

5. La progettazione logica

5.3 traduzione diretta nel modello relazionale

1. introduzione alla progettazione logica
2. ristrutturazione dello schema ER
3. **traduzione diretta nel modello relazionale**
4. ristrutturazione dello schema logico

Proprietà dello schema ristrutturato

La fase di ristrutturazione ha prodotto uno schema ER ristrutturato con le seguenti proprietà:

- preserva la semantica dello schema originale. Intuitivamente, esiste una funzione che associa ad ogni istanza dello schema originale un'opportuna istanza dello schema ristrutturato e viceversa
- può contenere delle ridondanze, ma sono volute per motivi di efficienza e sono comunque documentate
- non contiene attributi multivalore
- non contiene attributi composti
- non contiene ISA o generalizzazione (né tra entità, né tra relazioni); quindi tutte le entità sono disgiunte a coppie
- tutte le entità hanno un unico identificatore principale

Lo schema ristrutturato è il punto di partenza per la traduzione nel modello relazionale.

Traduzione diretta

- La traduzione diretta ha lo scopo di tradurre lo schema ER ristrutturato (con vincoli) in uno schema relazionale con vincoli che rappresenti le stesse informazioni
- Non richiede di effettuare scelte (tranne in un caso), in quanto si basa sulle scelte fatte in fase di ristrutturazione
- Produce uno schema logico di massima, che può essere accettabile, ma che può richiedere successive ristrutturazioni
- Consiste delle seguenti **attività**:
 - traduzione delle **entità** in relazioni dello schema logico, con relativi vincoli
 - traduzione delle **relazioni** dello schema ER in relazioni dello schema logico, con relativi vincoli
 - traduzione dei **vincoli esterni**
 - riformulazione di **operazioni** e **specifiche** sul carico applicativo in termini dello schema logico

Per distinguere tra le due accezioni di relazione, useremo il termine ER-relazione per denotare le relazioni dello schema ER

Notazione

- Nella fase di traduzione diretta ed in quella di ristrutturazione dello schema logico, esprimeremo gli schemi relazionali mediante una notazione che prevede di descrivere le relazioni con nome e attributi, ed i vincoli ad esse associati in forma testuale
- Esempio:
 - Partecipa(Cognome, DataN, Progetto, OreSett, Iva*)
 - foreign key: Partecipa[Cognome,DataN] \subseteq Impiegato[Cognome,DataN]
 - inclusione: Partecipa[OreSett] \subseteq Orario[Ore]
 - chiave: Progetto
- Ovviamente, questa notazione si può tradurre senza difficoltà in termini di "create table" in SQL. Si lascia allo studente la verifica di come effettuare la traduzione in SQL

Traduzione di entità: regole generali

- Ogni entità **E** dello schema ER viene tradotta in una relazione **R_E** dello schema relazionale
- Gli **attributi** della relazione **R_E** sono:
 - gli attributi dell'entità **E** (tutti not null, tranne quelli opzionali)
 - gli attributi derivanti dall'accorpamento di ER-relazioni in **R_E** – per ogni ER-relazione **Q** accorpata in **R_E** vengono aggiunti ad **R_E** come attributi:
 - gli attributi della ER-relazione **Q**
 - le chiavi primarie delle relazioni che corrispondono alle altre entità che partecipano a **Q**
- Una relazione **Q** viene **accorpata** in **R_E** quando un ruolo di **Q** partecipa all'identificatore principale (esterno) di **E** (si noti che in questo caso **E** partecipa a **Q** con cardinalità (1,1), ed inoltre **E** è l'unica entità per cui un ruolo di **Q** partecipa all'identificatore principale esterno, altrimenti ci sarebbe un ciclo di identificazione principale esterna)
- La **chiave primaria** di **R_E** è determinata in base all'identificatore principale di **E** (attributi di **E** e/o derivanti dall'identificazione esterna)
- Agli altri identificatori di **E** corrispondono dei **vincoli di chiave** su **R_E**
- A seconda dei casi, possono essere necessari ulteriori vincoli

Traduzione di ER-relazioni: regole generali

- Ogni relazione **Q** dello schema ER che non è stata accorpata al passo precedente viene tradotta in una relazione **R_Q** dello schema relazionale.
- Gli **attributi** della relazione **R_Q** sono:
 - gli attributi della ER-relazione **Q**
 - le chiavi primarie delle entità che partecipano alla ER-relazione **Q**
- Scelta della **chiave primaria** di **R_Q**:
 - Se nessuna entità partecipa con cardinalità massima 1 a **Q**, allora la chiave primaria di **R_Q** è costituita dalla combinazione delle chiavi primarie delle entità partecipanti
 - Altrimenti, la chiave primaria di ogni entità che partecipa con cardinalità massima 1 a **Q** è chiave di **R_Q**, e la chiave primaria di **R_Q** va scelta tra queste chiavi candidate
- Le chiavi candidate rimanenti divengono **vincoli di chiave** su **R_Q**
- Le tipizzazioni delle componenti di **Q**, per ogni ruolo, con le entità partecipanti divengono in **R_Q** vincoli di **foreign key** verso le relazioni che corrispondono alle entità partecipanti
- A seconda dei casi, possono essere necessari ulteriori vincoli

Traduzione di vincoli: regole generali

Questi sono i vincoli da considerare:

