

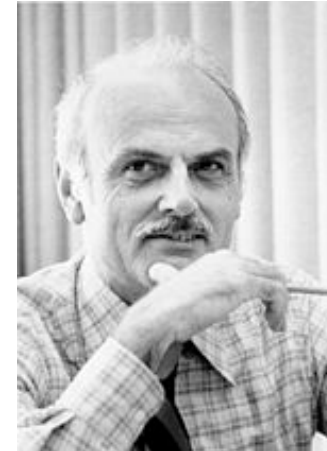
The left side of the slide features a series of vertical stripes in various shades of blue and grey. Overlaid on these stripes are several blue circles of different sizes. One circle contains the number '1'.

# IL MODELLO RELAZIONALE

1

# IL MODELLO RELAZIONALE

- Proposto da E. F. Codd nel 1970 per favorire l'indipendenza dei dati e reso disponibile come modello logico in DBMS reali nel 1981
- Oggi è il modello logico più diffuso ed è adottato dalla larga maggioranza dei DBMS disponibili a livello commerciale



## Information Retrieval

### A Relational Model of Data for Large Shared Data Banks

E. F. CODD  
*IBM Research Laboratory, San Jose, California*

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed. Changes in data representation will often be needed as a result of changes in query, update, and report traffic and natural growth in the types of stored information.

Existing noninferential, formatted data systems provide users with tree-structured files or slightly more general network models of the data. In Section 1, inadequacies of these models are discussed. A model based on  $n$ -ary relations, a normal form for data base relations, and the concept of a universal data sublanguage are introduced. In Section 2, certain operations on relations (other than logical inference) are discussed and applied to the problems of redundancy and consistency in the user's model.

KEY WORDS AND PHRASES: data bank, data base, data structure, data organization, hierarchies of data, networks of data, relations, derivability,

The relational view (or model Section 1 appears to be superior in graph or network model [3, 4] pres inferential systems. It provides a n with its natural structure only—tl posing any additional structure for purposes. Accordingly, it provides data language which will yield ma tween programs on the one hand a tion and organization of data on tl

A further advantage of the rel forms a sound basis for treating d and consistency of relations—these 2. The network model, on the oth number of confusions, not the leas the derivation of connections for tions (see remarks in Section 2 on

Finally, the relational view pern of the scope and logical limitation data systems, and also the relative standpoint) of competing represen single system. Examples of this cited in various parts of this pap systems to support the relational

#### 1.2. DATA DEPENDENCIES IN P

The provision of data descriptio veloped information systems repre toward the goal of data independe facilitate changing certain characte sentation stored in a data bank. I

# IL MODELLO RELAZIONALE

## ○ Caratteristiche:

- È basato su una semplice struttura dati – la *relazione*
- È caratterizzato da precise basi matematiche
  - teoria degli insiemi
  - logica dei predicati del primo ordine

## ○ Vantaggi

- Semplice rappresentazione dei dati → linguaggi dichiarativi
- Facilità con cui possono essere espresse interrogazioni anche complesse



# PRIMA DI INTRODURRE LE RELAZIONI ...

## ... UN RIPASSINO DI TEORIA DEGLI INSIEMI

- Dominio
- Prodotto Cartesiano



# DOMINIO

Un dominio è un insieme (anche infinito) di valori

## Esempi

- l'insieme dei numeri interi
- l'insieme delle stringhe di caratteri
- l'insieme  $\{0,1\}$

## Nel seguito

- D insieme di tutti i domini
  - i suoi elementi sono domini, quindi insiemi
- int: numeri interi
- real: numeri reali
- string: stringhe
- date: date



## RELAZIONE – PRODOTTO CARTESIANO

- $D_1, D_2, \dots, D_n \in D$  (n insiemi anche non distinti)

il prodotto Cartesiano  $D_1 \times D_2 \times \dots \times D_n$  è definito come:

$$\{(v_1, v_2, \dots, v_n) \mid v_1 \in D_1, \dots, v_n \in D_n\}$$

- Il prodotto Cartesiano  $D_1 \times D_2 \times \dots \times D_n$  ha **grado**  $n$
- Ogni elemento del prodotto cartesiano è detto **tupla**



# RELAZIONE

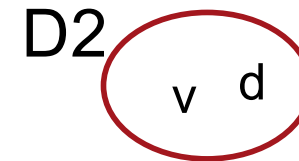
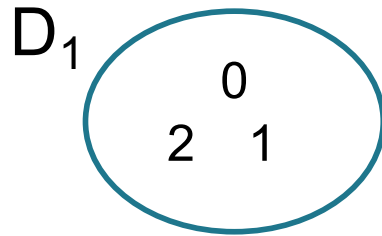
Mentre ciascun dominio può essere infinito

- Siano  $D_1, D_2, \dots, D_n \in D$  domini

Una relazione su  $D_1, D_2, \dots, D_n$  è un sottoinsieme **finito** del prodotto cartesiano  $D_1 \times D_2 \times \dots \times D_n$

