

Strutture di accesso ai dati

1

ALBERTO BELUSSI

PARTE II

ANNO ACCADEMICO 2010-2011

Gestore dei metodi di accesso

2

E' il modulo del DBMS che trasforma il piano di esecuzione prodotto dall'ottimizzatore in sequenze di accessi alle pagine della base di dati presenti in memoria secondaria.

piano di esecuzione



Gestore dei metodi
d'accesso



Richieste di accesso alle
pagine DATI/INDICI

Gestore dei metodi di accesso

3

Metodi d'accesso

Sono i moduli software che implementano gli algoritmi di accesso e manipolazione dei dati organizzati in specifiche strutture fisiche.

Esempio:

- Scansione sequenziale
- Accesso via indice
- Ordinamento
- Varie implementazioni del join

Metodi di accesso

4

Ogni metodo d'accesso ai dati conosce:

- L'organizzazione delle tuple nelle pagine DATI salvate in memoria secondaria
- L'organizzazione fisica della pagine DATI e delle pagine che memorizzano le strutture fisiche di accesso (INDICI).

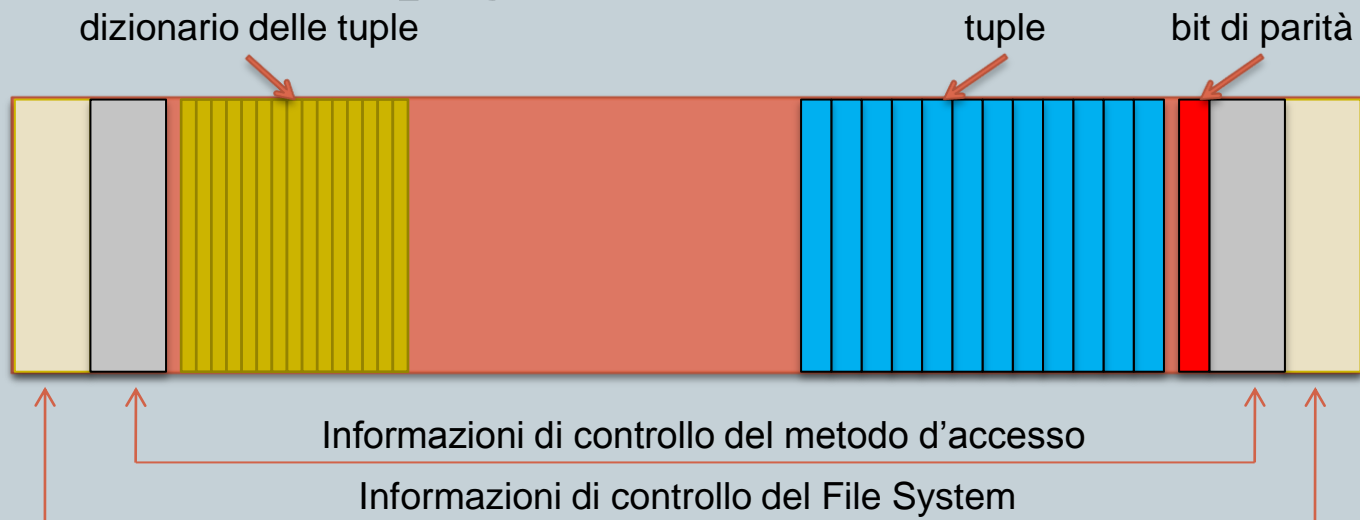
Organizzazione di una pagina DATI

5

In una pagina DATI sono presenti informazioni utili e informazioni di controllo:

- Informazioni utili: tuple della tabella
- Informazioni di controllo: dizionario pagina, bit di parità, altre informazioni del file system o del metodo d'accesso specifico.

STRUTTURA della pagina DATI



Struttura del dizionario

- tuple di lunghezza fissa: il dizionario non è necessario, si deve solo memorizzare la dimensione delle tuple e l'offset del punto iniziale
- Tuple di lunghezza variabile: il dizionario memorizza l'offset di ogni tupla presente nella pagina e di ogni attributo di ogni tupla

Lunghezza massima di una tupla = dimensione massima dell'area disponibile su una pagina, altrimenti va gestito il caso di tuple memorizzate su più pagine.

