

Introduzione alle basi di dati e ai DBMS (riferimento: [SA15, Cap. 13])

EUGENIO OMODEO
Università degli Studi di Trieste.

Trieste, 13.11.2018

Citazione del giorno

Una base di dati (database) è una collezione di dati correlati creata con lo scopo specifico di rappresentare adeguatamente e memorizzare le informazioni relative a una realtà di interesse.

La base di dati del sito cinematografico conterrà dati su attori, registi, film e così via;

quella relativa agli esami conterrà dati su studenti, corsi di laurea, esami, docenti e così via.

Un'altra caratteristica fondamentale delle basi di dati è che i dati memorizzati vengono mantenuti costantemente aggiornati nel tempo, per riflettere le modifiche che man mano si verificano nella realtà ...

[CFM09, pag. 2]

Scaletta

DBMS

Mansioni

Gestione dell'informazione su piú livelli

La progettazione concettuale

Il modello ER in generale

I costrutti del modello ER

Sistemi di gestione di basi di dati

*Un **DBMS** (Database Management System) è un sistema software specificamente realizzato per supportare la definizione e la manipolazione della base di dati da parte degli utenti.*

[CFM09, pag. 2]

A che serve un DBMS ?

- ▶ Fornisce agli utenti una **visione di alto livello** dei dati nel DB, nascondendo i dettagli relativi alla loro memorizzazione fisica.
- ▶ Sostiene la **condivisione** dati da parte di più utenti e applicaz., garantendo corretta gestione di accessi contemporanei.
- ▶ Garantisce la **sicurezza** dei dati, permettendo l'accesso ai dati e la loro modifica solo agli utenti autorizzati (offrendo a ciascuno solo le operazioni 'lecite').
- ▶ Fornisce **viste** personalizzate dei dati alle diverse tipologie di utenti e applicazioni.
- ▶ Offre funzionalità di **salvataggio** e **ripristino** dei dati, affinché non possano perdersi dati neppure in caso di malfunzionam. o guasti.
- ▶ Assicura la coerenza fra i dati e i **vincoli d'integrità**.

Intensione ed estensione di un DB

- ▶ Un DBMS realizza in termini concreti un **modello dei dati**.
- ▶ La descrizione delle caratteristiche dei dati di un DB utilizzando un modello dei dati costituisce lo **schema del DB**.
- ▶ I dati veri e propri contenuti in un certo momento nella base di dati costituiscono lo **stato corrente**, o *istanza* del DB.

Schema e contenuto

Modello: Il m. oggi prevalente è quello **relazionale**.¹

Schema: Lo s. fornisce una descrizione del contenuto del DB, sostanzialm. **invariante** nel tempo.

Stato: Lo s. **varia** di frequente, per riflettere gli aggiornamenti apportati ai dati.

¹Storicam., sono apparsi prima lo schema detto *gerarchico* e quello *reticolare*.

Es. di: oggetti / insiemi di oggetti / attributi / associaz.

Informazione	Descrizione	Istanza
Oggetti	Elem. individuali e particolari della realtà, cui riconosciamo un'esistenza	L'attrice <i>Uma Thurman</i> , il film <i>Pulp Fiction</i>
Insiemi	formati di oggetti accomunati da certe caratteristiche	L'insieme degli <i>attori</i> , l'insieme dei <i>film</i>
Attributi	Caratteristiche secondarie (proprietà) degli oggetti	Il <i>nome</i> degli attori, il <i>titolo</i> dei film
Associazioni	Legami logici fra gli oggetti	<i>Uma Thurman ha recitato in Pulp Fiction</i>

Il modello relazionale

Nel *m. relazionale*:

- ▶ i dati sono memorizzati in **tabelle**, dette appunto *relazioni*,
- ▶ composte da un certo numero di **colonne**, dette *attributi*,
- ▶ e di un certo numero di **righe**, dette *tuple*.

Quest'organizzaz. tabellare risulta intuitiva e fa sí che i DB relazionali possano essere interrogati con relativa facilità anche da utenti non esperti della tecnologia relazionale.

