Bill Gates, the Revolution, and the Art of Gardening

Based on a true story
Outline

• What are databases about?
• A history of database systems (abridged)
• Why are databases so successful?
• Why does Bill Gates appear in the title?
Databases

- Database Management System (DBMS): an integrated suite of tools to manage (store, query, update) a large collection of data

- Databases are used everywhere
  - ATMs, Banking Systems, Web Applications, ...

- Lucrative career path
  - Skilled DB programmers and administrators are very well paid
It looks so simple...

- **Input**: a query (in SQL)
- **Output**: results

```sql
SELECT title
FROM Movies, Actors
WHERE Actors.mid = Movies.mid AND Actors.name = "Pitt"
```

SQL Query → DBMS → Results
...but it is very complex!

- Transactional semantics
- Recovery
- Queries are declarative and must be optimized
- Data on disk is much larger than main memory
- Varying system characteristics
- Extensible type system
- Lots more!
Just how good is a DBMS?

Here are some examples:

- A DBMS can sort efficiently 700 gigabytes of data using 80MB of main memory.
- A DBMS can sustain hundreds of thousands of transactions per second.
- A DBMS can mine information from terabytes of data.

A DBMS can also do a lot more:

- Execute “statistical” queries over terabytes of data
- Support user-defined types and functions
- ...
History
circa 1960

Network Data Model (CODASYL)
Key idea: a database is a network of records
Tuple-at-a-time processing
1970: The revolution

- Ted Codd proposes the Relational Model
- Strong mathematical foundation
- Set-a-time processing
- Two query languages: RA and RC

<table>
<thead>
<tr>
<th>mid</th>
<th>year</th>
<th>genre</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2005</td>
<td>&quot;action&quot;</td>
</tr>
<tr>
<td>2</td>
<td>2004</td>
<td>&quot;drama&quot;</td>
</tr>
<tr>
<td>3</td>
<td>2000</td>
<td>&quot;drama&quot;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>aid</th>
<th>sex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>male</td>
</tr>
<tr>
<td>2</td>
<td>female</td>
</tr>
<tr>
<td>3</td>
<td>male</td>
</tr>
</tbody>
</table>

\[ \sigma_{\text{name}="Pitt"}(\text{Actor}) \otimes \text{Cast} \otimes \text{Movie} \]
CODASYL vs. Relational

CODASYL:
- Relational is too mathematical!
- Relational is not good for record-oriented apps!
- There is no efficient implementation for RA or RC!

Relational:
- CODASYL is too complex!
- Set-oriented queries are so much better!
- CODASYL does not have any math behind it!
mid 70s: the turn of the tide

- System-R and Ingres emerge as the first relational systems
- SQL and QUEL appear as user-friendly versions of RA and RC
- End result: the relational model starts looking more attractive!
Ingres

Led by UC (Berkeley) researchers
Research system
Descendant: Postgres
System-R

- Developed at IBM San Jose lab
- High impact work
- Descendants: Oracle, DB2, ...
- SQL => System R version 1 => System R version 2
80s: Commercialization

- Oracle beats System-R to the market
- IBM releases many systems and settles on DB2
- Informix is started by Roger Sippl
- 1985: SQL Standard
90s–today: Domination

- Relational systems are the standard
- CODASYL survives only on legacy systems
- SQL standard is > 1000 pages (started at 20!)
- Billion dollar industry
- Major players: Oracle, IBM, Microsoft
Why are databases so good?
Separate application code from data

SELECT title
FROM Movie, Actor
WHERE Movie.mid=Actor.mid AND Actor.name="Pitt"

SELECT Movie.year, COUNT(*)
FROM Movie, Actor
WHERE Movie.mid=Actor.mid AND Actor.name="Pitt"
GROUP BY Movie.year
Query Optimization

- SQL is declarative => User describes *what* data to retrieve (but not how!)
- The system determines the best plan to evaluate the query

 Scalable processing algorithms

SQL Query → Optimizer → Plan

Plan A

Plan B
Data Independence

- **External Schema**
- **Conceptual Schema**
- **Physical Schema**

- YoungActors(name, age)
- PittMovies(title)
- Movie(mid, title)
- Actor(name, mid, age)
- Movie(mid, title) + Index on mid + Index on title
- Actor(name, mid, age) + Index on age
ACID

Transactional Consistency:

- Atomicity
- Consistency
- Isolation
- Durability

Completely transparent to the application

Jim Gray won a Turing Award for his work on transactions
Theory

- Database systems have strong theoretical foundations
- Prominent example: schema normalization

\[
\begin{align*}
\text{Actor}(\text{name}, \text{mid}) & \quad \text{vs.} \quad \text{Actor}(\text{name}, \text{aid}) \\
\text{Movie}(\text{mid}, \text{title}, \text{year}) & \quad \text{vs.} \quad \text{Cast}(\text{aid}, \text{mid}) \\
& \quad \text{vs.} \quad \text{Movie}(\text{mid}, \text{title}, \text{year})
\end{align*}
\]
New Trends
Motivating example: run SQL query over the packet stream of a network router

Compute the number of packets that originate from hosts A and B and target the same machine

Problem #1: data is infinite!