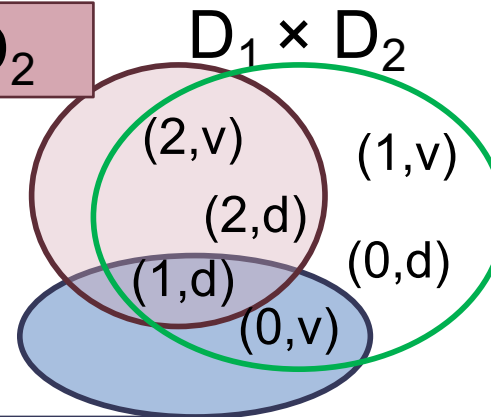
- Una relazione su  $D_1, D_2, \dots, D_k$  ha **grado**  $n$
- Ogni tupla di una relazione di grado  $n$  ha  $n$  componenti, una per ogni dominio su cui è definita la relazione cui la tupla appartiene
- La **cardinalità** di una relazione è il numero di tuple appartenenti alla relazione

# ESEMPIO



relazione  $r_1 \subseteq D_1 \times D_2$

$r_1$  ha cardinalità 3



relazione  $r_2 \subseteq D_1 \times D_2$

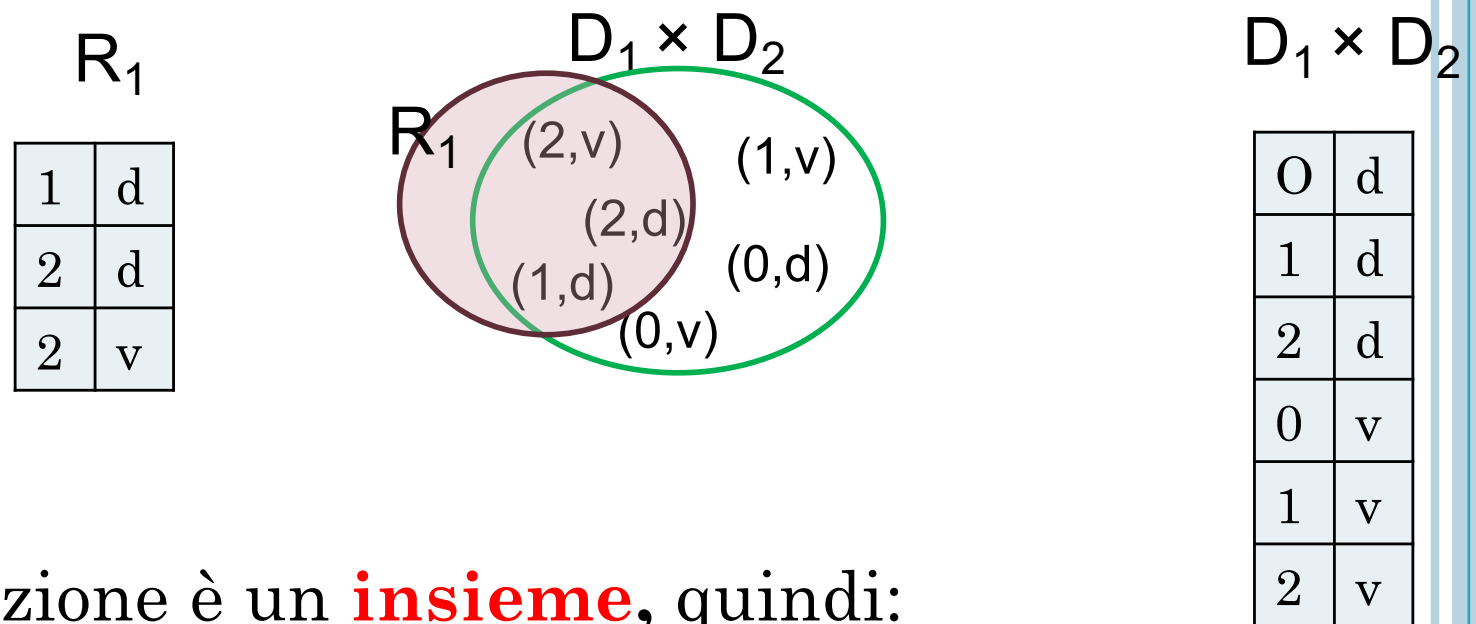
$r_2$  ha cardinalità 2





## NOTAZIONE TESTUALE

- Per semplicità e maggiore leggibilità le relazioni e i prodotti cartesiani si rappresentano come tabelle



- Una relazione è un **insieme**, quindi:
  - non è definito alcun ordinamento fra le tuple
  - le tuple di una relazione sono distinte l'una dall'altra

# RELAZIONE – NOTAZIONE POSIZIONALE

- In generale l'unico modo formalmente corretto di definire il prodotto cartesiano

$$D_1 \times \dots \times D_n$$

è come insieme delle funzioni che associano ad ogni indice un valore nell'insieme che ha quella posizione

$$\{t:[1,n] \rightarrow D_1 \cup \dots \cup D_n \mid t(i) \in D_i \ \forall i \in [1,n]\}$$