- Vincoli **not null** per gli attributi obbligatori
- Vincoli di interdipendenza di valori nulli (provenienti da attributi composti opzionali), formulati come vincoli di tupla
- Vincoli di **chiave** (primarie e non)
- Vincoli di **foreign key** che provengono
 - dalla tipizzazione di relazioni (incluse quelle che sono state accorpate in entità)
 - dai vincoli esterni derivanti dall'ISA di relazioni
- Vincoli di **generalizzazione**, formulati come vincoli insiemistici
- Vincoli di cardinalità:
 - partecipazione obbligatoria (cardinalità minima 1) diventa **vincolo di inclusione** o **foreign key** dalla relazione che corrisponde all'entità verso quella che corrisponde alla ER-relazione
 - funzionalità (cardinalità massima 1) diventa **vincolo di chiave** sulla relazione che corrisponde alla ER-relazione
 - gli altri vincoli di cardinalità diventano vincoli esterni
- Gli altri vincoli esterni vanno opportunamente tradotti

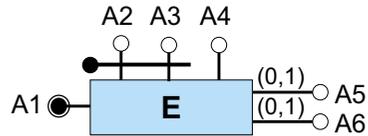
Riformulazione di operazioni e carico applicativo: regole generali

- Le operazioni e le informazioni sul carico applicativo sono state espresse all'inizio della progettazione logica sulla base dello schema concettuale, e poi modificate per renderle coerenti con lo schema concettuale ristrutturato. È ora necessario riformulare le operazioni e le informazioni sul carico applicativo in modo che siano coerenti con lo schema logico
- La riformulazione viene condotta semplicemente tenendo presente come le entità e le relazioni dello schema Entità-Relazione ristrutturato sono state tradotte nello schema relazionale

Traduzione di entità senza accorpamento

Consideriamo per ora un'entità per cui non si effettua accorpamento di relazioni (in particolare, un'entità che non ha identificatori esterni).

- L'entità si traduce in una **relazione** dello schema relazionale
- Gli **attributi** della relazione corrispondente all'entità sono quelli dell'entità.
 - se un attributo è opzionale diventa un attributo della relazione che può assumere valore nullo (tali attributi sono indicati con * nello schema logico)
 - altrimenti l'attributo non può assumere valore nullo
- L'identificatore principale dell'entità si traduce nella **chiave primaria** della relazione
- Gli altri identificatori interni si traducono in **chiavi** della relazione
- Ricordarsi dei vincoli esterni per identificatori opzionali correlati (derivanti da attributi composti opzionali)

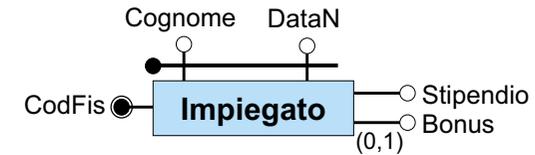


$E(\underline{A1}, A2, A3, A4, A5^*, A6^*)$

chiave: A2, A3

vincolo: A5 è NULL se e solo se A6 è NULL

Traduzione di entità senza accorpamento: esempio



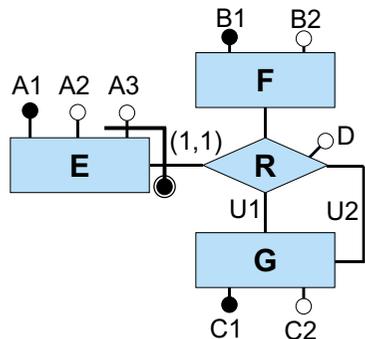
$Impiegato(\underline{CodFis}, Cognome, DataN, Stipendio, Bonus^*)$

chiave: Cognome, DataN

Traduzione di entità con accorpamento: caso 1

Consideriamo il caso in cui una ER-relazione **R** è **parte dell'identificatore principale esterno di un'entità E con ruolo U**, e in tutti gli altri ruoli di **R** la cardinalità è (0,n).

- La ER-relazione **R** viene **accorpata** nell'entità **E**. Questo significa che tutti gli attributi della ER-relazione e le chiavi primarie delle altre entità partecipanti diventano attributi della relazione RE che corrisponde all'entità **E**. Tali chiavi primarie fanno parte della chiave primaria della relazione RE.
- Si noti che eventuali altri identificatori dell'entità in cui la relazione è stata accorpata si traducono in vincoli di chiave sull'entità.



