Il modello relazionale

corso di basi di dati e laboratorio

Prof. Alfio Ferrara

Anno Accademico 2020/2021

Indice

1 Il modello relazionale

1.1 Definizione

Il modello relazionale

- Proposto da E. F. Codd nel 1970 per favorire l'indipendenza dei dati. Codd, Edgar F. "A relational model of data for large shared data banks." Communications of the ACM 13.6 (1970): 377-387.
- Reso disponibile come modello logico in DBMS reali nel 1981.
- Basato sulla nozione di **relazione matematica**, intesa come sottoinsieme del prodotto cartesiano fra due o più insiemi di dati, detti **domini**.
- Il modello relazionale definisce anche un insieme di **vincoli** sui dati ed è associato ad un linguaggio per l'interrogazione delle basi di dati relazionali denominato **algebra relazionale**.

Base di dati relazionale

- La rappresentazione più intuitiva di una relazione è la *tabella*.
- Una bd relazionale è quindi rappresentata come una collezione di tabelle.
- Ogni tabella ha un nome unico nella bd e:
  - una riga di tabella rappresenta una corrispondenza fra valori;
  - ogni colonna ha associato un nome distinto di attributo A_k; ad ogni attributo A_k corrisponde un insieme D_k di possibili valori detto **dominio**.
- In generale, per dominio si intende una collezione di valori atomici. In termini pratici, i domini a partire dai quali sono costruite le relazioni nei DBMS sono definiti a partire da tipi di dati, come ad esempio stringhe di caratteri, interi, date.

Tabella

- Dati n domini $D_1, D_2, \ldots, D_n$,
- ogni riga di una tabella è una *ennupla* ordinata di valori $(d_1, d_2, \ldots, d_n)$ con d_k appartenente al dominio D_k del corrispondente attributo A_k.
- Una tabella contiene un sottoinsieme di tutte le righe possibili, cioè un sottoinsieme del prodotto cartesiano: $D_1 \times D_2 \times \cdots \times D_n$.

Relazione matematica

- Siano $D_1, D_2, \ldots, D_n$ n insiemi di valori anche non distinti.
- Il **prodotto cartesiano** $D_1 \times D_2 \times \cdots \times D_n$ è definito come:

$$D_1 \times D_2 \times \cdots \times D_n = \{(d_1, d_2, \ldots, d_n) \mid d_1 \in D_1, d_2 \in D_2, \ldots, d_n \in D_n\}$$

- Una relazione matematica $\mathcal{R}$ su $D_1, D_2, \ldots, D_n$ è definita come:

$$\mathcal{R} \subseteq D_1 \times D_2 \times \cdots \times D_n$$

- $D_1, D_2, \ldots, D_n$ sono i **domini** di $\mathcal{R}$. Una relazione su n domini è detta avere **grado** n.
- Il numero di ennuple di una relazione è detto **cardinalità** della relazione. Nelle applicazioni reali la cardinalità di una relazione è sempre finita.

Esempio

Dati i domini $D_1 = \{a, b\}$ e $D_2 = \{x, y, z\}$:
Prodotto cartesiano $D_1 \times D_2 = \{(a, x), (a, y), (a, z), (b, x), (b, y), (b, z)\}$

a	x
a	y
a	z
b	x
b	y
b	z

Una relazione $r \subseteq D_1 \times D_2 = \{(a, x), (a, y), (b, x), (b, y)\}$

a	x
a	y
b	x
b	y

Esempio

Dati i domini $D_1 = \{corvo, gatto\}$ e $D_2 = \{bianco, nero, verde\}$: *Prodotto cartesiano* $D_1 \times D_2 =$
$\{(corvo, bianco), (corvo, nero), (corvo, verde), (gatto, bianco),$
$(gatto, nero), (gatto, verde)\}$

corvo	bianco
corvo	nero
corvo	verde
gatto	bianco
gatto	nero
gatto	verde

Una relazione $r \subseteq D_1 \times D_2 = \{(corvo, nero), (gatto, bianco), (gatto, nero)\}$

corvo	nero
gatto	bianco
gatto	nero

1.2 Proprietà delle relazioni

Relazione con attributi

- Si associa ad ogni occorrenza di dominio nella relazione un nome detto **attributo** che descrive il ruolo del dominio nella relazione.
- Una ennupla su un insieme di attributi X è una funzione $t[A_k] \rightarrow D_k$ che associa a ciascun attributo $A_k \in X$ un valore del dominio D_k di A_k.
- $t[A_k]$ denota quindi il valore della ennupla t sull'attributo A_k.

