

Voli aerei: specifica

Descrivere lo schema concettuale corrispondente ad un'applicazione riguardante voli aerei, per la quale valgono le seguenti specifiche.

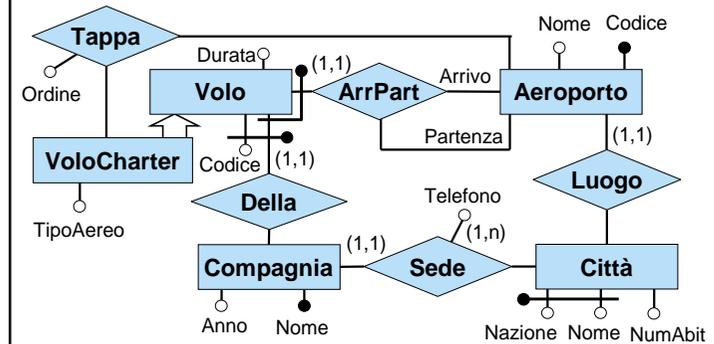
Dei voli interessa: codice, durata in minuti, compagnia aerea, aeroporto di partenza e aeroporto di arrivo. Ogni volo ha un codice unico nella compagnia. Due voli diversi della stessa compagnia devono differire o nell'aeroporto di arrivo o nell'aeroporto di partenza. Degli aeroporti interessa: codice, nome, città (con nome e numero di abitanti) e nazione. Delle compagnie aeree interessa il nome, l'anno di fondazione, e la città in cui ha sede la direzione, con i numeri di telefono della sede. Si suppone che ogni sede abbia almeno un numero di telefono. I voli charter sono particolari voli, che prevedono tappe intermedie in aeroporti. Delle tappe intermedie di un volo charter interessa l'ordine con cui esse si susseguono (ad esempio, il volo 124, che parte da "Milano Linate" e arriva a "Palermo Punta Raisi", prevede prima l'aeroporto di Bologna e poi quello di Napoli come tappe intermedie). Infine, dei voli charter interessa anche il tipo di aereo utilizzato per il volo.

Diego Calvanese

Basi di Dati - A.A. 2002/2003

Esercizio voli - 1

Schema concettuale



Vincoli esterni: 1) per ogni v in VoloCharter, se $(v, a_1), \dots, (v, a_n)$ sono tutte le coppie in Tappa alle quali partecipa v , e se o_1, \dots, o_n sono i valori assegnati a tali coppie dall'attributo Ordine, allora per $i=1, \dots, n$ esiste un o_i tale che $o_i = i$.

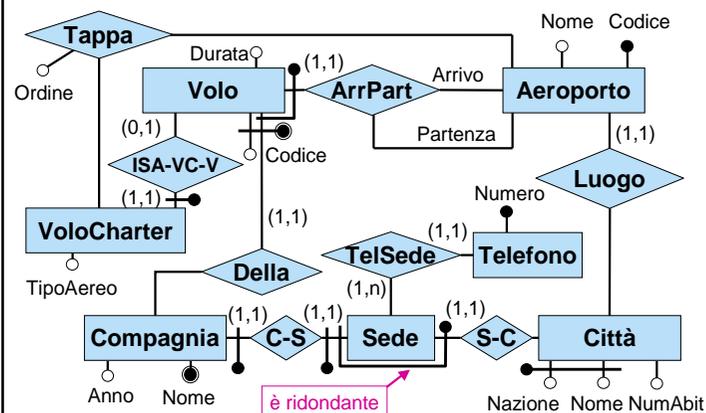
2) Un telefono è di una sola sede.

Diego Calvanese

Basi di Dati - A.A. 2002/2003

Esercizio voli - 2

Schema concettuale ristrutturato



Vincolo esterno: 1) vincolo su ordine in Tappa (2 è diventato interno allo schema)

Diego Calvanese

Basi di Dati - A.A. 2002/2003

Esercizio voli - 3

Schema logico

Volo(Codice, Comp, Durata, Arrivo, Partenza)
 foreign key: Volo[Comp] \subseteq Compagnia[Nome]
 foreign key: Volo[Arrivo] \subseteq Aeroporto[Codice]
 foreign key: Volo[Partenza] \subseteq Aeroporto[Codice]
 chiave: Comp, Arrivo, Partenza

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

Aeroporto(Codice, Nome, NomeCittà, NazCittà)
 foreign key: Aeroporto[NomeCittà, NazCittà] \subseteq Città[Nome, Nazione]

