

Basi di dati

Esercitazione 13 – Esercitazione su esercizi d'esame

Anno Accademico 2023/2024

Maurizio Lenzerini

Esercizio 1 (sul modello relazionale)

Si consideri la relazione con schema $R(A,B,C,D)$ per la quale sappiamo che per ogni sua istanza valgono le seguenti condizioni:

- le tuple di R possono avere il valore NULL nell'attributo C
- prese due tuple t_1 e t_2 qualunque di R , valgono entrambe le seguenti asserzioni:
 - i. almeno una delle condizioni tra $t_1[A] = t_2[A]$ e $t_1[B] = t_2[B]$ non è verificata;
 - ii. almeno una delle condizioni tra $t_1[C] = t_2[C]$ e $t_1[B] = t_2[B]$ non è verificata.

Rispondere alle seguenti domande:

1. È vero che in ogni istanza di R , l'attributo B soddisfa la condizione di chiave?
2. Mostrare tutti gli insiemi K di attributi di R che soddisfano la condizione di superchiave in ogni istanza di R .
3. Quale vincolo di chiave primaria definiresti per R ?

Tutte le risposte devono essere motivate in dettaglio.

1) NO

	A	B	C	D
τ_1	a ₁	b	c ₁	d ₁
τ_2	a ₂	b	c ₂	d ₂

✓

X C PUÒ ESSERE NULL

2)

A	B	C	D	AB	AC	AD	BC	BD	CD	ABC	BCD	ACD	ABCD
N	N	N	N	S	N	N	S	N	N	S	S	N	N
N	N	N	N	S	N	N	S	N	N	N	N	N	N

ABD
S
N

3) AB CHIAVE PRIMARIA

Esercizio 1 (sul modello relazionale) - soluzione

1. È vero che in ogni istanza di R, l'attributo B soddisfa la condizione di chiave?

No, basta considerare che una possibile istanza di R che soddisfa tutte le condizioni date è quella che contiene le due tuple $t1 = \langle A:a1, B:b, C:c1, D:d \rangle$ e $t2 = \langle A:a2, B:b, C:c2, D:d \rangle$ e queste coincidono nell'attributo B, che perciò non soddisfa la condizione di superchiave e quindi nemmeno di chiave.

2. Mostrare tutti gli insiemi K di attributi di R che soddisfano la condizione di superchiave in ogni istanza di R.

Per mostrare tutte le superchiavi della relazione è sufficiente osservare che in ogni istanza di R

- l'asserzione i. implica che non possono esistere due tuple in R con gli stessi valori per A e B,
- l'asserzione ii. implica che non possono esistere due tuple in R con gli stessi valori per C e B.

Ne segue che $\{A,B\}$ e $\{C,B\}$ sono entrambe superchiavi di R. Osserviamo che sia per A sia per C valgono considerazioni analoghe a quelle che abbiamo svolto per B al punto 1. Ne segue che sia $\{A,B\}$ e sia $\{C,B\}$ sono chiaramente minimali e quindi soddisfano la condizione di chiave in R. A questo punto è facile concludere che le superchiavi di R, essendo tutti i soprainsiemi di $\{A,B\}$ e $\{C,B\}$, sono: $\{A,B\}$, $\{A,B,C\}$, $\{A,B,D\}$, $\{A,B,C,D\}$, $\{C,B\}$, $\{C,B,D\}$

3. Quale vincolo di chiave primaria definiresti per R?

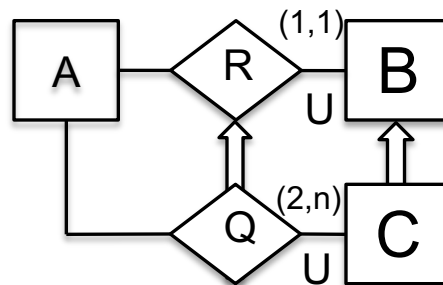
Siccome C può avere valori nulli, il vincolo da definire è quello che stabilisce che la chiave primaria è $\{A,B\}$.