Rappresentazione tabellare

- I valori di ciascuna colonna sono fra loro omogenei: i valori di un attributo appartengono allo stesso dominio.
  - Si noti che se non si definisse un nome univoco per ogni dominio della relazione occorrerebbe fare riferimento all'ordine dei domini per interpretare correttamente i dati. Grazie all'introduzione degli attributi invece tale ordine è irrilevante.
- Le righe sono diverse fra loro: una relazione non contiene mai ennuple identiche (orientamento ai valori).
- L'ordinamento delle colonne è irrilevante poiché esse sono sempre identificate per nome e non per posizione.
- L'ordinamento delle righe è irrilevante poiché sono identificate per contenuto e non per posizione.

Orientamento ai valori

- Mentre è possibile fare riferimento al valore di una ennupla in corrispondenza di un attributo sulla base del nome dell'attributo (usando la notazione $t[A_k]$), non è possibile riferirsi a una specifica ennupla per mezzo di un nome o del suo ordine.
- In altri termini, le ennuple (righe della tabella) non hanno un "nome" né un qualsivoglia sistema di identificazione estraneo ai dati.
- Per denotare una ennupla specifica di una relazione è necessario perciò riferirsi **esclusivamente** ai valori dei dati che essa contiene.
- Questa caratteristica del modello relazionale prende il nome di **orientamento ai valori**.

Esempio

NO

	A	B	C
1	AFG	Afghanistan	islamic republic
2	ALB	Albania	parliamentary democracy
3	DZA	Algeria	republic

individuazione di un valore => (2, C)

SI

iso3	name	government
AFG	Afghanistan	islamic republic
ALB	Albania	parliamentary democracy
DZA	Algeria	republic

individuazione di un valore => (iso3=ALB, government)

Livello intensionale

- **Schema di relazione** $R(A_1, \ldots, A_n)$**:** un nome di relazione R e un insieme di attributi $A_1, \ldots, A_n$
- **Schema di base di dati** $BD = \{R_1(X_1), \ldots, R_2(X_2)\}$**:** insieme di schemi di relazione, con $X_1, \ldots, X_n$ insiemi di attributi.

- Esempio:

  COUNTRY(ISO3, name, government, currency) WEB(website, country, official) WEBSITE(url, title, description)

Livello intensionale: esempio

iso3	name	government	currency
AFG	Afghanistan	islamic republic	Af
ALB	Albania	parliamentary democracy	lek
DZA	Algeria	republic	DA
ASM	American Samoa	NULL	NULL
AND	Andorra	parliamentary democracy...	€
AGO	Angola	republic	Kz
AIA	Anguilla	NULL	NULL
ATA	Antarctica	antarctic treaty summary...	NULL
ATG	Antigua and Barbuda	constitutional monarchy...	NULL
ARG	Argentina	republic	$Arg

Livello estensionale

- Si definisce istanza di relazione su uno schema $R(X)$ l'insieme r di ennuple su X.
- Si definisce istanza di base di dati su uno schema $BD = \{R_1(X_1, \ldots, R_n(X_n))\}$ l'insieme di relazioni $r = \{r_1, \ldots, r_n\}$ con r_i definita su R_i

Livello estensionale: esempio

iso3	name	government	currency
AFG	Afghanistan	islamic republic	Af
ALB	Albania	parliamentary democracy	lek
DZA	Algeria	republic	DA
ASM	American Samoa	NULL	NULL
AND	Andorra	parliamentary democracy...	€
AGO	Angola	republic	Kz
AIA	Anguilla	NULL	NULL
ATA	Antarctica	antarctic treaty summary...	NULL
ATG	Antigua and Barbuda	constitutional monarchy...	NULL
ARG	Argentina	republic	$Arg

2 Informazione incompleta

2.1 Cause e soluzioni

Informazione incompleta

- Una relazione rappresenta la conoscenza acquisita su una certa realtà applicativa di interesse.
- E' possibile che non tutti gli aspetti della realtà applicativa da rappresentare nella base di dati siano noti.
- Il modello relazionale impone ai dati una struttura rigida: le informazioni sono rappresentate per mezzo di ennuple, con formati predefiniti.
- E' ragionevole ammettere che una relazione contenga valori al momento non specificati.

Soluzioni

- Utilizzare valori ordinari del dominio "non utilizzati" (es., 0, stringa vuota, 99).
- NO, perchè:
  - Possono non esistere valori non utilizzati.
  - I valori non utilizzati possono diventare significativi.
  - I programmi dovrebbero tenerne conto.

