CS 525: Advanced Database Organization

01: Introduction

Boris Glavic

Slides: adapted from a course taught by Hector Garcia-Molina, Stanford InfoLab

Advanced Database Organization?

- =Database Implementation
- =How to implement a database system
- … and have fun doing it ;-)

Isn’t Implementing a Database System Simple?

Relations → Statements → Results

Introducing the

MEGATRON 3000

Database Management System

- The latest from Megatron Labs
- Incorporates latest relational technology
- UNIX compatible

Megatron 3000
Implementation Details

! First sign non-disclosure agreement !

Megatron 3000
Implementation Details

- Relations stored in files (ASCII)
  e.g., relation R is in /usr/db/R

```
Smith # 123 # CS
Jones # 522 # EE
   :
   :
```

Megatron 3000

Implementation Details

- Directory file (ASCII) in /usr/db/directory

```
R1 # A # INT # B # STR …
R2 # C # STR # A # INT …
   ⋮
```

Megatron 3000
Sample Sessions

```
% MEGATRON3000
   Welcome to MEGATRON 3000!
&
    .
    .
    .

& quit
%
```

Megatron 3000
Sample Sessions

```
& select *
  from R #

    Relation R
   A        B    C
   SMITH   123   CS


&
```

Megatron 3000

Sample Sessions

```
& select A,B
  from R,S
  where R.A = S.A and S.C > 100 #

   A    B
  123  CAR
  522  CAT


&
```

Megatron 3000
Sample Sessions

```
& select *
  from R | LPR #
&
```

Result sent to LPR (printer).

Megatron 3000
Sample Sessions

```
& select *
  from R
  where R.A < 100 | T #
&
```

New relation T created.

Megatron 3000

- To execute "**`select * from R where`** ***`condition`***":
  - (1) Read dictionary to get R attributes
  - (2) Read R file, for each line:
    - (a) Check condition
    - (b) If OK, display

Megatron 3000

- To execute "**`select * from R`**
  `where` ***`condition`*** **`| T`**":

  (1) Process select as before

  (2) Write results to new file T

  (3) Append new line to dictionary

Megatron 3000

- To execute “**select A,B from R,S where *condition***”:
    - (1) Read dictionary to get R,S attributes
    - (2) Read R file, for each line:
        - (a) Read S file, for each line:
            - (i) Create join tuple
            - (ii) Check condition
            - (iii) Display if OK

What' s wrong with the Megatron 3000 DBMS?

What’s wrong with the Megatron 3000 DBMS?

- Tuple layout on disk

e.g., - Change string from ‘Cat’ to ‘Cats’ and we have to rewrite file

- ASCII storage is expensive
- Deletions are expensive

What' s wrong with the Megatron 3000 DBMS?

- Search expensive; no indexes

e.g., - Cannot find tuple with given key quickly

- Always have to read full relation

What’s wrong with the Megatron 3000 DBMS?

- Brute force query processing

e.g.,

```
select *
from R,S
where R.A = S.A and S.B > 1000
```

- Do select first?
- More efficient join?

What’ s wrong with the Megatron 3000 DBMS?

- No buffer manager

e.g., Need caching

What’s wrong with the Megatron 3000 DBMS?

- No concurrency control

What’s wrong with the Megatron 3000 DBMS?

- No reliability

e.g., - Can lose data

- Can leave operations half done

What’s wrong with the Megatron 3000 DBMS?

- No security

e.g., - File system insecure

- File system security is coarse

What' s wrong with the Megatron 3000 DBMS?

- No application program interface (API)

e.g., How can a payroll program get at the data?

What’s wrong with the Megatron 3000 DBMS?

- Cannot interact with other DBMSs.

What’ s wrong with the Megatron 3000 DBMS?

- Poor dictionary facilities

What’s wrong with the Megatron 3000 DBMS?

- No GUI

What's wrong with the Megatron 3000 DBMS?

- Lousy salesman!!

