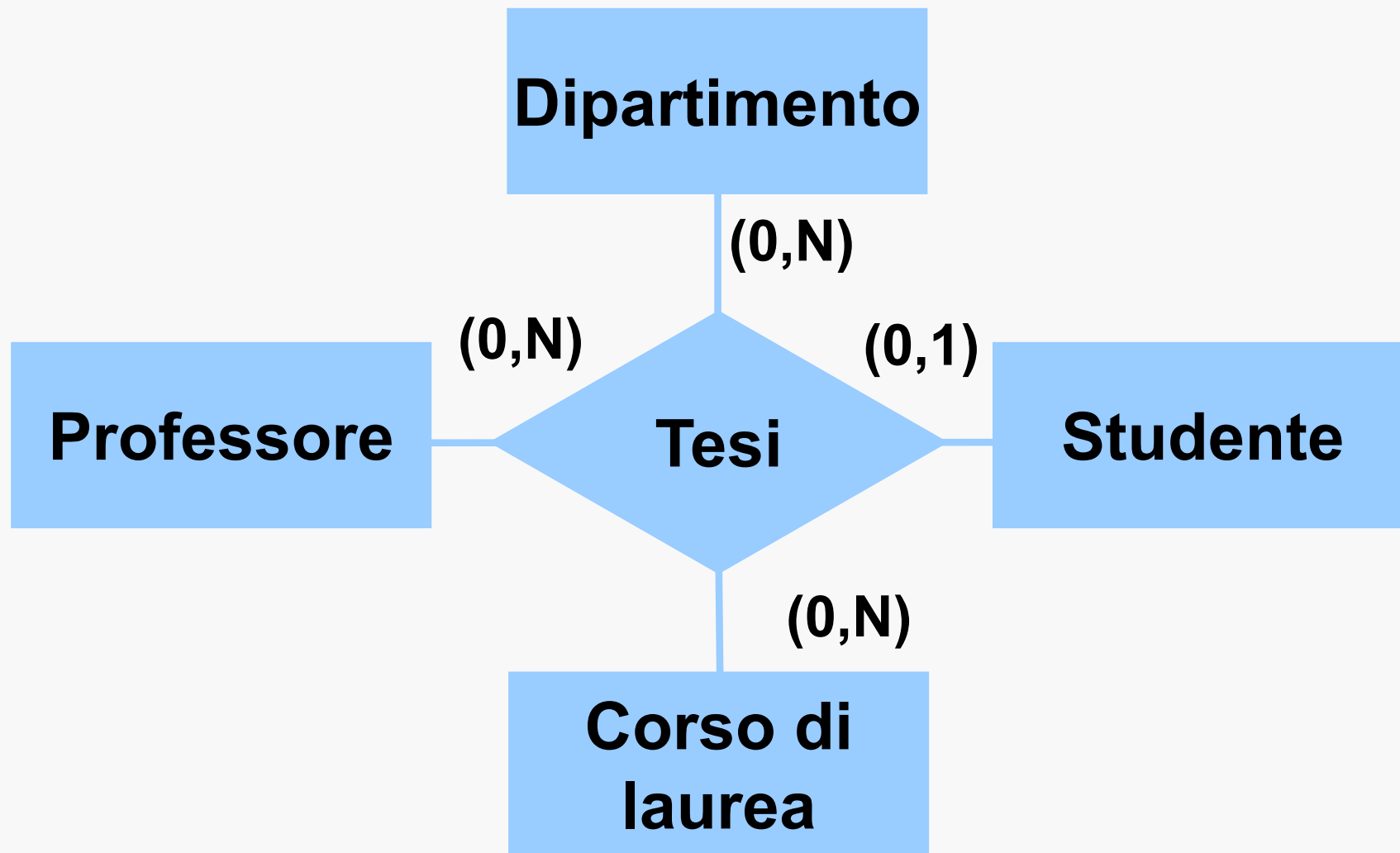


PRODOTTO

<u>CODICE</u>	NOME P	PREZZO	P.IVA

FORNITORE

<u>P.IVA</u>	NOME F	IND



Assumiamo che TESI coinvolga

S C P D quando

S svolge una tesi per il
CdL C con il prof P
presso il dip. D (che è
il dip del prof P)

