How to determine which algorithm, index, data layout is better?
What we considered:

- amount of data read and written
- random I/O vs sequential
What we ignored:

DB buffer scheduling strategy
L1, L2, L<x> cache misses
different bandwidth among CPU, caches, and main memory
CPU costs
multiple cores
multithreading
What we ignored:

DB buffer scheduling strategy
L1, L2, L<x> cache misses
different bandwidth among CPU, caches, and main memory
CPU costs
multiple cores
multithreading
hyperthreading

\[ 8 = 4 + 4 \]

4 physical cores
4 hyperthread
What we ignored:

DB buffer scheduling strategy
L1, L2, L<x> cache misses
different bandwidth among CPU, caches, and main memory
CPU costs
multiple cores
multithreading
hyperthreading
scheduling effects
multiple disks/RAID level
data distributions/data skew
What we ignored:

DB buffer scheduling strategy
L1, L2, L\(<x>\) cache misses
different bandwidth among CPU, caches, and main memory
CPU costs
multiple cores
multithreading
hyperthreading
scheduling effects
multiple disks/RAID level
data distributions/data skew
workload distribution/workload skew

\[ \text{WHERE } A = x \]
What we ignored:

- DB buffer scheduling strategy
- L1, L2, L<x> cache misses
- Different bandwidth among CPU, caches, and main memory
- CPU costs
- Multiple cores
- Multithreading
- Hyperthreading
- Scheduling effects
- Multiple disks/RAID level
- Data distributions/data skew
- Workload distribution/workload skew
- SIMD

...
How to Measure Performance?

Three ways to measure the performance of a computer program:

1. Analytical Modelling,
How to Measure Performance?

Three ways to measure the performance of a computer program:

(1.) Analytical Modelling,

(2.) Simulation,
How to Measure Performance?

Three ways to measure the performance of a computer program:

1. Analytical Modelling,
2. Simulation,
3. Experiment.
(1a) Asymptotic Complexity, O-Notation

count each primitive operation as “1“
(arithmetic operations, node hop, memory accesses, page accesses, etc.)
(1a) Asymptotic Complexity, O-Notation

count each primitive operation as “1“ (arithmetic operations, node hop, memory accesses, page accesses, etc.)

determine asymptotic complexity of the index

\[ f(N) = N^2 + N \cdot (\log N) \]

\[ O(N^2) \]
(1a) Asymptotic Complexity, O-Notation

count each primitive operation as “1”
(arithmetic operations, node hop, memory accesses, page accesses, etc.)
determine asymptotic complexity of the index
worst, best and/or average case
example:

AVL-tree search has $O(\log n)$ average and worst case time.

\[
\begin{array}{c}
\text{Model} \\
\text{Time}
\end{array} \neq \begin{array}{c}
\text{Reality} \\
\text{Time}
\end{array}
\]
(1a) Asymptotic Complexity, O-Notation

count each primitive operation as “1”
(arithmetic operations, node hop, memory accesses, page accesses, etc.)

determine asymptotic complexity of the index

worst, best and/or average case

example:

    AVL-tree search has $O(\log n)$ average and worst case time

problem:

    completely ignores memory hierarchy
(1a) Asymptotic Complexity, $O$-Notation

$\sum_{k=0}^{N} k = O(N^2)$

$\sum_{k=0}^{N} 1 = O(N)$
Scenario
(1b.1) Cost Models

also consider number of random I/Os

example:

    count "100" for each random read or write of a page
    count "1" for each sequential read or write of a page

problem:

    still ignores CPU time and cache misses...
(1b.1) Cost Models
(1b.2) Cost Models

count number of cache misses as well

example:

CSS-tree has O(\log_m n) cache misses (m=fan-out)

\[ \rightarrow \beta \text{-tree in main memory} \]
(1b.2) Cost Models

count number of cache misses as well

example:

CSS-tree has $O(\log_m n)$ cache misses ($m=\text{fan-out}$)

problem:

ignores CPU time...
(1b.2) Cost Models
(1b.2) Cost Models
strong winds

sweat, dehydration

painful

worker has a bad day

easy until here
Simulation

run part of the actual algorithm or system

simulate other parts

problem:

might still oversimplify reality

might miss some important effect from reality...
strong winds

sweat, dehydration

worker has a bad day

easy till here
strong winds

sweat, dehydration

worker has a bad day

easy till here

painful
(3) Experiment

implement it
run it
measure it

problems:

lacking abstraction → over-fitting
(3) Experiment

implement it
run it
measure it

problems:
- lacking abstraction
- lacking bounds
- lacking theory
  (lacking insight)
<table>
<thead>
<tr>
<th>method</th>
<th>effort/cost</th>
<th>reality</th>
<th>generalizability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analytical Modelling</td>
<td></td>
<td>![Image]</td>
<td></td>
</tr>
<tr>
<td>Simulation</td>
<td></td>
<td>![Image]</td>
<td></td>
</tr>
<tr>
<td>Experiment</td>
<td></td>
<td>![Image]</td>
<td></td>
</tr>
</tbody>
</table>
Copyrights and Credits

© iStock.com:
Freerick_k

CC:
Ricardo Liberato
http://creativecommons.org/licenses/by-sa/2.0/legalcode

iconshock
http://commons.wikimedia.org/wiki/File:Desk_lamp_icon.png
http://creativecommons.org/licenses/by-sa/3.0/legalcode

User Smial on de.wikipedia
http://commons.wikimedia.org/wiki/
File:Luefter_y.s.tech_pd1270153b-2f.jpg
http://creativecommons.org/licenses/by-sa/2.0/de/legalcode

other:
http://openclipart.org/image/800px/svg_to_png/26274/Anonymous_Right_Footprint.png

http://openclipart.org/detail/26217/left-footprint-by-anonymous

http://openclipart.org/detail/22012/weather-symbols:-sun-by-nicubunu

public domain:
http://commons.wikimedia.org/wiki/
File:The_Blue_Marble.jpg