Valore NULL

- Valore nullo (NULL): denota l'assenza di un valore del dominio.
- Non è un valore del dominio; i domini di definizione delle relazioni vengono estesi.

Semantica del valore NULL

- Il valore nullo non ha una semantica definita, poiché può rappresentare:
- un valore **sconosciuto**;
- un valore **inesistente**;
- un valore **senza informazione**.

Esempio

Cosa possiamo dire della moneta usata nelle American Samoa? E a Antigua and Bermuda?

iso3	name	government	currency
AFG	Afghanistan	islamic republic	Af
ALB	Albania	parliamentary democracy	lek
DZA	Algeria	republic	DA
ASM	American Samoa	NULL	NULL
AND	Andorra	parliamentary democracy...	€
AGO	Angola	republic	Kz
AIA	Anguilla	NULL	NULL
ATA	Antarctica	antarctic treaty summary...	NULL
ATG	Antigua and Barbuda	constitutional monarchy...	NULL
ARG	Argentina	republic	$Arg

3 Vincoli di integrità

3.1 Vincoli di integrità

Esempio: cosa non va in queste relazioni?

COUNTRY

iso3	name	goverment
FRA	France	republic
ITA	Italy	republic
ITA	Italy	NULL
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	-66028467	Population	FRA
1345	2813	59831093	Population	ITA
2848	2013	64097085	Population	GBR
2861	2013	316128839	Population	USA

Vincoli di integrità

Esistono istanze di basi di dati che, pur sintatticamente corrette, non rappresentano informazioni possibili per l'applicazione di interesse.

Vincolo di integrità
Proprietà che deve essere soddisfatta dalle istanze che rappresentano informazioni corrette per l'applicazione.

Un vincolo è una **funzione** booleana che associa ad ogni istanza i valori **vero** o **falso**.

Tipi di vincoli

I vincoli si distinguono in due macro-tipologie:

- Vincoli **intrarelazionali**
  - vincoli su valori o vincoli di **dominio**
  - vincoli di **ennupla**
  - vincoli di **chiave**
- vincoli **interrelazionali**

Esempio: necessità di vincoli di dominio

COUNTRY

iso3	name	government
FRA	France	republic
ITA	Italy	republic
NULL	Italy	NULL
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	-66028467	Population	FRA
1345	2813	59831093	Population	ITA
2848	2013	64097085	Population	GBR
2861	2013	316128839	Population	USA

Vicoli di dominio

- Un vincolo di dominio può essere espresso per mezzo di predicati che utilizzano operatori booleani e coinvolgono gli attributi come variabili.
- Tali predicati sono valutati al momento dell'inserimento e della modifica di una ennupla, garantendo che i dati siano memorizzati solo se il predicato è valutato vero.
- Esempio: supponiamo che i dati statistici siano disponibili solo a partire dal 1950 e non oltre il 2015.
- Forziamo dunque la relazione **STATISTICS** a tollerare solo valori superiori a 1950 e inferiori a 2015 come anno di rilevazione della statistica.

$$\text{year} \geq 1950 \text{ AND year} \leq 2015$$

Esempio: necessità di vincoli di ennupla

COUNTRY

iso3	name	government
FRA	France	republic
ITA	Italy	republic
NULL	Italy	NULL
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	-66028467	Population	FRA
1345	2813	59831093	Population	ITA
2848	2013	64097085	Population	GBR
2861	2013	316128839	Population	USA

Vincoli di ennupla

- Nell'esempio, un valore negativo è perfettamente ammissibile per una rilevazione statistica, ma non se tale rilevazione è riferita alla popolazione. L'errore è relativo alla relazione fra i due dati all'interno della stessa ennupla.
- I vicoli di ennupla esprimono dunque condizioni sui valori di ciascuna ennupla, indipendentemente dalle altre ennuple.

- I vincoli di dominio sono vincoli di ennupla che coinvolgono un solo attributo.
- Una possibile sintassi: espressione booleana (con AND, OR e NOT) di atomi che confrontano valori di attributo o espressioni aritmetiche su di essi (estensione dei vincoli di dominio coinvolgendo più possibili attributi).

Esempio

- Vogliamo che il valore dell'attributo **value** sia positivo ogni volta che la statistica di riferisce alla popolazione (ovvero che l'attributo **label** ha valore Population)
- Pertanto possiamo dire che "Se label = 'Population', allora value deve avere un valore positivo".

$$\text{NOT(label = 'Population') OR value} \geq 0$$

3.2 Vincoli di chiave

Definizione di chiave

Una chiave è un insieme di attributi che identificano univocamente le ennuple di una relazione.