$E(\underline{A1}, A2, A3, \underline{F}, \underline{U1}, \underline{U2}, D)$

foreign key: $E[F] \subseteq F[B1]$

foreign key: $E[U1] \subseteq G[C1]$

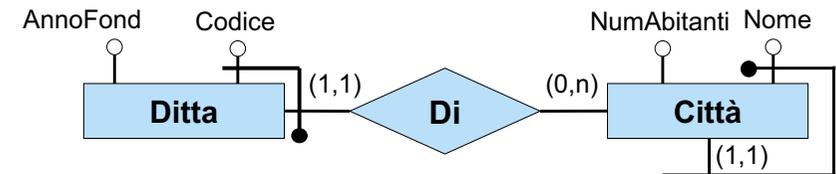
foreign key: $E[U2] \subseteq G[C2]$

chiave: A1

$F(\underline{B1}, B2)$

$G(\underline{C1}, C2)$

Traduzione di entità con accorpamento - caso 1: esempio



$Ditta(\underline{Codice}, \underline{NomeCit}, \underline{Nazione}, AnnoFond)$

foreign key: $Ditta[NomeCit, Nazione] \subseteq Città[Nome, Nazione]$

$Città(\underline{Nome}, \underline{Nazione}, NumAbitanti)$

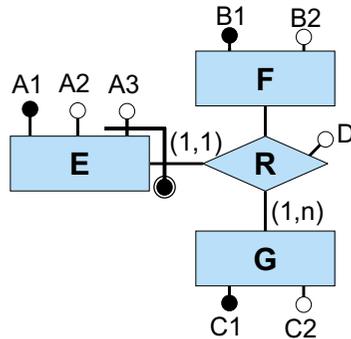
foreign key: $Città[Nazione] \subseteq Nazione[Nome]$

$Nazione(\underline{Nome}, Fascia)$

Traduzione di entità con accorpamento: caso 2

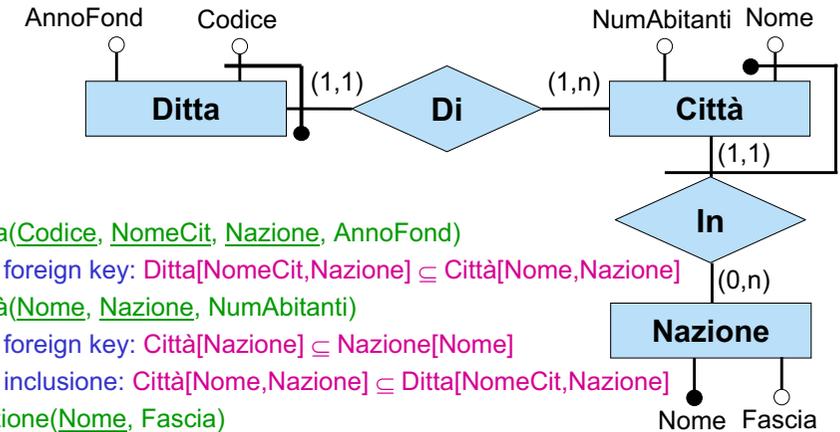
Consideriamo il caso in cui una ER-relazione **R** è **parte dell'identificatore principale esterno di un'entità E con ruolo U**, e in alcuni degli altri ruoli di **R** la cardinalità è (1,n).

- La ER-relazione **R** viene **accorpata** nell'entità **E** in modo analogo al caso 1
- Per ogni ruolo **U** con cardinalità (1,n), si aggiunge un vincolo di inclusione tra la chiave dell'entità corrispondente al ruolo **U** e l'attributo (o gli attributi, se la chiave è composta) di **E** corrispondente ad **U**



$E(A1, A2, A3, E, G, D)$
 foreign key: $E[F] \subseteq F[B1]$
 foreign key: $E[G] \subseteq G[C1]$
 chiave: $A1$
 $F(B1, B2)$
 $G(C1, C2)$
 inclusione: $G[C1] \subseteq E[G]$

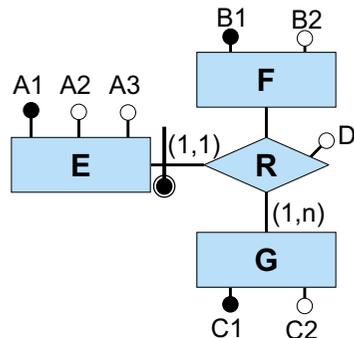
Traduzione di entità con accorpamento - caso 2: esempio



$Ditta(Codice, NomeCit, Nazione, AnnoFond)$
 foreign key: $Ditta[NomeCit, Nazione] \subseteq Città[Nome, Nazione]$
 $Città(Nome, Nazione, NumAbitanti)$
 foreign key: $Città[Nazione] \subseteq Nazione[Nome]$
 inclusione: $Città[Nome, Nazione] \subseteq Ditta[NomeCit, Nazione]$
 $Nazione(Nome, Fascia)$

Traduzione di entità con accorpamento: caso 3

Se una ER-relazione **R** è identificatore principale esterno (non parte di identificatore) di un'entità **E**, allora **R** viene accorpata in **E** secondo modalità analoghe a quelle viste nel caso 1 e nel caso 2.

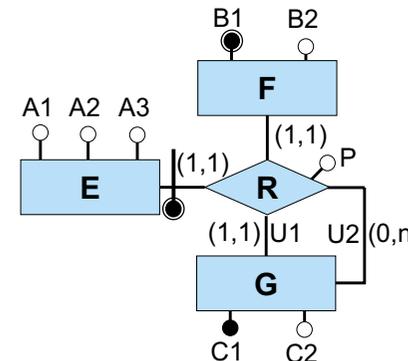


$E(A1, A2, A3, E, G, D)$
 foreign key: $E[F] \subseteq F[B1]$
 foreign key: $E[G] \subseteq G[C1]$
 chiave: $A1$
 $F(B1, B2)$
 $G(C1, C2)$
 inclusione: $G[C1] \subseteq E[G]$

Traduzione di entità con accorpamento: caso 4

Consideriamo il caso in cui una ER-relazione **R** è **identificatore principale esterno di un'entità E con ruolo U**, e in alcuni degli altri ruoli di **R** la cardinalità massima è 1.

