CompSci 590.6

Understanding Data:

Theory and Applications

Lecture 1

Introduction and Review of SQL

Instructor: Sudeepa Roy

Email: *sudeepa@cs.duke.edu*

Class Information

- https://www.cs.duke.edu/courses/fall15/compsci590.6/

- Class meets twice a week
  - LSRC A247
  - Tuesday-Thursday 11:45 am – 1:00 pm

- Class mailing list
  - compsci590.6@cs.duke.edu
  - Please send me an email if you are not taking the course, but would like to receive emails

- Instructor
  - SUDEEPA ROY
  - Homepage: http://www.cs.duke.edu/~sudeepa/
  - Email: sudeepa@cs.duke.edu
  - Office Hour: Tuesdays 2:00-2:50 pm LSRC D124 (Temporary), and by appointments

Instructor

- SUDEEPA ROY
  - Assistant Professor at Duke CS from Fall 2015
  - Postdoc at the Univ. of Washington, CSE (Seattle, WA)
  - Ph.D. from the Univ. of Pennsylvania, CIS (Philadelphia, PA)

- Research Interests
  - Databases (broadly)
  - Theoretical + Applied problems
  - Current research focus: Causality/Explanations for Databases
    - a new approach for deep analysis of data and query answers
  - Other research:
    - Provenance
    - Probabilistic Databases
    - Crowd-sourcing
    - Information Extraction

Why should we care about “Understanding Data”?

Big Data is changing the world

Technology

Science

Business

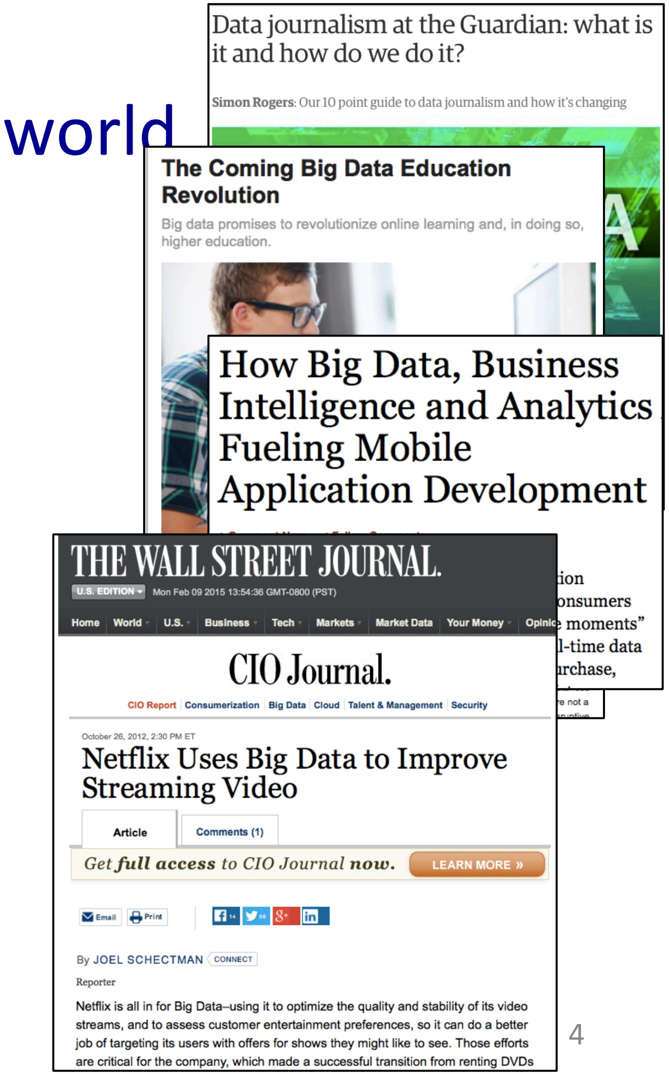
Manufacturing

Healthcare

Education

Journalism

Government



Data journalism at the Guardian: what is it and how do we do it?

Simon Rogers: Our 10 point guide to data journalism and how it's changing

The Coming Big Data Education Revolution

Big data promises to revolutionize online learning and, in doing so, higher education.

How Big Data, Business Intelligence and Analytics Fueling Mobile Application Development

ion
onsumers
moments”
l-time data
rchase,

THE WALL STREET JOURNAL.

U.S. EDITION Mon Feb 09 2015 13:54:36 GMT-0800 (PST)

Home | World | U.S. | Business | Tech | Markets | Market Data | Your Money | Opinio

CIO Journal.