Superchiave

Un insieme K di attributi è **superchiave** di una relazione r se non contiene due ennuple distinte t_1 e t_2 tali che $t_1[K] = t_2[K]$.

Identificazione univoca

Chiave

Un insieme K di attributi è **chiave** di una relazione r se è una **superchiave minimale** per r, ovvero non esiste alcun insieme S tale che S sia superchiave di r **e** $S \subset K$.

Minimalità

Presenza di chiavi

- Una relazione non può contenere ennuple distinte ma uguali.
- Ogni relazione ha come superchiave l'insieme degli attributi su cui è definita.
- Quindi ogni relazione ha (almeno) una chiave.

Esempio

NAMES

name	language	lang	country
France	english	en-us	FRA
France (la)	latin	fr-un	FRA
Italy	english	en-us	ITA
Italie (l')	latin	fr-un	ITA
United States	english	en-us	USA
États-Unis d'Amérique (les)	latin	fr-un	USA

A = name, language, lang, country $\rightarrow$ **superchiave** B = language, lang, country $\rightarrow$ **superchiave** C = lang, country $\rightarrow$ **superchiave**, ne consegue che B non è **chiave** Poiché C non contiene altre superchiavi è minimale, perciò C è una **chiave** E' facile verificare che anche **name** è **chiave**

Proprietà delle chiavi

- L'esistenza delle chiavi garantisce l'accessibilità a ciascun dato della base di dati.
- Le chiavi permettono di correlare i dati in relazioni diverse: **il modello relazionale è basato su valori**.

Chiavi e valori nulli

- In presenza di valori nulli, i valori della chiave non permettono:
  - di identificare le ennuple;
  - di realizzare facilmente i riferimenti da altre relazioni.
- La presenza di valori nulli nelle chiavi deve essere limitata.
- In generale, vogliamo imporre a ogni relazione la presenza di almeno una chiave priva di valori nulli, in modo che ci sia sicuramente una combinazione di dati che consenta l'accesso univoco alle ennuple della base di dati.

Chiave primaria

- Una relazione può avere più chiavi candidate (ne ha almeno una).
- E' opportuno che almeno una chiave consenta di identificare univocamente **ogni** ennupla della relazione e quindi non ammetta valori nulli (**vincolo di entity integrity**).

- **Chiave primaria**: chiave su cui non sono ammessi valori nulli.
- Notazione: sottolineatura.

Esempio: assenza di vincoli di chiave primaria

COUNTRY

iso3	name	goverment
FRA	France	republic
ITA	Italy	republic
NULL	Italy	NULL
USA	United States	constitution-based federal republic

Esempio: effetto del vincolo di chiave primaria

COUNTRY

iso3	name	goverment
FRA	France	republic
ITA	Italy	republic
USA	United States	constitution-based federal republic

L'introduzione del vincolo di chiave primaria sull'attributo **iso3** implica che l'inserimento della seconda ennupla relativa all'Italia non sia possibile, poiché non si possono avere né valori ripetuti su **iso3** (chiave) né valori nulli su **iso3**. In tal modo, a un oggetto reale (il Paese Italia) corrisponderà sempre una e una sola ennupla della base di dati. Inoltre, il valore di **iso3** consente di individuare con esattezza una sola ennupla della relazione.

3.3 Integrità referenziale

Esempio

COUNTRY

iso3	name	government
FRA	France	republic
ITA	Italy	republic
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	66028467	Population	FRA
1345	2013	59831093	Population	Italy
2848	2013	64097085	Population	GBR
2861	2013	316128839	Population	USA

Vincoli di integrità referenziale

- Informazioni in relazioni diverse sono correlate attraverso valori comuni (conseguenza dell'orientamento ai valori).
- In particolare, valori delle chiavi (primarie).

Integrità referenziale

Vincolo di integrità referenziale

Un vincolo di integrità referenziale (**foreign key**) fra gli attributi X di una relazione R_1 e un'altra relazione R_2 impone ai valori su X in R_1 di comparire come valori della chiave primaria di R_2.

Esempio

- Imponiamo un vincolo di integrità referenziale sull'attributo **country** della relazione **STATISTICS** e la relazione **COUNTRY**.