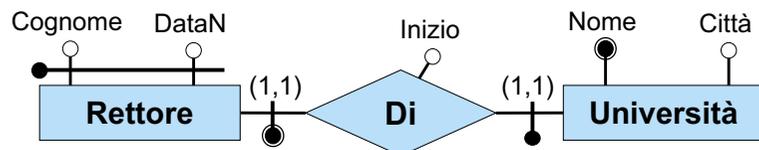
- La ER-relazione **R** viene **accorpata** nell'entità **E** in modo analogo al caso 1
- Ma in questo caso la chiave primaria della relazione che rappresenta **E** deve essere scelta tra le chiavi primarie delle altre entità coinvolte in **R** i cui ruoli hanno cardinalità massima 1



Si sceglie la chiave primaria di **F** come parte di chiave primaria di **E**:

$E(A1, A2, A3, E, U1, U2, P)$
 foreign key: $E[F] \subseteq F[B1]$
 foreign key: $E[U1] \subseteq G[C1]$
 foreign key: $E[U2] \subseteq G[C1]$
 chiave: $A3, U1$
 $F(B1, B2)$
 foreign key: $F[B1] \subseteq E[F]$
 $G(C1, C2)$
 inclusione: $G[C1] \subseteq E[U1]$

Traduzione di entità con accorpamento - caso 4: esempio



Rettore(Cognome, DataN, Inizio, Università)

foreign key: Rettore[Università] \subseteq Università[Nome]

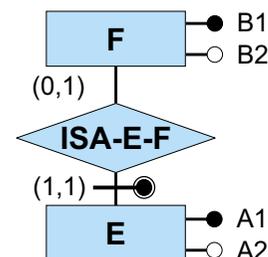
chiave: Cognome, DataN

Università(Nome, Città)

foreign key: Università[Nome] \subseteq Rettore[Università]

Accorpamento di relazione derivante da ISA

Un caso di ER-relazione che è identificatore principale esterno di un'entità può essere quello derivante dalla ristrutturazione di un'ISA nello schema ER originale.



$E(A1, A2, \underline{B1})$

foreign key: $E[B1] \subseteq F[B1]$

chiave: A1

$F(\underline{B1}, B2)$

Si noti come la traduzione della parte di schema ER che si ottiene dalla ristrutturazione di **E ISA F** corrisponda ad aggiungere agli attributi di **E** la chiave primaria di **F**, ed a rendere tali attributi anche chiave primaria di **E**. Il vincolo derivante dall'ISA dello schema ER originario diventa quindi un vincolo di foreign key dello schema logico.

Accorpamento di relazione derivante da ISA: esempio



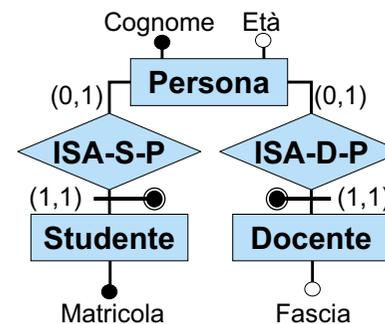
Studente(Cognome, DataN, Matricola, MediaVoti)

foreign key: Studente[Cognome, DataN] \subseteq Persona[Cognome, DataN]

chiave: Matricola

Persona(Cognome, DataN, Indirizzo)

Traduzione di vincoli di generalizzazione



Vincolo di generalizzazione:

nessuna istanza di Persona partecipa sia a ISA-S-P sia a ISA-D-P

Diventa sullo schema logico:

$Studente[Cognome] \cap Docente[Cognome] = \emptyset$

Persona(Cognome, Età)

Studente(Cognome, Matricola)

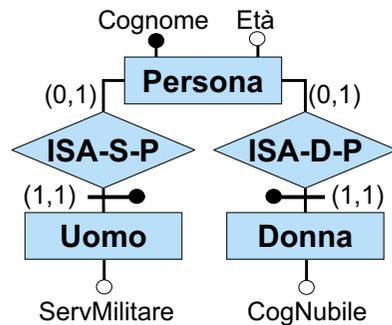
foreign key: Studente[Cognome] \subseteq Persona[Cognome]

chiave: Matricola

Docente(Cognome, Fascia)

foreign key: Docente[Cognome] \subseteq Persona[Cognome]

Traduzione di vincoli di generalizzazione completa



Vincolo di generalizzazione:

ogni istanza di **Persona** partecipa ad ISA-U-P oppure ad ISA-D-P, ma non ad entrambi

Diventa sullo schema logico:

$Uomo[Cognome] \cap Donna[Cognome] = \emptyset$
 $Persona[Cognome] \subseteq Uomo[Cognome] \cup Donna[Cognome]$

Persona(Cognome, Età)

Uomo(Cognome, ServMilitare)

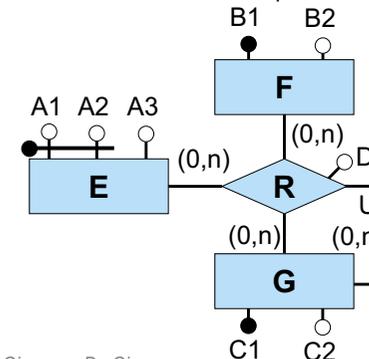
foreign key: Uomo[Cognome] \subseteq Persona[Cognome]