CIO Report | Consumerization | Big Data | Cloud | Talent & Management | Security

October 26, 2012, 2:30 PM ET

Netflix Uses Big Data to Improve Streaming Video

Article | Comments (1)

*Get **full access** to CIO Journal **now.*** LEARN MORE »

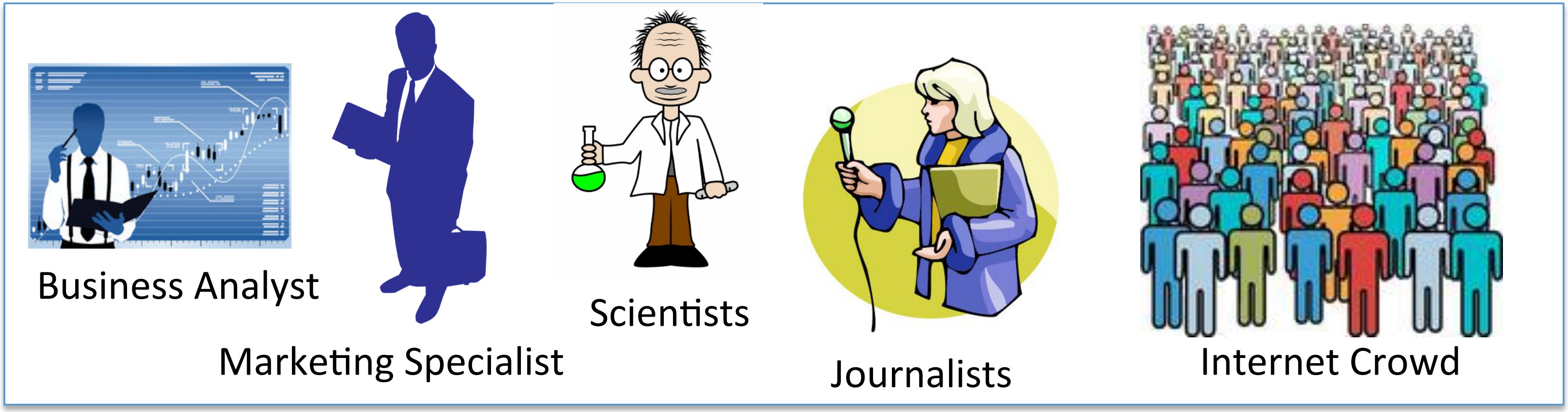
Email | Print

By JOEL SCHECTMAN CONNECT

Reporter

Netflix is all in for Big Data–using it to optimize the quality and stability of its video streams, and to assess customer entertainment preferences, so it can do a better job of targeting its users with offers for shows they might like to see. Those efforts are critical for the company, which made a successful transition from renting DVDs