Course Overview

- File & System Structure

  Records in blocks, dictionary, buffer management,…

- Indexing & Hashing

  B-Trees, hashing,…

- Query Processing

  Query costs, join strategies,…

- Crash Recovery

  Failures, stable storage,…

Course Overview

- Concurrency Control

  Correctness, locks,…

- Transaction Processing

  Logs, deadlocks,…

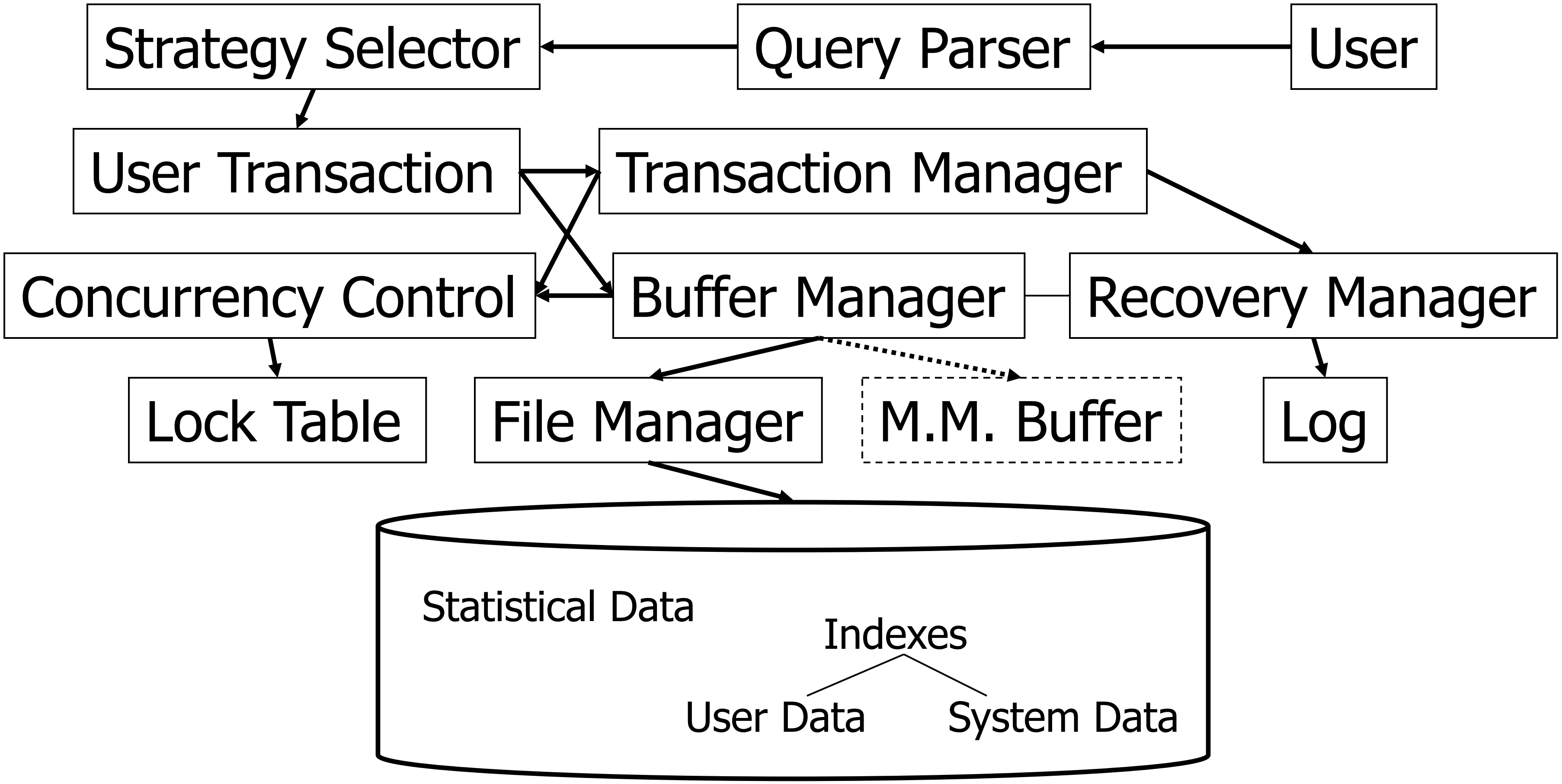
- Security & Integrity

  Authorization, encryption,…

- Advanced Topics

  Distribution, More Fancy Optimizations, …

System Structure



Some Terms

- Database system
- Transaction processing system
- File access system
- Information retrieval system

Course Information

- **Webpage**: http://www.cs.iit.edu/~cs525/
- **Instructor**: Boris Glavic
  - http://www.cs.iit.edu/~glavic/
  - **DBGroup:** http://www.cs.iit.edu/~dbgroup/
  - **Office Hours: Thurdays, 1pm-2pm**
  - **Office:** Stuart Building, Room 226 C
- **TA:** Ma Di (dma2@hawk.iit.edu)
- **Time:** Mon + Wed 3:15pm – 4:30pm

Google Group

- https://groups.google.com/forum/#!forum/cs525-2013-spring-group
- Mailing-list for annoucements
- Discussion forum
  - Student - Instructor/TA
  - Student – Student
- ->please accept invite to keep up to date

Workload and Grading

- Schedule and Important Dates
  - On webpage & updated there
- Programming Assignments (50%)
  - 4 Assignments
  - Groups of 3 students
  - Plagiarism -> 0 points and administrative action
- Quizzes (10%)
- Mid Term (20%) and Final Exam (20%)

Textbooks

- Elmasri and Navathe , **Fundamentals of Database Systems**, 6th Edition , Addison-Wesley , 2003
- Garcia-Molina, Ullman, and Widom, **Database Systems: The Complete Book**, 2nd Edition, Prentice Hall, 2008
- Ramakrishnan and Gehrke , **Database Management Systems**, 3nd Edition , McGraw-Hill , 2002
- Silberschatz, Korth, and Sudarshan , **Database System Concepts**, 6th Edition , McGraw Hill , 2010

Programming Assignments

- 4 assignments one on-top of the other
- Optional 5th assignment for extra credit
- Code has to compile & run on server account
  - **Email-ID**@fourier.cs.iit.edu
  - Linux machine
  - SSH with X-forwarding
- Source code managed in **git** repository on Bitbucket.org
  - Handing in assignments = commit to repository
  - One repository per student
  - You should have gotten an invitation (if not, contact me/TA)
  - Git tutorials linked on course webpage!

Next time:

- Hardware