Rappresentazione tabellare dei dati

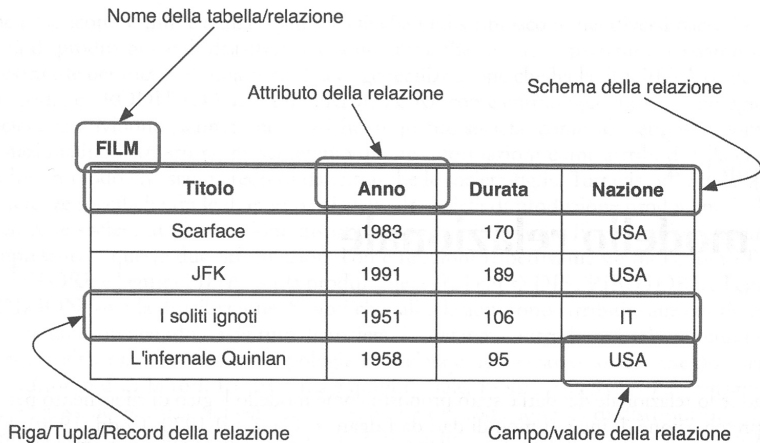


Figura 4.1 Esempio di rappresentazione tabellare dei dati.

Livelli di astrazione

Un DB può essere modellato a tre livelli di astrazione:

Schema logico: fornisce una descrizione dell'intero DB vicina alla *concettualizzazione* della realtà per mezzo del modello dei dati utilizzato nel DBMS.

Schema fisico: troppo tecnico e complesso per interessare l'utente *finale*, cui viene *nascosto* (viene gestito internamente dallo stesso DBMS).

Viste esterne: a varie *tipologie di utenti* i dati possono essere presentati secondo schemi diversi, che fanno riferimento solo ai dati che sono d'interesse specifico per quella particolare utenza.

Modello Entità-Relazione (ER)

Chen, 1976

- ▶ La *realizzazione*, come sistema *software*, di un DB relativo a una **realtà di interesse** dev'essere preceduta da una fase di *progettazione*.
- ▶ Tale **progettazione**, detta **concettuale**, culmina in un diagramma che, utilizzando uno specifico formalismo, schematizza la realtà d'interesse.
- ▶ Uno dei formalismi più usati per tali scopi è il **modello Entità-Relazione**.

Una regola pragmatica

“lo schema non rappresenta la realtà di riferimento in termini assolutamente veri, ma solo funzionali all'applicazione che necessita della base di dati.

... nella progettazione delle basi di dati l'obiettivo è la definizione della realtà di riferimento dell'applicazione e non la mimesi della realtà in quanto tale”.

[CFM09, pag. 18]

Entità, relazioni, attributi

I principali costrutti del modello ER sono le:

entità

relazioni (dette anche ‘*associazioni*’) e gli

attributi

A questi costrutti si aggiungono notazioni per indicare *vincoli* e *proprietà*, fra cui:

cardinalità delle relazioni

gerarchie di generalizzazione

Entità e attributi

Un'**entità**, nel modello ER, rappresenta un insieme di oggetti della realtà che hanno proprietà comuni ed *esistenza autonoma* ai fini dell'applicazione d'interesse.

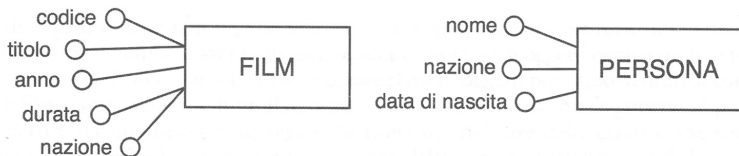


Figura 3.1 Esempio di entità con attributi.

Entità e attributi



Ogni entità, in uno schema ER,

- ▶ ha un **nome** che univocamente la identifica;
- ▶ è raffigurata da un **rettangolo**;
- ▶ è caratterizzata da una collezione di *proprietà elementari*, chiamate **attributi**,² comuni agli oggetti della realtà rappresentati dall'entità stessa.

