



Introduzione Archivi e database

A. Lorenzi, E. Cavalli

INFORMATICA PER ISTITUTI TECNICI TECNOLOGICI

Archivi

Archivi

Archivio: un insieme organizzato di informazioni caratterizzate da:

- un nesso logico che le accomuna, ovvero le informazioni riguardano il medesimo argomento;
- sono rappresentate con un formato che ne rende possibile l'interpretazione;
- sono registrate su un supporto che ne permette l'aggiornamento;
- sono persistenti, cioè possono essere lette anche dopo molto tempo, idealmente sono memorizzate per sempre ... ;
- sono organizzate in modo da permettere una facile consultazione;
- Esempio: elenco telefonico degli abbonati di una città
- Il nostro interesse è per gli archivi informatizzati ...
- Si parla di **file** come collezione di **record** composti da **campi**.

File

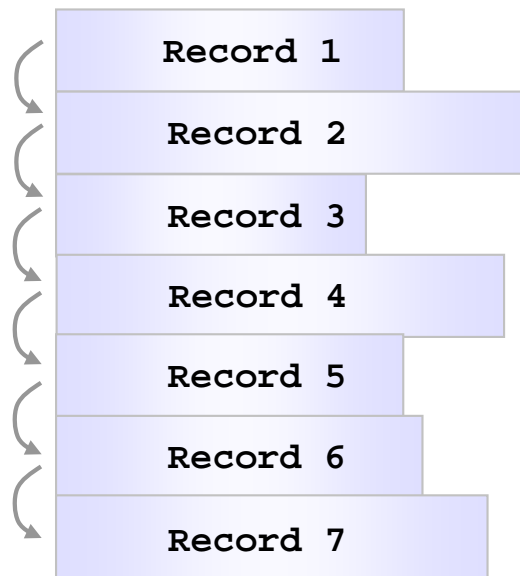
Un **file** è una collezione di **record**, cioè di informazioni logicamente omogenee, che descrivono istanze di una entità. Ogni record è composto da **campi** con i valori assunti dalle caratteristiche scelte per descrivere le entità.



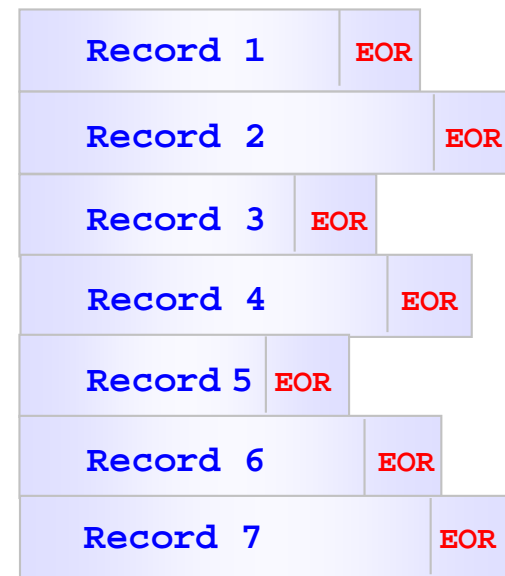
Organizzazione dei file

Organizzazione dei file (1)

Sequenziale: i record sono registrati uno di seguito all'altro. L'organizzazione sequenziale ha come modello di file il pacco di schede o un file su nastro magnetico dove è possibile accedere a un record solo dopo aver visitato tutti i record che lo precedono.



