

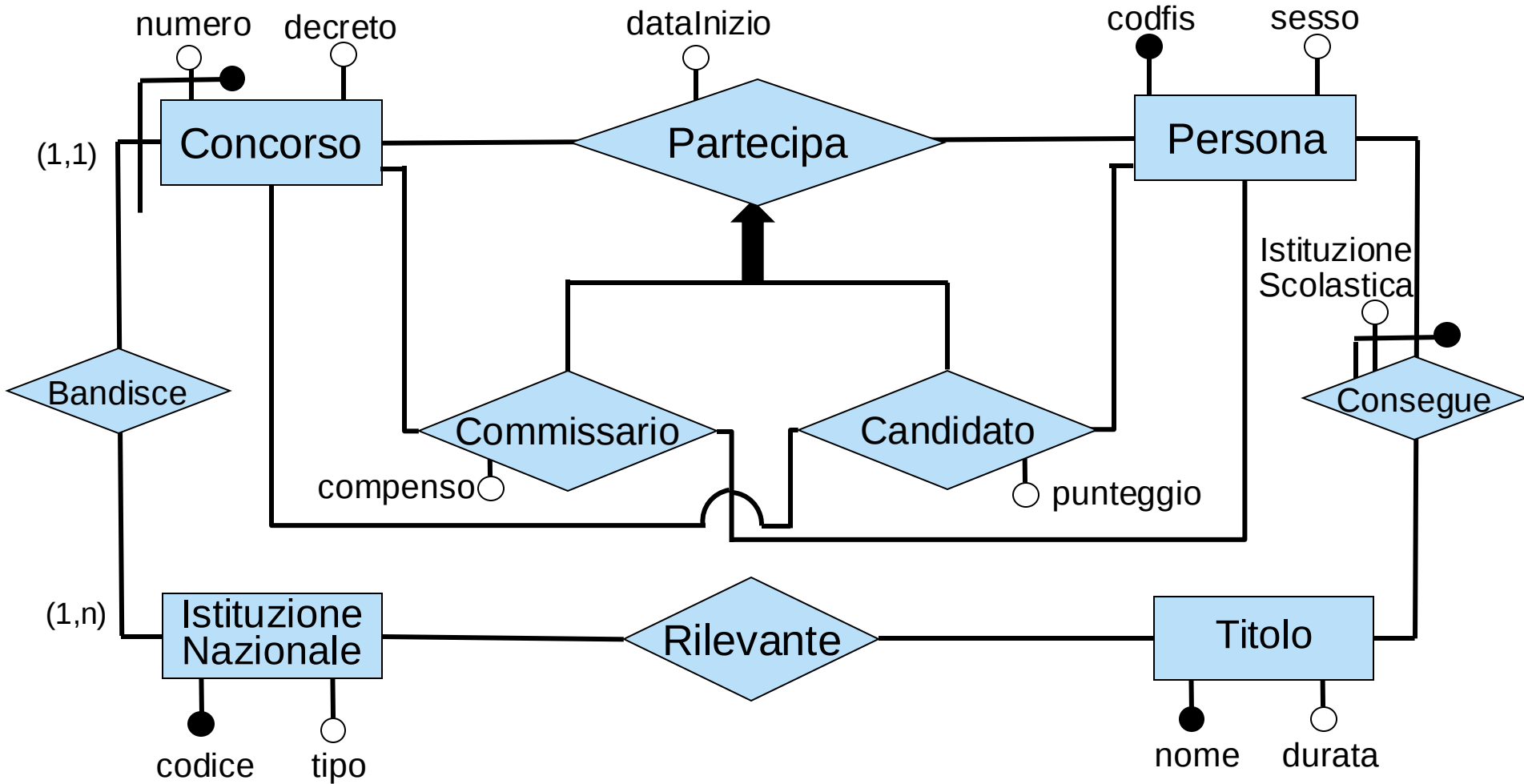
Basi di dati

**Soluzioni dei problemi proposti
nell'appello del 13-02-2025
Compito A**

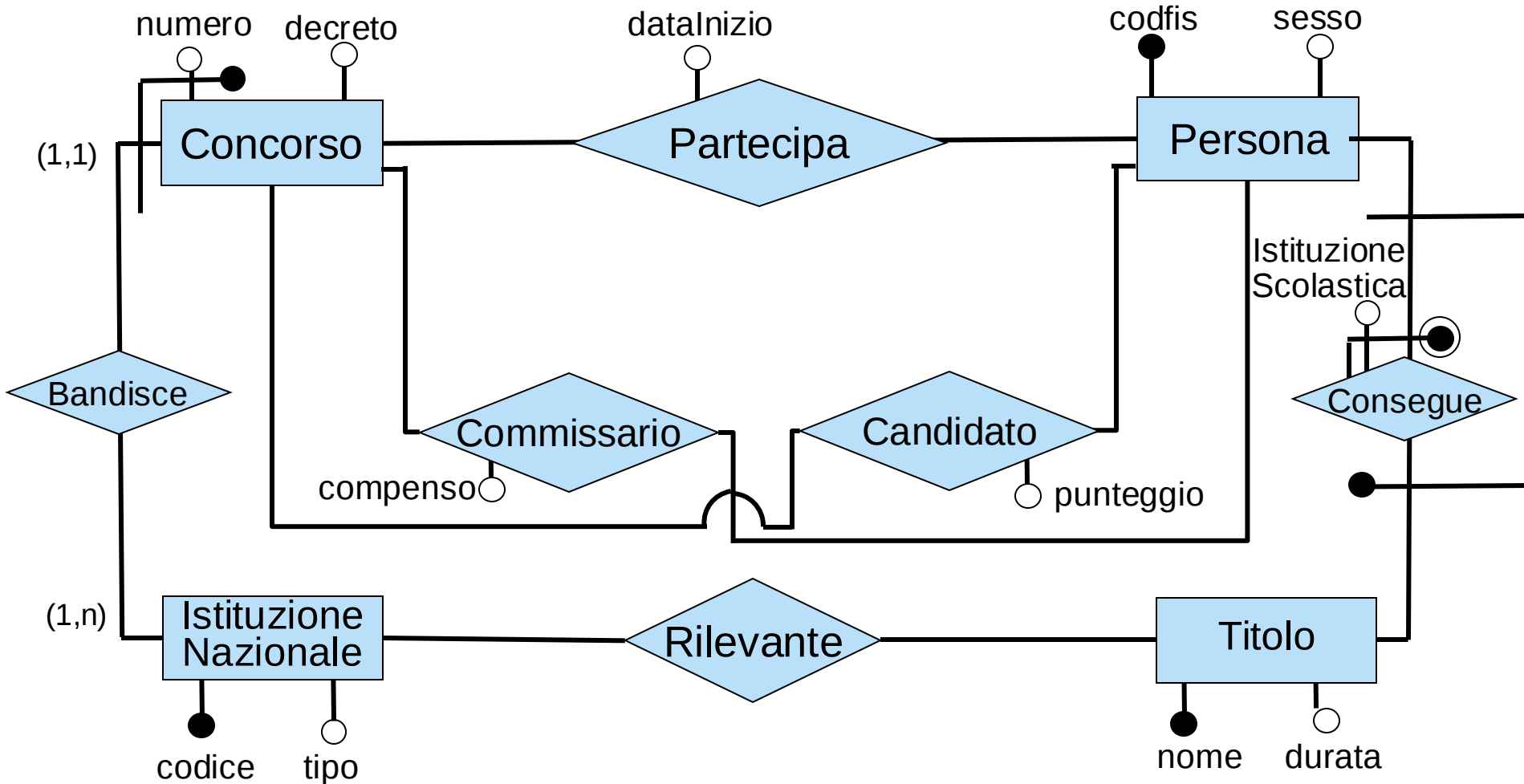
Maurizio Lenzerini

Anno Accademico 2024/25

Problema 1 – Schema ER



Problema 1 – Schema ER ristrutturato



Vincolo esterno: Commissario e Candidato formano una partizione (disgiunta) di partecipa

Problema 2 – Traduzione diretta

Schema logico prodotto dalla traduzione diretta:

Concorso(numero,istNazionale,decreto)

foreign key: Concorso[istNazionale] $\subseteq$ IstituzioneNazionale[codice]

IstituzioneNazionale(codice,tipo)

inclusione: IstituzioneNazionale[codice] $\subseteq$ Concorso[istNazionale]

Partecipa(numero,istNazionale,persona,dataInizio)

foreign key: Partecipa[numero,istNazionale] $\subseteq$ Concorso[numero,istNazionale]

foreign key: Partecipa[persona] $\subseteq$ Persona[codfis]

vincolo esterno: Partecipa[numero,istNazionale,persona] $\subseteq$
Commissario[numero,istNazionale,persona] $\cup$
Candidato[numero,istNazionale,persona]