T

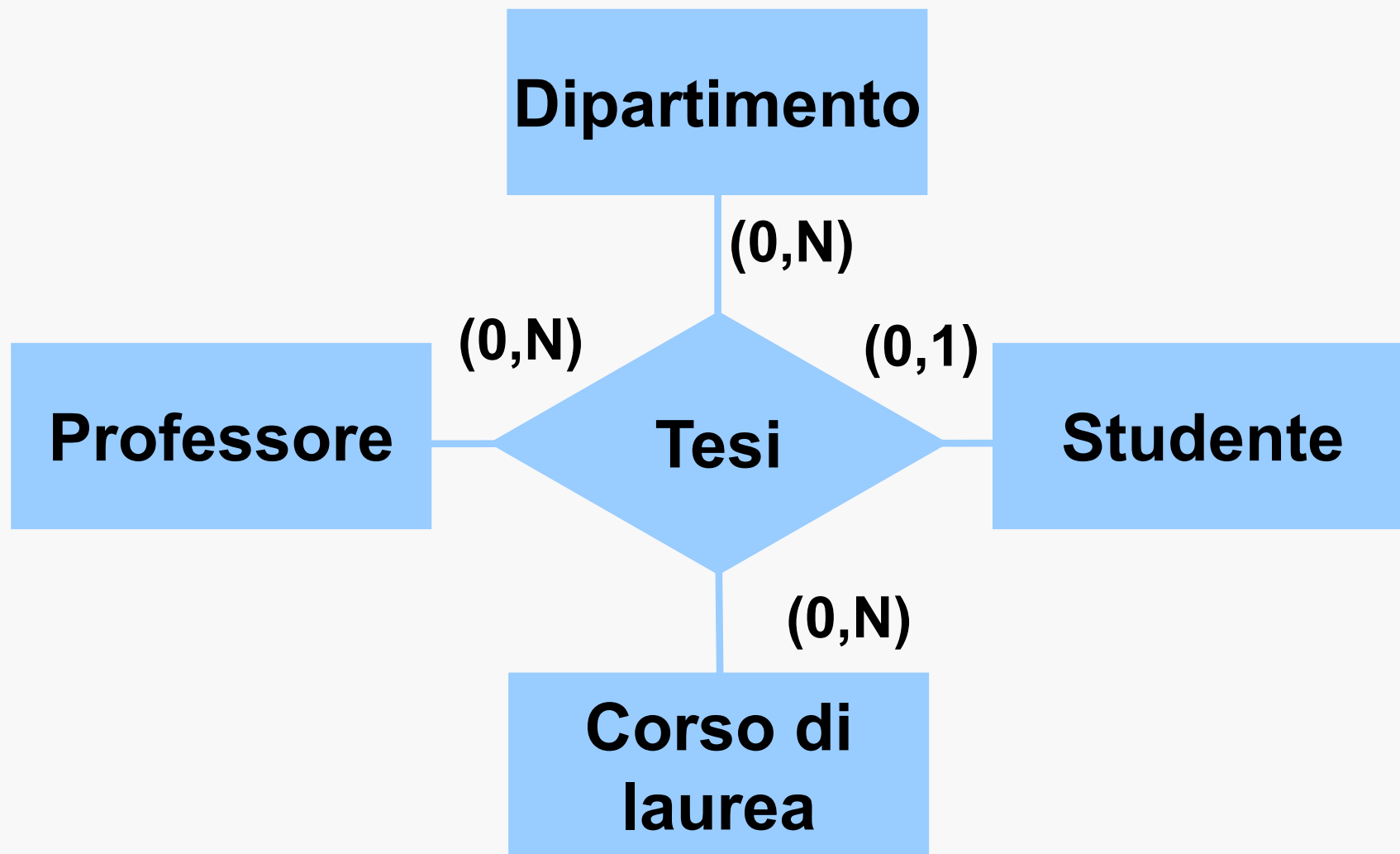
S	C	P	D

$S \rightarrow C$

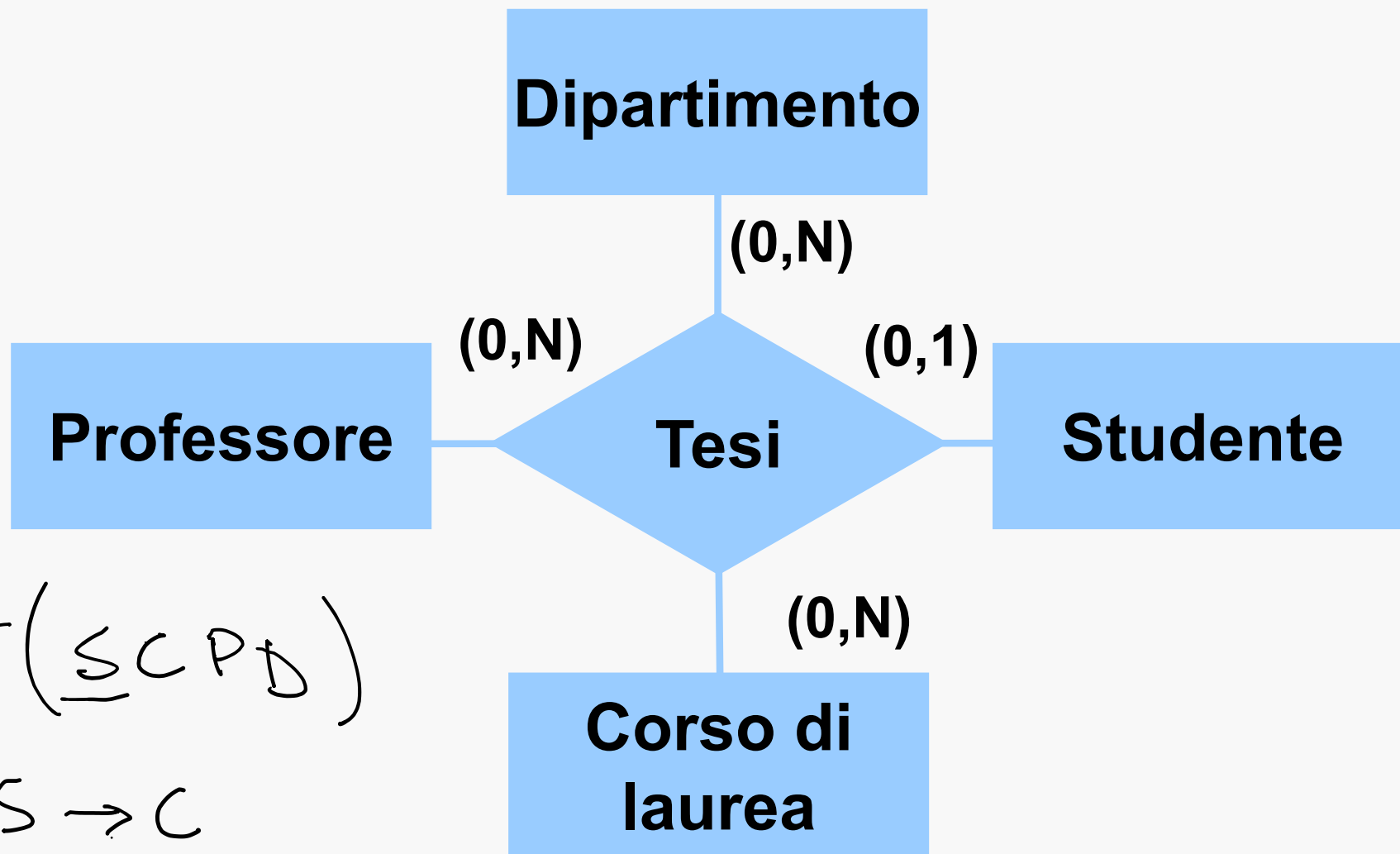
$S \rightarrow P$

$P \rightarrow D$

CHIAVE
S



Studente → Corso di laurea
Studente → Professore
Professore → Dipartimento



$T(\underline{SCPD})$

✓ $S \rightarrow C$

✓ $S \rightarrow P$

✗ $P \rightarrow D$

Studente → Corso di laurea
Studente → Professore
Professore → Dipartimento

T (S, C, P, D)