Donna(Cognome, CogNubile)

foreign key: Donna[Cognome] \subseteq Persona[Cognome]

Traduzione di relazione non accorpata

- Una ER-relazione che non è stata accorpata ad alcuna entità si traduce in una relazione
- Gli **attributi** della relazione sono quelli della ER-relazione, più le chiavi primarie delle relazioni corrispondenti alle entità partecipanti (per ogni ruolo)
- Se nessuna cardinalità massima è 1, allora la **chiave primaria** della relazione è data dalle chiavi primarie delle entità partecipanti
- Si definiscono vincoli di foreign key dalla relazione verso le entità partecipanti per dar conto dei vincoli di tipizzazione nello schema ER



E(A1, A2, A3)

F(B1, B2)

G(C1, C2)

R(EA1, EA2, F, G, U, D)

foreign key: R[EA1,EA2] \subseteq E[A1,A2]

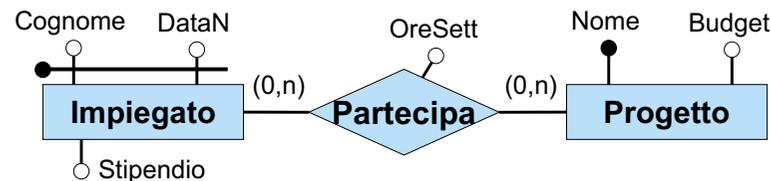
foreign key: R[F] \subseteq F[B1]

foreign key: R[G] \subseteq G[C1]

foreign key: R[U] \subseteq G[C1]

Traduzione di relazione non accorpata: esempio

Nello scegliere per una relazione il nome di un attributo che rappresenta la chiave primaria di un'entità che partecipa alla relazione, può essere opportuno utilizzare il nome del ruolo con cui l'entità partecipa alla relazione (invece del nome che l'attributo ha per l'entità).



Impiegato(Cognome, DataN, Stipendio)

Progetto(Nome, Budget)

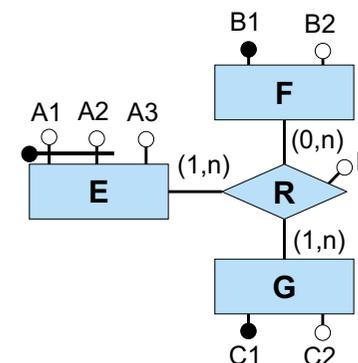
Partecipa(Cognome, DataN, Progetto, OreSett)

foreign key: Partecipa[Cognome,DataN] \subseteq Impiegato[Cognome,DataN]

foreign key: Partecipa[Progetto] \subseteq Progetto[Progetto]

Traduzione di relazione con cardinalità minima 1

- Un vincolo di cardinalità minima 1 per la partecipazione di un'entità (in un ruolo) ad una relazione non accorpata si traduce in un **vincolo di inclusione** dall'entità verso la relazione.
- Il vincolo di inclusione non è in generale di foreign key (si vedano in seguito i casi in cui l'inclusione diventa in realtà una foreign key).



E(A1, A2, A3)

inclusione: E[A1,A2] \subseteq R[A1,A2]

F(B1, B2)

G(C1, C2)

inclusione: G[C1] \subseteq R[C1]

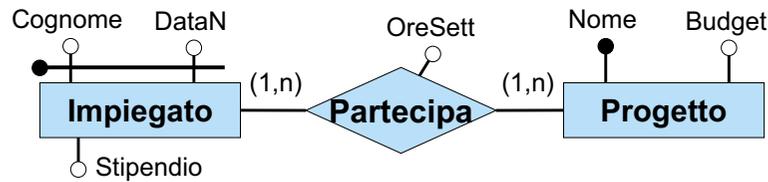
R(A1, A2, B1, C1, D)

foreign key: R[A1,A2] \subseteq E[A1,A2]

foreign key: R[B1] \subseteq F[B1]

foreign key: R[C1] \subseteq G[C1]

Traduzione di relazione con cardinalità minima 1: esempio



Impiegato(Cognome, DataN, Stipendio)

inclusione: Impiegato[Cognome,DataN] \subseteq Partecipa[Cognome,DataN]

Progetto(Nome, Budget)

inclusione: Progetto[Nome] \subseteq Partecipa[Progetto]

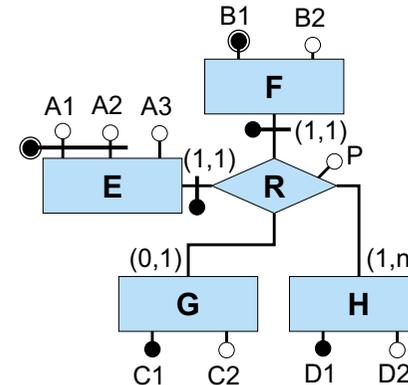
Partecipa(Cognome, DataN, Progetto, OreSett)

foreign key: Partecipa[Cognome,DataN] \subseteq Impiegato[Cognome,DataN]

foreign key: Partecipa[Progetto] \subseteq Progetto[Nome]

Traduzione di relazione con cardinalità massima 1

- Se per una ER-relazione (non accorpata) una entità partecipa (in un ruolo) con cardinalità massima 1, la chiave primaria dell'entità diventa una **chiave della relazione**. Si noti che, se l'entità ha anche cardinalità minima 1, il vincolo di inclusione corrispondente è in realtà un vincolo di foreign key.
- Se vi è più di una di tali entità, **bisogna scegliere la chiave primaria** della relazione tra le chiavi primarie di tali entità. Le chiavi primarie delle entità diverse da quella scelta si traducono in vincoli di chiave per la relazione.



