CSCI317 Database Performance Tuning

Quantitative Analysis of Indexing

Dr Janusz R. Getta

School of Computing and Information Technology -
University of Wollongong

Quantitative Analysis of Indexing

Outline

Cost model ? What is it ?

A model used for estimation of the cost of various database operations

Notation

- `b` => the total number of fully packed blocks
- `r` => an average number of rows per block
- `f` => fanout of B*-tree index
- `k` => number of keys in an index

Quantitative Analysis of Indexing

Outline

Storage organizations and models

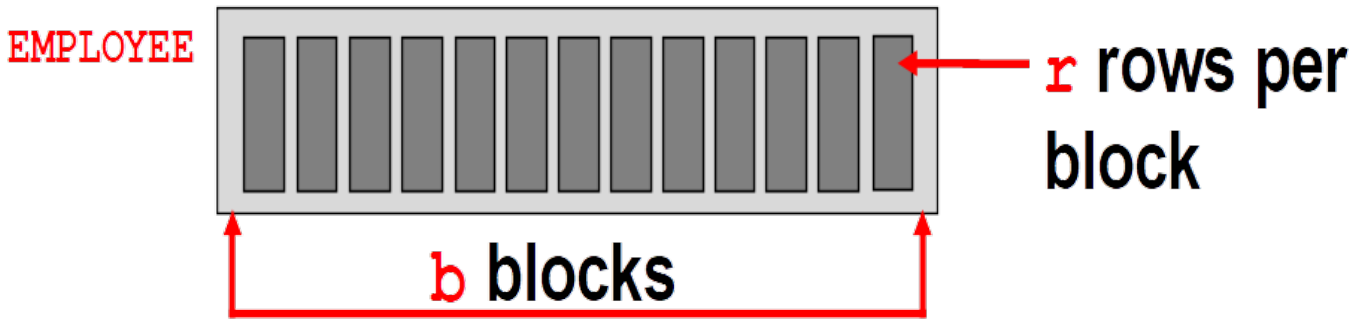
- Tables with the randomly ordered rows
- Tables with the clustered B*-tree indices
- Tables with the unclustered B*-tree indices