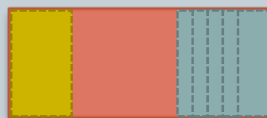
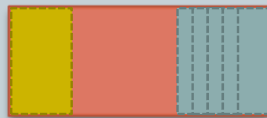
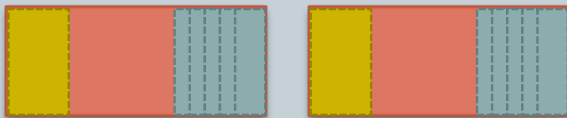
Operazioni

- Inserimento di una tupla
 - Se esiste spazio contiguo sufficiente: inserimento semplice
 - Se non esiste spazio contiguo ma esiste spazio sufficiente: riorganizzare lo spazio e eseguire un inserimento semplice
 - Se non esiste spazio sufficiente: operazione rifiutata
- Cancellazione: sempre possibile anche senza riorganizzare
- Accesso ad una tupla
- Accesso ad un attributo di una tupla
- Accesso sequenziale (in ordine di chiave primaria)
- Riorganizzazione

Rappresentazione di una tabella a livello fisico

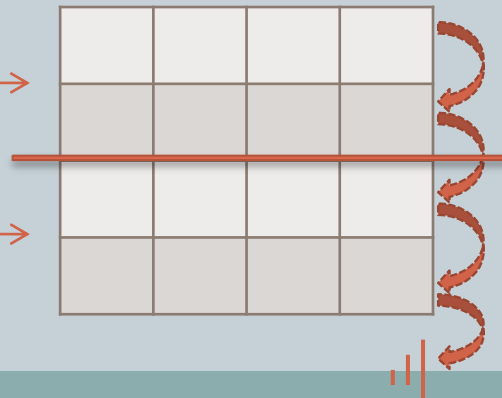
8

Livello fisico (memoria secondaria)



Struttura FISICA:

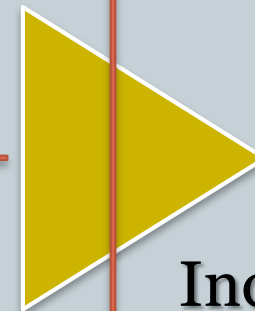
- Strutture ad accesso calc.
- Seriale
- Array
- SEQUENZIALE ORDINATA
o FILE SEQUENZIALE



Livello logico (modello dei dati relazionale)

Tabella

Indice



Strutture fisiche di rappresentazione dei dati

9

File sequenziale (struttura sequenziale ordinata in base alla chiave di ordinamento)

Caratteristica fondamentale: è un file sequenziale dove le tuple sono ordinate secondo una chiave di ordinamento.

Esempio

Pagina 1

Filiale	Conto	Cliente	Saldo	
A	102	Rossi	1000	
B	110	Rossi	3020	
B	198	Bianchi	500	

Pagina 2

E	17	Neri	345	
E	102	Verdi	1200	
E	113	Bianchi	200	
H	53	Neri	120	
F	78	Verdi	3400	



FILE SEQUENZIALE

10

Operazioni

- Inserimento di una tupla
 - Individuare la pagina P che contiene la tupla che precede, nell'ordine della chiave, la tupla da inserire.
 - Inserire la tupla nuova in P; se l'operazione non va a buon fine aggiungere una nuova pagina (overflow page) alla struttura: la pagina contiene la nuova tupla, altrimenti si prosegue.
 - Aggiustare la catena di puntatori.
- Scansione sequenziale ordinata secondo la chiave
- Cancellazione di una tupla
 - Individuare la pagina P che contiene la tupla da cancellare.
 - Cancellare la tupla da P.
- Riorganizzazione: si assegnano le tuple alle pagine in base ad opportuni coefficienti di riempimento.

INDICI

11

Per aumentare le prestazioni degli accessi alle tuple memorizzate nelle strutture fisiche (FILE SEQUENZIALE), si introducono strutture ausiliarie (dette strutture di accesso ai dati o INDICI).

Tali strutture velocizzano l'accesso casuale via chiave di ricerca. La chiave di ricerca è un insieme di attributi utilizzati dall'indice nella ricerca.

Indici su file sequenziali

- INDICE PRIMARIO: in questo caso la chiave di ordinamento del file sequenziale coincide con la chiave di ricerca dell'indice.
- INDICE SECONDARIO: in questo caso invece la chiave di ordinamento e la chiave di ricerca sono diverse.