COUNTRY

iso3	name	government
FRA	France	republic
ITA	Italy	republic
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	66028467	Population	FRA
1345	2013	59831093	Population	Italy
2848	2013	64097085	Population	GBR
2861	2013	316128839	Population	USA

Esempio finale corretto

COUNTRY

iso3	name	government
FRA	France	republic
ITA	Italy	republic
USA	United States	constitution-based federal republic

STATISTICS

id	year	value	label	country
952	2013	66028467	Population	FRA
2861	2013	316128839	Population	USA

4 Forme di relazione

Anomalie dovute alla cattiva strutturazione dei dati

- L'introduzione dei vincoli di integrità garantisce la correttezza e buona qualità dei dati, ma non risolve tutte le anomalie possibili.
- Eccesso di valori ridondanti nelle ennuple.
- Eccesso di valori nulli nelle ennuple.
- Possibilità di generare ennuple spurie.

Esempio

COUNTRY_DATA

iso3	government	currency	c_name	lang	language	name	formal_name
DEU	federal republic	€	euro	en-us	english	Germany	Federal Republic of Germany
DEU	federal republic	€	euro	en-un	english	Germany	the Federal Republic of Germany
DEU	federal republic	€	euro	en-iso	latin	Germany	NULL
DEU	federal republic	€	euro	fr-iso	latin	Allemagne	NULL
DEU	federal republic	€	euro	es-iso	latin	Alemania	NULL
DEU	federal republic	€	euro	fr-un	latin	Allemagne (l')	la République fédérale d'Allemagne
DEU	federal republic	€	euro	es-un	latin	Alemania	la República Federal de Alemania
DEU	federal republic	€	euro	en-gb	english	Germany	The Federal Republic of Germany
DEU	federal republic	€	euro	et	latin	Saksamaa	NULL
DEU	federal republic	€	euro	es-fao	latin	NULL	la República Federal de Alemania
DEU	federal republic	€	euro	it-fao	latin	NULL	Repubblica federale di Germania
DEU	federal republic	€	euro	en-fao	english	NULL	the Federal Republic of Germany
DEU	federal republic	€	euro	fr-fao	latin	NULL	la République fédérale d'Allemagne
ITA	republic	€	euro	en-us	english	Italy	Italian Republic
ITA	republic	€	euro	en-un	english	Italy	the Republic of Italy
ITA	republic	€	euro	en-iso	latin	Italy	NULL
ITA	republic	€	euro	fr-iso	latin	Italie	NULL
ITA	republic	€	euro	es-iso	latin	Italia	NULL
ITA	republic	€	euro	de	latin	Italien	NULL
ITA	republic	€	euro	fr-un	latin	Italie (l')	la République italienne

Soluzioni intuitive

- Questo genere di anomalie conduce frequentemente a errori e ulteriori anomalie nelle operazioni di **inserimento**, **aggiornamento**, e **cancellazione** dei dati.
- Intuitivamente vi sono alcuni principi generali che possono aiutare a risolvere tale genere di anomalie:
- un singolo oggetto reale deve corrispondere a una e una sola ennupla.
- una classe di oggetti con le medesime proprietà deve corrispondere a una sola relazione.
- ogni cella di una tabella deve contenere un valore atomico.
- relazionare in modo appropriato i dati di tabelle diverse in modo da evitare di generare corrispondenze scorrette in fase di join.

Dipendenze funzionali

Lo strumento concettuale attraverso il quale analizzare questo genere di anomalie è costituito dalla nozione di **dipendenza funzionale**

Dipendenza funzionale

Una dipendenza funzionale, denotata $X \to Y$, tra due insiemi di attributi X e Y che siano sottoinsiemi di una relazione R specifica un *vincolo* sulle ennuple che possono formare uno stato di relazione *r* di R. Il vincolo stabilisce che, per ogni coppia di ennuple t_1 e t_2 in r per cui $t_1[X] = t_2[X]$, si ha $t_1[Y] = t_2[Y]$, ovvero $t_1[X] = t_2[X] \to t_1[Y] = t_2[Y]$.

Ne consegue che:

- I valori di R su Y dipendono (ovvero sono determinati) dai valori di R su X.
- Se X è una chiave candidata di R, allora $X \to Y$ per ogni sottoinsieme Y di attributi di R.
- Se $X \to Y$, ciò **non** ci dice se $Y \to X$ è vero o no.

Significato delle dipendenze funzionali

- Le dipendenze funzionali sono vincoli relativi alla semantica degli attributi
- Devono essere definite, come gli altri vincoli, dal progettista sulla base della conoscenza della realtà di interesse
- Servono a definire gli **stati validi** di una relazione, ovvero specificare legami logici fra dati che devono valere sempre, per qualsiasi istanza della relazione

Regole di inferenza per dipendenze funzionali

- In genere un progettista individua le dipendenze funzionali più evidenti, sulla base della propria conoscenza del dominio.
- Dato l'insieme F delle dipendenze iniziali individuate, è possibile dedurre altre dipendenze sulla base delle seguenti regole di inferenza.
- L'insieme F^+ delle dipendenze funzionali individuate dal progettista e di tutte quelle inferite prende il nome di **chiusura** di F.