Quantitative Analysis of Indexing

Outline

Tables with randomly ordered rows

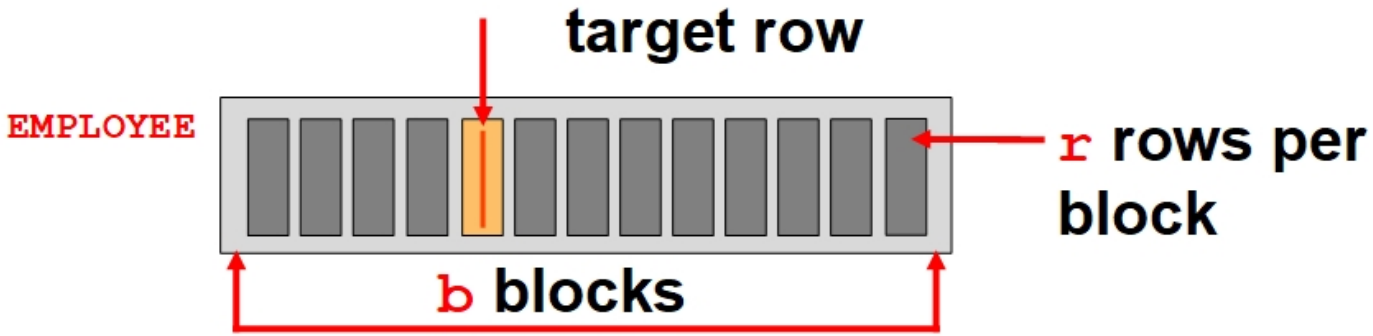
Full scan



Cost = b read block operations

Tables with randomly ordered rows

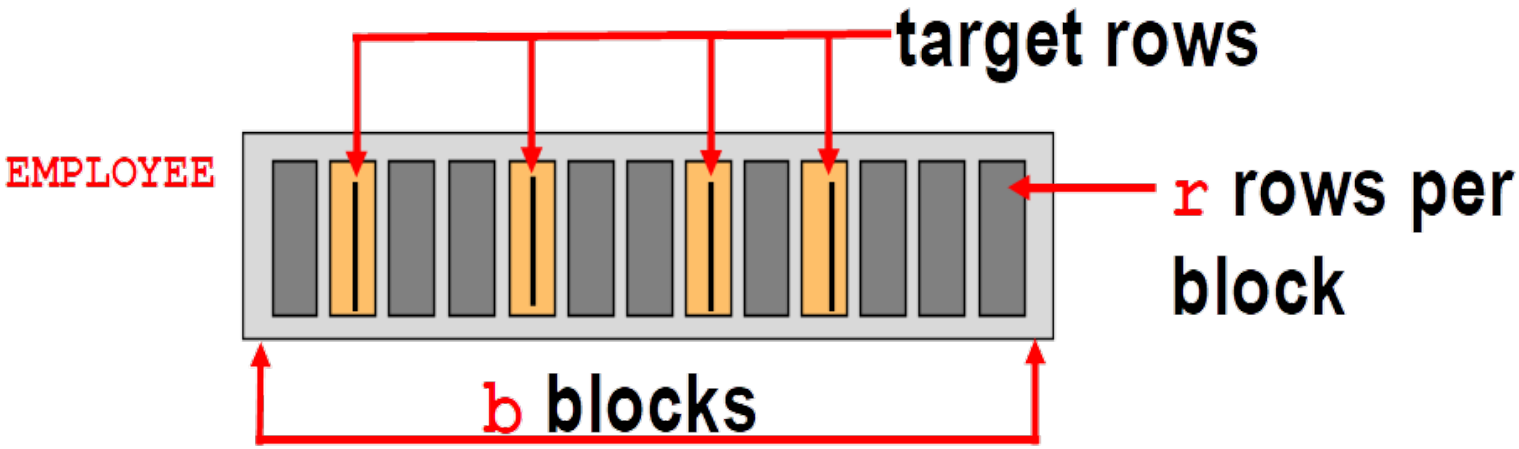
Full scan with an equality condition on a key



Cost = `(1 + b)/2` read block operations

Tables with randomly ordered rows

Full scan with a range condition



Cost = `b` read block operations

Quantitative Analysis of Indexing

Outline

Cost model ? What is it ?