$E(\underline{A1}, \underline{A2}, A3)$

foreign key: $E[A1,A2] \subseteq R[EA1,EA2]$

$F(\underline{B1}, B2)$

foreign key: $F[B1] \subseteq R[F]$

$G(\underline{C1}, C2)$

$H(\underline{D1}, D2)$

inclusione: $H[D1] \subseteq R[H]$

$R(EA1, EA2, F, \underline{G}, H, P)$

foreign key: $R[EA1,EA2] \subseteq E[A1,A2]$

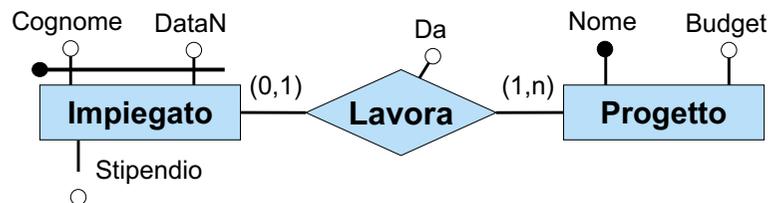
foreign key: $R[F] \subseteq F[B1]$

foreign key: $R[G] \subseteq G[C1]$

foreign key: $R[H] \subseteq H[D1]$

chiave: EA1, EA2 chiave: F

Traduzione di relazione con cardinalità massima 1: esempio



Impiegato(Cognome, DataN, Stipendio)

Progetto(Nome, Budget)

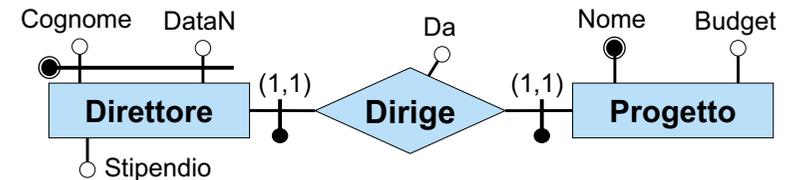
inclusione: Progetto[Nome] \subseteq Lavora[Progetto]

Lavora(Cognome, DataN, Progetto, Da)

foreign key: Lavora[Cognome,DataN] \subseteq Impiegato[Cognome,DataN]

foreign key: Lavora[Progetto] \subseteq Progetto[Nome]

Traduzione di relazione non accorpata con cardinalità (1,1): esempio



Direttore(Cognome, DataN, Stipendio)

foreign key: Direttore[Cognome,DataN] \subseteq Dirige[Cognome,DataN]

Progetto(Nome, Budget)

foreign key: Progetto[Nome] \subseteq Dirige[Progetto]

Dirige(Cognome, DataN, Progetto, Da)

chiave: Cognome, DataN

foreign key: Dirige[Cognome,DataN] \subseteq Direttore[Cognome,DataN]

foreign key: Dirige[Progetto] \subseteq Progetto[Nome]

In alternativa:

Dirige(Cognome, DataN, Progetto, Da)

chiave: Progetto foreign key: ...

Traduzione di relazione non accorpata derivante da ISA : esempio

Una relazione derivante da ISA che non è stata accorpata (perché l'identificatore principale non è quello esterno) si traduce nel modo mostrato in questo esempio.



ISA-S-P(Matricola, Cognome, DataN)

foreign key: $StudentePersona[Cognome, DataN] \subseteq Persona[Cognome, DataN]$

chiave: Cognome, DataN

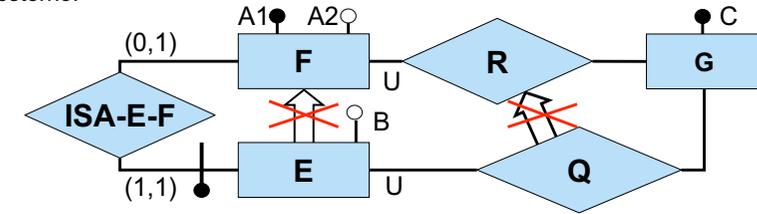
Studente(Matricola, MediaVoti)

foreign key: $Studente[Matricola] \subseteq StudentePersona[Matricola]$

Persona(Cognome, DataN, Indirizzo)

Traduzione di vincoli derivanti da ISA tra relazioni

Si ricordi che la ristrutturazione di una ISA tra relazioni ha prodotto un vincolo esterno.



Vincolo esterno: per ogni istanza (e,g) di Q, sia f l'istanza di F tale che (e,f) è un'istanza di ISA-E-F (si noti che f esiste sempre ed è unica). Allora (f,g) deve essere un'istanza di R.