²Gli a. dipendono dall'e.; non viene loro riconosciuta 'esistenza autonoma'

Entità e attributi

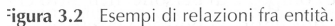


Gli attributi sono raffigurati come **cerchietti**, ciascuno associato al nome dell'attr. e collegato con una linea all'entità cui s'intende riferirlo.

Relazioni fra entità

Una **relazione**, nel modello ER, rappresenta un legame logico fra due o piú entità.

- ▶ Ha un **nome** che univocamente la identifica;
- ▶ è raffigurata da un **rombo**, collegato con linee alle entità che partecipano alla relazione;
- ▶ può possedere *attributi*.



◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

Indicatori di cardinalità

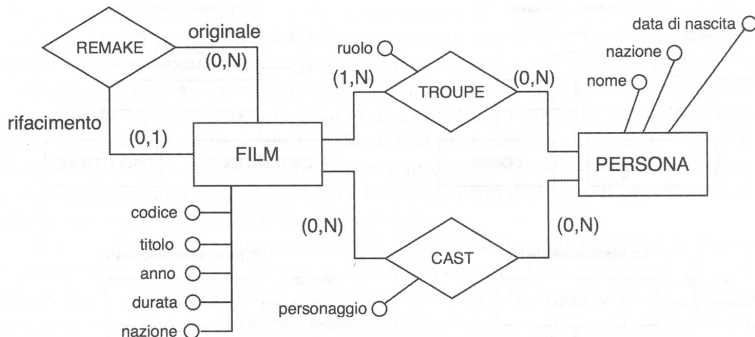


Figura 3.3 Esempi di relazioni fra entità con l'indicazione della cardinalità.

(Un CAST con 0 attori sarà un cartone animato;
una PERSONA può non lavorare ad alcun film)

Indicatori di cardinalità

Spiegazione

Sono importanti il *numero minimo* e *massimo* di corrispondenze che si possono avere fra gli oggetti delle *due* entità nella realtà di riferimento.

Dal momento che ogni relazione ha due sensi di lettura, è importante indicare tali numeri di corrispondenze per entrambe le direzioni.

Conveniamo che la lettera *N* indichi l'assenza di un limite prefissato.

Identificatori

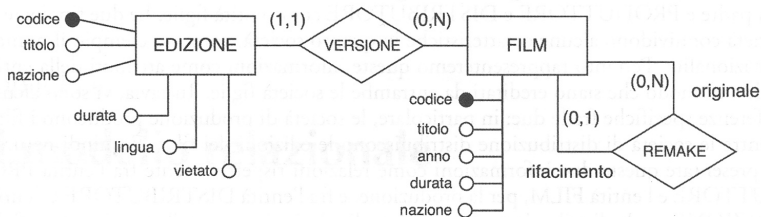


Figura 3.6 Prima porzione di schema ER.

Un **identificatore** è una collezione di attributi di E che consente di identificare univocam. gli elementi di E .



Gerarchie di generalizzazione

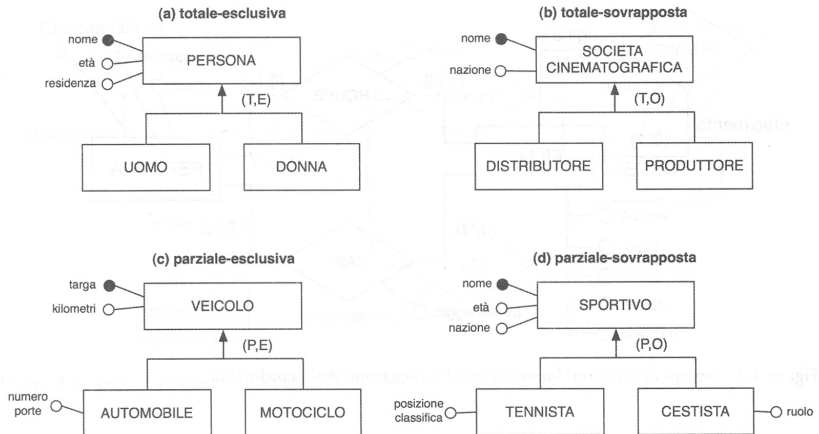


Figura 3.4 Esempi di diverse tipologie di gerarchie di generalizzazione.