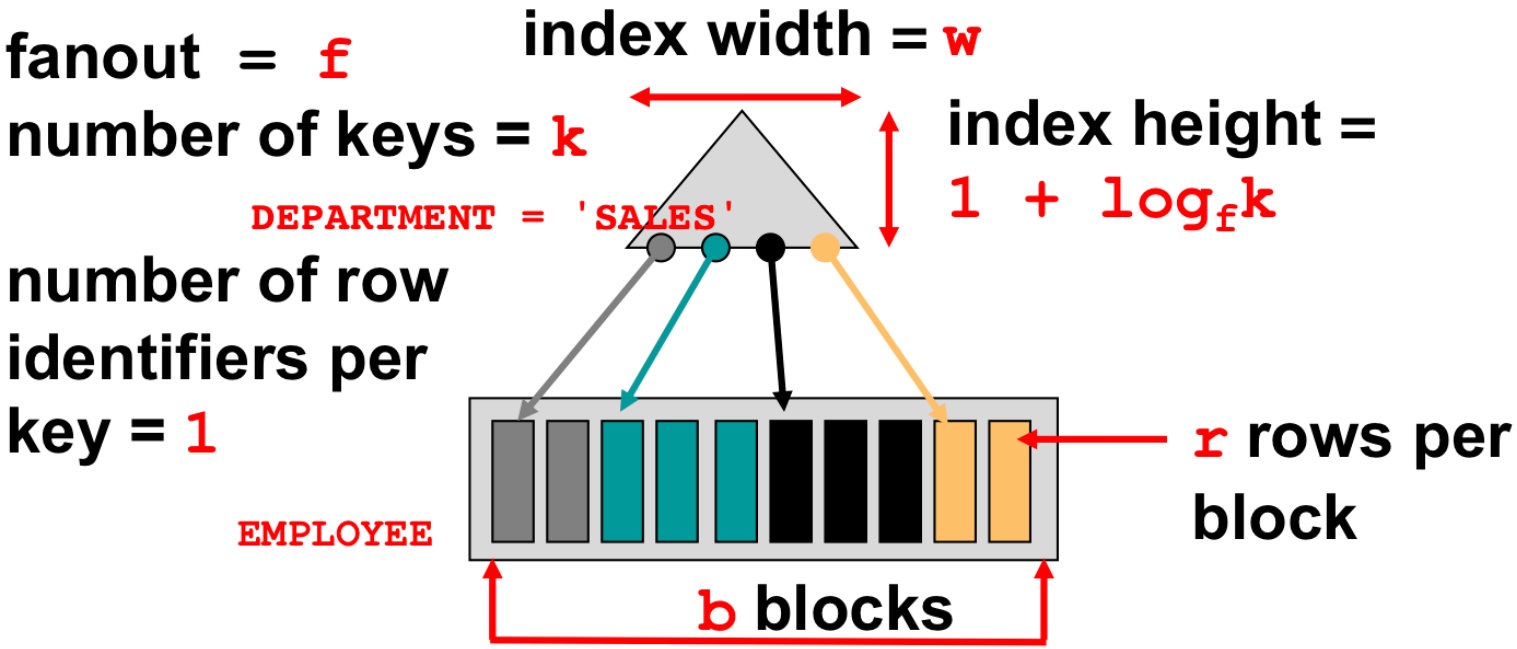
Storage organization and models

Tables with randomly ordered rows

Tables with clustered B*-tree index

Tables with not clustered B*-tree index

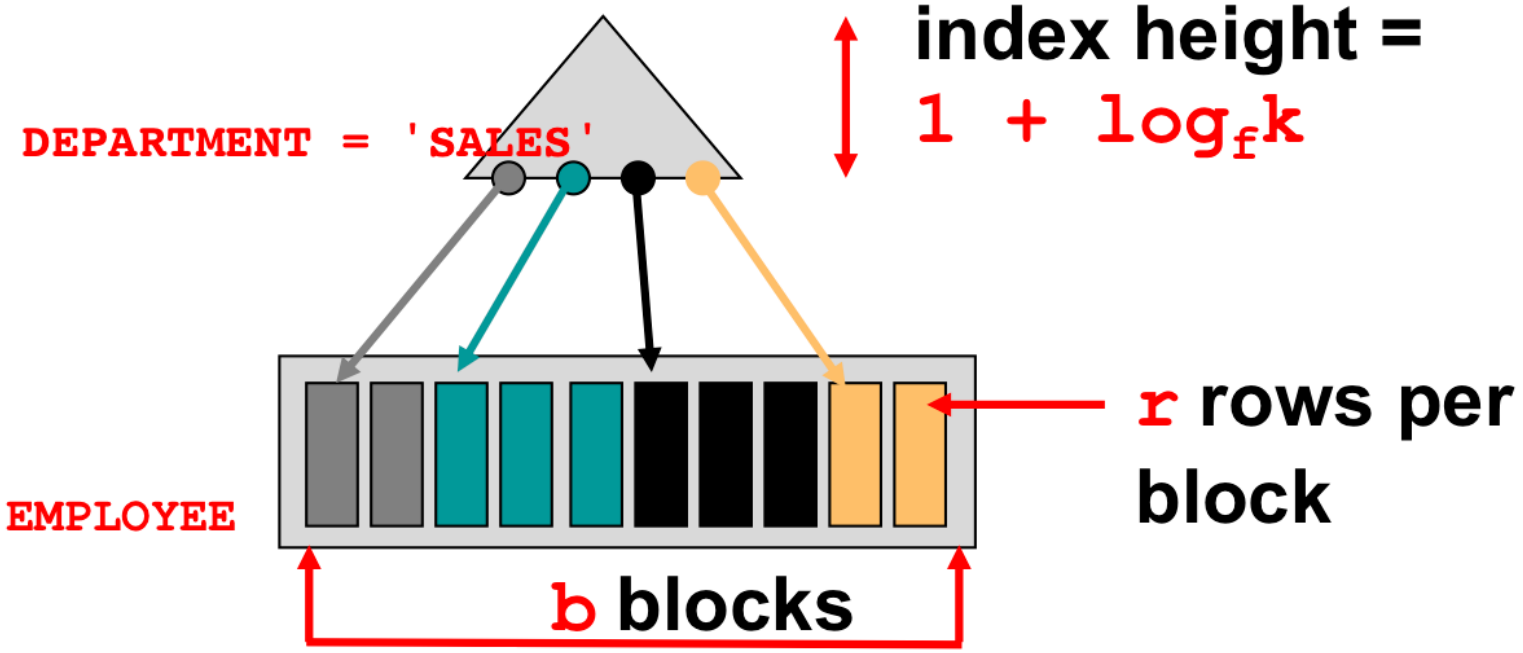