Persona(codfis,sesso)

Commissario(numero,istNazionale,persona,compenso)

foreign key: Commissario[numero,istNazionale,persona] $\subseteq$
Partecipa[numero,istNazionale,persona]

vincolo esterno: Commissario[numero,istNazionale,persona] $\cap$
Candidato[numero,istNazionale,persona] = $\emptyset$

Candidato(numero,istNazionale,persona,punteggio)

foreign key: Candidato[numero,istNazionale,persona] $\subseteq$
Partecipa[numero,istNazionale,persona]

TITOLO(nome,durata)

Consegue(persona,istScolastica,titolo)

foreign key: Consegue[persona] $\subseteq$ Persona[codfis]

foreign key: Consegue[titolo] $\subseteq$ Titolo[nome]

chiave: persona, titolo

Rilevante(titolo,istNazionale)

foreign key: Rilevante[titolo] $\subseteq$ Titolo[nome]

foreign key: Rilevante[titolo] $\subseteq$ IstituzioneNazionale[codice]

Problema 2 – Ristrutturazione dello schema logico

L'indicazione di progetto induce un accorpamento tra la tabella Concorso e la tabella IstNazionale debolmente accoppiate: di fatto è un accorpamento per denormalizzazione a fronte del quale Concorso e IstituzioneNazionale diventano:

Concorso(numero,istNazionale,decreto,tipo)

foreign key: Concorso[istNazionale,tipo] $\subseteq$ IstituzioneNazionale[codice,tipo]

IstituzioneNazionale(codice,tipo)

inclusione: IstituzioneNazionale[codice,tipo] $\subseteq$ Concorso[istNazionale,tipo]

Ma a questo punto possiamo anche eliminare la relazione inutile IstituzioneNazionale, a patto di specificare in Concorso il vincolo di dipendenza funzionale IstNazionale $\rightarrow$ tipo:

Concorso(numero,istNazionale,decreto,tipo)

foreign key: Concorso[istNazionale] $\subseteq$ IstituzioneNazionale[codice]

vincolo esterno: vale la dipendenza funzionale IstNazionale $\rightarrow$ tipo

Partecipa(numero,istNazionale,persona,dataInizio)

foreign key: Partecipa[numero,istNazionale] $\subseteq$ Concorso[numero,istNazionale]

foreign key: Partecipa[persona] $\subseteq$ Persona[codfis]

vincolo esterno: Partecipa[numero,IstNazionale,persona] $\subseteq$
Commissario[numero,IstNazionale,persona] $\cup$ Candidato[numero,IstNazionale,persona]

Persona(codfis,sesso)

Commissario(numero,istNazionale,persona,compenso)

foreign key: Commissario[numero,IstNazionale,persona] $\subseteq$ Partecipa[numero,IstNazionale,persona]

vincolo esterno: Commissario[numero,IstNazionale,persona] $\cap$ Candidato[numero,IstNazionale,persona] = $\emptyset$

Candidato(numero,istNazionale,persona,punteggio)

foreign key: Candidato[numero,IstNazionale,persona] $\subseteq$ Partecipa[numero,IstNazionale,persona]

Titolo(nome,durata)

Consegue(persona,istScolastica,titolo)

foreign key: Consegue[persona] $\subseteq$ Persona[codfis]

foreign key: Consegue[titolo] $\subseteq$ Titolo[nome]