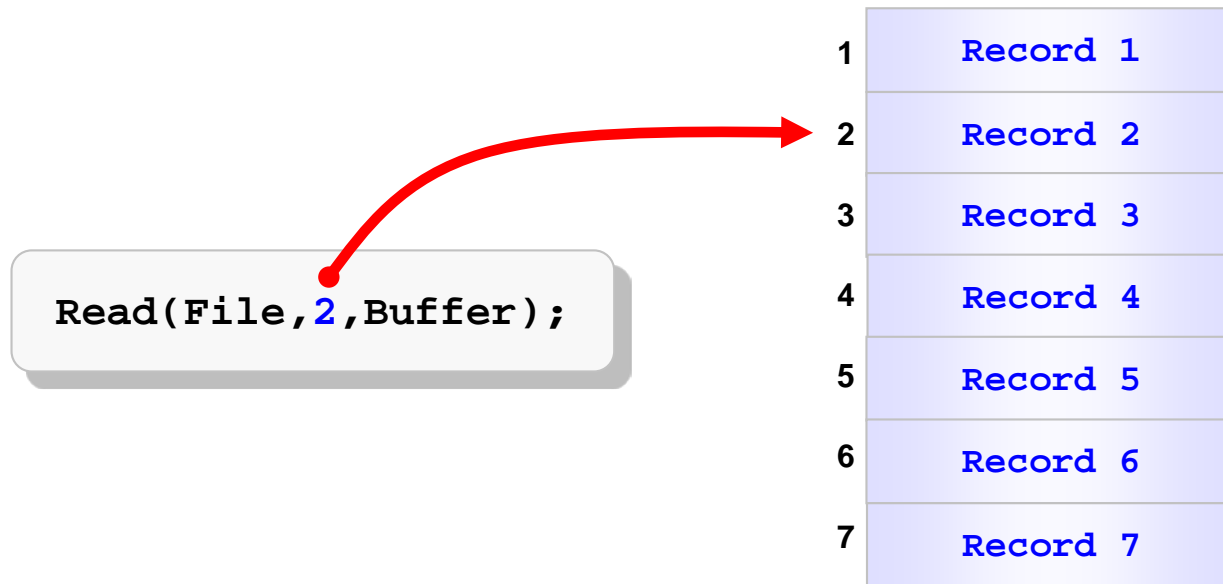
Accesso sequenziale



Organizzazione sequenziale

Organizzazione dei file (2)

Ad **Accesso diretto** o **Random**. I record, tutti della medesima lunghezza, possono essere acceduti, ossia letti o scritti, in base alla posizione che occupano nel file. E' possibile eseguire operazioni del tipo: **Read (File, Pos, Dato)**



Organizzazione dei file (3)

Con **accesso a chiave**. L'accesso avviene in base al valore assunto da un particolare campo detto campo **chiave**. La chiave ha la caratteristica di identificare univocamente il record cercato. Esempio di chiave è la matricola di uno studente.

```
Read("Anagrafe","ROS2",Buffer);
```

Anagrafe

CAR	Carrara	Alessandro
ANZA	Anzani	Antonio
CATT	Cattaneo	Mirella
GANAP	Ganapini	Walter
ROSS	Rossi	Giuliano
BIAN	Bianchi	Francesca
CAVA	Cavallotti	Ennio
ERMO	Ermolli	Marco
ROS2	Rossi	Piercarlo
BERG	Bergantini	Mario
MAGN	Magnani	Gianni
DOTT	Dotti	Laura
MARE	Marenzi	Giuliana
LORE	Lorenzetti	Carla
VENE	Venezian	Luca

Organizzazione dei file (4)

Organizzazione **sequenziale a indici**, o **indexed sequential**. I dati sono registrati nell'ordine di immissione e viene gestita una tabella delle chiavi o **file indice**. La ricerca del record avviene leggendo la tabella delle chiavi mantenuta ordinata secondo il valore delle chiavi. Nell'esempio i dati sono registrati in un file ad accesso diretto.

```
Read ("Cittadini", "RSSFBA75R05F205Q", Buffer);
```

BNCLRA54C65F3007S	550
CVLRCM88D22A205T	088
...	...
...	...
...	...
...	...
...	...
RSSFBA75R05F205Q	370
RSSNCN55A65S407R	045
SPDMLE91S21G135L	620
...	...
...	...