Regole di inferenza

1. Regola *riflessiva*: se $X \supseteq Y$, allora $X \to Y$
2. Regola di *arricchimento*: $\{X \to Y\} \models XZ \to YZ$
3. Regola *transitiva*: $\{X \to Y, Y \to Z\} \models X \to Z$
4. Regola di *decomposizione*: $\{X \to YZ\} \models X \to Y$

5. Regola di *unione*: $\{X \to Y, X \to Z\} \models X \to YZ$
6. Regola *pseudo-transitiva*: $\{X \to Y, WY \to Z\} \models WX \to Z$

Esempio

COUNTRY_DATA

iso3	government	currency	c_name	lang	language	name	formal_name

- Es. 1: {**iso3** $\to$ **currency**, **currency** $\to$ **c_name**}
- Applicando (3): **iso3** $\to$ **c_name**
- Es. 2: {**iso3** $\to$ **name**, **formal_name**}
- Applicando (4): {**iso3** $\to$ **name**}, {**iso3** $\to$ **formal_name**}
- Es. 3: { **name** $\to$ **formal_name**, **name** $\to$ **lang** }
- Applicando (5): **name** $\to$ **formal_name**, **lang**

Normalizzazione delle relazioni

- Le dipendenze funzionali sono utilizzate per effettuare dei test su relazioni dotate di vincoli di chiave.
- A questo scopo si definiscono delle **forme normali** di relazione.
- Se una relazione non è compatibile con una forma normale, la di **decompone** in relazioni più piccole che rispettino la forma normale data.
- In questo modo si ottiene uno schema che è associato a un certo livello di normalizzazione secondo le necessità del progetto di basi di dati.
- In particolare si mira a ottenere uno schema che soddisfi le seguenti proprietà:
- Garantire **join senza perdita** $\to$ se ricostruiamo una relazione dalle sue parti decomposte non dobbiamo generare ennuple non inizialmente presenti.
- Garantire la **conservazione delle dipendenze** $\to$ ogni dipendenza funzionale deve essere rispettata nello schema normalizzato.

4.1 Prima forma normale

Struttura degli attributi

Negli esempi visti il valore di ogni attributo in ogni ennupla era atomico (unico e indivisibile nella bd).

- **Attributo semplice:** costituito da valori atomici.
- **Attributo multivalore:** in cui un possibile valore è un insieme di valori. *Esempio:* **name** $\rightarrow$ {Italy, Italie, Italia}
- **Attributo strutturato:** in cui un possibile valore è una ennupla di valori. *Esempio:* **name** $\rightarrow \langle$Italy, en-us, english$\rangle$

Prima forma normale

Prima forma normale

Uno schema di relazione $R(X)$ è detto in **prima forma normale** (**1NF** o *flat*) se ogni attributo appartenente a X è un attributo semplice.

Altrimenti lo schema è detto in forma strutturata o *nested*.

Prima forma normale

- Nel modello relazionale la 1NF deve essere garantita per ogni relazione che risulta così semplice da interpretare e da gestire.
- Nella rappresentazione tabellare: se i valori sono atomici è facile realizzare le operazioni di manipolazione ed è facile interpretare le ennuple risultato.
- Se una relazione ammette valori multipli o strutturati non è sempre possibile rispettare correttamente le dipendenze funzionali.

Risoluzione di anomalie basate sulla 1NF

In presenza di attributi strutturati o multivalore occorre sostituire l'attributo strutturato con attributi atomici e l'attributo multivalore con una relazione separata che contenga la chiave primaria della relazione originaria.

Esempio

COUNTRY_DATA (relazione originaria)

iso3	**...**	**names**
DEU	...	⟨Germany, Federal Republic of Germany, en-us⟩ ⟨Germany, the Federal Republic of Germany, en-un⟩

COUNTRY_DATA (attributo strutturato)

iso3	**...**	**name**	**formal_name**	**lang**
DEU	Germany	...	Federal Republic of Germany	en-us
	Germany	...	the Federal Republic of Germany	en-un

(attributo multivalore)

COUNTRY

iso3	**...**
DEU	...

