CSCI317 Database Performance Tuning

Indexing Examples

Dr Janusz R. Getta

School of Computing and Information Technology - University of Wollongong

1 of 21 25/6/22, 9:50 pm

Indexing Examples

Outline

Examples

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

Consider a relational table

```
PRODUCT(PNUM, PNAME, PRICE, MANUFACTURER)
```

Assume that

- the table contains 10⁵ rows,
- an attribute **PNUM** is a primary key,
- the products are manufactured by 100 manufacturers,
- the table contains 1000 product names,
- an average number of rows per disk block (blocking factor) is 50
- a database administrator created a non-clustered B*-tree index on attribute MANUFACTURER,
- a fanout of B*-tree is equal to **f**
- a width (total number of blocks at leaf level) of B*-tree implementing an index on the primary key is equal to \overline{w}_{p}
- a width (total number of blocks at leaf level) of B*-tree implementing an index on an attribute MANUFACTURER is equal to \mathbf{w}_m

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

3/21

To compute a query

```
SELECT PNAME, PRICE FROM PRODUCT;

Projection query, no index can be used
```

we have to read10⁵/50 (total number of rows/blocking factor) blocks because the entire relational table PRODUCT must be read

A fragment of respective query processing plan

To compute a query

```
Selection and projection query, index on primary key can be used

SELECT PNAME, PRICE
FROM PRODUCT
WHERE PNUM = 12345;
```

we have to read $[\log_f 10^5 + 1] + 1$ blocks because the primary key PNUM is automatically indexed

A fragment of respective query processing plan

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

5/21

To compute a query

```
Selection and projection query, index on primary key can be used

SELECT PNAME, PRICE
FROM PRODUCT
WHERE PNUM = 1234567 AND
PRICE <= 20.0;
```

we have to read $[\log_f 10^5 + 1] + 1$ blocks because the primary key PNUM is automatically indexed

A fragment of respective query processing plan

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

To compute a query

```
Selection and projection query, index on primary key cannot be used due to disjunction (OR)

SELECT PNAME, PRICE

FROM PRODUCT

WHERE PNUM = 1234567 OR

PRICE <= 20.0;
```

we have to read 10⁵/50 (total number of rows/blocking factor) blocks because the entire relational table PRODUCT must be read

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

7/21

7 of 21

To compute a query

```
Selection and projection guery, index on MANUFACTURER can be used
SELECT PNAME, PRICE
FROM PRODUCT
WHERE MANUFACTURER = 'GoldenBolts';
```

```
we have to read [\log_{\rm f}10^2 + 1] + ((10^5/100) + 10^5/(100*50))/2 data blocks
```

Index processing: $[log_f 10^2 + 1]$ read block operations

Total number of row identifiers found: 10⁵/100

Table processing:

- The best case (full clustering: $10^5/(100*50)$) read block operations
- The worst case (each row in a different block): 10^{5} /100 read block operations

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022 TOP

8/21

Example 5 (continuation)

To compute a query

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

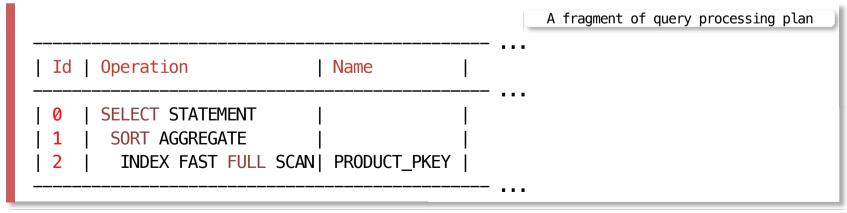
To compute a query

```
Counting the rows, and index on primry key can be used SELECT COUNT(*)
FROM PRODUCT;
```

we have to read w_p data blocks (width of leaf level on an index on primary key)

Index processing: w_p read block operations

Counting row identifiers: 0 read block operations, counting is performed in transient memory



Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

10/21

To compute a query

```
Counting the rows with selection, an index on MANUFACTURER can be used

SELECT COUNT(*)

FROM PRODUCT;

WHERE MANUFACTURER = 'GoldenBolts';
```

we have to read $[\log_f 10^2 +1]$ data blocks

Index processing: [log_f10² +1] read block operations

Counting row identifiers: 0 read block operations, counting is performed in transient memory

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

11/21

11 of 21

To compute a query

```
SELECT MANUFACTURER
FROM PRODUCT
WHERE MANUFACTURER LIKE 'M%';
```

we have to read w_m data blocks (width of a leaf level of an index on MANUFACTURER)