Esercizio 2 (sul modello ER)

Facendo riferimento allo schema concettuale S riportato qui sotto,

1. dire se esiste una istanza I dello schema S in cui la relazione R ha almeno una istanza.
2. dire se esiste una istanza I dello schema S in cui la relazione Q ha almeno una istanza.

In entrambi i casi, se la risposta è positiva, descrivere una tale istanza I, altrimenti spiegare perché una tale istanza non esiste.

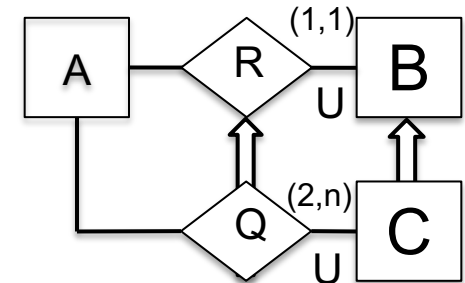


Esercizio 2 (sul modello ER) - soluzione

Facendo riferimento allo schema concettuale S riportato qui sotto,

1. dire se esiste una istanza I dello schema S in cui la relazione R ha almeno una istanza.
2. dire se esiste una istanza I dello schema S in cui la relazione Q ha almeno una istanza.

In entrambi i casi, se la risposta è positiva, descrivere una tale istanza I, altrimenti spiegare perché una tale istanza non esiste.



1. Risposta positiva. Mostriamo l'istanza I1 di S in cui R ha una istanza:

$Istanze(I1, A) = \{ a \}$, $Istanze(I1, B) = \{ b \}$, $Istanze(I1, R) = \{ \langle A:a, U:b \rangle \}$

2. Risposta negativa. Ci sono due modi per motivare la risposta negativa. Il primo modo è il seguente.

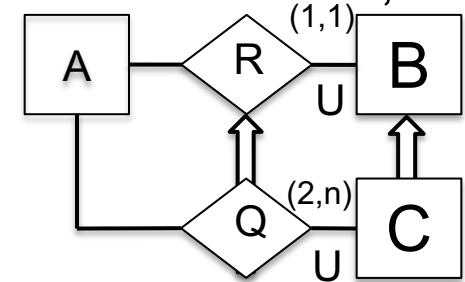
Sappiamo che un vincolo di cardinalità massima su un ruolo U di una relazione R si propaga anche nello stesso ruolo di qualunque relazione che sia in ISA con R. Poiché nello schema in questione nel ruolo U di R abbiamo un vincolo di cardinalità massima pari ad 1, questo si propaga sullo stesso ruolo della relazione Q, che è in ISA con R. Ne segue che sul ruolo U di Q abbiamo un vincolo di cardinalità minima pari a 2 ed un vincolo di cardinalità massima pari ad 1 e ciò significa che in ogni istanza dello schema S, ogni istanza di C dovrebbe partecipare contemporaneamente ad almeno 2 istanze di Q ed al massimo 1 istanza di Q. Poiché questo è impossibile, concludiamo che non possono esistere istanze dello schema S in cui esiste una istanza di C e quindi non esiste alcuna istanza I di S in cui esiste una istanza di Q.

Esercizio 2 (sul modello ER) – altra soluzione

Facendo riferimento allo schema concettuale S riportato qui sotto,

1. dire se esiste una istanza I dello schema S in cui la relazione R ha almeno una istanza.
2. dire se esiste una istanza I dello schema S in cui la relazione Q ha almeno una istanza.

In entrambi i casi, se la risposta è positiva, descrivere una tale istanza I, altrimenti spiegare perché una tale istanza non esiste.