NAME

iso3	**lang**	**formal_name**	**name**
DEU	en-us	Federal Republic of Germany	Germany
DEU	en-un	the Federal Republic of Germany	Germany

4.2 Seconda forma normale

Seconda forma normale 2NF

- La seconda forma normale si basa sul concetto di **dipendenza funzionale completa**.
- Una dipendenza funzionale $X \to Y$ è completa se la rimozione di qualsiasi attributo A da X comporta che la dipendenza non sussista più.
- Al contrario, una dipendenza $X \to Y$ è **parziale** se $\exists A \in X : (X - A) \to Y$.

Seconda forma normale 2NF

Uno schema di relazione R è in 2NF se ogni attributo non primo A di R dipende funzionalmente in modo completo dalla chiave primaria di R

- **Normalizzazione**: data una chiave primaria composta X, decomporre R realizzando una relazione che conservi X e, per ogni dipendenza parziale $(X - A) \to Y$, una specifica relazione con schema $(X - A) \cup Y$ e chiave primaria $X - A$.

Esempio

COUNTRY_DATA

iso3	currency	cur_name	government
ITA	€	euro	republic
DEU	€	euro	federal republic
AUS	$A	Australian dollar	federal parliamentary democracy...

- **iso3** $\rightarrow$ **government**
- **iso3** $\rightarrow$ **currency**
- **currency** $\rightarrow$ **cur_name**

COUNTRY

iso3	currency	government
ITA	€	republic
DEU	€	federal republic
AUS	$A	federal parliamentary democracy...

CURRENCY

currency	cur_name
€	euro
$A	Australian dollar

4.3 Terza forma normale e BCNF

Terza forma normale 3NF

- La terza forma normale si basa sul concetto di **dipendenza transitiva**.
- Una dipendenza $X \rightarrow Y$ è transitiva se esiste un insieme di attributi Z che non è né chiave né un sottoinsieme di una chiave per cui valgono $X \rightarrow Z$ e $Z \rightarrow Y$.

Terza forma normale 3NF

Uno schema di relazione R è in 3NF se soddisfa la 2NF e nessun attributo non primo di R dipende in modo transitivo dalla chiave primaria

- **Normalizzazione**: decomporre la relazione definendo una nuova relazione che contenga l'attributo (o attributi) non chiave che determinano funzionalmente un altro (o altri) attributi non chiave (mantenendo l'attributo Z nella relazione originaria come "ponte" fra le nuove relazioni).

Esempio

COUNTRY_DATA

iso3	currency	population	continent	area_population
ITA	€	60,795,612	Europe	742,452,000
DEU	€	81,083,600	Europe	742,452,000
AUS	$A	23,916,300	Oceania	36,659,000

- **iso3** $\rightarrow$ **currency**
- **iso3** $\rightarrow$ **population**
- **iso3** $\rightarrow$ **continent**, **area_population**
- **continent** $\rightarrow$ **area_population**

COUNTRY

iso3	**currency**	**population**	**continent**
ITA	€	60,795,612	Europe
DEU	€	81,083,600	Europe
AUS	$A	23,916,300	Oceania

CONTINENT

continent	**area_population**
Europe	742,452,000
Oceania	36,659,000

Forma normale di Boyce-Codd BCNF

La forma normale di Boyce-Codd (BCNF) è una forma normale più restrittiva della 3NF. Uno schema può essere in 3NF senza essere in BCNF. Ma una relazione BCNF è necessariamente in 3NF.

Forma normale di Boyce-Codd BCNF

Uno schema di relazione R è in BCNF se, ogni volta che sussiste in R una dipendenza funzionale *non banale* $X \rightarrow A$, X è una superchiave di R

Esempio

CORPORATIONS

country	**company**	**manager**
ITA	Bayer	Saris
GRC	Bayer	Robertson
DEU	Bayer	Robertson
ITA	Bracco	Rossi
FRA	Bracco	Rossi

- In questo esempio, la sede locale di una azienda determina il dirigente di riferimento. Un dirigente è legato a una sola azienda, ma può essere responsabile di diverse sedi nazionali.
- In altri termini si hanno le seguenti dipendenze funzionali:
- **country**, **company** $\rightarrow$ **manager**
- **manager** $\rightarrow$ **company**

Esempio

- Nella precedente relazione, **country**, **company** è una chiave candidata ma **manager** non lo è. La relazione è in 3NF, ma non in BCNF.
- Possibili decomposizioni:

1. R1(country, manager), R2(country, company)
2. R1(manager, company), R2(company, country)
3. R1(manager, company), R2(manager, country)

Notando che tutte perdono la dipendenza **country**, **company** $\rightarrow$ **manager**. Occorre verificare però se la decomposizione è non additiva (ovvero non genera ennuple non presenti nella relazione originaria).