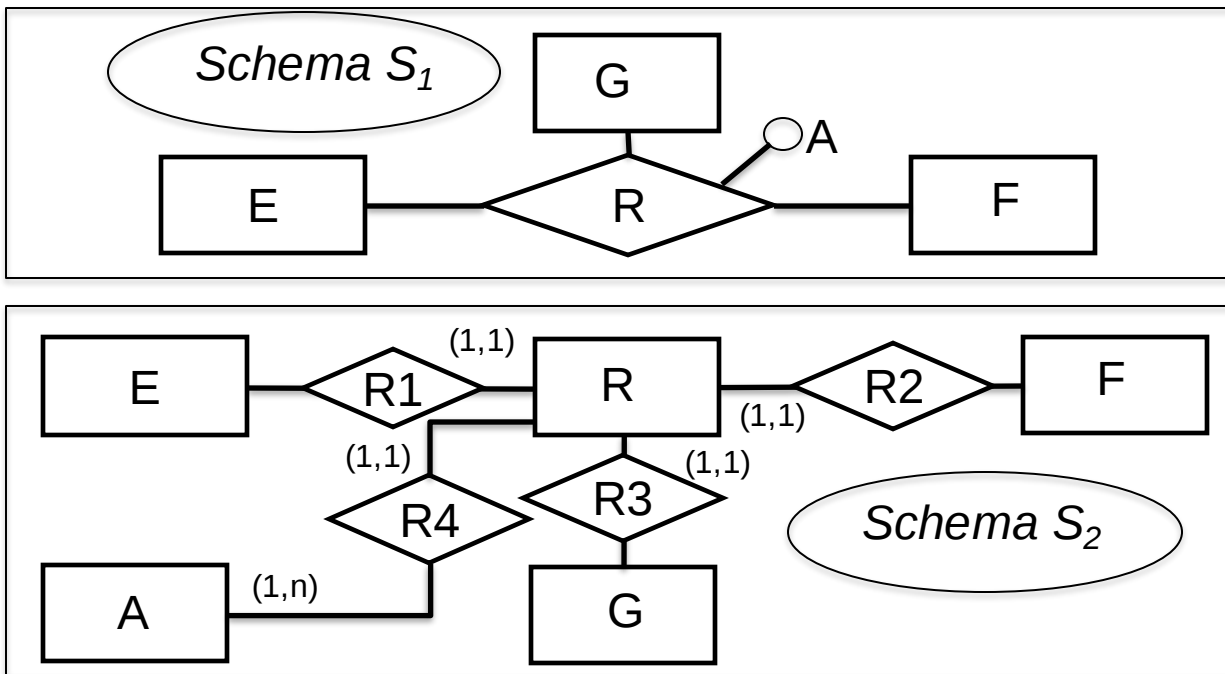
chiave: persona, titolo

Rilevante(titolo,istNazionale)

foreign key: Rilevante[titolo] $\subseteq$ Titolo[nome]

foreign key: Rilevante[titolo] $\subseteq$ IstNazionale[codice]

Problema 3: testo e soluzione



Si vuole sapere se gli schemi S_1 e S_2 mostrati qui a sinistra sono equivalenti, cioè se esiste una corrispondenza tra le istanze dei due schemi che specifichi come ogni istanza dell'uno si possa rappresentare correttamente mediante una istanza dell'altro. Se la risposta è positiva, motivarla in dettaglio. Se la risposta è negativa, mostrare una possibile istanza di uno dei due schemi che non potrà corrispondere ad alcuna istanza dell'altra, spiegando in dettaglio il motivo.

La risposta è chiaramente negativa, perché nello schema S_1 nessuna terna di istanze di E , G ed F può comparire più volte nel legame rappresentato da R , mentre nello schema S_2 nulla vieta che questo accada. Consideriamo una istanza I di S_2 tale che:

$Istanze(I, E) = \{e\}$, $Istanze(I, F) = \{f\}$, $Istanze(I, G) = \{g\}$, $Istanze(I, R) = \{r_1, r_2\}$, $Istanze(I, A) = \{a\}$,
 $Istanze(I, R1) = \{ \langle E:e, R:r_1 \rangle, \langle E:e, R:r_2 \rangle \}$, $Istanze(I, R2) = \{ \langle F:f, R:r_1 \rangle, \langle F:f, R:r_2 \rangle \}$,
 $Istanze(I, R3) = \{ \langle G:g, R:r_1 \rangle, \langle G:g, R:r_2 \rangle \}$, $Istanze(I, R4) = \{ \langle A:a, R:r_1 \rangle, \langle A:a, R:r_2 \rangle \}$

Notiamo che nella istanza I di S_2 , r_1 ed r_2 sono legate alla stessa terna di E, F e G mediante $R1$, $R2$ e $R3$. Quindi, qualunque funzione dalle istanze di S_2 alle istanze di S_1 che si volesse usare per dimostrare l'equivalenza di S_2 ed S_1 , quando applicata ad I , dovrebbe scegliere tra r_1 ed r_2 una sola istanza di R ed usarla per definire una tupla istanza di R in S_1 . Ma questo significherebbe perdere una informazione presente in S_2 passando a S_1 . Non è difficile concludere che l'equivalenza varrebbe invece se in S_2 definissimo un identificatore per R formato dal ruolo R nelle relazioni $R1$, $R2$ e $R3$.