More people are working with Data



Business Analyst

Marketing Specialist

Scientists

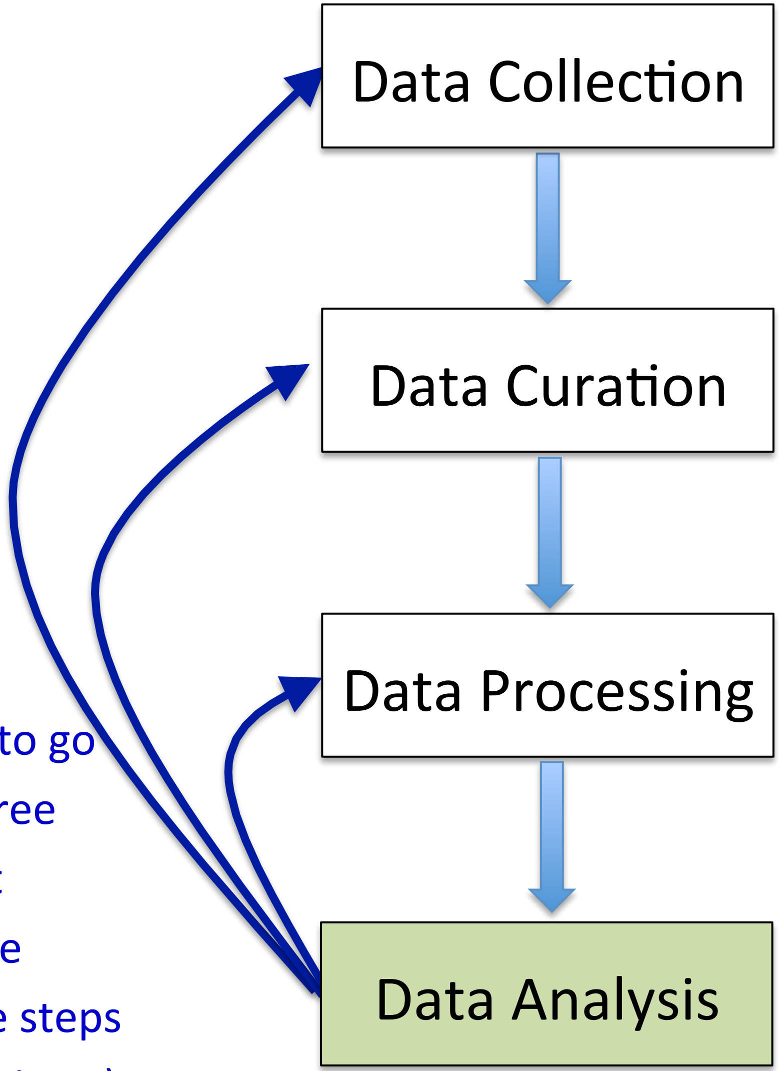
Journalists

Internet Crowd

Today's data analysts and consumers

Need new tools to handle, analyze, and understand data

Data Analysis Pipeline



- Acquire data from different sources

- Clean/Format
- Integrate with other datasets
- Store in database

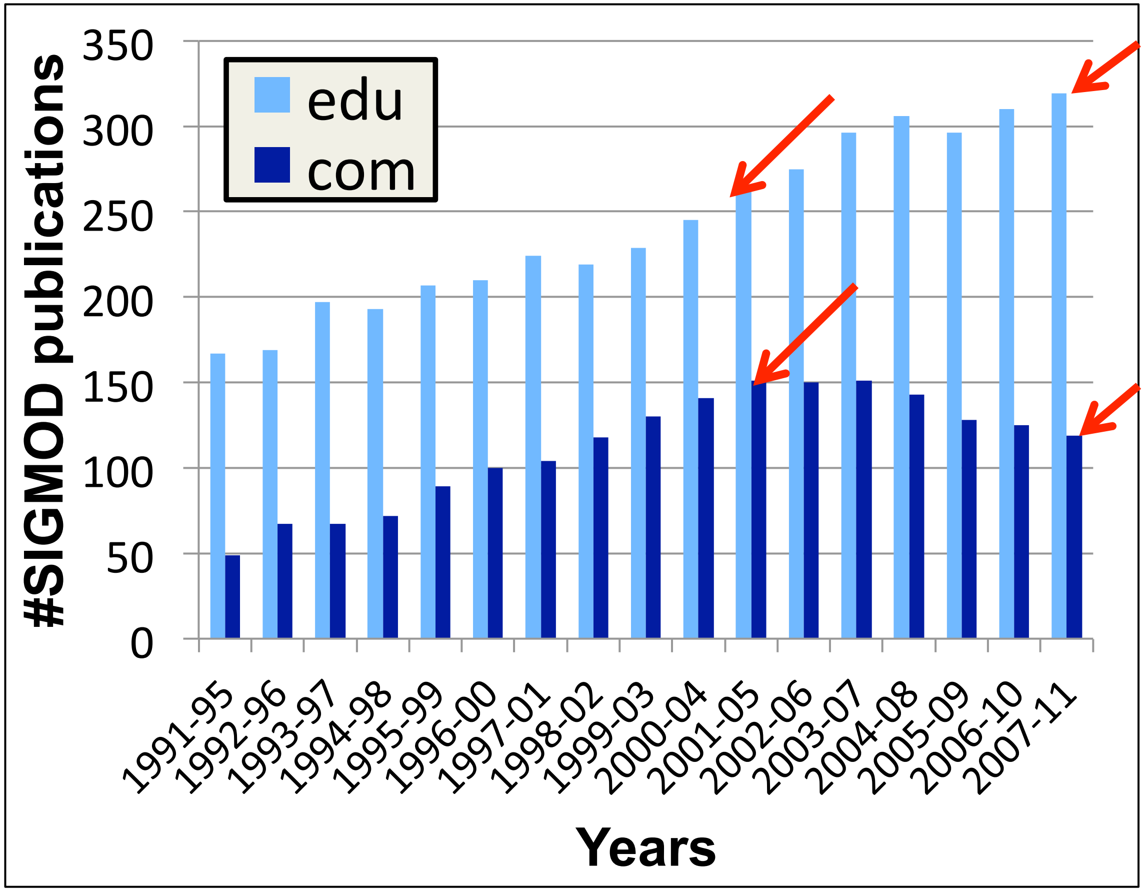
- Run queries (aggregate)
- Plot graphs

- Examine trends and anomalies
- Understand results

This course mostly focuses on the last step, but you need to go through the first three steps at some point anyway (e.g. you are likely to do all these steps in your research projects)