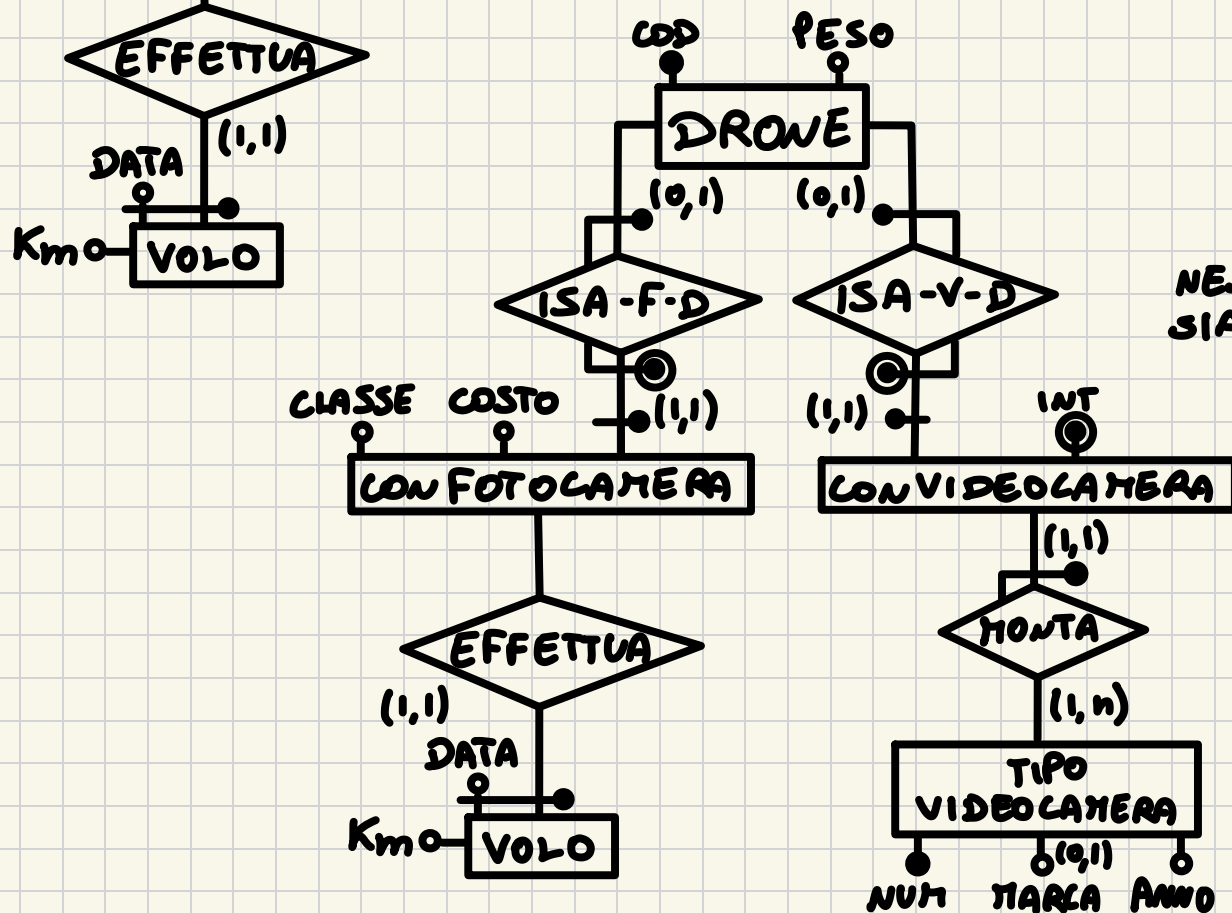
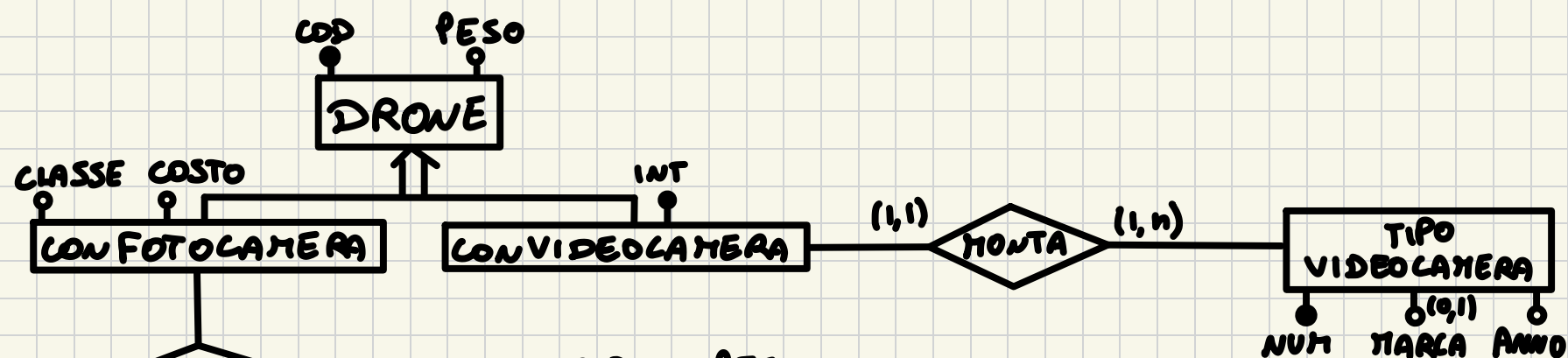
2. Risposta negativa. Il secondo modo per motivare la risposta negativa è il seguente.