# RELAZIONE – NOTAZIONE POSIZIONALE

Siano

- $R$  una relazione di grado  $n$
- $t = (d_1, d_2, \dots, d_n)$  una tupla di  $R$
- $i \in \{1, \dots, n\}$

$t[i]$  denota la  $i$ -esima componente di  $t$  cioè  $t[i] = d_i$

$$t = (0, d)$$

- $t[1] = 0$
- $t[2] = d$

- Quindi ogni tupla è una funzione e la notazione è quella naturale
- In realtà oltre a  $t[i]$  useremo anche  $t(i)$ ,  $t.i$



# RELAZIONE – NOTAZIONE POSIZIONALE

Nella notazione posizionale lo schema è costituito da

- Nome di relazione  $R$
- $D_1, \dots, D_n$



## RELAZIONI – NOTAZIONE PER NOME

- Si associa un **nome di attributo** ad ogni componente delle tuple in una relazione

La coppia (nome di attributo, dominio) è detta **attributo**

- L'uso degli attributi permette di
  - denotare le componenti di ogni tupla per nome piuttosto che per posizione
  - fornire maggiori informazioni semantiche sulle proprietà che ogni componente delle tuple in una relazione modella
- Useremo questa notazione



## RELAZIONE - SCHEMA

- $R$  nome di relazione
- $\{A_1, A_2, \dots, A_n\}$  un **insieme** di nomi di attributi
- $\text{dom} : \{A_1, A_2, \dots, A_n\} \rightarrow D$  una funzione totale che associa ad ogni nome di attributo in  $\{A_1, A_2, \dots, A_n\}$  il corrispondente dominio

La coppia  $(R(A_1, A_2, \dots, A_n), \text{dom})$  è uno  
**schema di relazione**

- $U_R$  denota l'insieme dei nomi di attributi di  $R$ 
  - $U_R = \{A_1, A_2, \dots, A_n\}$



# SCHEMA DI BASE DI DATI

- Siano  $S_1, S_2, \dots, S_m$  schemi di relazioni, con nomi di relazione distinti

$S = \{S_1, S_2, \dots, S_m\}$  è detto  
**schema di base di dati**



# TUPLA E RELAZIONE

- Sia  $S = (R(A_1, A_2, \dots, A_n), \text{dom})$  uno schema di relazione

Una **tupla**  $t$  definita su  $S$  è una funzione totale

$$\{t: U_R \rightarrow \text{dom}(A_1) \cup \dots \cup \text{dom}(A_n) \mid t(A_i) \in \text{dom}(A_i) \forall i \in [1, n]\}$$

- Una tupla  $t$  su  $S$  può essere rappresentata come

$$[A_1 : v_1, A_2 : v_2, \dots, A_n : v_n]$$

$$v_i = t[A_i] \in \text{dom}(A_i), i = 1, \dots, n$$

- Sottotupla: se  $A = \{A_{i_1}, \dots, A_{i_k}\} \subseteq \{A_1, \dots, A_n\}$  sottoinsieme dei nomi di attributi

$$t[A] = [A_{i_1} : v_{i_1}, \dots, A_{i_k} : v_{i_k}] \text{ ovvero } t[A](A_{i_j}) = v_{i_j}$$

- Una relazione definita su uno schema di relazione è un insieme finito di tuple definite su tale schema
  - tale relazione è anche detta **istanza dello schema**





# ESEMPIO

## Film

titolo	regista	anno	genere	valutaz
underground	emir kusturica	1995	drammatico	3.20
edward mani di forbice	tim burton	1990	fantastico	3.60
nightmare before christmas	tim burton	1993	animazione	4.00
ed wood	tim burton	1994	drammatico	4.00
mars attacks	tim burton	1996	fantascienza	3.00
il mistero di sleepy hollow	tim burton	1999	horror	3.50
big fish	tim burton	2003	fantastico	3.10
la sposa cadavere	tim burton	2005	animazione	3.50
la fabbrica di cioccolato	tim burton	2005	fantastico	4.00
io non ho paura	gabriele salvatores	2003	drammatico	3.50
nirvana	gabriele salvatores	1997	fantascienza	3.00
mediterraneo	gabriele salvatores	1991	commedia	3.80
pulp fiction	quentin tarantino	1994	thriller	3.50
le iene	quentin tarantino	1992	thriller	4.00

**Schema:** Film(titolo,regista,anno,genere,valutaz)  
dom(titolo) = dom(regista) = dom(genere) = string  
dom(anno) = int  
dom(valutaz) = real

**Relazione** (= istanza dello schema)  $\subseteq$  string  $\times$  string  $\times$  int  $\times$  string  $\times$  real