INDICE PRIMARIO

12

Usa una chiave di ricerca che coincide con la chiave di ordinamento del file sequenziale.

Ogni record dell'indice primario contiene una coppia $\langle v_i, p_i \rangle$:

- v_i : valore della chiave di ricerca;
- p_i : puntatore al primo record nel file sequenziale con chiave v_i

Esistono due varianti dell'indice primario:

- **INDICE DENSO**: per ogni occorrenza della chiave presente nel file esiste un corrispondente record nell'indice.
- **INDICE SPARSO**: solo per alcune occorrenze della chiave presenti nel file esiste un corrispondente record nell'indice, tipicamente una per pagina.

INDICE PRIMARIO

13

Esempio

Indice denso

A	
B	
E	
H	
M	

Pagina 1

A	
E	

Pagina 2

Indice sparso

Filiale	Conto	Cliente	Saldo	
A	102	Rossi	1000	
B	110	Rossi	3020	
B	198	Bianchi	500	
E	17	Neri	345	
E	102	Verdi	1200	
E	113	Bianchi	200	
H	53	Neri	120	
M	78	Verdi	3400	



INDICE PRIMARIO

14

Operazioni

- Ricerca di una tupla con chiave di ricerca K .
 - DENSO ($\Rightarrow K$ è presente nell'indice)
 - ✦ Scansione sequenziale dell'indice per trovare il record (K, p_k)
 - ✦ Accesso al file attraverso il puntatore p_kCosto: 1 accesso indice + 1 accesso pagina dati
 - SPARSO ($\Rightarrow K$ potrebbe non essere presente nell'indice)
 - ✦ Scansione sequenziale dell'indice fino al record $(K', p_{k'})$ dove K' è il valore più grande che sia minore o uguale a K
 - ✦ Accesso al file via $p_{k'}$ e scansione del file per trovare le tuple con chiave K .Costo: 1 accesso indice + 1 accesso pagina dati

INDICE PRIMARIO

15

Operazioni

- Inserimento di un record nell'indice

Come inserimento nel FILE SEQUENZIALE

- DENSO

- ✦ L'inserimento nell'indice avviene solo se la tupla inserita nel file ha un valore di chiave K che non è già presente.

- SPARSO

- ✦ L'inserimento avviene solo quando, per effetto dell'inserimento di una nuova tupla, si aggiunge una pagina dati alla struttura.

INDICE PRIMARIO

16

Operazioni

- Cancellazione di un record nell'indice

Come cancellazione nel FILE SEQUENZIALE

- DENSO

- ✦ La cancellazione nell'indice avviene solo se la tupla cancellata nel file è l'ultima tupla con valore di chiave K.

- SPARSO

- ✦ La cancellazione nell'indice avviene solo quando K è presente nell'indice e la corrispondente pagina viene eliminata, altrimenti, se la pagina sopravvive, va sostituito K nel record dell'indice con il primo valore K' presente nella pagina.

INDICE SECONDARIO

17

Usa una chiave di ricerca che NON coincide con la chiave di ordinamento del file sequenziale.

Ogni record dell'indice primario contiene una coppia $\langle v_i, p_i \rangle$:

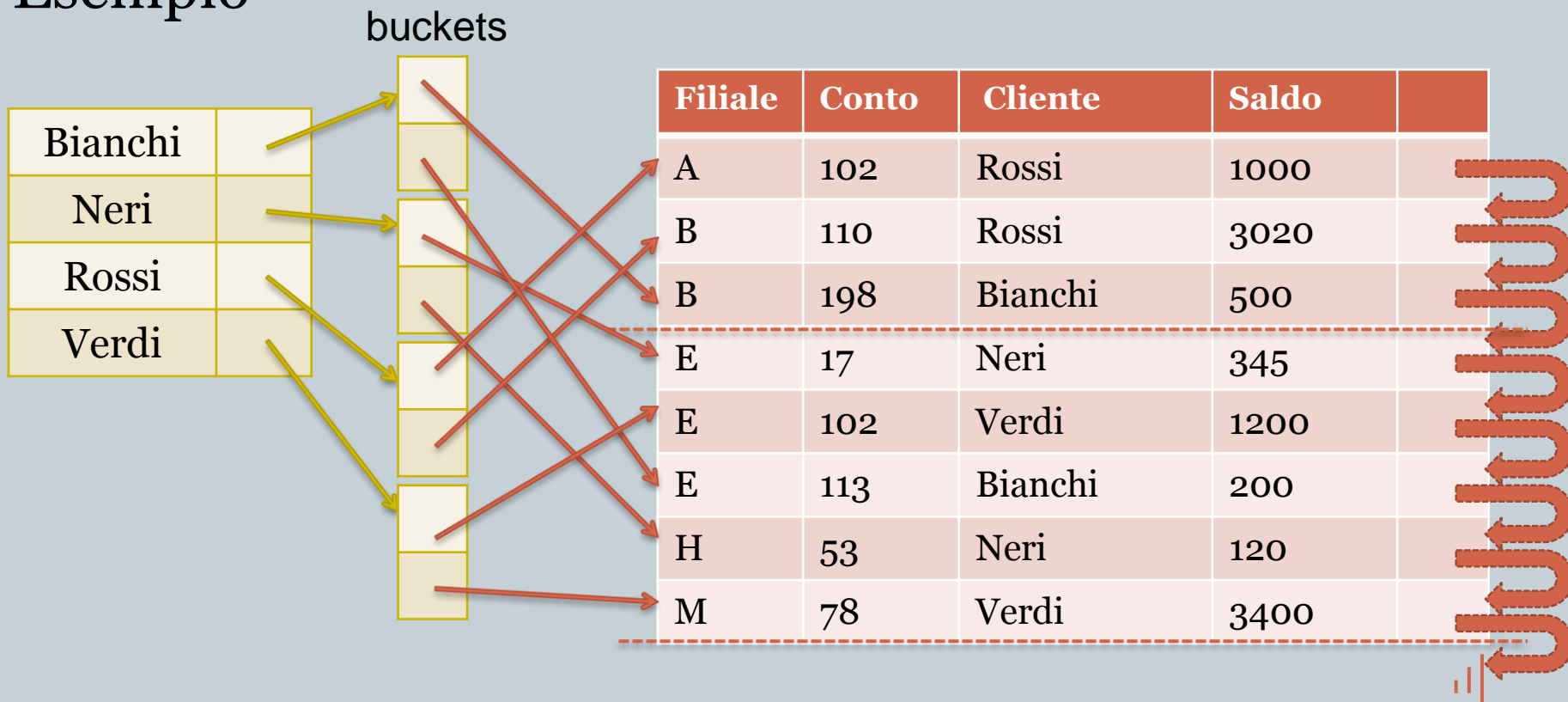
- v_i : valore della chiave di ricerca;
- p_i : puntatore al bucket di puntatori che individuano nel file sequenziale tutte le tuple con valore di chiave v_i .

Gli indici secondari sono sempre DENSI.

INDICE SECONDARIO

18

Esempio



INDICE SECONDARIO

19

Operazioni

- Ricerca di una tupla con chiave di ricerca K .
 - ✦ Scansione sequenziale dell'indice per trovare il record (K, p_k)
 - ✦ Accesso al bucket B di puntatori attraverso il puntatore p_k
 - ✦ Accesso al file attraverso i puntatori del bucket B .

Costo: 1 accesso indice + 1 accesso al bucket + n accessi pagine dati
- Inserimento e cancellazione: come indice primario denso con in più l'aggiornamento dei bucket.