In qualunque istanza dello schema S in cui ci fosse una istanza di Q, ci sarebbe anche una istanza di C, che dovrebbe partecipare almeno due volte alla relazione Q (vedi vincolo di cardinalità minima su Q nel ruolo U) e quindi almeno due volte anche alla relazione R (visto che Q è in ISA con R). Però, l'istanza di C in questione è anche istanza di B (visto che C è in ISA con B) e quindi questa istanza non può partecipare più di una volta ad R (visto il vincolo di cardinalità massima su R nel ruolo U). Possiamo concludere che una qualunque istanza di C dovrebbe contemporaneamente partecipare almeno due volte ad R (per il vincolo di cardinalità minima su Q nel ruolo U) e al massimo una volta ad R (per il vincolo di cardinalità massima su R nel ruolo U). Siccome questo è contraddittorio, concludiamo che non esiste alcuna istanza I dello schema S in cui esiste una istanza di C e quindi non esiste alcuna istanza I dello schema S in cui esiste una istanza di Q.

Esercizio 3: progettazione concettuale

Definire lo schema concettuale relativo ai seguenti requisiti per un sistema informativo sui droni.

Di ogni drone interessano il codice identificativo ed il peso. Alcuni droni montano a bordo una fotocamera (e non una videocamera), altri montano invece una videocamera (e non una fotocamera) e altri ancora non montano alcun dispositivo. Dei droni che montano una fotocamera interessano la classe, il costo ed i vari voli effettuati, ciascuno con il numero di chilometri percorsi e la data in cui è stato effettuato (ogni drone con fotocamera effettua al massimo un volo al giorno). Dei droni che montano una videocamera interessa il codice internazionale (identificativo) ed il tipo di videocamera che ha a bordo. Di ogni tipo di videocamera montata a bordo di un drone interessa il numero di serie (identificativo), la marca (se nota) e l'anno di produzione.



VINCOLO DI GEN:

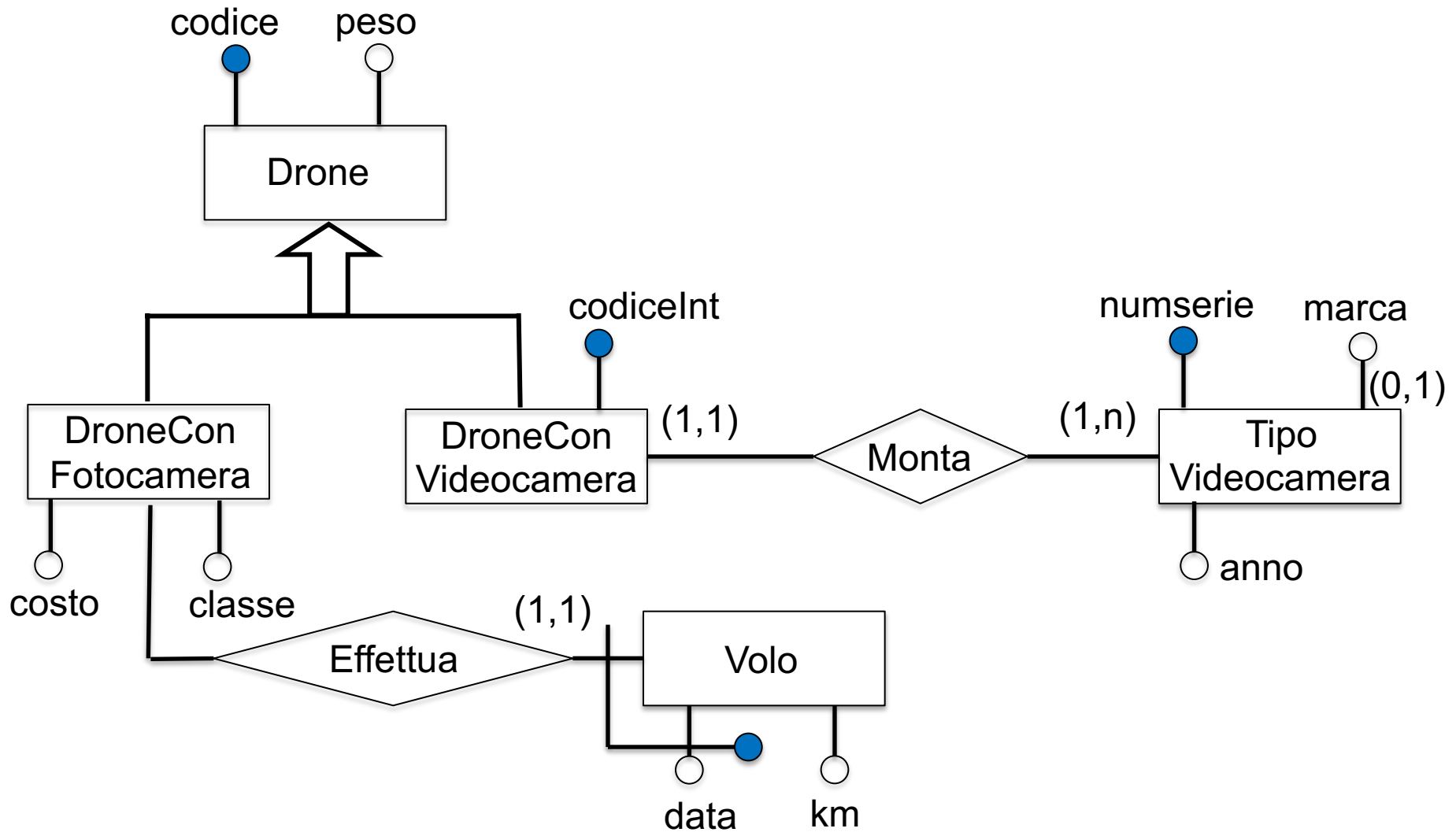
NESSUNA ISTANZA PARTECIPA
SIA AD ISA-F-D SIA AD ISA-V-D

Esercizio 3: progettazione logica

Indicazioni sul carico applicativo:

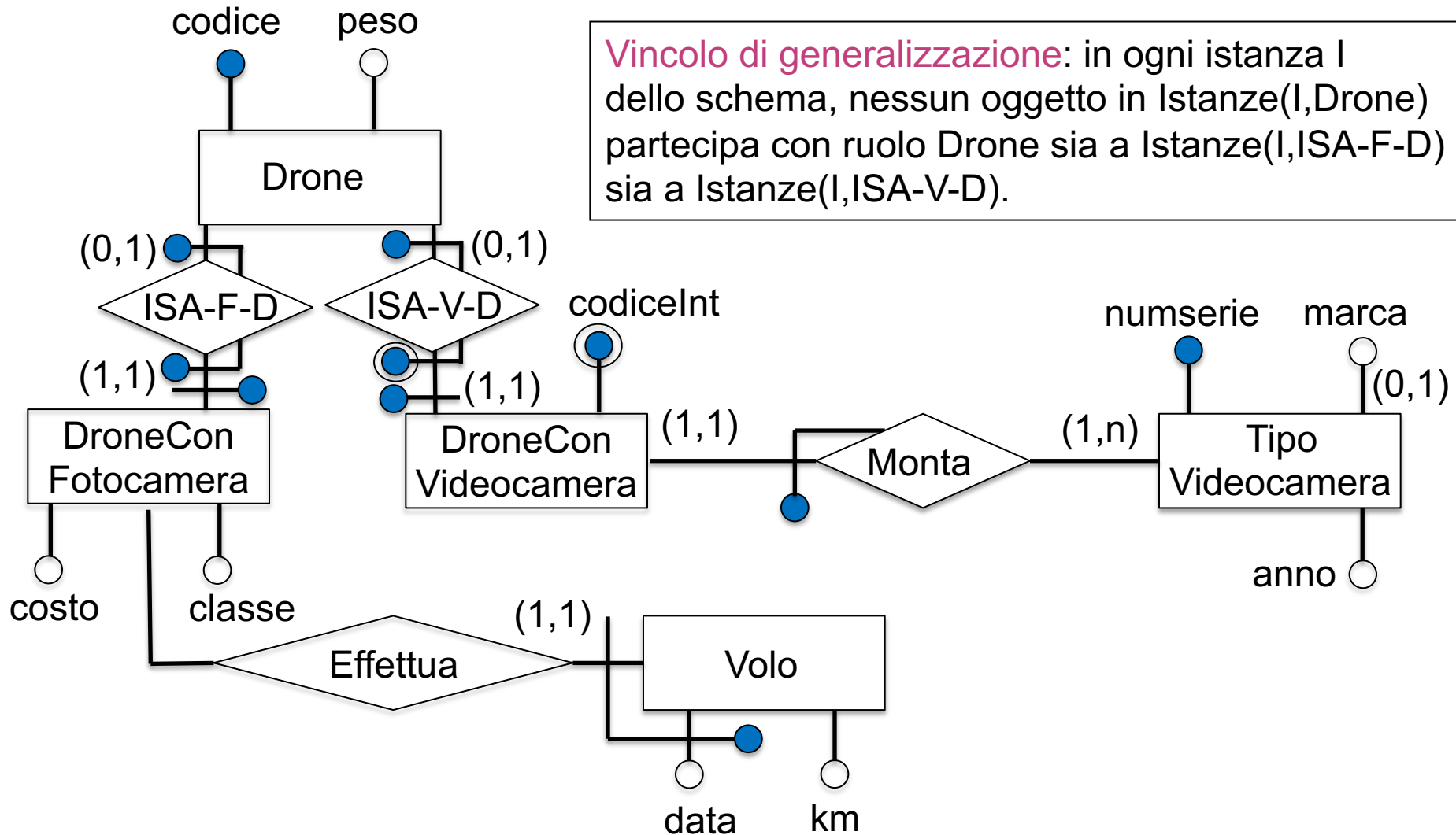
1. Ai droni con videocamera si accede prevalentemente mediante il codice internazionale.
2. Quando si accede ad un drone con la videocamera si vuole sempre conoscere il tipo di videocamera che essa monta.
3. Quando si accede ad un drone con videocamera si vuole sempre conoscere l'anno di produzione del tipo di videocamera che esso monta e, se nota, la marca di tale tipo di videocamera (anno e marca non cambiano mai).
4. Quando si accede ad un drone si vuole sempre sapere se è un drone con la fotocamera ed in tale caso si vuole sempre sapere sia il costo sia il tipo; inoltre, quando si accede ad un drone con fotocamera si vuole sempre saperne il peso.

Soluzione esercizio 3: Schema concettuale



Soluzione esercizio 3: schema concettuale ristrutturato

La ristrutturazione tiene conto dell'indicazione di progetto N.1, ossia del fatto che ai droni con videocamera si accede prevalentemente mediante il codice internazionale.



DRONE (CODICE, PESO)

DRONE CON FOTOCAMERA (DRONE, COSTO, CLASSE)

FOREIGNKEY: DLF [DRONE] \subseteq DRONE [CODICE]

DRONE CON VIDEOCAMERA (INT)

FOREIGNKEY: DLV [INT] \subseteq ISA-V-D [DLV]

FOREIGNKEY: DLV [INT] \subseteq MONTA [DLV]

ISA-V-D (DRONE CON VIDEOCAMERA, DRONE)

FOREIGNKEY: ISA-V-D [DLV] \subseteq DLV [INT]

FOREIGNKEY: ISA-V-D [DRONE] \subseteq DRONE [CODICE]