**Tupla:** [titolo:'ed wood',regista:'tim burton',anno:1994,  
genere:'drammatico', valutaz:4.00]



# VALORI NULLI

- Non sempre sono disponibili informazioni complete su una qualche istanza di entità
- Per la tupla corrispondente non si sa quale valore assegnare per un qualche attributo
- Si introduce un valore speciale (**valore nullo**) che denota la mancanza di valore
  - dovuta alla nostra mancanza di informazione



# VALORI NULLI – QUALE VALORE?

- Usare un valore  $v$  legale per almeno un dominio non è una soluzione
  - il suo uso non permetterebbe di distinguere il caso in cui  $v$  sia effettivamente il valore dell'attributo dal caso in cui  $v$  indichi il valore nullo
- Assumiamo di denotare il valore nullo con il simbolo '?'
- Il valore nullo è un valore ammissibile per ogni dominio
- I linguaggi come SQL permettono di specificare nella definizione di una relazione quali attributi non possono mai assumere valore null
- Notazione
  - negli schemi evidenziamo con un circoletto gli attributi che possono assumere valori nulli



## ESEMPIO

Noleggio(colloc,dataNol, codCli,**dataRest**o)  
dom(codCli) = dom(colloc) = int  
dom(dataNol)= dom(dataRest)= date

Noleggio

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1111	04-Mar-2006	6642	05-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1116	08-Mar-2006	6642	09-Mar-2006
1118	10-Mar-2006	6642	11-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6610	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?

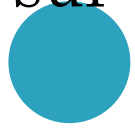
Noleggio



# CARATTERIZZAZIONE DELLE TUPLE

- Per ogni entità di un dominio applicativo vi sono uno o più sottoinsiemi dei suoi attributi che **caratterizzano** le sue istanze
  - nel senso che non possono esistere due istanze diverse con gli stessi valori su tutti quegli attributi

## Esempio

- Cliente(codCli, nome, cognome, telefono, dataN, residenza, CodFiscale)
  - Su tutte le entità l'insieme di tutti gli attributi sicuramente distingue le tuple
  - Determinare quali insiemi di attributi caratterizzano le entità è fondamentale per catturare vincoli di integrità sul dominio applicativo
- 

# CHIAVI

Una chiave è un insieme **minimale** di nomi di attributi che **caratterizzano le istanze di un'entità**

Sia  $R(A_1, \dots, A_n)$  uno schema di relazione

○ Un insieme  $X \subseteq U_R$  di attributi di  $R$  è **chiave** di  $R$  se verifica **entrambe** le seguenti proprietà

1. qualsiasi sia l'istanza di  $R$ , non esistono due tuple distinte di  $R$  che abbiano lo stesso valore per tutti gli attributi in  $X$

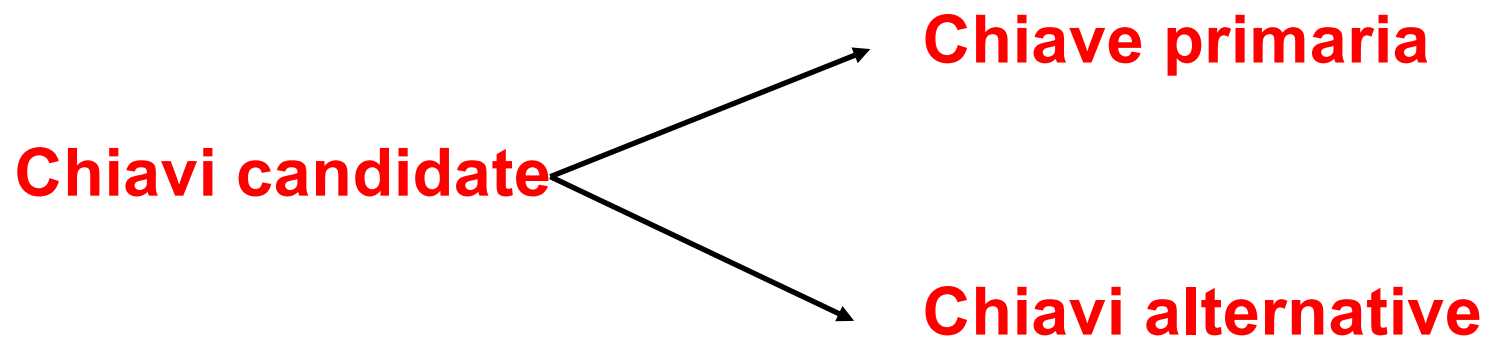
2. nessun sottoinsieme **proprio** di  $X$  verifica la proprietà (1)

○ Un insieme di nomi di attributi che verifica la proprietà (1) ma non la proprietà (2), è detto **super-chiave** di  $R$



# CHIAVI CANDIDATE

- Una relazione può avere più di un insieme  $S$  di (nomi di) attributi che verificano le proprietà (1) e (2)
  - chiavi candidate
- Le chiavi delle relazioni vengono individuate mediante esame del dominio applicativo e dei relativi vincoli
- Una relazione ha sicuramente almeno una chiave
  - $U_R$  soddisfa sempre la proprietà (1)



# CHIAVI

- Una tupla è identificata dal valore di una qualunque chiave candidata
- La chiave primaria viene utilizzata dal DBMS per ottimizzare le operazioni
- Criteri di scelta della chiave primaria
  - Chiave candidata contenente il minor numero di attributi
  - Chiave candidata più frequentemente utilizzata nelle interrogazioni
- Le chiavi primarie **non** possono assumere valori nulli
- Le chiavi alternative possono assumere valori nulli





## ESEMPIO

Cliente(codCli, nome, cognome, telefono, dataN, residenza)

Film(titolo, regista, anno, genere, valutaz.)