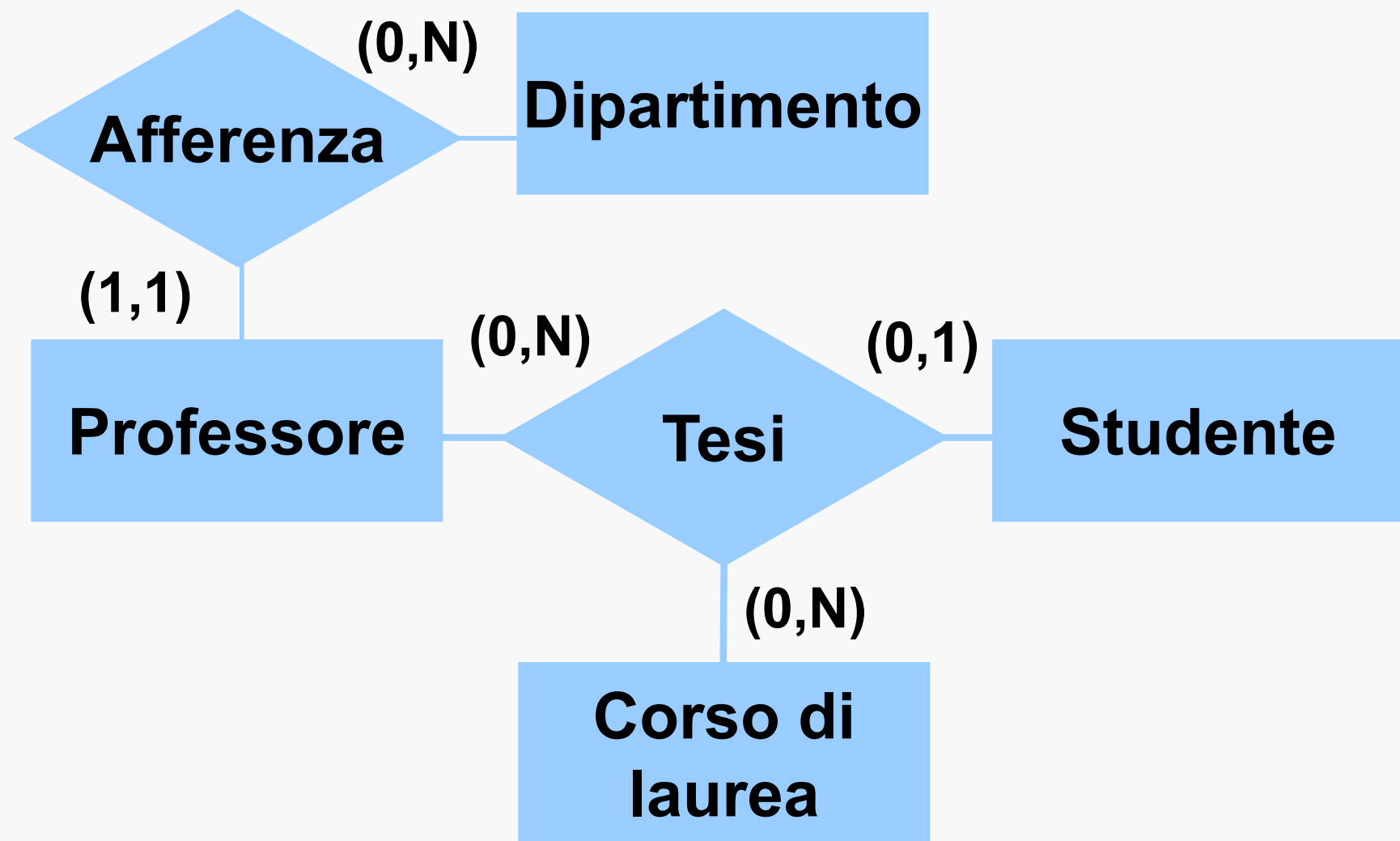
✓ $S \rightarrow C$

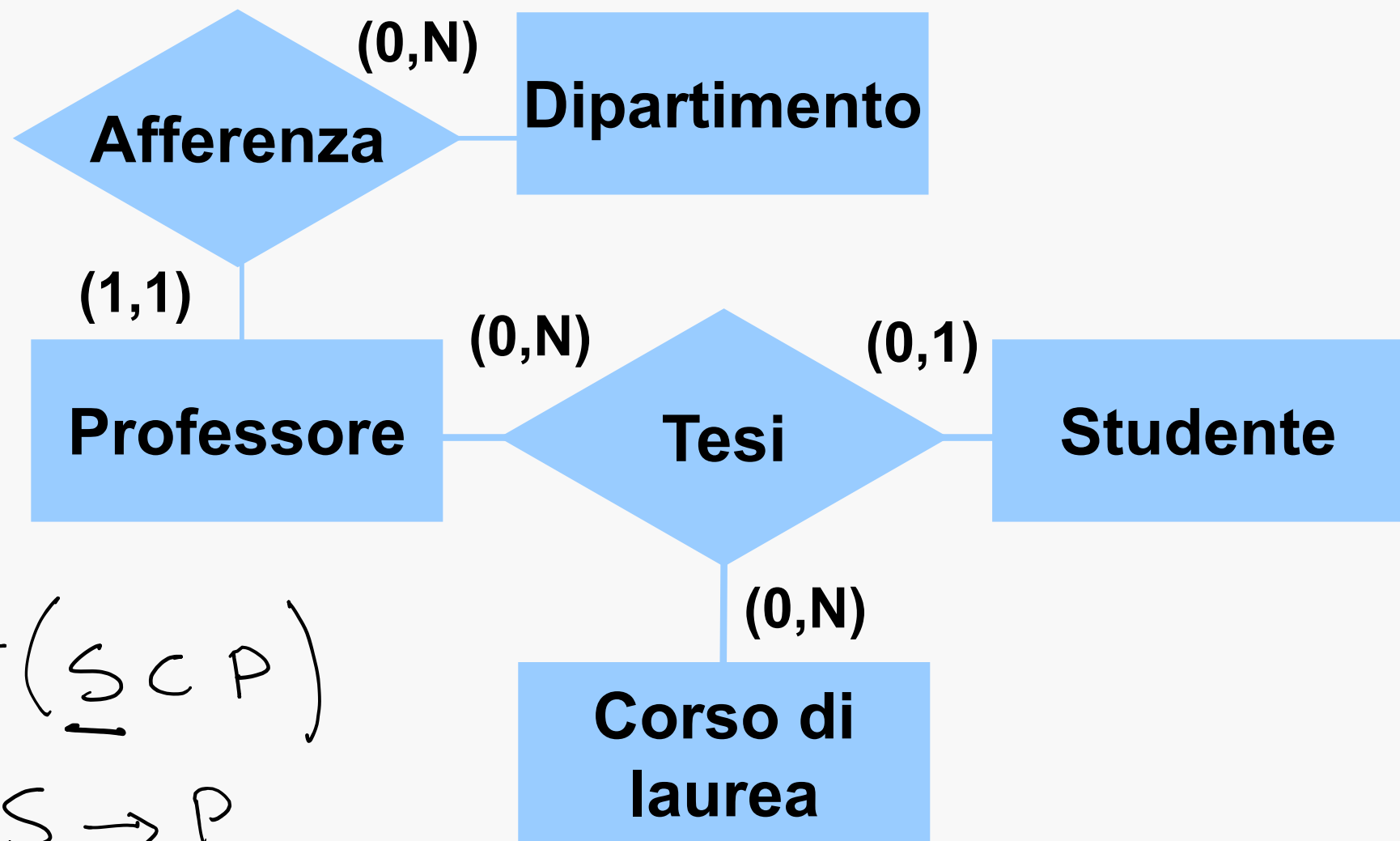
✓ $S \rightarrow P$

✗ $P \rightarrow D$

$R_1(\underline{S}, C, P)$