Città(Nome, Nazione, NumAbitanti)

Compagnia(Nome, AnnoFond)
 foreign key: Compagnia[Nome] \subseteq SedeCompagnia[Comp]

SedeCompagnia(Comp, NomeCittà, NazCittà)
 foreign key: SedeCompagnia[Comp] \subseteq Compagnia[Nome]
 foreign key: SedeCompagnia[NomeCittà, NazCittà] \subseteq Città[Nome, Nazione]
 inclusione: SedeCompagnia[Comp] \subseteq Telefono[Comp]

Telefono(Numero, Comp)
 foreign key: Telefono[Comp] \subseteq SedeCompagnia[Comp]

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

Vincolo esterno:
 vincolo su ordine
 in Tappa.

Diego Calvanese

Basi di Dati - A.A. 2002/2003

Esercizio voli - 4

Specifiche per la ristrutturazione dello schema logico

- Non ci devono essere valori nulli.
- Si accede spesso per conoscere tutte le proprietà di un volo charter.
- Quando si accede alla compagnia si accede anche ai dati relativi alla sua sede.

La relazione Tappa deve essere accorpata in VoloCharter?
Perché?

Diego Calvanese

Basi di Dati – A.A. 2002/2003

Esercizio voli - 5

Ristrutturazioni dello schema logico

- Non ci devono essere valori nulli. →
 - non si possono fare accorpamenti deboli perché in questo caso introdurrebbero valori nulli
- Si accede spesso per conoscere tutte le proprietà di un volo charter. →
 - **decomposizione orizzontale** di Volo in
 - VoloNonCharter
 - DatiVoloCharter
 - **accorpamento forte** di DatiVoloCharter e VoloCharter
- Quando si accede alla compagnia si accede anche ai dati relativi alla sua sede. →
 - **accorpamento forte** di Compagnia e SedeCompagnia

La metodologia non consente di accorpate Tappa in VoloCharter. Sarebbe infatti sbagliato, perché porterebbe ad uno schema con ridondanze.

Diego Calvanese

Basi di Dati – A.A. 2002/2003

Esercizio voli - 6

Schema logico ristrutturato

VoloNonCharter(Codice, Comp, Durata, Arrivo, Partenza)
foreign key: VoloNonCharter[Comp] ⊆ Compagnia[Nome]
foreign key: VoloNonCharter[Arrivo] ⊆ Aeroporto[Codice]
foreign key: VoloNonCharter[Partenza] ⊆ Aeroporto[Codice]
chiave: Comp, Arrivo, Partenza

VoloCharter(Codice, Comp, TipoAereo, Durata, Arrivo, Partenza)
foreign key: VoloCharter[Comp] ⊆ Compagnia[Nome]
foreign key: VoloCharter[Arrivo] ⊆ Aeroporto[Codice]
foreign key: VoloCharter[Partenza] ⊆ Aeroporto[Codice]
chiave: Comp, Arrivo, Partenza

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

Città(Nome, Nazione, NumAbitanti)

Compagnia(Nome, AnnoFond, NomeCittà, NazCittà)
foreign key: Compagnia[NomeCittà, NazCittà] ⊆ Città[Nome, Nazione]
inclusione: Compagnia[Nome] ⊆ Telefono[Comp]

Telefono(Numero, Comp)
foreign key: Telefono[Comp] ⊆ Compagnia[Nome]

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

Diego Calvanese

Basi di Dati – A.A. 2002/2003

Esercizio voli - 7

Vincoli dello schema logico ristrutturato e viste per ricostruire le relazioni originali

Vincoli:

- VoloNonCharter e VoloCharter sono disgiunti:
 $VoloNonCharter[Codice, Comp] \cap VoloCharter[Codice, Comp] = \emptyset$
 $VoloNonCharter[Comp, Arrivo, Partenza] \cap VoloCharter[Comp, Arrivo, Partenza] = \emptyset$
- vincolo esterno: vincolo su ordine in Tappa

Viste per ricostruire le relazioni dello schema originario:

view Volo = PROJ_{Codice, Comp, Durata, Arrivo, Partenza}(VoloNonCharter) ∪ PROJ_{Codice, Comp, Durata, Arrivo, Partenza}(VoloCharter)
view SedeCompagnia = PROJ_{Nome, NomeCittà, NazCittà}(Compagnia)
view CompagniaNoSede = PROJ_{Nome, AnnoFond}(Compagnia)

Diego Calvanese

Basi di Dati – A.A. 2002/2003

Esercizio voli - 8