scans vs indexes

prof. Stratos Idreos

HTTP://DASLAB.SEAS.HARVARD.EDU/CLASSES/CS165/
midterm reviewing next week
project testing
automated testing & leaderboard biweekly
your tests
SQL tests as of M3

work with the TFs to stabilize testing procedures
clustered (all columns)

secondary indexes
subset of columns
**b-tree** - dynamic tree - always balanced
milestone 2

cache conscious b-tree-like index

node design, fill factor, etc

most differences with classic design you end up doing for the leaves

contiguous vs not, tuples vs single values, fill factor, need to maintain alignment info with the rest of the columns

insert & select now, updates later
declarative interface
ask what you want

indexes/views/tuning knobs

db system
select ... from R where A<v and ....

(secondary) **index vs scan**: the eternal battle
design/implement numerous possible algorithms + data representations

choose the best data source, algorithms and path for each query
random access to traverse the tree & need to sort result

sequential access pattern but needs to access all data
a query that select on A and then needs B

intermediate out of order
A secondary index on A values out of order with base data.

A query that selects on A and then needs B

Intermediate out of order.
secondary index on A
values out of order
with base data

a query that select on A and then needs B

intermediate out of order
covering indices

no need to go to base data but…
The diagram shows the relationship between response time, selectivity, and the percentage of values that qualify as a function of selectivity. There is a turning point where the performance changes from index access to scan access. The graph indicates that as selectivity increases, the response time decreases for index access and increases for scan access, with a turning point where the two methods' performances intersect.

% of values that qualify
the standard solution
1) maintain statistics,
2) optimizer chooses access path depending on estimated selectivity

what is wrong with that?
Motivation TPCH (SF10) 2/2

Original

Tuned
can we just recompute the statistics?

basic stats
- Full Scan
- Index Scan
- Optimizer decision
- Avg. statistics collection

per column

for pair

Execution time (sec)

Result selectivity (%)
2012, somewhere in Germany

if I keep 30 data systems researchers “trapped” in a castle for a week we might be able to define “robust query processing” and find a few solutions :)

CS165, Fall 2015
Stratos Idreos
robust query processing (best definition to date by Goetz)

graceful degradation when the environment changes
can we avoid bad access path selection even when we have wrong statistics?
SWITCH SCAN
while index probing
switch to scan
if cardinality > estimation

SMOOTH SCAN
goal avoid performance cliff
close to optimal

good: avoids worst case
bad: performance cliff
smooth scan
gradually morph from index scan to full scan

for each qualifying tupleID
- **mode1** fetch the respective page and get the row
- **mode2** check all tuples in a fetched page
- **mode3** fetch and check adjacent pages as well
- **mode3+** increase # of pages fetched
some design points

tuple cache to avoid producing the same tuple twice
page cache to avoid reading the same page twice
result cache to produce result in indexed order
when to morph
in order to achieve a smooth behavior

**optimizer** start when selectivity > estimation

**SLA** respect an upper threshold

**selectivity** morph when selectivity increases by z

**pessimistic** morph with every new probe
TPC-H Query

Execution time (sec)

- Q1 (98%)
- Q4 (<1%)
- Q6 (2%)
- Q14 (<1%)

PostgreSQL
PostgreSQL with Smooth Scan
random access & page-based access

need to only read $x$... but have to read all of page 1

data value $x$

page1  page2  page3  ...
**select** min(A) **from** R **where** B<10 and C<80
Efficient mid-query re-optimization of sub-optimal query execution plans
Navin Kabra and David DeWitt
ACM SIGMOD International Conference on Management of Data, 1998

Smooth Scan: Statistics-Oblivious Access Paths
Renata Borovica, Stratos Idreos, Anastasia Ailamaki, Marcin Zukowski and Campbell Fraser
IEEE International Conference on Data Engineering (ICDE), 2015

next: fast scans
scans vs indexes

DATA SYSTEMS

prof. Stratos Idreos

HARVARD
School of Engineering and Applied Sciences