$R_2(\underline{P}, D)$





$T(\underline{S}CP)$

✓ $S \rightarrow P$

✓ $S \rightarrow C$

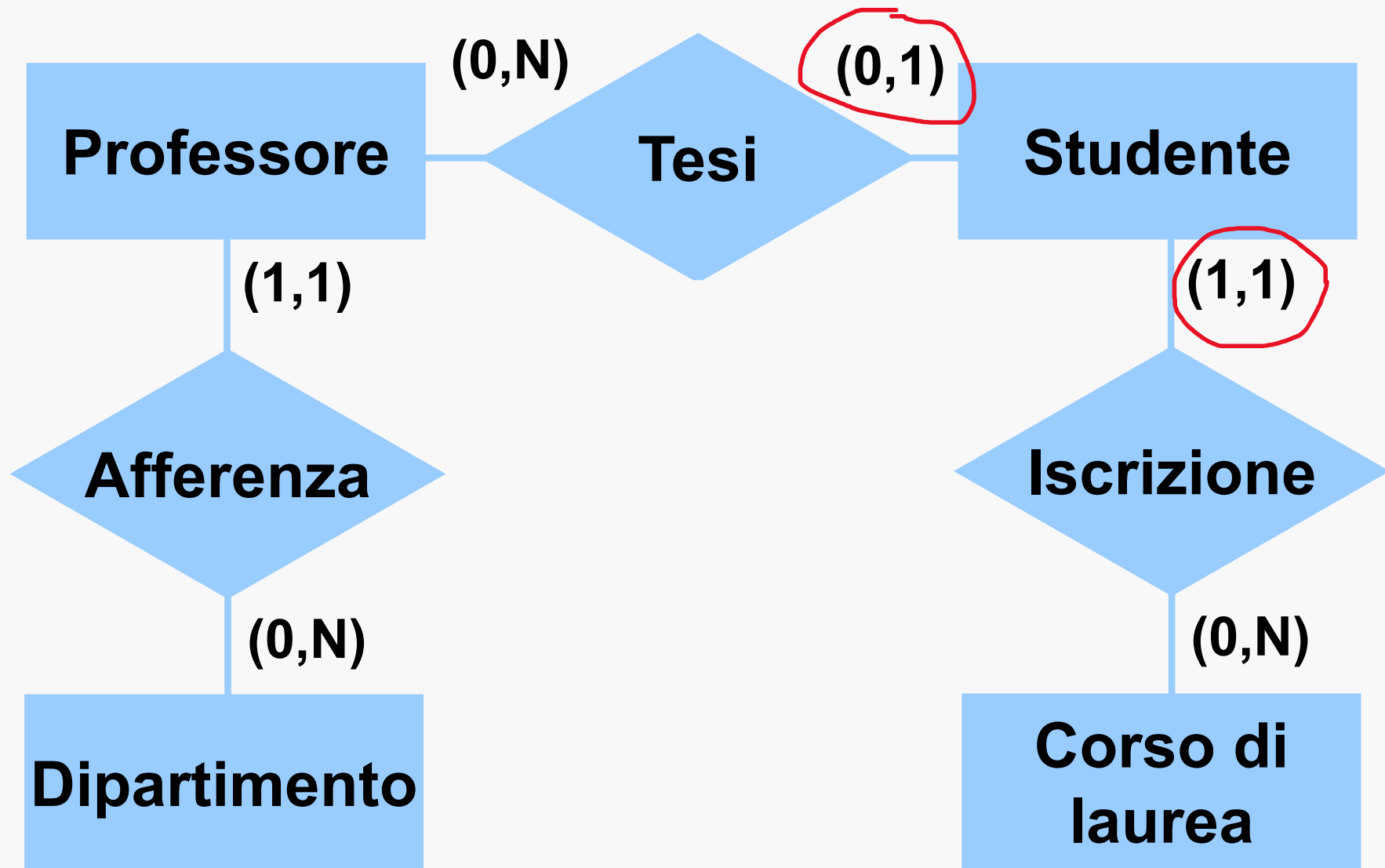
Ulteriore analisi sulla base delle dipendenze