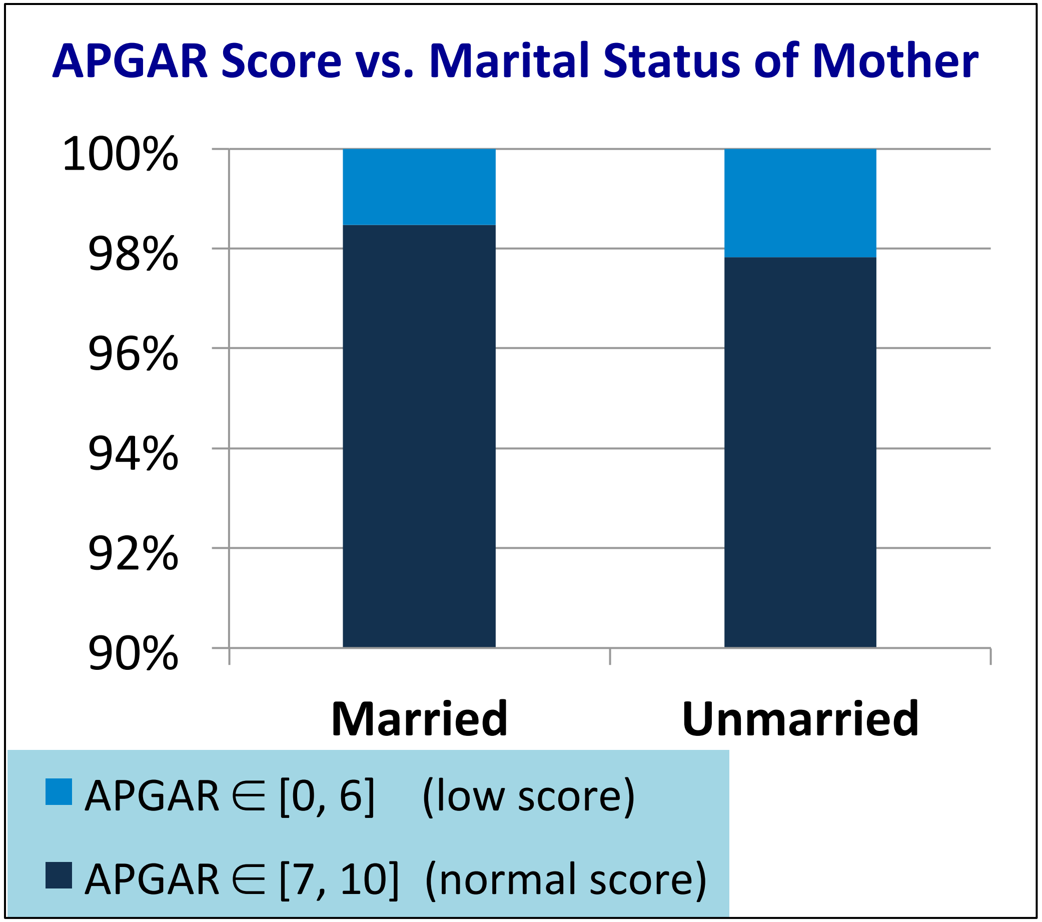
Example Data Analysis Question: Publication Data (DBLP)

Potential question from funding agencies, policy makers in industrial research



A peak for industry around 2000.
An increasing trend for academia. Explain why.

Example Data Analysis Question: Birth Records (Natality)



- Do different attributes of the mother (marital status, education, smoking habit) have a “causal effect” on the health of the baby?

Natality Dataset 2010 (from CDC/NCHS)

Single table, 233 attributes, ~4M entries, 2.89GB

Course Goals

1. Learn about research on data analysis in the database community

2. Practice reading, understanding, and presenting research papers

3. Have experience in doing research in data analysis

A Quick Survey

- Introduce yourself (name – UGRAD/GRAD/year – Main research area/ research ineterest – credit/audit)
- Have you taken an undergrad database course earlier (CS 316/eq.)?
- Have you taken a grad database course earlier (CS 516/eq.)?
- Are you familiar with
  - SQL?
  - RA? (σ, Π, ×, ⋈, ρ, ∪, ∩, -)
  - Index in databases?
  - Probability (conditional probability, Chernoff/union bounds)?
  - Polynomial time, NP-Hardness?
  - Logic: ∧, ∨, ∀, ∃, ¬, ∈
- Have you ever worked with a dataset? (relational database, text, csv, XML)
- Have you ever used a database system? (PostGres, MySQL, SQL Server, SQL Azure, ..)

Topics

- We will cover the seminal and state-of-the-art research papers on the following main topics

- Classical topics
  - OLAP Cube, data warehousing, data mining
  - Efficient exploration of multi-dimensional data
- Provenance
  - Origin of database query answers (or missing answers) in terms of input data and query
- Uncertain Data
  - Probabilistic, incomplete, and inconsistent data
- Causality and Explanations
  - Finding "causes" (more than association/correlation) and answering "why" questions in statistics, AI, and databases
- Data Analysis with Humans
  - crowdsourcing and usability
- Data Analysis Systems
  - ML, Large scale analytics, and visualization

Lectures

- Format
  - Presentation and discussions
  - Each of you will present one paper, I will present the rest
  - One main paper in depth in every class, sometimes one additional paper
  - Everyone should participate in the discussion

Grading

- Research Project: 50%
- Paper Review: 25%
- Class Participation: 10%
- Paper Presentation: 15%

Research Project

- In groups of 1 or 2
  - Form your groups soon
- I'll send out some project ideas by next Tuesday
  - Including instructions to install a DBMS and pointers to some datasets
- Start thinking about your project
  - On any topic in the course that interests you
  - Any data-oriented problem that you have faced in your own research (theory or implementation)
  - A new tool for data analysis for a specific application
    - UI, backend, algorithms, optimizations – all matter
    - Take a look at the demonstration track papers in recent SIGMOD/VLDB
  - The project cannot be
    - a survey
    - an implementation of known algorithms from the literature

      UNLESS (1) the problem is very important, and (2) you can achieve better performance and efficiency

More on Research Project

- The topic should be interesting, novel, and challenging
- Each group will try to meet me once in every two weeks to discuss progress
  - Let me know by email the weekly day/time(s) when your group can meet
- Your aim should be to have a research paper or a demonstration proposal (or both!) ready for submission at the end of the semester
  - VLDB/SIGMOD/ICDE – Research Track (for theory + experiments)
  - VLDB/SIGMOD/ICDE – Demonstration Track (prototype)
  - PODS/ICDT (for deep theoretical results)
  - the best conferences in your own research area
- Even if you are working on a theoretical problem, actually exploring datasets for finding interesting observations to analyze, and evaluating your algorithms on the dataset will be useful
- However, it is a "Research" project, so the outcome is not always predictable. Your approach, level of effort, and partial results will matter.