Prossime lezioni ed esercitazioni

20

28-mar-11	Strutture ad accesso calcolato: hashing come indice secondario. Confronto tra B+-tree e hashing. Applicazioni Web: introduzione, protocollo TCP/IP, protocollo HTTP, definizione di URL. Linguaggio HTML: struttura, definizione di LINK. Esempio di applicazione web: il sito della facoltà di Scienze. Architettura generale di sito statico e di sito dinamico centrato sui dati. Accesso a una base di dati dall'application server. Limitazioni dell'approccio CGI. Applicazioni Web: architettura basata sull'estensione dell'application server: Servlet Java. Esempio di servlet: oggetti <code>HttpServletRequest</code> e <code>HttpServletResponse</code> .	2	WEB-MM
29-mar-11	Interazione con una base di dati da un programma JAVA: classi della libreria java JDBC. Applicazioni Web: Esempio di servlet per l'estrazione dati da una base di dati gestita dal sistema postgresql via JDBC.	2	WEB-MM
29-mar-11	Laboratorio di basi di dati (applicazioni web): il linguaggio HTML. Struttura di un file HTML. Principali tag di formattazione del testo. Tabelle. Form. Controllo input con Javascript: cenni	3	LABWEB/MM
04-apr-11	Applicazioni Web: Una metodologia di progettazione di un sito web centrato sui dati: gli schemi di pagina. Metodologia di progettazione di un sito web centrato sui dati: schemi di pagina con link. Esempi e esercizi. Una metodologia di progettazione di un sito web centrato sui dati: corrispondenza con la base di dati.	2	WEB
05-apr-11	Tecnologie per applicazioni Web: Introduzione al costrutto "java bean". Uso dei java bean per l'interazione con il database. Classe Java per l'interazione con il DBMS. I metodi <code>makeBean</code> . Esercizi sulla progettazione logica di applicazioni web centrate sui dati.	2	WEB
05-apr-11	Laboratorio di basi di dati (applicazioni Web): installazione di un servlet engine (Tomcat). Servlet: funzionamento di base e in accoppiata con form HTML. Esempio di servlet semplice (HelloWorld) e servlet per il calcolo del contenuto di una tabella.	3	LABWEB/MM
11-apr-11	Java Server Pages (JSP): introduzione e sintassi di directives e scripting elements. Esempio di JSP semplice. Java Server Pages (JSP): marcatori per l'uso di java beans nelle JSP. Esempi di JSP con java beans.	2	WEB
12-apr-11	Architettura di un'applicazione Web-DBMS: il modello Model View Controller (MVC) servlet-centric. Java Server Pages (JSP): JSP action (forward, include). XML: introduzione. XML: sintassi base, nomi dei tag e struttura. XML: elementi vs attributi. XML: esempi di documenti semistrutturati.	2	WEB
12-apr-11	Laboratorio di Basi di dati (applicazioni Web): interazione con un DBMS dalle servlet: il package JDBC e i Java Data Beans.	4	LABWEB/MM
18-apr-11	XML: specifica della sintassi degli elementi attraverso uno schema XMLSchema. XMLSchema: namespaces, definizione di elementi, tipi semplici e specializzazione di tipi semplici.	2	WEB-MM
19-apr-11	XMLSchema: gerarchie di tipi e gerarchie di elementi (substitutiongroup). Dichiarazione di attributi. Dichiarazione di <code><xsd:complexType></code> varianti: <code><xsd:choice></code> , <code><xsd:sequence></code> e <code><xsd:all></code> . Dichiarazione di elementi vuoti ed elementi vuoti con attributi. File XSD senza namespace. Esempi di file XML e corrispondenti file XSD.	2	WEB-MM
19-apr-11	Laboratorio di Basi di dati (applicazioni Web): Java Server Pages. Esempi di JSP semplici. JSP e interazione con Bean e Database.	4	LABWEB/MM
02-mag-11	DTD: sintassi. XML: esercizi.	2	WEB-MM/MM
03-mag-11	Dati Multimediali: architettura dei DBMS multimediali, il dato multimediale. Concetti generali sulle sorgenti di segnali e messaggi: entropia e quantità di informazione, primo teorema di Shannon. Tecniche di compressione: lossless e lossy.	2	MM
03-mag-11	Laboratorio di applicazioni Web: esercitazione sull'architettura MVC-2 servlet centric.	4	LABWEB/MM
09-mag-11	Tecniche di compressione: Huffman e Lempel-Ziv-Welch. Esempi di codifica ed esercizi.	2	MM
10-mag-11	Tecniche di compressione dei dati multimediali: codifica JPEG. Dati MM: Indici per dati multimediali: il k-d-tree.	2	MM
10-mag-11	Laboratorio di applicazioni Web: Laboratorio di basi di dati e mm: gestione dei dati multimediali da servlet.	3	LABWEB/MM
17-mag-11	Laboratorio di basi di dati: architetture avanzate per applicazioni web. Command pattern e Enterprise Java Beans e Portlet (cenni).	4	
24-mag-11	Esercitazione finale	2	LABWEB/MM