Video(colloc, titolo, regista, tipo)

Noleggio(colloc, dataNol, codCli, dataResto)



# NOTAZIONE TESTUALE

Le chiavi primarie si rappresentano in sottolineato (underlined) quelle secondarie in *italico* (italic)

Cliente(codCli, *nome*, *cognome*, *telefono*, dataN, residenza)

Film(titolo, regista, anno, genere, valutaz<sub>0</sub>)

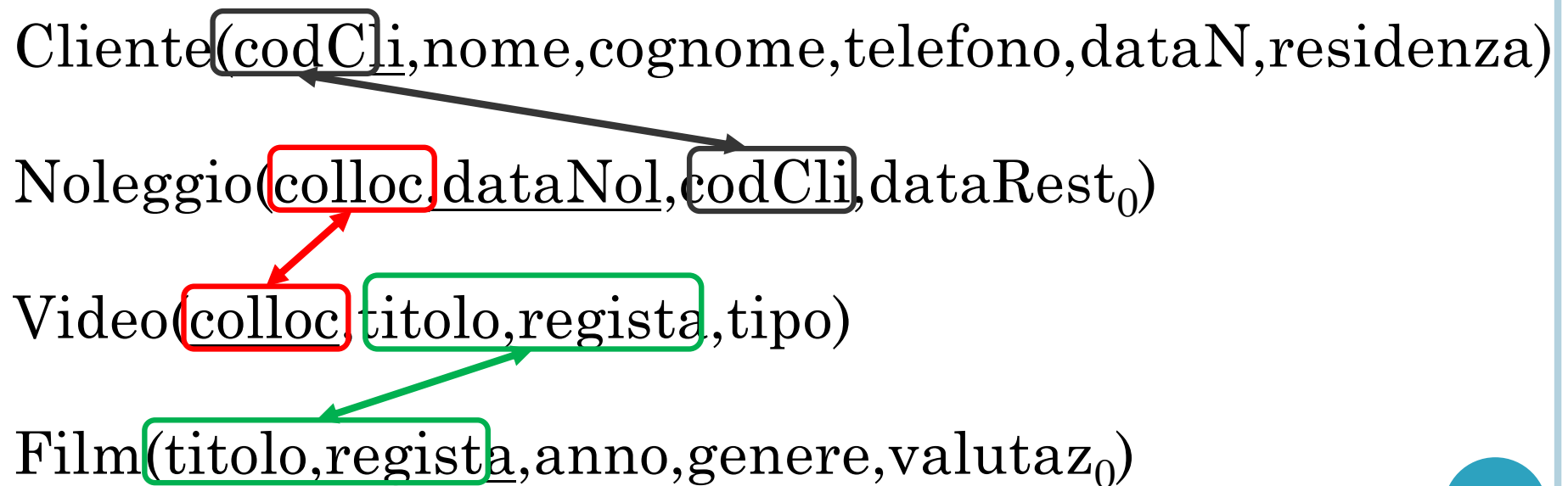
Video(colloc, titolo, regista, tipo)

Noleggio(colloc, dataNol, codCli, *dataRest<sub>0</sub>*)



# RAPPRESENTAZIONE DELLE ASSOCIAZIONI

- Rappresentazione per valore
- Tuple in relazioni diverse sono associate dal condividere gli stessi valori di attributi corrispondenti



# CHIAVI ESTERNE

- Permettono di modellare le associazioni
- Siano
  - $(R(A_1, \dots, A_n), \text{dom})$  ed  $(R'(B_1, \dots, B_m), \text{dom}')$  due schemi di relazione
  - $Y = \{Y_1, \dots, Y_k\} \subseteq U_{R'}$  una chiave per  $R'$
  - $f: Y \rightarrow U_R$  una funzione iniettivatali che  $\text{dom}'(Y_i) \supseteq \text{dom}(f(Y_i))$  per ogni  $i \in [1, k]$  (domini **compatibili**)

$X = \{f(Y_1), \dots, f(Y_k)\} \subseteq U_R$  è una **chiave esterna** di  $R$  su  $R'$  se qualsiasi siano gli stati di  $R$  ed  $R'$   
per ogni tupla  $t$  di  $R$  esiste una tupla  $t'$  di  $R'$  tale che  
 $t[f(Y_i)] = t'[Y_i]$  per ogni  $i \in [1, k]$

- $R$  viene detta **relazione referente**
- $R'$  viene detta **relazione riferita**