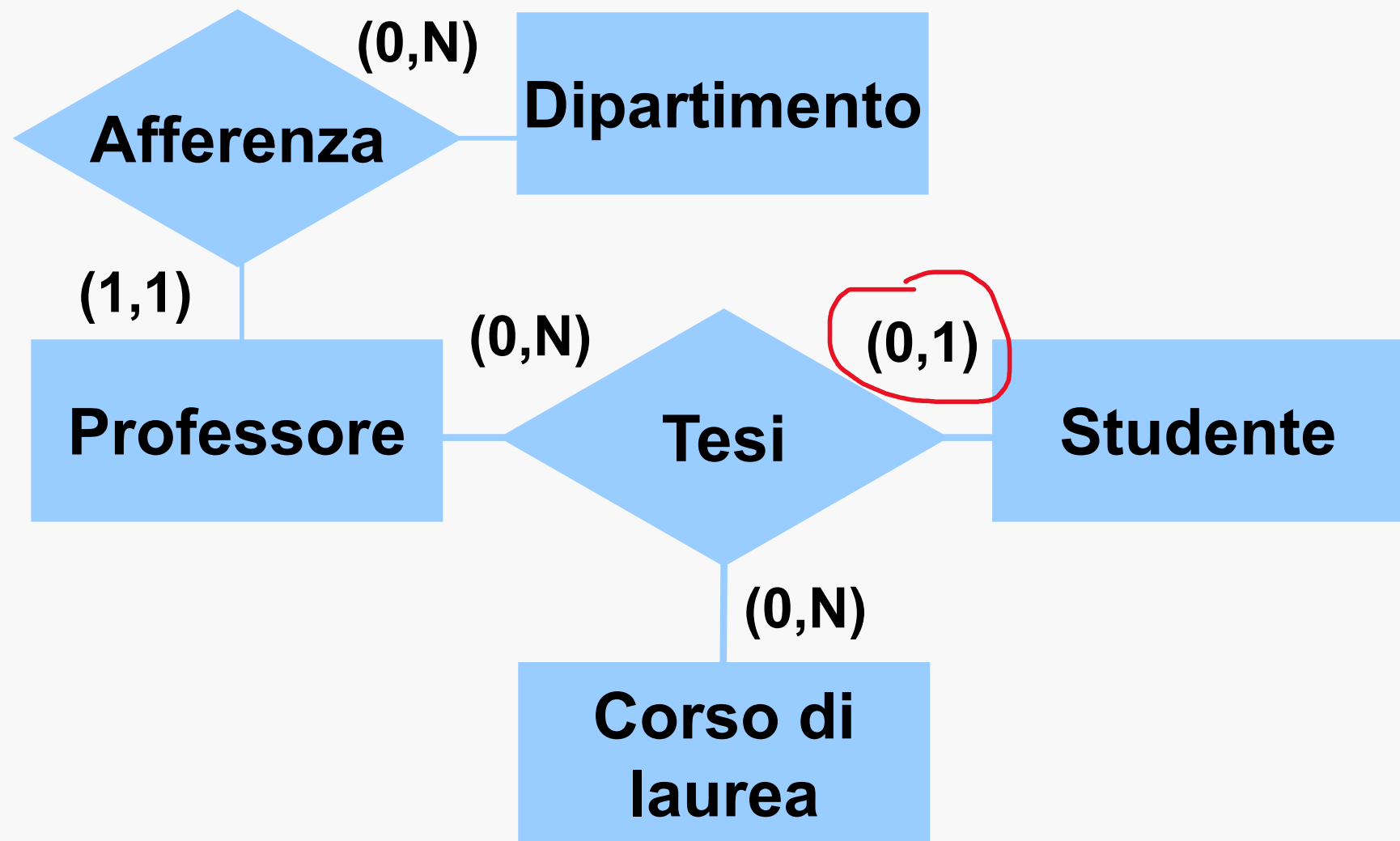
- La relationship **Tesi** (su S C P) è in BCNF sulla base delle dipendenze