Table with clustered B* tree index



Average number of blocks that contain all rows with the same value of index key = b/k

Table with clustered B* tree index

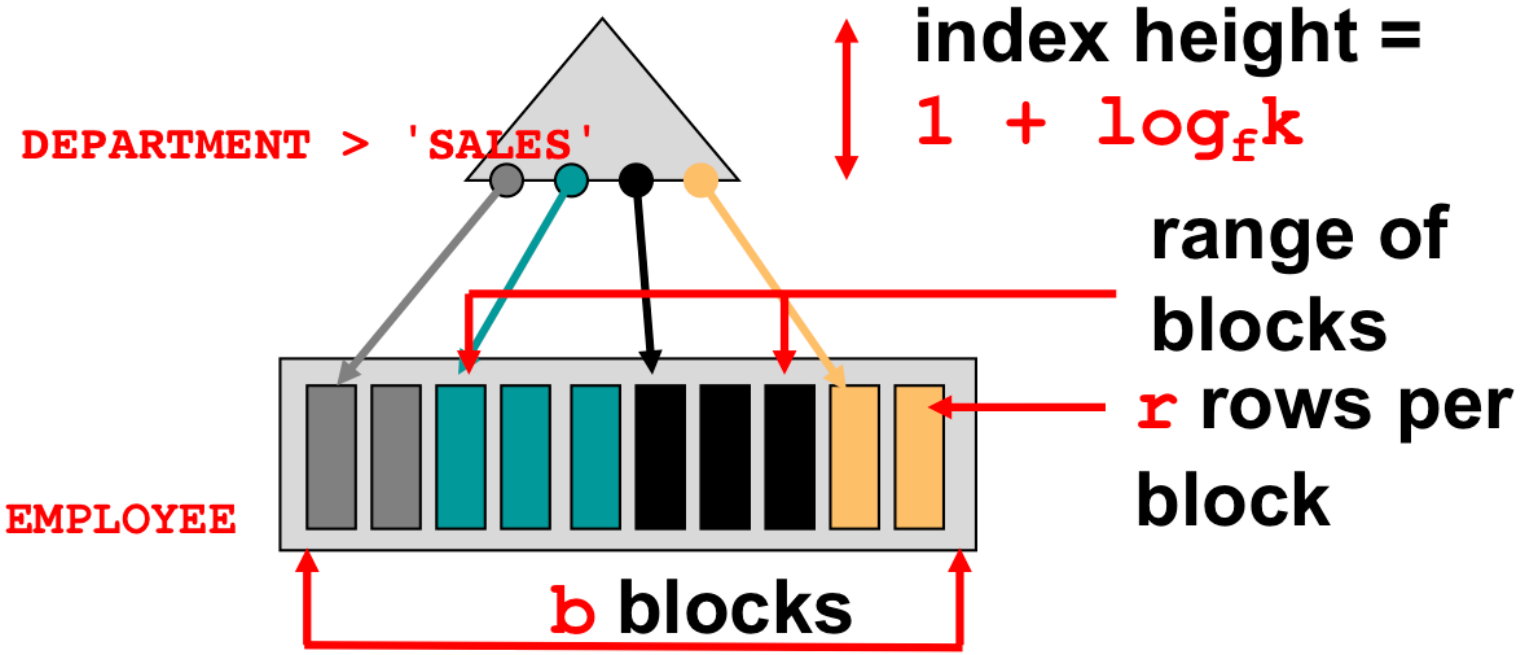
Full scan with an equality condition on a cluster key