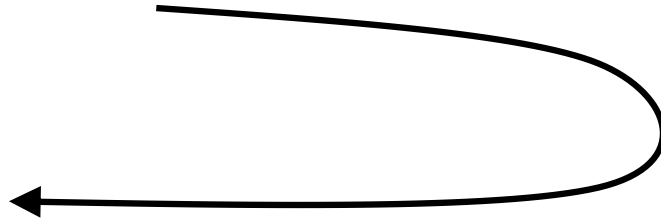


## CHIAVI ESTERNE - SCHEMA

$R(\underline{A}, B, C, \boxed{D}, E)$

X

$R'(\boxed{F}, G, H, I)$

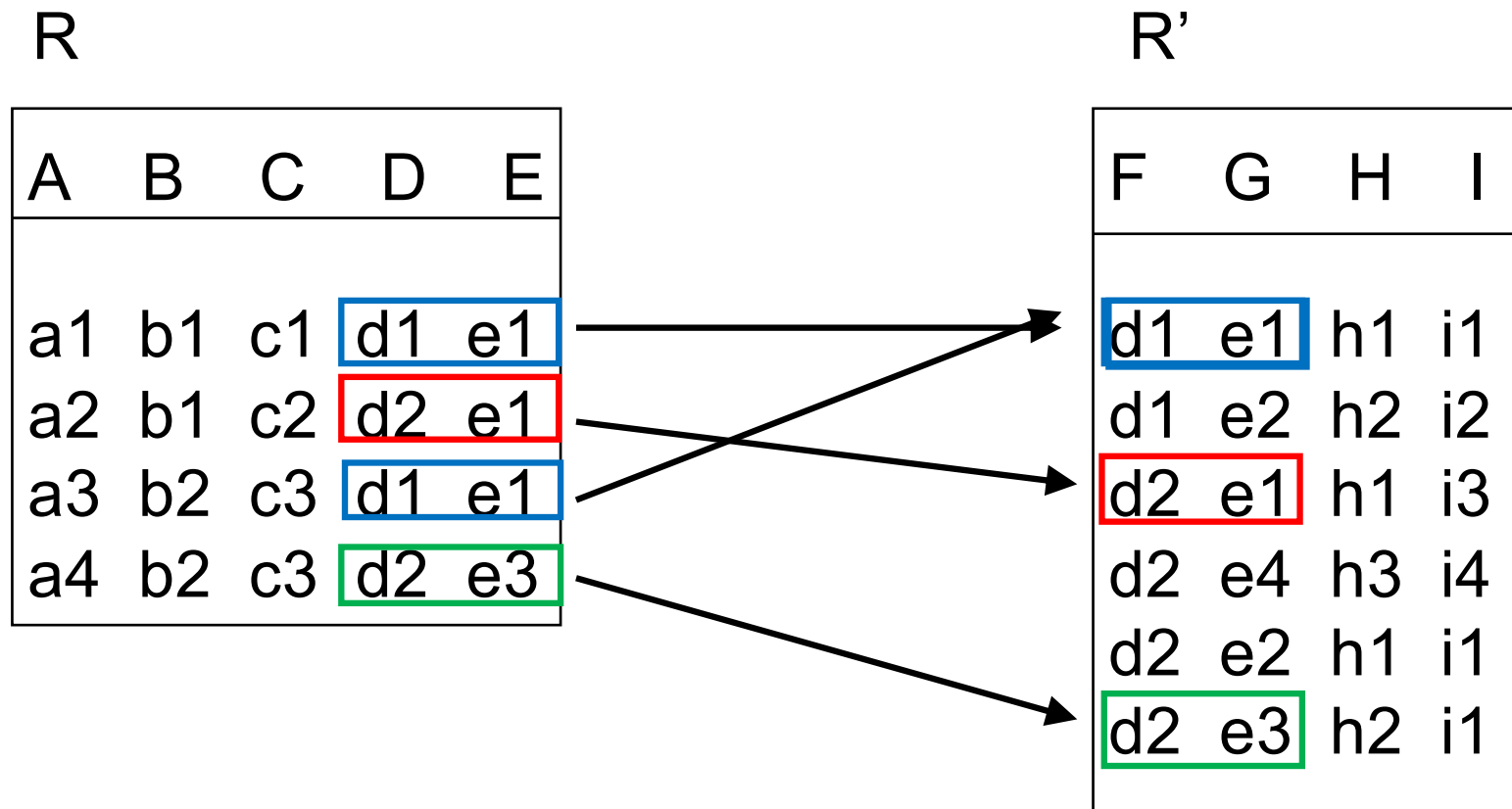


(D,E) e (F,G) domini compatibili  
(D,E) può essere definita come  
chiave esterna di R su R'

Y chiave per R'