Studiante → **CorsoDiLaurea**

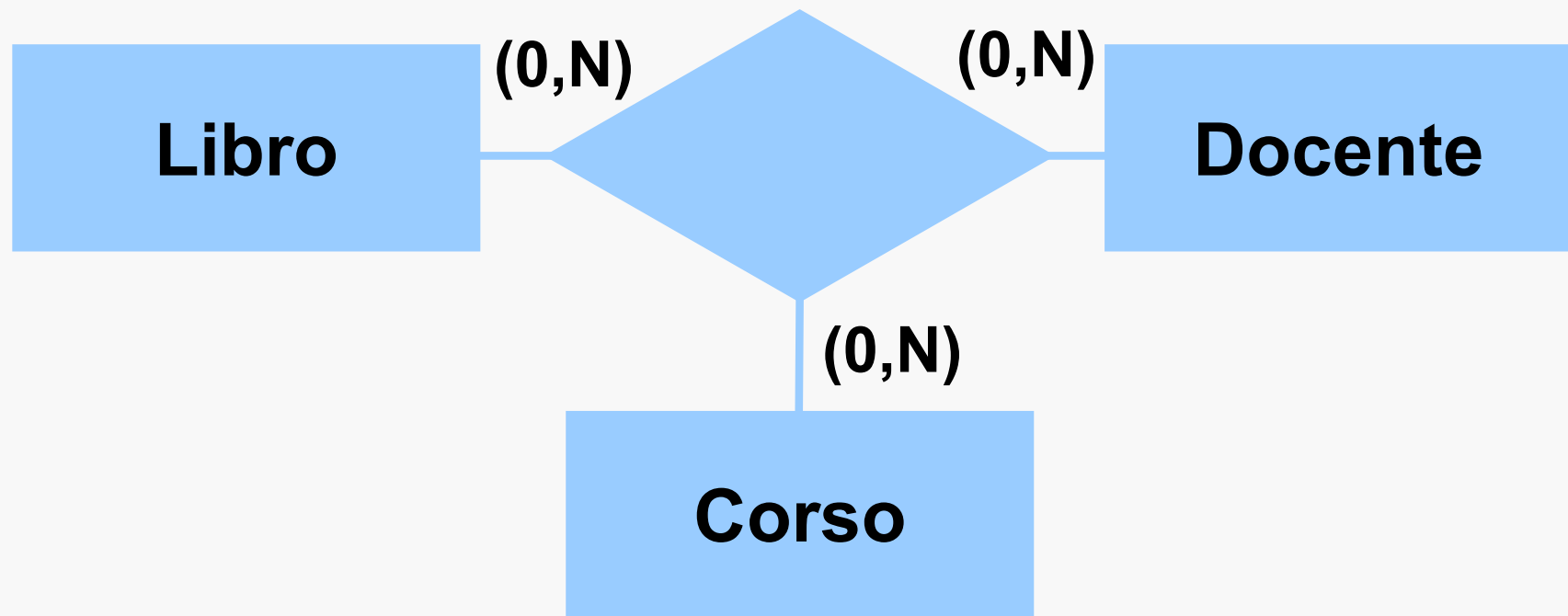
Studiante → **Professore**

- Però, le due proprietà sono indipendenti
- Questo suggerisce una ulteriore decomposizione





