db architectures
+ column-stores

BIG DATA SYSTEMS

prof. Stratos Idreos
plan for today

taking cs265 without cs165

class structure + logistics

db architecture + column-stores

project examples

first batch of papers
who can take the class

graduate students

senior undergrads

anyone with strong systems experience (cs165/cs161)
two student groups
with and without systems experience

we will adjust projects and grading accordingly

workload and goals
this is not an easy A class…

ideal result:
research & publication for systems people
strong introduction to systems design for non systems people
TFs
Manos Athanassoulis (as of Class 3)

+ 1 more if we are more than 10-15 students
what is this class about

about how to design data systems
(mongodb, hadoop, mySQL, db2, …)

not about how to use them
background reading (4 papers)

db architectures
column-stores
map-reduce
b-trees

extra: cs165 material

we will do extra meetings to review this material
first meeting in two weeks
declarative interface
ask what you want

↓

db system

↑

the db system decides how to “best store and access data”
select min(A) from R where B<10 and C<80


Diagram of a database kernel with client programs and SQL client programs. The database kernel includes a thread pool with threads 1 to 4. Applications are connected to client programs via SQL.
select name
from student
where GPA > 3.0

logical plan
give me all students enrolled in cs265

\textbf{select} student.name \\
\textbf{from} students, enrolled, courses \\
\textbf{where} courses.name="cs265" \\
and enrolled.courseld=course.id \\
and student.id=enrolled.studentld
applications

cpu - cpu - cpu - cpu

cpu registers

caches

memory

disk - disk - disk - ssd

system where db runs

sql

smaller/faster
100K  
- disk

100  
- memory

10  
- on board cache

2  
- on chip cache

1  
- registers

Pluto  
- 2 years

New York  
- 1.5 hr

this building  
- 10 min

this room  
- 1 min

my head
sequential vs random access
the way we store data defines the possible (efficient) access methods
row-store

file

one page contains all fields of multiple attributes

stored continuously

select A, B, C, D

select A
row-store

A B C D

stored continuously

column-store

A B C D

one page contains fields of a single attribute

select A, B, C, D

select A
virtual ids/ positional alignment

columns do not need to have the same width

positional lookups/joins

\[ A(i) = A + i \times \text{width}(A) \]
select min(C) from R where A<10 & B<20
possible data flow patterns
tuple at a time
block/vector at a time
bulk processing
systems project example

adaptive min-max indexes
next class:
9 slots: who is presenting?

1. Fast scans (SIMD, multicores) Yihe/Sam
2. Google megastore & spanner (distributed querying) Ore/Archie
3. H20 (adaptive systems) Wasay-Stella
4. SQLonHadoop Jennie/Alex
5. ADS (time series - adaptive indexing) Michael
6. SharedDB/CooperativeScans Wilson
7. Polybase (multistores) Tarik/Lukas
8. ART (adaptive radix tree) Mike/Sierra
9. BlinkDB & Researcher’s guide vision (data exploration) Yifan/Kristen
how to prepare for presentations

understand the paper
(you might have to read a couple of the cited papers)

discuss:
what is the problem?
why is it important?
why it is not easy to solve?
what is the main intuition of the proposed solution?
what are the technical challenges?
give examples
db architectures
+ column-stores

BIG DATA SYSTEMS

prof. Stratos Idreos