Problem #2: data arrives fast!

Need for new architecture:

- New query languages
- New optimization techniques
- New paradigms
Data Mining

Goal: extract knowledge from data

- Example: Walmart data has revealed that customers who buy beer also buy diapers

Challenge #1: scale!

- Walmart is already at terabytes and going for petabytes

Challenge #2: richer knowledge

- Frequent sub-sequences, association rules, ...
- What's next?
Sensor Databases

- Goal: deploy a large number of small devices that measure “stuff” (temperature, illumination, humidity, ...)
- Use SQL to manipulate sensor data
- Challenge #1: Low power consumption
- Challenge #2: Limited processing power
Peer-to-Peer

- P2P systems have been popular for file sharing
- What about running SQL over P2P?
- Challenge #1: Volatility
- Challenge #2: Locating the data
- Challenge #3: Consolidating heterogeneity
XML
Letter to Bill Gates
“Microsoft mailing address”
“Microsoft address”

Worm dupes with fake Microsoft address | CNET News.com
Worm dupes with fake Microsoft address | A new e-mail worm, which feigns a Microsoft.com origin, is spreading rapidly. ... Worm dupes with fake Microsoft address. ... news.com.com/2100-1002_3-1007603.html - 31k - Cached - Similar pages

Microsoft kills Net address to foil worm | CNET News.com
Microsoft kills Net address to foil worm | As part of its effort to stop the MSBlast worm, the software giant is eliminating the Windows Update address that ... news.com.com/2100-1002_3-5064433.html - 33k - Cached - Similar pages
[ More results from news.com.com ]

1999 09 14: Intel, Compaq, Entrust, Ibm And Microsoft Address ...

Portal: Portal Software and Microsoft Address WiFi Business ...
Web Search Today

- Web document: bag of words
- HTML: presentation language

```
<I>
  Microsoft<br>
  One Microsoft Way<br>
  Redmond, WA<br>
</I>

<I>
  Teriyaki sause<br>
  One egg<br>
  New York steak<br>
</I>

- Difficult to identify structure/semantics
A first step - XML

Focus on structure/semantics instead of layout

```
<I>
    Microsoft<br>
    One Microsoft Way<br>
    Redmond, WA<br>
</I>
```

"Microsoft mailing address"

```
<address>
    <company name="Microsoft"> <street>One Microsoft way</street> </company>
    <city>Redmond</city>
    <state>WA</state>
</address>
```

address[.*name="Microsoft"]
Example Query: “retrieve the experiments section of papers on XML, where the related work section references papers on histograms”
The Revolution

XML

XML

XML
Challenges for XML

- Data is tree-structured
- Data is self-describing \(\Rightarrow\) Schema Chaos!
- Need to revise several aspects of the system
  - New query language
  - New optimization techniques
  - New system architecture
DB Research at UCSC
Group

- 3 Faculty
- 5 PhD students
- Several MSc and undergraduates
Projects

- Data Integration
- Data Provenance
- Approximate Query Answering
- Self-Tuning Systems
Approximate Query Answering

Data

Data Synopsis

Query

Result

Expensive

Efficient

Query

Approximate Result
Data and Query Model

Tree data with heterogeneous value content
Tree-pattern queries with XPath expressions
Result: set of binding tuples
Structural Summarization

**Node <-> Elements of same tag**

**Statistical information: node- and edge-counts**

- **Node-count:** number of elements in cluster
- **Edge-count:** average number of children

Data

- **bib**
  - **p**
    - **y** 2000
    - **kw**...
  - **b**
    - **y** 1970
    - **kw**...

XCluster

- **BIB(1)**
  - **P(2)**
    - 1
  - **B(1)**
    - 1/2
    - 1

- **T(2)**
  - 1
  - 1/2

- **Y(2)**
  - 1

- **KW(2)**
  - 1

- **T(1)**
  - 1

- **BIB(1)**
  - 2
  - 1
Value Summarization

Value summary => Fractional value distribution

- Single-dimensional
- Approximation method depends on value type
Results

XMark

- XCluster
- TreeSketch
CHAMELEON-DB

Problem: users do not want to "lose" their data

Solution: make the system storage-less!
- The system processes data in-place
- No need to import/export

DBMS

Data

Trapped in the "box"

Data is free
Challenges

- How can the DBMS ensure efficiency?
  - Data has to be indexed on-the-fly
- What about query optimization?
  - Need to build cost models on top of files
- What about system administration?
  - Make the system self-tuning
COLT

- Continuous On-Line Tuning
  - Key idea: monitor the query load and install indexes on the fly
  - Challenge #1: Identify useful indices
  - Challenge #2: Control overhead
Results
Τέλος!

- Database classes
  - CMPS 180
  - CMPS 181
- Weekly database seminar
  - Subscribe to db-seminar@soe
- Come and talk to us!