Altro esempio, più sottile



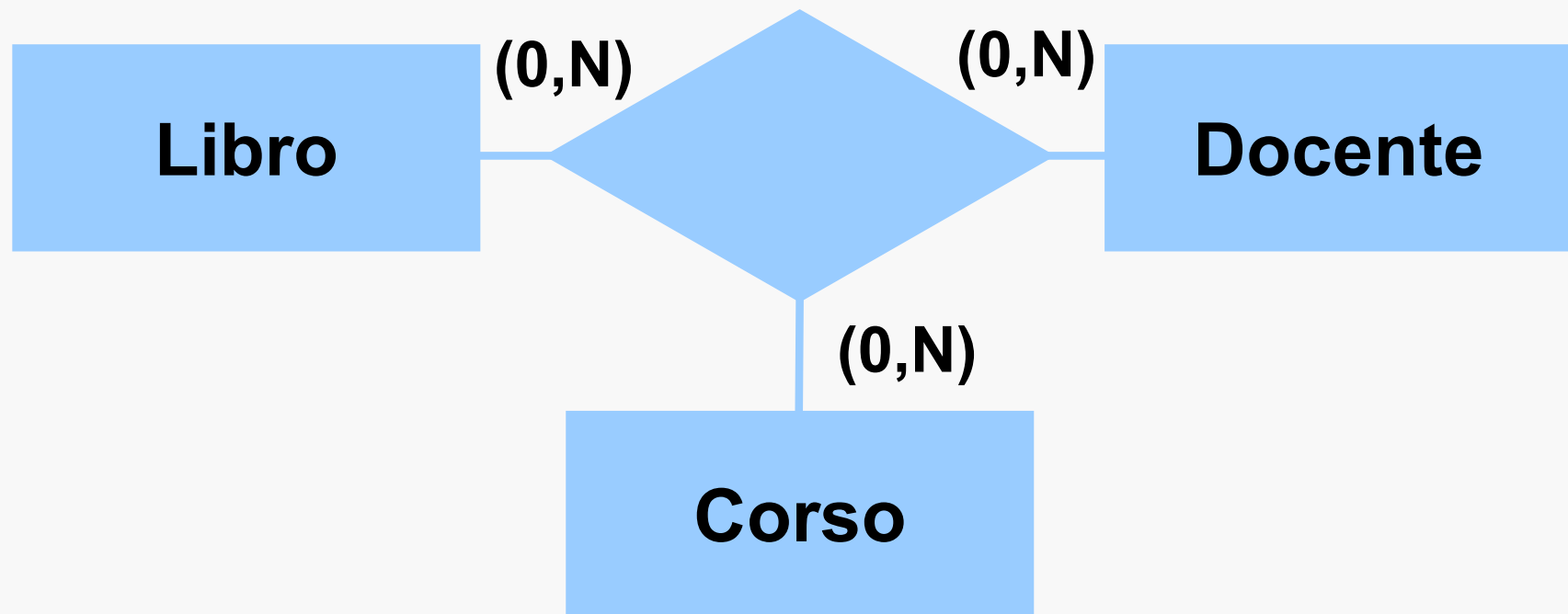
Dettagli

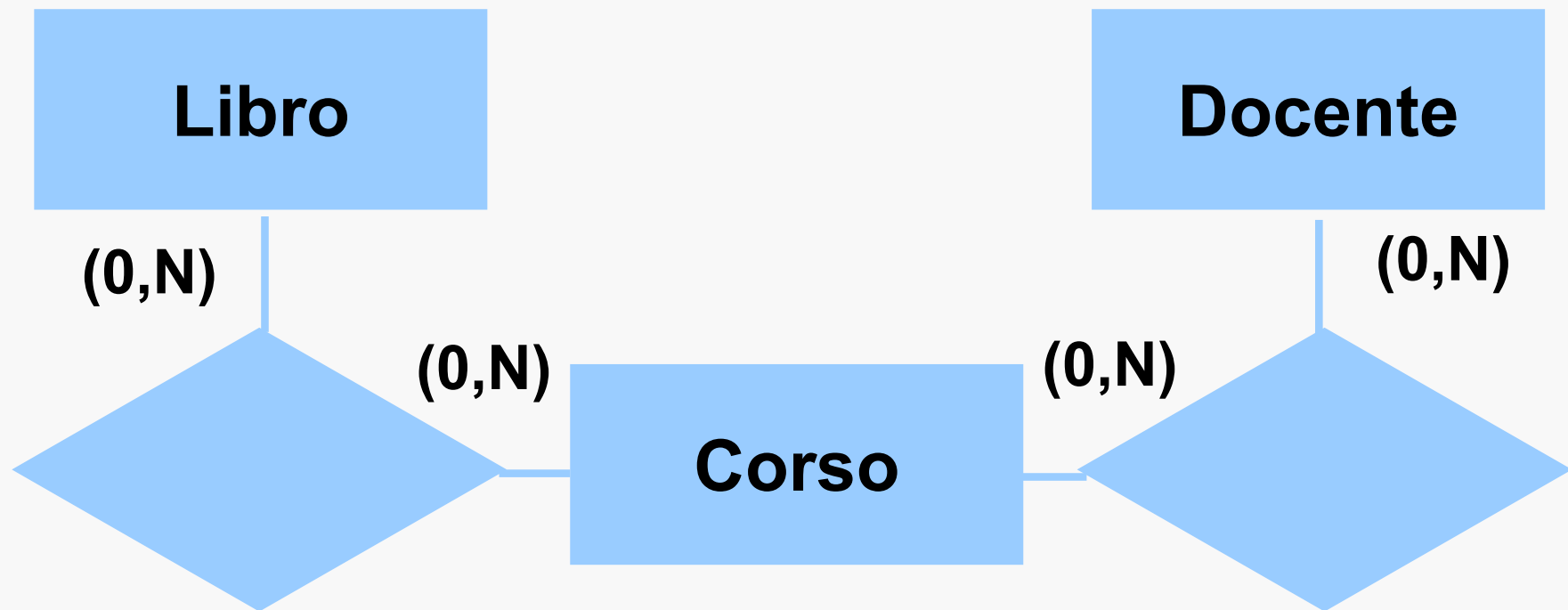
- Un corso ha uno o più docenti (ad esempio più canali) e uno o più libri di testo
 - ognuno dei docenti di un corso usa tutti i libri del corso
- Un docente può tenere più corsi
- Un libro può essere usato in più corsi

In sostanza

- Non c'è legame diretto fra docente e libro
- I legami sono
 - fra corso e docente
 - fra corso e libro

Quindi ...





Perché possiamo decomporre?

- Perché le due relationship sono indipendenti!

Abbiamo visto qualcosa di simile nel relazionale

Impiegato	Progetto	Sede
Rossi	Marte	Roma
Verdi	Giove	Milano
Verdi	Venere	Milano
Verdi	Giove	Roma
Verdi	Venere	Roma
Bruni	Saturno	Milano