Traduzione: il vincolo esterno diventa un vincolo di **foreign key**

$E(A1, B)$ foreign key: $E[A1] \subseteq F[A1]$

$F(A1, A2)$ $G(C)$

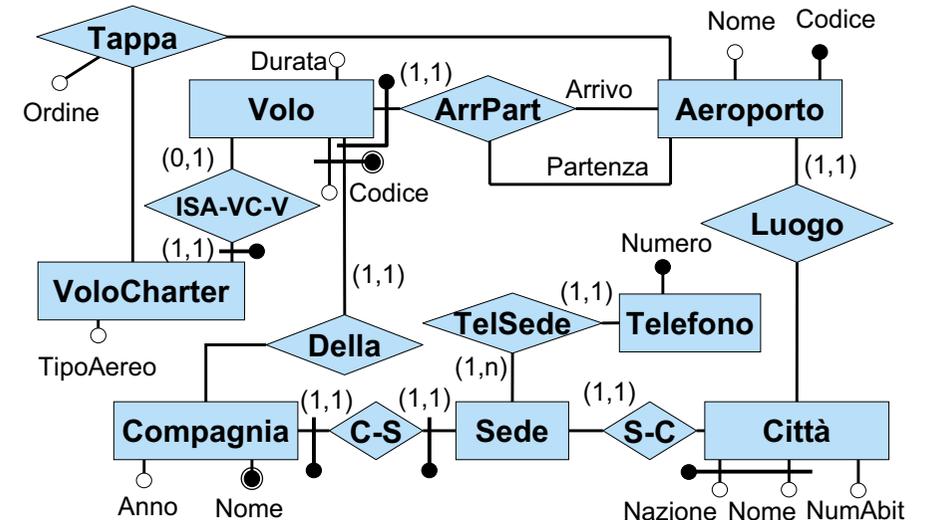
$R(A1, C)$ foreign key: $R[A1] \subseteq F[A1]$, $R[C] \subseteq G[C]$

$Q(A1, C)$ foreign key: $Q[A1] \subseteq E[A1]$, $Q[A1, C] \subseteq R[A1, C]$

Riassunto sulla traduzione diretta

- Traduzione di ogni entità in una relazione, con i seguenti attributi:
 - gli attributi dell'entità stessa
 - gli attributi delle relazioni che partecipano all'identificazione principale esterna dell'entità, insieme alle chiavi primarie, opportunamente nominate (possibilmente con ruolo) delle entità connesse a tali relazioni (si noti che in questo caso, per l'assenza di cicli sull'identificazione principale esterna, la relazione non può avere altre entità per le quali la relazione è parte di identificatore esterno)
- Traduzione di ogni ER-relazione (non accorpata al punto 1) in relazione, con opportuna chiave primaria, e con attributi:
 - gli identificatori principali delle entità partecipanti (con opportuno nome)
 - gli attributi della ER-relazione
- Traduzione di vincoli
 - not null per gli attributi obbligatori
 - chiavi (primarie e non)
 - foreign key che provengono dall'accorpamento (vedi punto 1), da tipizzazione di relazioni, da ISA di relazioni
 - vincoli di generalizzazione
 - vincoli di cardinalità (parte obbligatoria diventa vincolo di inclusione, parte di funzionalità diventa vincolo di chiave)
 - altri vincoli esterni
- Riformulazione di operazioni e specifiche sul carico applicativo in termini dello schema logico

Esercizio 5: tradurre il seguente schema



Vincolo esterno: vincolo su Ordine in Tappa.

Esercizio 5: soluzione

Volo(Codice, Comp, Durata)
 foreign key: Volo[Comp] ⊆ Compagnia[Nome]
 foreign key: Volo[Codice, Comp] ⊆ ArrPart[Codice, Comp]

ArrPart(Codice, Comp, Arrivo, Partenza)
 foreign key: ArrPart[Arrivo] ⊆ Aeroporto[Codice]
 foreign key: ArrPart[Partenza] ⊆ Aeroporto[Codice]
 foreign key: ArrPart[Codice, Comp] ⊆ Volo[Codice, Comp]
 chiave: Comp, Arrivo, Partenza

VoloCharter(Codice, Comp, TipoAereo)
 foreign key: VoloCharter[Codice, Comp] ⊆ Volo[Codice, Comp]

Aeroporto(Codice, Nome)
 foreign key: Aeroporto[Codice] ⊆ LuogoAeroporto[Aeroporto]

LuogoAeroporto(Aeroporto, NomeCittà, NazCittà)
 foreign key: LuogoAeroporto[Aeroporto] ⊆ Aeroporto[Codice]
 foreign key: LuogoAeroporto[NomeCittà, NazCittà] ⊆ Città[Nome, Nazione]

Città(Nome, Nazione, NumAbitanti)

Compagnia(Nome, AnnoFond)
 foreign key: Compagnia[Nome] ⊆ Sede[Comp]

Sede(Comp)
 foreign key: Sede[Comp] ⊆ Compagnia[Nome]
 foreign key: Sede[Comp] ⊆ CittàSede[Comp]

CittàSede(Comp, NomeCittà, NazCittà)
 foreign key: CittàSede[Comp] ⊆ Sede[Comp]
 foreign key: SedeCompagnia[NomeCittà, NazCittà] ⊆ Città[Nome, Nazione]
 inclusione: SedeCompagnia[Comp] ⊆ TelefonoComp[Numero]

TelefonoComp(Numero, Comp)
 foreign key: Telefono[Comp] ⊆ SedeCompagnia[Comp]