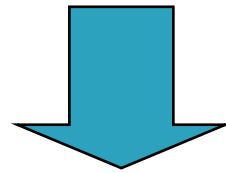


## CHIAVI ESTERNE - ISTANZA

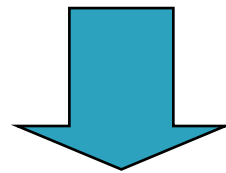


## CHIAVI ESTERNE

Le chiavi esterne permettono di collegare tra loro tuple di relazioni diverse



meccanismo per realizzare le associazioni per valore



Vincolo di **integrità referenziale**



# ESEMPIO

Video(colloc, titolo, regista, tipo)

Film(titolo, regista, anno, genere, valutaz<sub>0</sub>)

## Film

titolo	regista	anno	genere	valutaz
underground	emir kusturica	1995	drammatico	3.20
edward mani di forbice	tim burton	1990	fantastico	3.60
nightmare before christmas	tim burton	1993	animazione	4.00
ed wood	tim burton	1994	drammatico	4.00
mars attacks	tim burton	1996	fantascienza	3.00
il mistero di sleepy hollow	tim burton	1999	horror	3.50
big fish	tim burton	2003	fantastico	3.10
la sposa cadavere	tim burton	2005	animazione	3.50
la fabbrica di cioccolato	tim burton	2005	fantastico	4.00
io non ho paura	gabriele salvatores	2003	drammatico	3.50
nirvana	gabriele salvatores	1997	fantascienza	3.00
mediterraneo	gabriele salvatores	1991	commedia	3.80
pulp fiction	quentin tarantino	1994	thriller	3.50
le iene	quentin tarantino	1992	thriller	4.00

## Video

colloc	titolo	regista	tipo
1111	underground	emir kusturica	v
1112	underground	emir kusturica	d
1113	big fish	tim burton	v
1114	big fish	tim burton	d
1115	edward mani di forbice	tim burton	d
1116	nightmare before christmas	tim burton	v
1117	nightmare before christmas	tim burton	d
1118	ed wood	tim burton	d
1119	mars attacks	tim burton	d
1120	il mistero di sleepy hollow	tim burton	d
1121	la sposa cadavere	tim burton	d
1122	la fabbrica di cioccolato	tim burton	d
1123	la fabbrica di cioccolato	tim burton	d
1124	io non ho paura	gabriele salvatores	d
1125	nirvana	gabriele salvatores	d
1126	mediterraneo	gabriele salvatores	d
1127	pulp fiction	quentin tarantino	v
1128	pulp fiction	quentin tarantino	d
1129	le iene	quentin tarantino	d

Vincolo di  
integrità referenziale  
soddisfatto





## Noleggio

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
			16
			16
			16
			16
			16
			16
			16
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6610	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?
1126	22-Mar-2006	6655	?

Cliente(codCli,nome,cognome,telefono,dataN,residenza)

Noleggio(colloc,dataNol,codCli,dataRest<sub>0</sub>)

Vincolo di  
integrità referenziale  
NON soddisfatto

Cliente

codCli	nome	cognome	telefono	dataN	residenza
6610	anna	rossi	01055664433	05-Ott-1979	via scribanti 16 16131 genova
6635	paola	bianchi	0104647992	12-Apr-1976	via dodecaneso 35 16146 genova
6642	marco	verdi	3336745383	16-Ott-1972	via lagustena 35 16131 genova

# VIOLAZIONI INTEGRITÀ REFERENZIALE

- L'integrità referenziale può essere violata  
nella relazione referente da
  - inserimenti e modifiche (del valore della chiave esterna)  
nella relazione riferita da
  - cancellazioni e modifiche (del valore della chiave)
- I linguaggi per basi di dati quali SQL
  - garantiscono l'integrità referenziale
  - permettono all'utente, nella definizione di chiavi esterne, di specificare *come*, in caso di operazioni conflittuali



# VIOLAZIONI

Vincolo di  
integrità referenziale  
violato da inserimento  
in tabella referente

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1111	04-Mar-2006	6642	05-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1116	08-Mar-2006	6642	09-Mar-2006
1118	10-Mar-2006	6642	11-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6610	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?
1126	22-Mar-2006	6655	?

## Cliente

codCli	nome	cognome	telefono	dataN	residenza
6610	anna	rossi	01055664433	05-Ott-1979	via scribanti 16 16131 genova
6635	paola	bianchi	0104647992	12-Apr-1976	via dodecaneso 35 16146 genova
6642	marco	verdi	3336745383	16-Ott-1972	via lagustena 35 16131 genova

# VIOLAZIONI

Vincolo di  
integrità referenziale  
violato da modifica  
in tabella referente

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1111	04-Mar-2006	6642	05-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1116	08-Mar-2006	6642	09-Mar-2006
1118	10-Mar-2006	6642	11-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6660	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?

6610



6660

6660 non appare

Cliente

codCli	nome	cognome	telefono	dataN	residenza
6610	anna	rossi	01055664433	05-Ott-1979	via scribanti 16 16131 genova
6635	paola	bianchi	0104647992	12-Apr-1976	via dodecaneso 35 16146 genova
6642	marco	verdi	3336745383	16-Ott-1972	via lagustena 35 16131 genova



# VIOLAZIONI

Vincolo di  
integrità referenziale  
violato da cancellazione  
in tabella riferita

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1111	04-Mar-2006	6642	05-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1116	08-Mar-2006	6642	09-Mar-2006
1118	10-Mar-2006	6642	11-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6610	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?