File indice

Cittadini

369			
370	Rossi Fabio	Corso Garibaldi 55
371			

File ad accesso diretto

Indici

- Un **indice** permette di accedere rapidamente ai dati
- La costruzione di un indice ha un costo:
 - **per lo spazio occupato dal file indice**
 - **per il tempo necessario ad aggiornare il file indice e i dati**
- In genere i dati, dopo essere stati scritti negli archivi, sono letti molte volte: il costo pagato per inserirli è giustificato dal risparmio nel tempo di accesso quando li si legge.
- Oltre all'organizzazione sequenziale a indice ci sono altri modi per organizzare gli indici, per esempio con indici su più livelli ovvero con un albero di indici, ...

Archivi e database

Organizzazione convenzionale (1)

- Dati memorizzati in **file** gestiti da programmi ad hoc. Esempio: un archivio anagrafico dei clienti di una banca con il saldo dei conti, gestito con programmi scritti in Cobol

Cognome	Via	Città	CAP	NumConto
Bianchi	Mazzini	Milano	24127	9000
Neri	Garibaldi	Savona	24044	7000
Rossi	Rosmini	Milano	20125	4500
Gialli	Cavour	Padova	24047	5100
Rosa	Crispi	Terni	24129	8000
Verdi	Dante	Potenza	20148	5100
Rossi	Rosmini	Messina	20125	4310

NumConto	Saldo
9000	120345
7000	500
4500	3500
5100	58500
8000	6320
4310	37

Cliente	Codice	Data	Causale	Importo
Bianchi	9000	23/12/2010	Vers	2500,00
Neri	7000	20/04/2011	Prel	1250,00
Neri	7000	27/05/2011	Vers	550,00
...
...
...
...

- In un secondo tempo, un altro gruppo di programmatori sviluppa, in linguaggio C, un'applicazione per gestire i movimenti contabili con versamenti e prelievi

Organizzazione convenzionale (2)

Si nota:

- **NumConto** e **Codice**: nome diverso, stesso dato
- **Cognome** (50 caratteri) e **Cliente**: (40 caratteri): rappresentano il medesimo dato ma hanno formati differenti
- **Saldo** e **Importo**: stesso tipo di grandezze ma con differenti formati
- **Ridondanza dei dati**: il nome del cliente è presente in più archivi
- **Aggiungere un campo**: come fare ad aggiungere i C.F. dei clienti?
- **Inserire i vincoli di integrità dei dati**: non sono ammessi prelievi senza un'adeguata copertura, non si può cancellare un cliente se ci sono movimenti abbinati
- Sviluppare nuove applicazioni per ogni esigenza non prevista inizialmente, per esempio:
 - elenco dei clienti con saldo superiore a un dato importo
 - elenco dei clienti che effettuano un elevato numero di operazioni
 -

Organizzazione convenzionale (3)

L'approccio tradizionale ha evidenziato, nel tempo, molti inconvenienti:

- **Dipendenza dai dati:** i programmi sono dipendenti dagli archivi che gestiscono
- **Difficoltà di accesso ai dati:** è possibile accedere ai dati solo tramite le applicazioni previste, per ogni altra esigenza informativa bisogna sviluppare applicazioni ad hoc
- **Isolamento dei dati:** i dati sono dispersi tra molti file, in differenti formati, pertanto diventa difficile collegarli ed integrarli
- **Ridondanza e inconsistenza dei dati:** causata dallo sviluppo di molte applicazioni indipendenti, da parte di molti programmatori, in differenti linguaggi, con diversi formati dei dati.
- ... segue

Organizzazione convenzionale (4)

- **Difficoltà nel gestire l'integrità dei dati:** i vincoli di integrità dei dati (ad esempio: non sono ammessi prelievi senza un'adeguata copertura, ...) sono esprimibili unicamente scrivendo del codice nei programmi che manipolano i dati.
- **Anomalie dovute alla concorrenza:** ad esempio perdita di uno o più aggiornamenti
- **Problemi di sicurezza:** intesa sia come riservatezza sia come garanzia contro la possibile perdita di dati a causa del malfunzionamento dei dispositivi o del software

Organizzazione con basi di dati (1)

Il **database** è una collezione di dati logicamente correlati e condivisi, che ha lo scopo di soddisfare i fabbisogni informativi di una specifica organizzazione. I dati, congiuntamente con la loro descrizione, sono gestiti da un unico sistema, chiamato **DBMS** (DataBase Management System), che ne permette la gestione e ne regola gli accessi.