Research Project Deliverables

1. A project proposal (due: 9/8)

2. A midterm progress report (due: 10/15)

3. A final project report (due: 11/24)

You will update the same document

4. A final project presentation and/or demonstration (on: 11/24)

1. Project Proposal

- 3-4 pages, write the following sections (I will send you a blank latex format)
- Introduction
  - write a short introduction
  - motivation, why the problem is important
- Related Work
  - do a thorough survey of related work
- Proposed Research
  - write directions of proposed research, why they will be important + novel + challenging, and a tentative timeline
  - the timeline is for our convenience, it is ok if you do not finish everything or stick to the timeline (that is research!), but you should aim for more
  - how you are going to do these, what datasets and tools you will use
- PostGres, MySQL, and many visualization tools are free
- If you absolutely need any additional resources that you cannot find for free online, contact me

2. Midterm Progress Report

- 4-6 pages, **add** the following sections

- Preliminaries
  - Concepts, Problem Definition

- Current Status
  - What interesting observations you have by exploring datasets
  - What partial results you have obtained
  - How you are going to finish your project
  - What new directions you will explore

3. Final Report

- 8-12 pages
- **Update** Introduction, Related Work, Preliminaries
- **Remove** Current Status and Proposed Research
- **Add** an Abstract and the following sections
- Technical Contributions
  - Theorems, proofs, algorithms, analysis of correctness and time complexity, heuristic for optimizations (if any)
- Experiments (waived only for serious theoretical results)
  - Datasets, machines, software used
  - Details of the tool built (if any)
  - Experimental results as graphs, comparisons with other approaches (from previous work and naïve approaches)
  - Discussions on experimental results
- Conclusions
  - Summary – contributions -- what worked and what did not work -- what can be done in the future
- Now it should look like a research paper!

4. Project Presentation

- More about it later

Review Questions

- Answer five review questions, one per topic.
- Option 1:

Review one of the papers on that topic (0.5-1 page)

- Option 2:

Write about potential research directions on each topic that you can think of related to your own interests

  - 0.5 - 1 page
  - See the schedule for details. The review related to your own research interests
  - no need to solve them in this course!

- The first one is due on 9/13

Class Participation

- Think critically
- Ask questions, answer others' questions
- Point out weaknesses and possible future work
- Express your opinion

Paper Presentation

- You will present one paper from the reading list in this class
- 50-60 min presentation followed by 15-25 mins discussion
- Go into detail
- But do not copy-paste theorems/formulas/proofs/system details in the slides!
- Try to convey the main technical ideas with examples, figures, simplified proofs for special cases

More on Paper Presentation

- Send me a list of 4-5 class#+paper in your order of preference
  - First come first serve, name will be updated on the webpage

- We have two "TBD" classes
  - Feel free to suggest a new topic (can be related to your project) that you want to present

- If you plan to use one of the systems from Class#21-23 in your project, you might want to opt for these classes

Questions?

- Actively participate
- Give me feedback and suggestions
- Send me topics for review in the class

- Let’s get started!

Review of SQL

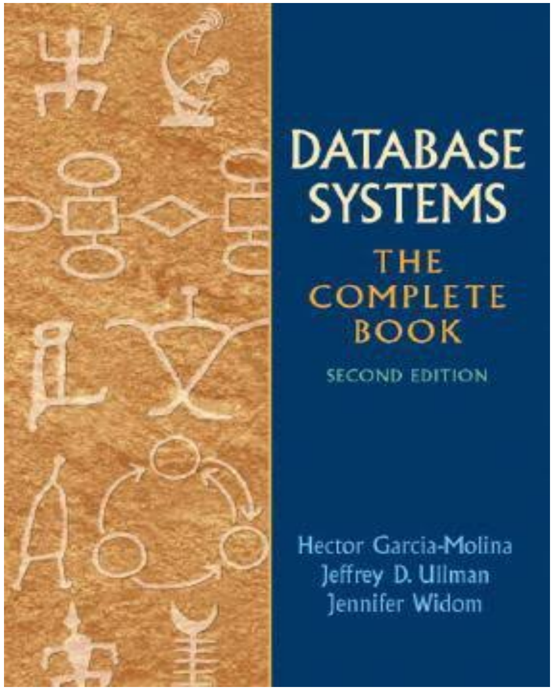
- Acknowledgement:
  - The following slides have been prepared using the lecture slides by Prof. Dan Suciu, University of Washington.