Cost = $1+\log_f k + b/k$ read block operations

Table with clustered B* tree index

Full scan with a range condition on a cluster key where `n` is a number of cluster keys that satisfy the range condition



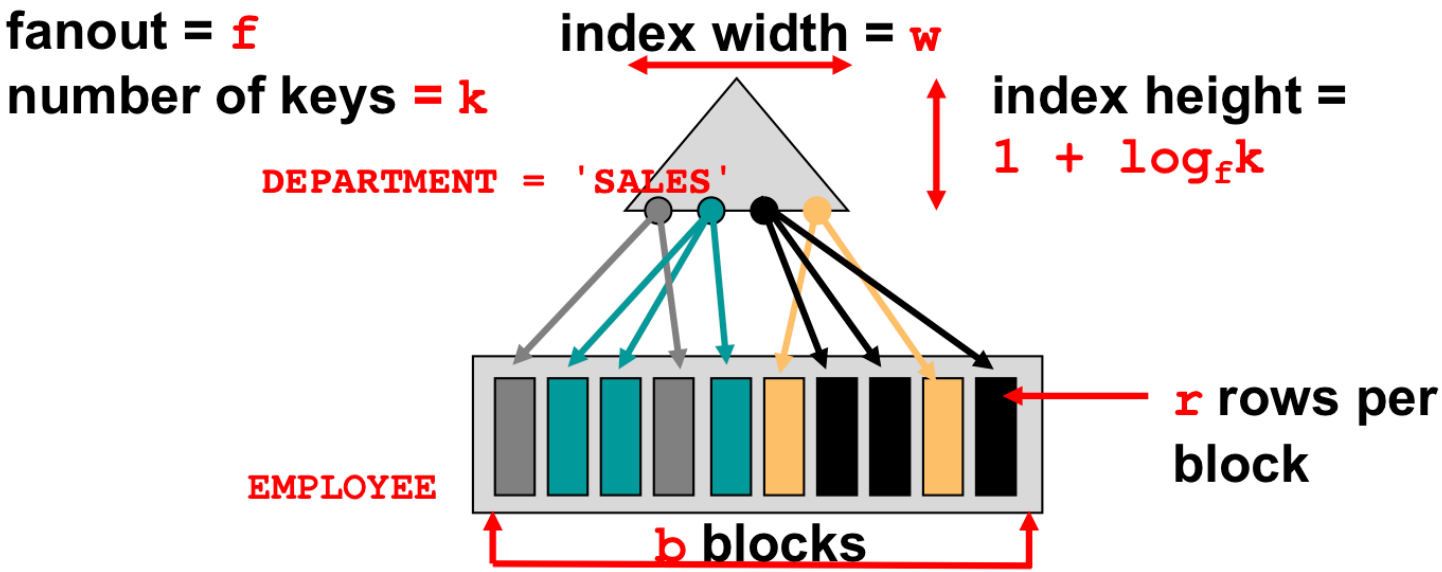
Cost = $1+\log_f k + n*b/k$ read block operations