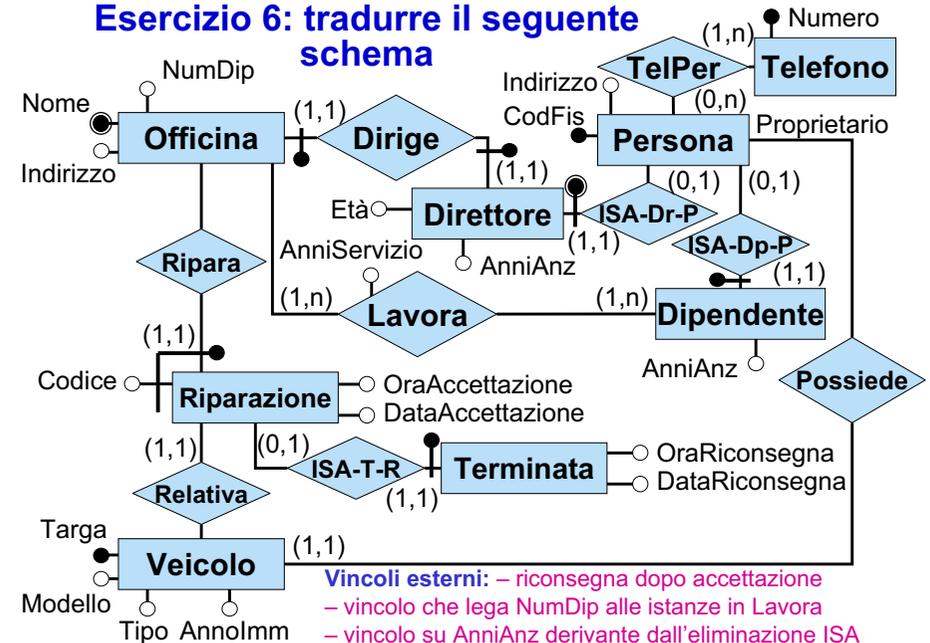
Telefono(Numero)

Tappa(CodVoloCharter, Comp, Aeroporto, Ordine)
 foreign key: Tappa[CodVoloCharter, Comp] ⊆ VoloCharter[Codice, Comp]
 foreign key: Tappa[Aeroporto] ⊆ Aeroporto[Codice]

Vincolo esterno:
 vincolo su
 Ordine in Tappa

Nota: viene
 dall'identificatore
 esterno di Volo

Esercizio 6: tradurre il seguente schema



Esercizio 6: soluzione (parte 1)

Officina(Nome, NumDip, Indirizzo)
 foreign key: Officina[NumDip] ⊆ Dirige[Officina]
 inclusione: Officina[Nome] ⊆ Lavora[Officina]

Persona(CodFis, Indirizzo)

Direttore(CodFis, Eta, AnniAnz)
 foreign key: Direttore[CodFis] ⊆ Persona[CodFis]
 foreign key: Direttore[CodFis] ⊆ Dirige[Direttore]

Dipendente(CodFis, AnniAnz)
 foreign key: Dipendente[CodFis] ⊆ Persona[CodFis]
 inclusione: Dipendente[CodFis] ⊆ Lavora[Dipendente]

Dirige(Officina, Direttore)
 foreign key: Dirige[Officina] ⊆ Officina[Nome]
 foreign key: Dirige[Direttore] ⊆ Direttore[CodFis]
 chiave: Direttore

Lavora(Officina, Dipendente, AnniServizio)
 foreign key: Lavora[Officina] ⊆ Officina[Nome]
 foreign key: Lavora[Dipendente] ⊆ Dipendente[CodFis]

TelPer(CodFis, Telefono)
 foreign key: TelPer[CodFis] ⊆ Persona[CodFis]
 foreign key: TelPer[Telefono] ⊆ Telefono[Numero]

Esercizio 6: soluzione (parte 2)

TelPer(CodFis, Telefono)
 foreign key: TelPer[CodFis] ⊆ Persona[CodFis]
 foreign key: TelPer[Telefono] ⊆ Telefono[Numero]

Telefono(Numero)
 inclusione: Telefono[Numero] ⊆ TelPer[Telefono]

Veicolo(Targa, Modello, Tipo, AnnoImm)
 foreign key: Veicolo[Targa] ⊆ Possiede[Targa]

Possiede(Targa, Proprietario)
 foreign key: Possiede[Targa] ⊆ Veicolo[Targa]
 foreign key: Possiede[Proprietario] ⊆ Persona[CodFis]

Riparazione(Codice, Officina, OraAcc, DataAcc)
 foreign key: Riparazione[Officina] ⊆ Officina[Nome]

Terminata(Codice, Officina, OraRic, DataRic)
 foreign key: Terminata[Codice, Officina] ⊆ Riparazione[Codice, Officina]

Relativa(Codice, Officina, Veicolo)
 foreign key: Relativa[Codice, Officina] ⊆ Riparazione[Codice, Officina]
 foreign key: Relativa[Veicolo] ⊆ Veicolo[Targa]

Vincoli esterni:

- riconsegna dopo accettazione
- vincolo che lega Officina[NumDip] alle istanze in Lavora
- vincolo su Direttore[AnniAnz] e Dipendente[AnniAnz] derivante dall'eliminazione ISA