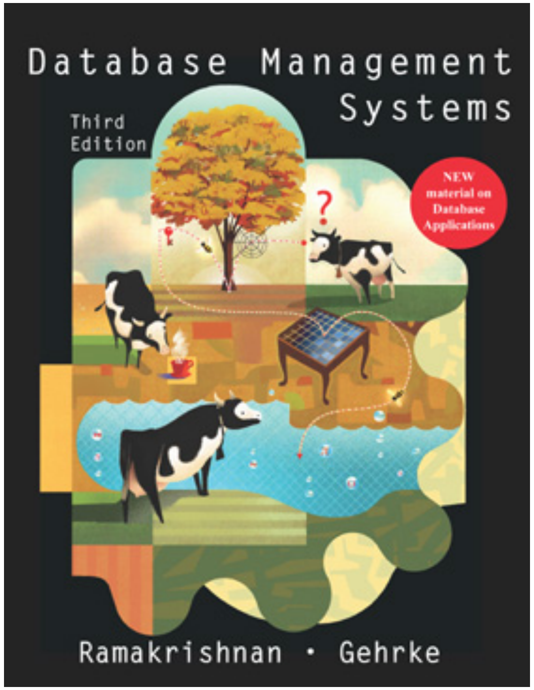
(CSE 544: Principles of Database Systems, Spring 2013

http://courses.cs.washington.edu/courses/cse544/13sp/)

Resources to review SQL and DBMS

- Lecture notes of Duke CompSci 316
  - (Instructor: Prof. Jun Yang)




Database Management Systems
Third Edition
NEW material on Database Applications
Ramakrishnan • Gehrke

Basic Notions

- Database:
  - A collection of files
- Relational Database:
  - Data is in flat table format (next slide)
- Q. Who were the Turing awardees from databases?
  - Bachman (1973), Codd (1981), Gray (1998), Stonebraker (2014)
- DBMS:
  - A program that allows us to efficiently manage and query relational databases
  - e.g. SQL Server, Oracle, DB2, MySQL, Postgres
- SQL (Structured Query Language)

Relation = Table

Tables in SQL

Attribute = Column

Key

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$9.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Tuples = rows

Purposes of SQL

- Data Manipulation Language (DML)
  - Querying: SELECT-FROM-WHERE
  - Modifying: INSERT/DELETE/UPDATE
- Data Definition Language (DDL)
  - CREATE/ALTER/DROP

Two types of query workloads

- OLTP (OnLine Transaction Processing)
  - Multiple concurrent read-write requests (transactions)
  - Commercial applications (banking, online shopping)
  - Data changes frequently
  - Concurrency control
    - Waiting for one to finish is not efficient
  - Very important, but not a topic of this course

- OLAP (OnLine Analytical Processing)
  - Many aggregate/group-by queries – multidimensional data
  - Data mostly static
  - Data analytics – a topic in this course
  - Will study OLAP Cube in Class#2-4

Toy IMDB Database

SQL Examples

Actor:

id	fName	lName	gender
195428	Tom	Hanks	M
645947	Amy	Hanks	F
. . .			

Casts:

pid	mid
195428	337166
. . .	

Movie:

id	Name	year
337166	Toy Story	1995
. . .	. . .	. ..

```
SELECT *
FROM  Actor
```

```
SELECT count(*)
FROM  Actor
```

```
SELECT *
FROM  Actor
WHERE lName = 'Hanks'
```

```
SELECT TOP 100 *
FROM  Actor
```

```
SELECT *
FROM  Movie

WHERE Name LIKE '%Toy%'
```

```
SELECT DISTINCT lName
FROM  Actor

ORDER BY lName DESC
```

- SELECT clause vs. SELECTION condition
- Foreign keys

Toy IMDB Database

SQL Examples

Actor:

id	fName	lName	gender
195428	Tom	Hanks	M
645947	Amy	Hanks	F
. . .			

Casts:

pid	mid
195428	337166
. . .	

Movie:

id	Name	year
337166	Toy Story	1995
. . .	. . .	. ..

```
SELECT *
FROM  Actor x, Casts y, Movie z
WHERE x.lname='Hanks'
      and x.id = y.pid
      and y.mid=z.id
      and z.year=1995
```

Join conditions

Selection condition

Semantic

```
Answer = {}
for $x_1$ in Actor do
   for $x_2$ in Casts do
      for $x_3$ in Movie do
         if (cond1) and (condn2) and (condn3)
            then Answer = Answer ∪ {($x_1$,$x_2$,$x_3$)}
return Answer
```

In the actual IMDB database,
>= 1.8M actors, 11M casts, 1.5M movies
How can a DBMS run so fast ?