Organizzazione con basi di dati (2)

Un **DBMS** deve:

- Permettere la **creazione** di basi di dati da parte degli utenti e di specificare la struttura logica del data base (**schema**) mediante un apposito linguaggio
- Dare la possibilità agli utenti di **interrogare** la base di dati per estrarre informazioni
- Permettere di **manipolare** i dati contenuti nel data base e di modificare lo schema della base di dati
- Effettuare le precedenti operazioni sfruttando i servizi di un linguaggio semplice da apprendere e standardizzato in modo che l'utente possa agevolmente passare da un DBMS ad un altro: **SQL** (Structured Query Language)

Organizzazione con basi di dati (3)

Caratteristiche di un **DBMS**:

- Facilità di accesso ai dati
- Indipendenza dalla struttura logica
- Indipendenza dalla struttura fisica dei dati
- Eliminazione della ridondanza
- Eliminazione della inconsistenza
- Integrità dei dati
- Utilizzo da parte di più utenti
- Controllo della concorrenza
- Sicurezza dei dati

Organizzazione con basi di dati (4)

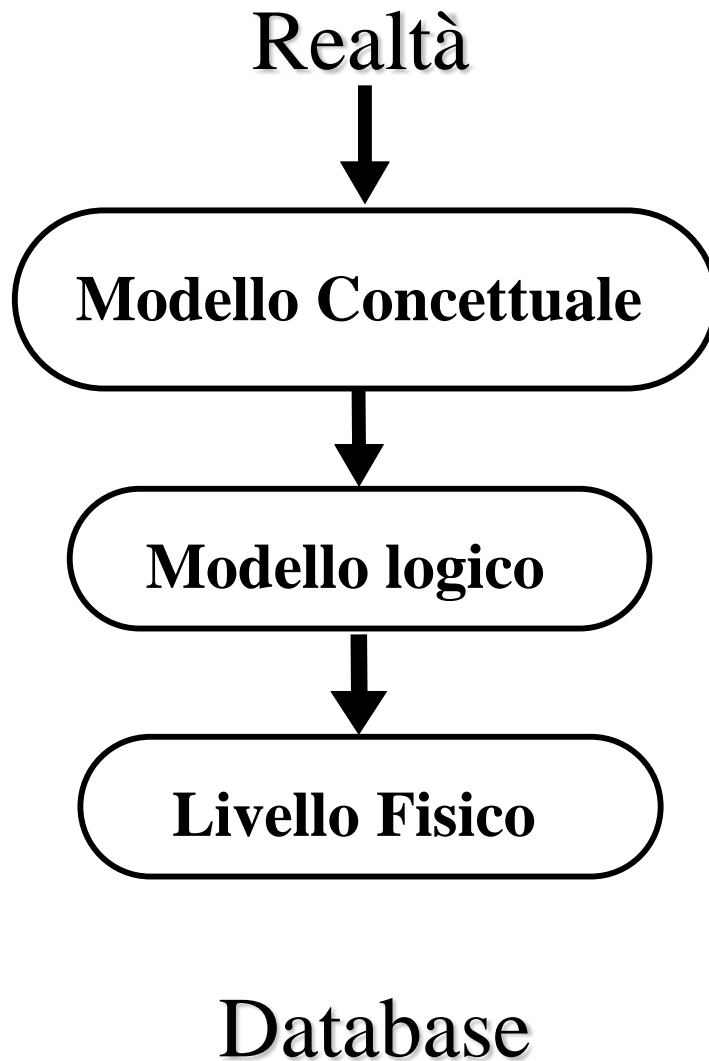
Un'importante ragione alla base delle caratteristiche possedute da archivi gestiti con la tecnologia dei database consiste nel fatto che le informazioni sulla natura degli archivi, la loro composizione, i vincoli sui dati, le limitazioni al loro accesso, in sintesi la descrizione dei dati, è memorizzata all'interno del database stesso con i dati.