Index processing: w_m read block operations

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

12/21

To compute a query

```
Projection query with elimination of duplicates, an index on MANUFACTURER can be used

SELECT DISTINCT MANUFACTURER

FROM PRODUCT;
```

we have to read w_m data blocks

Index processing: w_m read block operations (horizontal traversal through leaf level on an index on MANUFACTURER)

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

13/21

To compute a query

```
Aggregation query, an index on MANUFACTURER can be used

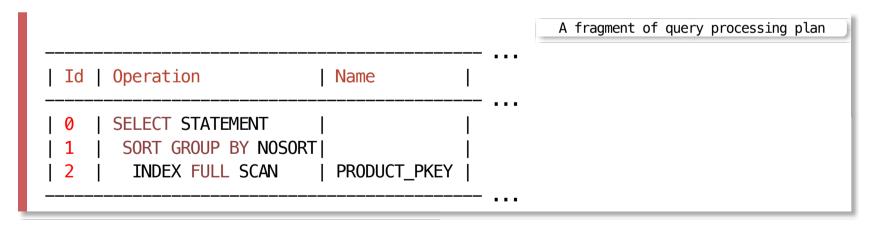
SELECT MANUFACTURER, COUNT(*)

FROM PRODUCT

GROUP BY MANUFACTURER;
```

we have to read w_m data blocks

Index processing: w_m read block operations



TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

Examples (continuation)

Consider a relational table

PRODUCT(PNUM, PNAME, PRICE, MANUFACTURER)

Assume that

- a database administrator created a non-clustered B*-tree index on attribute PNAME,
- the table contains 1000 distinct product name
- a fanout of B*-tree is equal to f
- a width (total number of blocks at leaf level) of B*-tree implementing an index on an attribute $\mbox{\tt PNAME}$ is equal to $\mbox{\tt w}_n$

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

15/21

To compute a query

```
Selection query with conjunction WHERE condition, both indexes on MANUFACTURER and PNAME can be used

SELECT *
FROM PRODUCT
WHERE PNAME = 'bolt' AND
MANUFACTURER = 'Golden Bolts';
```

we have to read $\lfloor \log_f 1000 + 1 \rfloor + \lfloor \log_f 100 + 1 \rfloor + 1$ data blocks

Index processing: [log_f1000 +1] read block operations to process an index on PNAME

The total number of row identifiers found: 10⁵/1000

Index processing: [log_f100 +1] read block operations to process an index on MANUFACTURER

The total number of row identifiers found: 10⁵/100

Intersection of the sets of row identifiers obtained from index processing: 0 read block operations

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

Total number of row identifiers found: $10^5/(1000*100) = 1$

Table processing: 1 read block operations

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

17/21

To compute a query

```
Selection query with disjunction WHERE condition, both indexes on MANUFACTURER and PNAME can be used

SELECT *
FROM PRODUCT
WHERE PNAME = 'bolt' OR
MANUFACTURER = 'Golden Bolts';
```

```
we have to read [\log_f 1000 + 1] + [\log_f 100 + 1] + ((10^2 + 10^3 - 1) + (10^2 + 10^3 - 1)/50)/2 data blocks
```

Index processing: $[log_f 1000 + 1]$ read block operations to process an index on PNAME

The total number of row identifiers found: 10⁵/1000

Index processing: [log_f100 +1] read block operations to process an index on MANUFACTURER

The total number of row identifiers found: $10^5/100$

TOP Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

18/21

Union of the sets of row identifiers obtained from index processing: 0 read block operations

```
Total number of row identifiers found: (10^5/1000) + (10^5/1000) - (10^5/(1000*100)) = 10^2 + 10^3 - 1

Table processing: ((10^2 + 10^3 - 1) + (10^2 + 10^3 - 1)/50)/2 read block operations
```

Created by Janusz R. Getta, CSCI317 Database Performance Tuning, SIM, Session 3, 2022

19/21

To compute a query

```
Selection query with a range condition, an index on PNAME can be used

SELECT *
FROM PRODUCT
WHERE PNAME > 'bolt';
```

we have to read $[\log_f 1000 +1] + w_p*n/10^5 + (n + n/50)/2$ data blocks

Vertical index processing: $[log_f1000 +1]$ read block operations to process an index on PNAME

Horizontal index processing: $w_p*n/10^5$ where w_p is the total number of leaf level blocks and n is the total number of rows that satisfy a condition PNAME > 'bolt'

20/21

References

Cookbook, How to measure and how to improve performance of database applications, how to choose the best index, how to analyze index structures?

Ramakrishnan R., J. Gehrke Database Management Systems, chapters 8.1-8.3

Lightstone, S., Teorey T., Nadeau T., Physical Database Design, The Database Professional's Guide to Exploiting

Indexes, Views, Storage, and More, Morgan Kaufmann Publishers, 2007, chapter 4

SIM, Session 3, 2022 21/21