Review Natural Join, Equi-join, Theta-Join, Left/Right/Full Outer Join

Relational Algebra and Logical Query Plans

Translating WHAT to HOW:

- SQL = WHAT we want = a declarative language
- Relational algebra = HOW to get it = algorithm/ logical query plan (like a pseudo code)
- RDBMS are about translating WHAT to HOW
  - SQL => logical plan + physical implementation

SQL Query => Multiple Logical Plans/RA Expressions

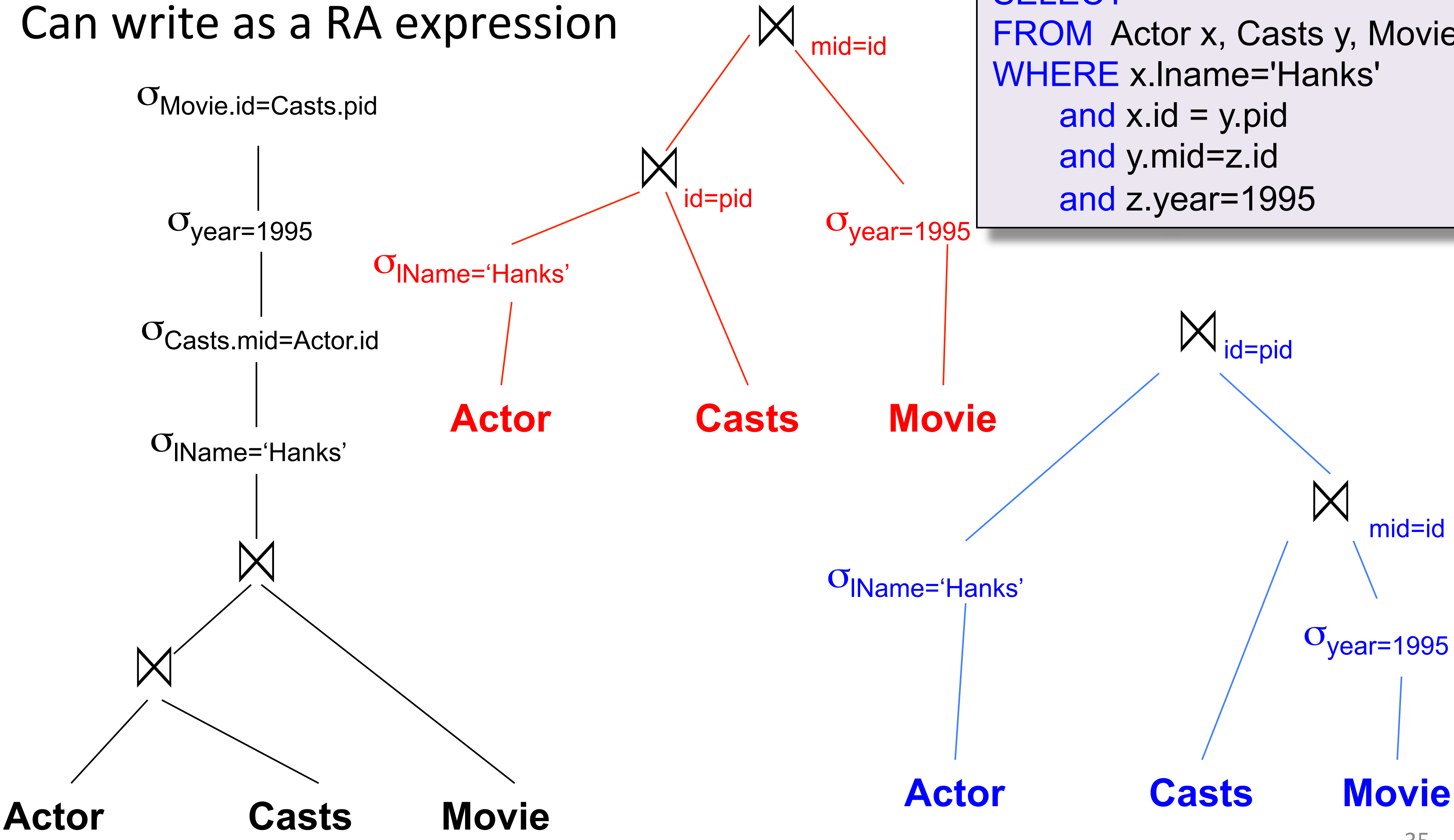
Many equivalent plans

Some more efficient

Can write as a RA expression

$[(\sigma_{lName=\text{'Hanks'}}\ Actor) \bowtie_{id=pid} Casts] \bowtie_{id=pid} (\sigma_{year=1995}\ Movie)$

```
SELECT *
FROM  Actor x, Casts y, Movie z
WHERE x.lname='Hanks'
      and x.id = y.pid
      and y.mid=z.id
      and z.year=1995
```



Logical Plan => Multiple Physical Plans

In practice, in a single step
SQL => RA + physical plan

Classical query execution

Index-based selection

Hash-join

Merge-join

Index-join

Query optimizations:

Pushing selections down

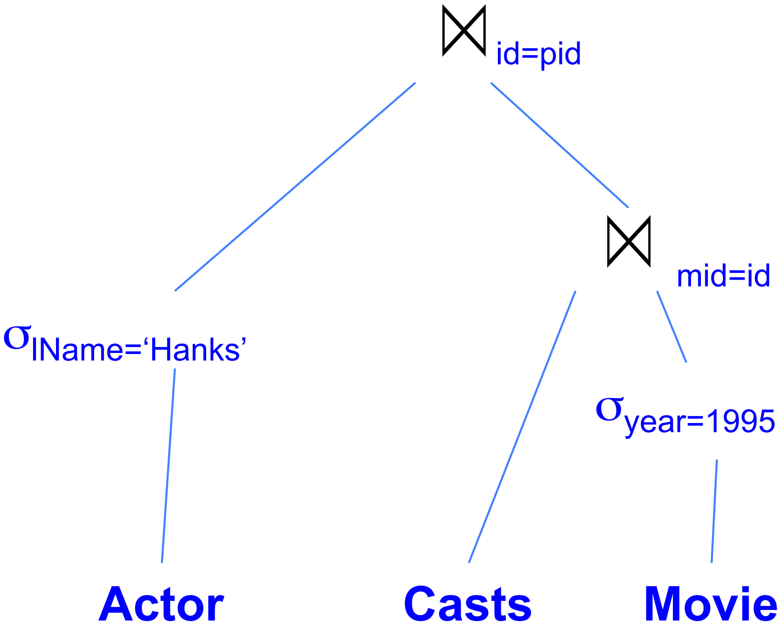
Join reorder

Statistical Information

Table cardinalities

distinct values

histograms



Data Independence

Physical data independence:

- Applications are isolated from changes to the physical organization (e.g. adding or dropping an index)

Logical data independence:

- Existing “views” (query output) are unaffected by change of schema

Purchase(product, quantity, price)

Simple Aggregations

Basic aggregate operations in SQL

```
select count(*) from Purchase
select count(distinct quantity) from Purchase
select sum(quantity) from Purchase
select avg(price) from Purchase
select max(quantity) from Purchase
select min(quantity) from Purchase
```

Purchase(product, quantity, price)

Counting Duplicates

COUNT applies to duplicates, unless otherwise stated:

```
SELECT  Count(product)
FROM    Purchase
WHERE   price > 4.99
```

same as Count(*) except NULL values won't be counted

We probably want:

```
SELECT  Count(DISTINCT product)
FROM    Purchase
WHERE   price> 4.99
```

Purchase(product, quantity, price)

More Examples

Purchase

Product	Price	Quantity
Bagel	3	20
Bagel	1.50	20
Banana	0.5	50
Banana	2	10
Banana	4	10

```
SELECT  Sum(price * quantity)
FROM     Purchase
WHERE   product = ‘bagel’
```

90 (= 60+30)

```
SELECT  product, Sum(price * quantity) as total
FROM     Purchase
GROUP BY product
```

Product	total
Bagel	90
Banana	85

Purchase(product, quantity, price)

Semantic: Grouping and Aggregation

Product	Price	Quantity
Bagel	3	20
~~Bagel~~	~~1.50~~	~~20~~
~~Banana~~	~~0.5~~	~~50~~
Banana	2	10
Banana	4	10

```
SELECT      product, Sum(price * quantity) as total
FROM         Purchase
WHERE        price > 1.7
GROUP BY product
```

Product	total
Bagel	60
Banana	60

1. Compute the FROM and WHERE clauses.

2. Group by the attributes in the GROUPBY

3. Compute the SELECT clause:
   grouped attributes and aggregates.

Purchase(product, quantity, price)

HAVING Clause

Product	Price	Quantity
Bagel	3	20
~~Bagel~~	~~1.50~~	~~20~~
~~Banana~~	~~0.5~~	~~50~~
~~Banana~~	~~2~~	~~10~~
~~Banana~~	~~4~~	~~10~~

```
SELECT      product, Sum(price * quantity) as total
FROM        Purchase
WHERE       price > 1.7
GROUP BY    product
HAVING      min(quantity) > 15
```

Product	total
Bagel	60

1. Compute the FROM and WHERE clauses.

2. Group by the attributes in the GROUPBY

3. **Apply the HAVING clause**

4. Compute the SELECT clause: grouped attributes and aggregates.

- WHERE (to each row, no aggregate)
- HAVING (to each group, aggregate, either entire group is returned or no tuple from that group).

Purchase(product, quantity, price)

Next few classes: Data Cube

Product	Price	Quantity
Bagel	3	20
Bagel	1.50	20
Banana	0.5	50
Banana	2	10
Banana	4	10

```
SELECT   product, quantity, sum(price) as total
FROM      Purchase
GROUP BY   product, quantity
```

Product	quantity	total
Bagel	20	4.5
Banana	50	0.5
Banana	10	6

```
SELECT   product, quantity, sum(price) as total
FROM      Purchase
GROUP BY   product, quantity
WITH CUBE
```

Product	quantity	total
Bagel	20	4.5
Banana	50	0.5
Banana	10	6
Bagel	*	4.5
Banana	*	6.5
*	20	4.5
…	…	…

DeptStoreSales(storeid, city, state, manager,
day, date, month, year, foundedin, productSold, salesUSD)

The above schema is not good. Why?