La descrizione dei dati è formata dai **metadati**, cioè dati che descrivono i dati, e prende il nome di **dizionario dei dati** o anche **catalogo dei dati** (*in inglese, data dictionary*).

Organizzazione con basi di dati (5)

- Nell'approccio con **Basi di Dati**: i metadati (i dati che descrivono i dati) sono memorizzati assieme ai dati nel database e sono utilizzati dal DBMS
- Nell'approccio tradizionale **File Based**: La descrizione dei dati è memorizzata separatamente rispetto ai dati ed è dispersa e distribuita nei programmi che implementano il sistema informativo
- Non basta usare un database per eliminare gli inconvenienti descritti in precedenza: serve una loro accurata progettazione

Modellazione dei dati: livelli di analisi

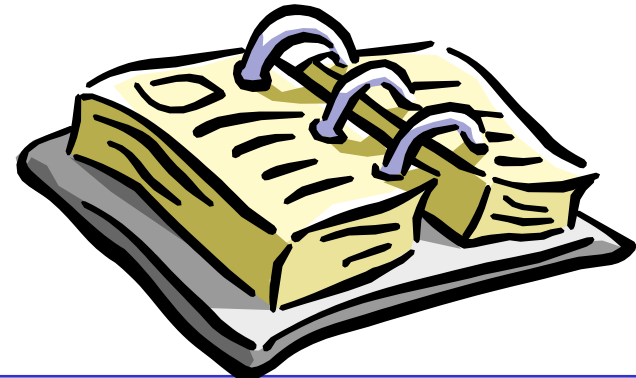


Amico:

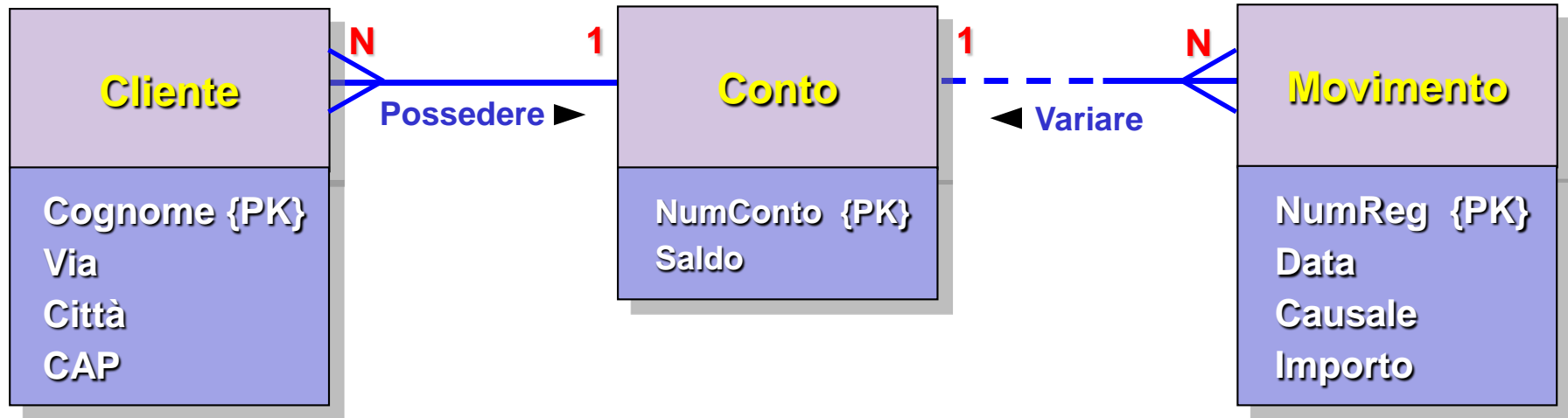
Nome _____

Cognome _____

Telefono _____



Livello concettuale – il modello E/R



Il modello E/R è un utile strumento di comunicazione

- Ogni conto deve essere posseduto da uno o più clienti
- Ogni cliente deve possedere un solo conto
- Ogni conto può essere variato da uno o più movimenti
- Ogni movimento deve variare un solo conto