Impiegato	Progetto
Rossi	Marte
Verdi	Giove
Verdi	Venere
Bruni	Saturno

Impiegato	Sede
Rossi	Roma
Verdi	Milano
Verdi	Roma
Bruni	Milano

Corso, libro e docente nel modello relazionale

Corso	Docente	Libro
BD	Atzeni	ACFPT1
BD	Atzeni	ACFPT2
BD	Torlone	ACFPT1
BD	Torlone	ACFPT2
APS	Cabibbo	Larman

Corso	Docente
BD	Atzeni
BD	Torlone
APS	Cabibbo

Corso	Libro
BD	ACFPT1
BD	ACFPT2
APS	Larman

Nota bene, riesco a ricostruire

Corso	Docente	Libro
BD	Atzeni	ACFPT1
BD	Atzeni	ACFPT2
BD	Torlone	ACFPT1
BD	Torlone	ACFPT2
APS	Cabibbo	Larman

Corso	Docente
BD	Atzeni
BD	Torlone
APS	Cabibbo

Corso	Libro
BD	ACFPT1
BD	ACFPT2
APS	Larman

Corso	Docente	Libro
BD	Atzeni	ACFPT1
BD	Atzeni	ACFPT2
BD	Torlone	ACFPT1
BD	Torlone	ACFPT2
APS	Cabibbo	Larman