CHIAVE: DRONE

MONTA (DLV, TIPOVC)

FOREIGNKEY: MONTA [DLV] \subseteq DLV [INT]

FOREIGNKEY: MONTA [TIPOVC] \subseteq TIPOVC [NUMSERIE]

TIPO VIDEOCAMERA (NUMSERIE, ANNO, MARCA*)

INCLUSIONE: TIPOVC [NUMSERIE] \subseteq MONTA [TIPOVC]

VOLO (DATA, CODICE, Km)

FOREIGNKEY: VOLO [COD] \subseteq DLF [DRONE]

Soluzione esercizio 3: Schema logico risultante dalla traduzione diretta

Drone(codice, peso)

DroneConFotocamera(codice, costo, classe)

foreign key: DroneConFotocamera[codice] \subseteq Drone[codice]

vincolo di generalizzazione: DroneConFotocamera[codice] \cap ISA-V-D[drone] = \emptyset

DroneConVideocamera(codiceInt)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[codiceInt] \subseteq Monta[droneconvideocamera]

ISA-V-D(droneconvideocamera, drone)

foreign key: ISA-V-D[drone] \subseteq Drone[codice]

foreign key: ISA-V-D[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

chiave: drone

Monta(droneconvideocamera, tipovideocamera)

foreign key: Monta[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

foreign key: Monta[tipovideocamera] \subseteq TipoVideocamera[numserie]

TipoVideocamera(numserie, anno, marca*)

foreign key: TipoVideocamera[numserie] \subseteq Monta[tipovideocamera]

Volo(codice, data, km)

foreign key: Volo[codice] \subseteq DroneConFotocamera[codice]

Soluzione esercizio 3: ristrutturazione schema logico

Analizziamo una per una le operazioni di ristrutturazione a fronte delle indicazioni di progetto:

2. Quando si accede ad un drone con videocamera si vuole sempre conoscere il numero di serie del tipo della videocamera che esso monta a bordo.

- Accorpamento delle due tabelle fortemente accoppiate “Monta” e “DroneConVideocamera”:

DroneConVideocamera(codiceInt,tipovideocamera)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[tipoVideocamera] \subseteq TipoVideocamera[numserie]

3. Quando si accede ad un drone con videocamera si vuole sempre conoscere l’anno di produzione del tipo di videocamera che esso monta e, se nota, la marca di tale tipo di videocamera (anno e marca non cambiano mai).

- Accorpamento delle due tabelle lascamente accoppiate “Videocamera” e “DroneConVideocamera” (appena modificata) senza eliminazione di tabelle:

DroneConVideocamera(codiceInt,tipoVideocamera,anno,marca*)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[tipoVideocamera,anno,marca] \subseteq
TipoVideocamera[numserie,anno,marca]

TipoVideocamera(numserie,anno,marca*)

foreign key: TipoVideocamera[numserie] \subseteq DroneConVideocamera[tipoVideocamera]

Soluzione esercizio 3: ristrutturazione schema logico

Situazione attuale dello schema:

Drone(codice, peso)

DroneConFotocamera(codice, costo, classe)

foreign key: DroneConFotocamera[codice] \subseteq Drone[codice]

vincolo di generalizzazione: DroneConFotocamera[codice] \cap ISA-V-D[drone] = \emptyset

ISA-V-D(droneconvideocamera, drone)

foreign key: ISA-V-D[drone] \subseteq Drone[codice]

foreign key: ISA-V-D[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

chiave: drone

DroneConVideocamera(codiceInt, tipoVideocamera, anno, marca*)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[tipoVideocamera, anno, marca] \subseteq
TipoVideocamera[numserie, anno, marca]

TipoVideocamera(numserie, anno, marca*)

foreign key: TipoVideocamera[numserie] \subseteq
DroneConVideocamera[tipoVideocamera]

Volo(codice, data, km)

foreign key: Volo[codice] \subseteq DroneConFotocamera[codice]

Deriva
dall'accorpamento di
DroneConVideoCamera,
Monta e TipoVideocamera

Nota: la dipendenza funzionale tipoVideocamera \rightarrow anno, marca in DroneConVideocamera è implicata dalla foreign key verso TipoVideocamera e dalla chiave numserie

Soluzione esercizio 3: ristrutturazione schema logico

4. Quando si accede ad un drone si vuole sempre sapere se è un drone con la fotocamera ed in tale caso si vuole sempre sapere sia il costo sia la classe; inoltre, quando si accede ad un drone con fotocamera si vuole sempre saperne il peso.