Il livello logico: il modello relazionale (1)

Basi di Dati Relazionali (Codd 1970): i dati sono visti dall'utente sotto forma di tabelle dette, come vedremo, relazioni. Dalle due entità **Cliente** e **Conto** si derivano le tabelle in figura.

Clienti

Nome	Via	Città	Cap	NumConto
Bianchi	Mazzini	Milano	24127	9000
Neri	Garibaldi	Savona	24044	7000
Rossi	Rosmini	Milano	20125	4500
Gialli	Cavour	Padova	24047	5100
Rosa	Crispi	Terni	24129	8000
Verdi	Dante	Potenza	20148	5100
Rossi	Rosmini	Messina	20125	4310

Conti

NumConto	Saldo
9000	120345
7000	500
4500	3500
5100	58500
8000	6320
4310	37

- Il **modello relazionale** è un modello basato sui valori
- Il legame tra cliente e conto è descritto con la presenza del numero di conto nella riga del cliente
- Non ci sono puntatori

Il livello logico: il modello relazionale (2)

Clienti

Nome	Via	Città	Cap
Bianchi	Mazzini	Milano	24127
Neri	Garibaldi	Savona	24044
Rossi	Rosmini	Milano	20125
Gialli	Cavour	Padova	24047
Rosa	Crispi	Terni	24129
Verdi	Dante	Potenza	20148
Rossi	Rosmini	Messina	20125

5385
5386
6227
6780
4535
6780

Conti

	NumConto	Saldo
5385	9000	120345
5386	7000	500
6227	4500	3500
6780	5100	58500
4535	8000	6320
6500	4310	37

- **Modello Relazionale dei dati:** modello basato sui valori, non ci sono collegamenti tra i record delle tabelle per stabilire legami tra i dati
- **Modello Gerarchico e Reticolare dei dati:** i record associati sono collegati tramite puntatori alla collocazione fisica dei record su disco
 - Sistema più rigido
 - Difficoltà nella manutenzione e nel porting dei dati

Il livello fisico

- **Il livello fisico** riguarda la effettiva rappresentazione dei dati nei dischi del computer:
 - **Come sono rappresentate le tabelle**
 - **Come sono costruiti gli indici**
 - ...
- Non fa parte della progettazione di uno specifico database
- La progettazione del livello fisico esula dagli scopi del corso
- Interessa i progettisti di uno specifico DBMS
- Useremo Access per implementare i database progettati
- Access è la somma di
 - **il DBMS Jet**
 - **un generatore di applicazioni**

Approccio dichiarativo di SQL

- SQL è facile perché ha un approccio dichiarativo
- Si dice cosa si vuole ottenere e non come ottenerlo

Clienti

Nome	Via	Città	Cap	NumConto
Bianchi	Mazzini	Milano	24127	9000
Neri	Garibaldi	Savona	24044	7000
Rossi	Rosmini	Milano	20125	4500
Gialli	Cavour	Padova	24047	5100
Rosa	Crispi	Terni	24129	8000
Verdi	Dante	Potenza	20148	5100
Rossi	Rosmini	Messina	20125	4310

Conti

NumConto	Saldo
9000	120345
7000	500
4500	3500
5100	58500
8000	6320
4310	37

I conti di Rossi

```
SELECT NumConto
FROM Clienti
WHERE Nome = 'Rossi';
```

I clienti con saldo > 50000

```
SELECT Nome
FROM Clienti C, Conti D
WHERE C.NumConto = D.NumConto
AND Saldo > 50000;
```