## Cliente

codCli	nome	cognome	telefono	dataN	residenza
6610	anna	rossi	01055664433	05-Ott-1979	via scribanti 16 16131 genova
6635	paola	bianchi	0104647992	12-Apr-1976	via dodecaneso 35 16146 genova
6642	marco	verdi	3336745383	16-Ott-1972	via lagustena 35 16131 genova



# VIOLAZIONI

Vincolo di  
integrità referenziale  
violato da modifica  
in tabella riferita

6610



6630

Cliente

colloc	dataNol	codCli	dataRest
1111	01-Mar-2006	6635	02-Mar-2006
1115	01-Mar-2006	6635	02-Mar-2006
1117	02-Mar-2006	6635	06-Mar-2006
1118	02-Mar-2006	6635	06-Mar-2006
1111	04-Mar-2006	6642	05-Mar-2006
1119	08-Mar-2006	6635	10-Mar-2006
1120	08-Mar-2006	6635	10-Mar-2006
1116	08-Mar-2006	6642	09-Mar-2006
1118	10-Mar-2006	6642	11-Mar-2006
1121	15-Mar-2006	6635	18-Mar-2006
1122	15-Mar-2006	6635	18-Mar-2006
1113	15-Mar-2006	6635	18-Mar-2006
1129	15-Mar-2006	6635	20-Mar-2006
1119	15-Mar-2006	6642	16-Mar-2006
1126	15-Mar-2006	6610	16-Mar-2006
1112	16-Mar-2006	6610	18-Mar-2006
1114	16-Mar-2006	6610	17-Mar-2006
1128	18-Mar-2006	6642	20-Mar-2006
1124	20-Mar-2006	6610	21-Mar-2006
1115	20-Mar-2006	6610	21-Mar-2006
1124	21-Mar-2006	6642	22-Mar-2006
1116	21-Mar-2006	6610	?
1117	21-Mar-2006	6610	?
1127	22-Mar-2006	6635	?
1125	22-Mar-2006	6635	?
1122	22-Mar-2006	6642	?
1113	22-Mar-2006	6642	?

codCli	nome	cognome	telefono	dataN	residenza
6630	anna	rossi	01055664433	05-Ott-1979	via scribanti 16 16131 genova
6635	paola	bianchi	0104647992	12-Apr-1976	via dodecaneso 35 16146 genova
6642	marco	verdi	3336745383	16-Ott-1972	via lagustena 35 16131 genova



## OSSERVAZIONE 1

- I nomi degli attributi nella chiave e nella chiave esterna non devono necessariamente essere gli stessi
  - Se lo sono, semplificano alcune operazioni (join naturale)
- Esempio
  - Cliente(codCli, nome, cognome, telefono, dataN, residenza)
  - Noleggio(colloc<sup>Video</sup>, dataNol, cliente<sup>Cliente</sup>, dataRest<sub>o</sub>)
- Per indicare le chiavi esterne si usa il nome della relazione riferita come apice dei nomi degli attributi usati nella tabella referente, quando non ci siano ambiguità



# OSSERVAZIONE 1

- Se relazione referente = la relazione riferita
  - cioè la chiave esterna contiene un riferimento alla relazione stessa  
gli attributi devono avere nomi diversi
- Esempio
  - $\text{Film}(\underline{\text{titolo}}, \underline{\text{regista}}, \text{anno}, \text{genere}, \text{valutaz}_o, \text{titoloPre}_o^{\text{Film}}, \text{registaPre}_o^{\text{Film}})$
  - $(\text{titoloPre}, \text{registaPre})$  contengono titolo e regista del film di cui il film è eventualmente il seguito
  - il collegamento  $f: \{\text{titolo}, \text{regista}\} \rightarrow U_{\text{Film}}$  dovrebbe essere esplicitato
    - nella comunicazione fra umani si omette se è ovvio dal contesto



## OSSERVAZIONE 2

- Una relazione può contenere più chiavi esterne
  - anche sulla stessa relazione
- Sia le chiavi che le chiavi esterne devono essere esplicitamente specificate nello schema di relazione
  - La presenza di attributi con lo stesso nome e domini compatibili in relazioni diverse non implica siano chiavi esterne
- Le chiavi esterne possono assumere valore nullo



# VINCOLI DI INTEGRITÀ

- Vincoli visti finora
  - di dominio
  - di obbligatorietà
  - di chiave (primaria, alternativa)
  - di integrità referenziale (chiave esterna)
- Non bastano per modellare tutte le informazioni del dominio applicativo da rappresentare
  - come garantire che ogni film sia contenuto in al più 10 video?
- Discuteremo nel seguito altre categorie di vincoli