Quantitative Analysis of Indexing

Outline

- Cost model ? What is it ?
- Storage organization and models
- Tables with randomly ordered rows
- Tables with clustered B*-tree index
- Tables with not clustered B*-tree index

Table with not clustered B* tree index



Total number of row identifiers per index key =

```
total number of rows / total number of keys = b*r/k
```

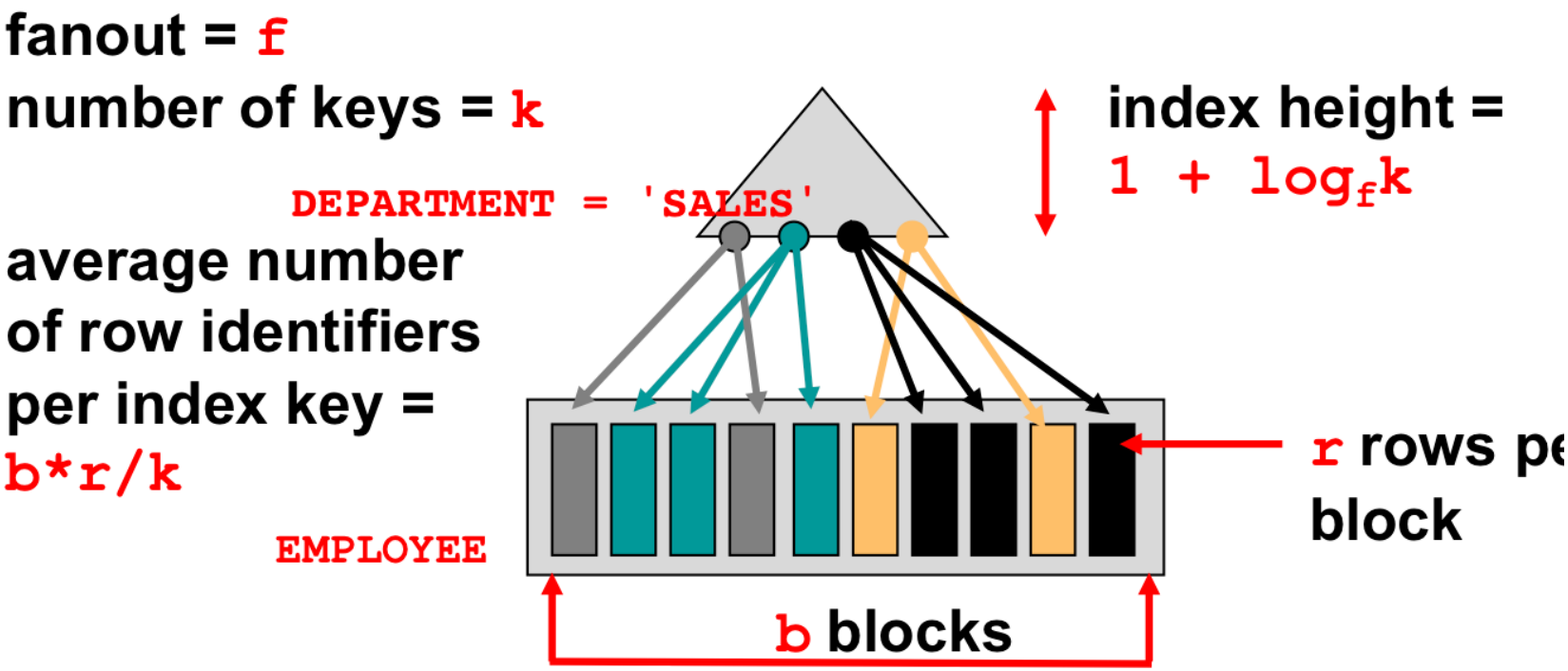
Average number of read block operations to read `b*r/k` rows =

```
(the worst case+the best case)/2 = (b*r/k +(b*r/k)/r)/2
```

```
=(b*r/k + b/k)/2 = b*(r+1)/(2*k) ≈ b*r/(2*k)
```

