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 hyperthreaded
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

\[
\begin{align*}
\text{uniform} & \rightarrow \frac{4}{2} \text{sec} \\
\text{skewed} & \rightarrow \eta \geq 0 \text{ sec}
\end{align*}
\]
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

[Diagram]\(0 \leq A < 100\)

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,

\[ \Theta \text{- Notation} \rightarrow \text{Mathematical Model} \]
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.
count each primitive operation as “1”
(arithmetic operations, node hop, memory accesses, page accesses, etc.)
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

\[
\begin{array}{c}
\text{model} \\
time \\
\hline
\text{reality} \\
time \neq time
\end{array}
\]
(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

\[ T \leq 5 \text{ steps} \]
\[ f(n) = n \]
\[ O(n) \]
Scenario

Explosive

Painful

Easy until here

Chaos
(1b.1) Cost Models

also consider number of random I/Os

eexample:

  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)

$\rightarrow$ B-tree in main memory
(1b.2) Cost Models

count number of cache misses as well

eample:

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

problem:

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

[Diagram of a cost model]
(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

outfit influence

painful
strong winds

sweat, dehydration

worker has a bad day

easy till here

painful
strong winds

worker has a bad day

sweat, dehydration

painful

easy till here
(3) Experiment

implement it
run it
measure it

problems:

lacking abstraction \rightarrow over fitting
(3) Experiment

implement it
run it
measure it

problems:

lacking abstraction
lacking bounds
lacking theory
(lacking insight)
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