Soluzione 1

R1

country	manager
ITA	Saris
GRC	Robertson
DEU	Robertson
ITA	Rossi
FRA	Rossi

R2

country	company
ITA	Bayer
GRC	Bayer
DEU	Bayer
ITA	Bracco
FRA	Bracco

Se ricostruiamo la relazione usando l'attributo comune **country**

CORPORATIONS

country	company	manager
ITA	Bayer	Saris
ITA	Bracco	Saris
GRC	Bayer	Robertson
DEU	Bayer	Robertson
ITA	Bayer	Rossi
ITA	Bracco	Rossi
FRA	Bracco	Rossi

Soluzione 2

R1

manager	company
Saris	Bayer
Robertson	Bayer
Rossi	Bracco

R2

country	company
ITA	Bayer
GRC	Bayer
DEU	Bayer
ITA	Bracco
FRA	Bracco

Se ricostruiamo la relazione usando l'attributo comune **company**

CORPORATIONS

country	company	manager
ITA	Bayer	Saris
GRC	Bayer	Saris
DEU	Bayer	Saris
ITA	Bayer	Robertson
GRC	Bayer	Robertson
DEU	Bayer	Robertson
ITA	Bracco	Rossi
FRA	Bracco	Rossi

Soluzione 3

R1

manager	company
Saris	Bayer
Robertson	Bayer
Rossi	Bracco

R2

country	manager
ITA	Saris
GRC	Robertson
DEU	Robertson
ITA	Rossi
FRA	Rossi

Se ricostruiamo la relazione usando l'attributo comune **manager**

CORPORATIONS

country	company	manager
ITA	Bayer	Saris
GRC	Bayer	Robertson
DEU	Bayer	Robertson
ITA	Bracco	Rossi
FRA	Bracco	Rossi

5 Proprietà delle decomposizioni

5.1 Proprietà

Decomposizione delle relazioni

- Usando la teoria della normalizzazione si può pensare a una base di dati come un prodotto originato da una relazione universale $R = \{A_1, A_2, \ldots, A_n\}$ contenente tutti gli attributi dello schema.
- Da questa relazione universale è poi possibile ottenere uno schema composto da più relazioni applicando delle decomposizioni.
- Lo schema finale ottenuto è detto **decomposizione** di R e denotato $D = \{R_1, R_2, \ldots, R_m\}$.
- Ogni attributo di R deve essere presente in almeno una relazione R_i di D (**conservazione degli attributi**), in modo che:

$$\bigcup_{i=1}^{m} R_i = R$$

Conservazione delle dipendenze

Una buona decomposizione dovrebbe rispettare alcune proprietà.

Conservazione delle dipendenze

Dato un insieme di dipendenze F su R, si definisce **proiezione** di F su R_i ($\pi_{R_i}(F)$) l'insieme delle dipendenze $X \rightarrow Y$ di F^+ tali che gli attributi di $X \cup Y$ siano tutti contenuti in R_i. Una decomposizione D **conserva le dipendenze** di R se:

$$(\pi_{R_1}(F) \cup \cdots \cup \pi_{R_m}(F))^+ = F^+$$

Proprietà di join non-additivo (senza perdita)

Join non-additivo (senza perdita)

Una decomposizione D è senza perdita rispetto all'insieme F di dipendenze di R se, per ogni stato di relazione r di R che soddisfa F, data l'operazione di JOIN $*$ vale che:

$$*(\pi_{R_1}(r), \ldots, \pi_{R_m}(r)) = r$$

5.2 Algoritmi di progettazione

Sintesi relazionale in 3NF con conservazione delle dipendenze

Input: una relazione universale R e un insieme F di dipendenze funzionali su R

1. Si trovi una copertura minimale G per F
2. Per ogni parte sinistra X di una dipendenza in G si crei uno schema di relazione con attributi $\{X \cup \{A_1\} \cup \cdots \cup \{A_k\}\}$ dove $X \to A_1, \ldots, X \to A_k$ sono le sole dipendenze in G con X come parte sinistra
3. Si pongano tutti gli attributi restanti in un unico schema di relazione per assicurare la conservazione degli attributi

Decomposizione in BCNF senza perdita

Input: una relazione universale R e un insieme F di dipendenze funzionali su R

1. $D = \{R\}$
2. Ripetere le seguenti finché c'è uno schema di relazione Q in D che non è in BCNF
3. Si trovi una dipendenza $X \to Y$ in Q che violi la BCNF
4. Si sostituisca Q in D con due schemi di relazione $(Q - Y)$ e $X \cup Y$