Table with not clustered B* tree index

Full scan with an equality condition on an index key

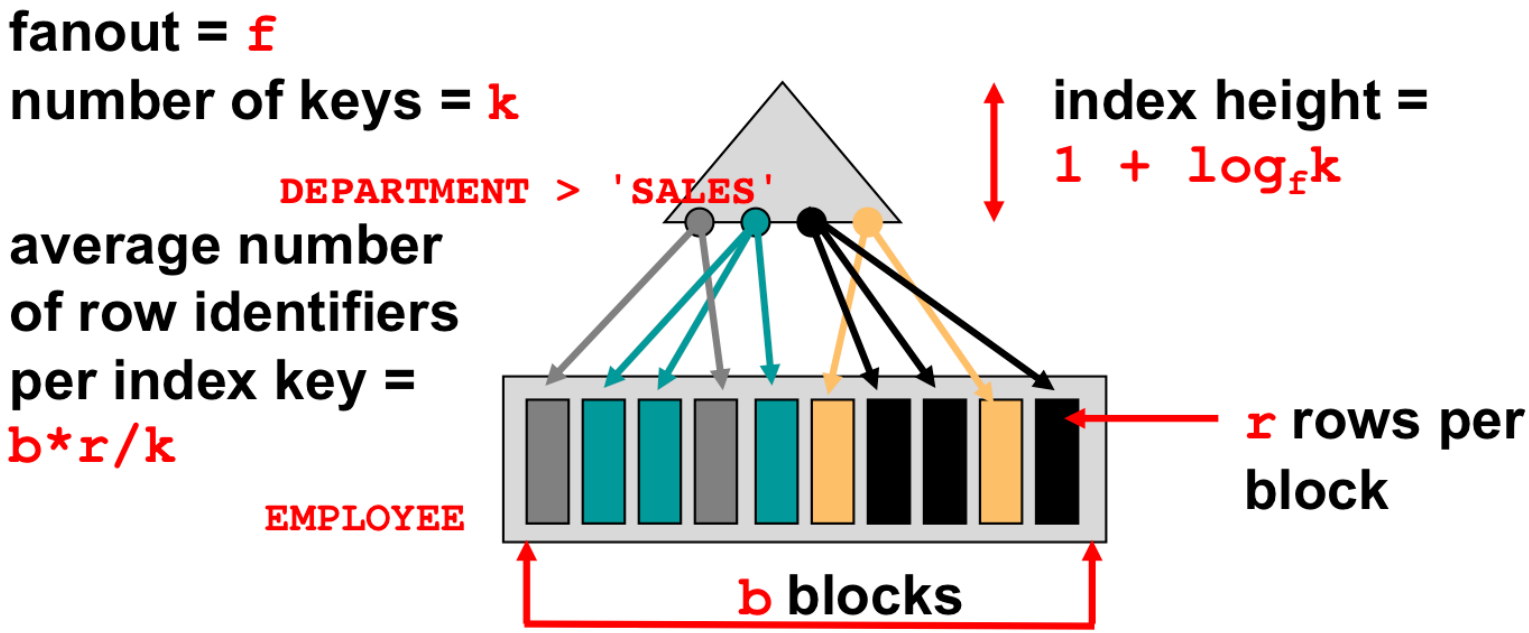


Cost = $t_{\text{traverse index}} + t_{\text{read data blocks}} =$

$1+\log_f k + (b*r/k + b/k)/2$

Table with not clustered B* tree index

Full scan with a range condition on an index key where n is a total number of index keys that satisfy a range condition



Cost = $t_{traverse\ index} + t_{read\ data\ blocks} =$

$1+\log_f k + n*((b*r/k + b/k)/2)$

References

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