- Accorpamento delle tabelle debolmente accoppiate Drone e DroneConFotocamera ed eliminazione della tabella DroneConFotocamera

Drone(codice, peso, costo*, classe*)

Deriva dall'accorpamento di Drone e DroneConFotoCamera

foreign key: DroneConFotocamera[codice] \subseteq Drone[codice]

vincolo di tupla: costo is null se e solo se classe is null

vincolo di gen.: $PROJ_{codice}(SEL_{costo \text{ is not null}}(Drone)) \cap ISA-V-D[drone] = \emptyset$

ISA-V-D(droneconvideocamera, drone)

foreign key: ISA-V-D[drone] \subseteq Drone[codice]

foreign key: ISA-V-D[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

chiave: drone

DroneConVideocamera(codiceInt, tipoVideocamera, anno, marca*)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[tipoVideocamera, anno, marca] \subseteq TipoVideocamera[numserie, anno, marca]

TipoVideocamera(numserie, anno, marca*)

foreign key: TipoVideocamera[numserie] \subseteq DroneConVideocamera[tipoVideocamera]

Volo(codice, data, km)

foreign key: Volo[codice] \subseteq $PROJ_{codice}(SEL_{costo \text{ is not null}}(Drone))$

Deriva dall'accorpamento di DroneConVideoCamera, Monta e TipoVideocamera

Soluzione esercizio 3: effetto ristrutturazione schema logico

Drone(codice, peso)

DroneConFotocamera(codice, costo, classe)

foreign key: DroneConFotocamera[codice] \subseteq Drone[codice]

vincolo di generalizzazione: DroneConFotocamera[codice] \cap ISA-V-D[drone] = \emptyset

DroneConVideocamera(codiceInt)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[codiceInt] \subseteq Monta[droneconvideocamera]

Monta(droneconvideocamera, tipovideocamera)

foreign key: Monta[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

foreign key: Monta[tipovideocamera] \subseteq TipoVideocamera[numserie]

ISA-V-D(droneconvideocamera, drone)

foreign key: ISA-V-D[drone] \subseteq Drone[codice]

foreign key: ISA-V-D[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

chiave: drone

TipoVideocamera(numserie, anno, marca*)

foreign key: TipoVideocamera[numserie] \subseteq Monta[tipovideocamera]

Volo(codice, data, km)

foreign key: Volo[codice] \subseteq DroneConFotocamera[codice]

Schema logico
dalla traduzione
diretta

Drone(codice, peso, costo*, classe*)

foreign key: DroneConFotocamera[codice] \subseteq Drone[codice]

vincolo di tupla: costo is null se e solo se classe is null

vincolo di gen.: $\text{PROJ}_{\text{codice}}(\text{SEL}_{\text{costo is not null}}(\text{Drone})) \cap$
ISA-V-D[drone] = \emptyset

ISA-V-D(droneconvideocamera, drone)

foreign key: ISA-V-D[drone] \subseteq Drone[codice]

foreign key: ISA-V-D[droneconvideocamera] \subseteq DroneConVideocamera[codiceInt]

chiave: drone

DroneConVideocamera(codiceInt, tipoVideocamera, marca*)

foreign key: DroneConVideocamera[codiceInt] \subseteq ISA-V-D[droneconvideocamera]

foreign key: DroneConVideocamera[numserie, anno, marca] \subseteq TipoVideocamera[numserie, anno, marca]

TipoVideocamera(numserie, anno, marca*)

foreign key: TipoVideocamera[numserie] \subseteq
DroneConVideocamera[tipoVideocamera]

Volo(codice, data, km)

foreign key: Volo[codice] \subseteq
 $\text{PROJ}_{\text{codice}}(\text{SEL}_{\text{costo is not null}}(\text{Drone}))$

Schema
logico finale