Gerarchie di generalizzazione

Spiegazione

- ▶ Le **relazioni** viste sopra rappresentano un *generico* legame fra entità;
- ▶ ma le **gerarchie** rappresentano un legame di *specializzazione* di un'entità rispetto a un'altra.

L'entità più specifica, detta **figlia**, 'eredita' tutti gli attributi dell'entità **madre** e partecipa a tutte le relazioni cui essa partecipa; il vice-versa non è vero.

Tipologie di gerarchia di generalizzazione

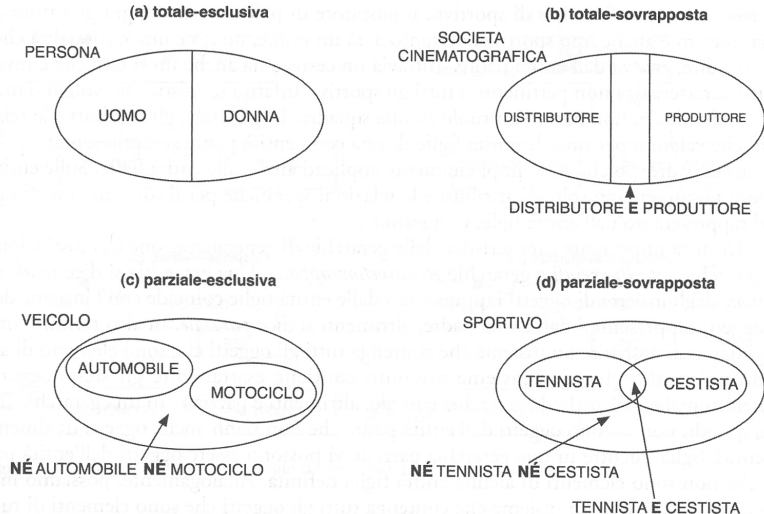


Figura 3.5 Rappresentazione insiemistica delle tipologie di gerarchie di generalizzazione

Esempio di schema ER

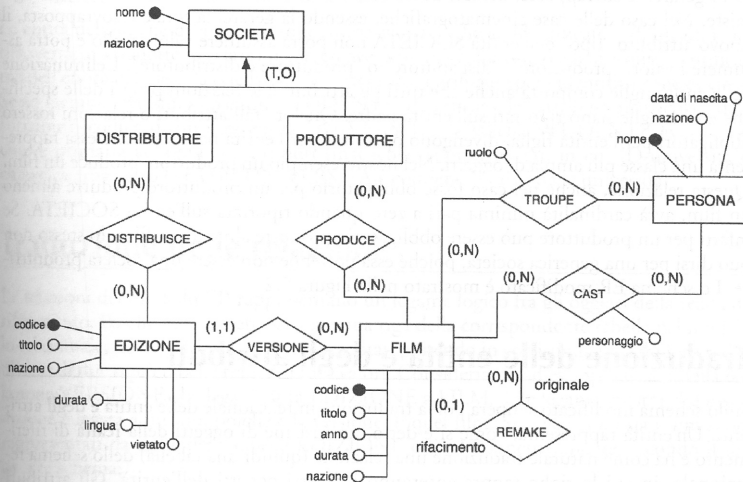


Figura 5.1 Esempio di schema ER.

Riferimenti bibliografici



Silvana Castano, Alfio Ferrara, and Stefano Montanelli.
Informazione, conoscenza e web — per le scienze umanistiche.
Pearson / Addison Wesley, 2009.



Lawrence Snyder and Alessandro Amoroso.
FLUENCY –Conoscere e usare l'informatica.
Pearson Italia, Milano-Torino, 5^a edition, 2015.