Problema 4 – testo e soluzione

Si consideri la relazione `Laboratorio(codice, categoria)`, che memorizza i dati dei laboratori di analisi di uno studio medico, e la relazione `Analisi(paziente, codice, data, tipo, risultato)`, che registra le analisi cliniche effettuate dai pazienti nei laboratori nelle varie date, ciascuna con tipo di analisi e risultato.

- 4.1 Scrivere una query in algebra relazionale che per ogni tipo di analisi calcoli il codice dei laboratori che non hanno mai effettuato analisi di quel tipo.
- 4.2 Scrivere una query in SQL che per ogni categoria di laboratorio calcoli qual'è il tipo di analisi (o quali sono i tipi di analisi, se sono più d'uno) che è stato maggiormente effettuato nei laboratori di quella categoria.

(4.1)

$(\text{PROJ}_{\text{tipo}}(\text{Analisi}) \text{ JOIN } \text{PROJ}_{\text{codice}}(\text{Analisi})) - \text{PROJ}_{\text{tipo, codice}}(\text{Analisi})$

(4.2)

with vista as

```
(select lab.categoria cat, a.tipo tip, count(*) quanti
from Laboratorio lab join Analisi a on lab.codice = a.codice
group by lab.categoria, a.tipo)
```

```
select v.cat, v.tip
```

```
from vista v
```

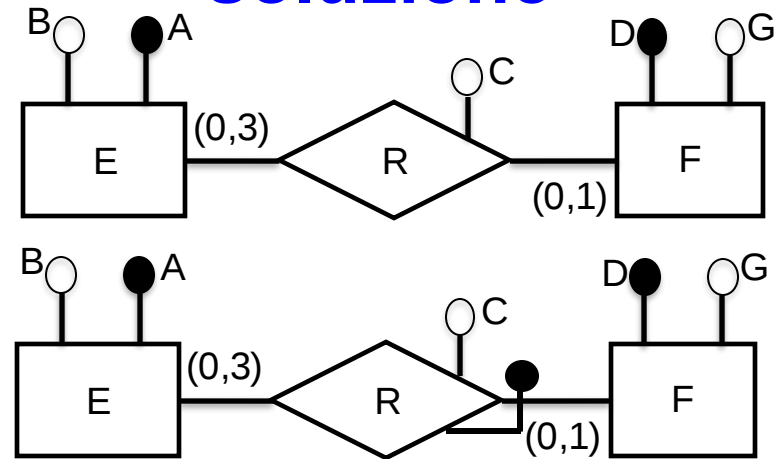
```
where v.quanti = (select max(v2.quanti)
```

```
from vista v2
```

```
where v2.cat=v1.cat and v2.tip=v1.tip)
```

Si consideri lo schema concettuale S mostrato qui a destra, per il quale sappiamo che, dopo il caricamento iniziale dei dati, non verranno eseguite operazioni di insert sulla relazione R. Si chiede di eseguire i vari passi della progettazione logica e produrre il corrispondente schema logico relazionale espresso nella nostra abituale rappresentazione. Si chiede poi di mostrare anche la definizione della tabella R in SQL, corredata da eventuali trigger

Problema 5: testo e soluzione



SOLUZIONE:

Lo schema concettuale ristrutturato è mostrato qui sopra e rende esplicito che R è identificata dal ruolo F.

La definizione dello schema logico con la nostra usuale notazione è:

$F(\underline{D}, G)$

$E(\underline{A}, B)$

$R(\underline{FK}, EK, C)$

foreign key $R[FK] \subseteq F[F]$

foreign key $R[EK] \subseteq E[A]$

vincolo esterno nessun valore appare più di 3 volte in $R[EK]$

Una definizione corretta di R in SQL è (assumendo il tipo int per tutti gli attributi):

```
create table R (
  FK int not null references F primary key,
  EK int not null references E,
  C int not null);
```

```
create function aggiornaR() returns trigger as
$$
begin
  if ((select count(*) from where EK = new.EK) > 3)
  then rollback; -- aggiornamento bloccato
  else return new; -- aggiornamento da eseguire
  end if;
end; $$ language plpgsql;
```

```
create trigger triggerForR after update on R for each row
execute procedure aggiornaR();
```