Introduction to Data Management

Practical Data Management

Alyssa Pittman
Based on slides by Jonathan Leang, Dan Suciu, et al

Paul G. Allen School of Computer Science and Engineering
University of Washington, Seattle
Goals for Today

- Talk about the parts of data management you might encounter in the real world without having theory to back you up
Outline

- Views
- Data Cleaning
  - ETL
  - Data wrangling on GCP Dataprep (Trifacta)
- Data Management Ethics and Best Practices
A view is a relation defined by a query:

```sql
CREATE VIEW StorePrice AS
SELECT x.store, y.price
FROM Purchase x, Product y
WHERE x.pid = y.pid
```

This is like a new relation:

```
StorePrice(store, price)
```

---

**Customer(cid, name, city)**

**Purchase(cid, pid, store)**

**Product(pid, name, price)**
Views can be queried just like tables:

```
SELECT DISTINCT z.name, u.store
FROM Customer z, Purchase u, StorePrice v
WHERE z.cid = u.cid AND u.store = v.store
    AND v.price > 1000
```

- The semantics are the same as using a subquery:

```
SELECT DISTINCT z.name, u.store
FROM Customer z, Purchase u,
    (SELECT x.store, y.price
     FROM Purchase x, Product y) v
WHERE x.pid = y.pid
    AND z.cid = u.cid AND u.store = v.store
    AND v.price > 1000
```
Applications of views

- Logical data independence
- Security
- Increased physical data independence
Applications of views

- **Logical data independence**

Say I want to normalize my database, but have many old queries using the original schema.

...create views consistent with the old schema!
Applications of views

- Security

Give users access to only the data they need.

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>Houston</td>
<td>450.99</td>
</tr>
<tr>
<td>Sue</td>
<td>Seattle</td>
<td>-240</td>
</tr>
<tr>
<td>Joan</td>
<td>Seattle</td>
<td>60.23</td>
</tr>
<tr>
<td>Ann</td>
<td>Portland</td>
<td>-23.50</td>
</tr>
</tbody>
</table>

CREATE VIEW PublicCustomers
SELECT Name, Address
FROM Customers

But they can see this.

Advertising team shouldn’t see the balances.
Applications of views

- Security

Give users access to only the data they need.

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>Houston</td>
<td>450.99</td>
</tr>
<tr>
<td>Sue</td>
<td>Seattle</td>
<td>-240</td>
</tr>
<tr>
<td>Joan</td>
<td>Seattle</td>
<td>60.23</td>
</tr>
<tr>
<td>Ann</td>
<td>Portland</td>
<td>-23.50</td>
</tr>
</tbody>
</table>

CREATE VIEW NegativeBalanceCustomers
SELECT * FROM Customers WHERE Balance < 0

Collections team shouldn’t see people with positive balances
Applications of views

- Increased physical data independence
  - Vertical partitioning

<table>
<thead>
<tr>
<th>Resumes</th>
<th>SSN</th>
<th>Name</th>
<th>Address</th>
<th>Resume</th>
<th>Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>234234</td>
<td>Mary</td>
<td>Huston</td>
<td>Clob1…</td>
<td>Blob1…</td>
</tr>
<tr>
<td></td>
<td>345345</td>
<td>Sue</td>
<td>Seattle</td>
<td>Clob2…</td>
<td>Blob2…</td>
</tr>
<tr>
<td></td>
<td>345343</td>
<td>Joan</td>
<td>Seattle</td>
<td>Clob3…</td>
<td>Blob3…</td>
</tr>
<tr>
<td></td>
<td>234234</td>
<td>Ann</td>
<td>Portland</td>
<td>Clob4…</td>
<td>Blob4…</td>
</tr>
</tbody>
</table>

T1

<table>
<thead>
<tr>
<th>SSN</th>
<th>Name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>234234</td>
<td>Mary</td>
<td>Huston</td>
</tr>
<tr>
<td>345345</td>
<td>Sue</td>
<td>Seattle</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>

T2

<table>
<thead>
<tr>
<th>SSN</th>
<th>Resume</th>
</tr>
</thead>
<tbody>
<tr>
<td>234234</td>
<td>Clob1…</td>
</tr>
<tr>
<td>345345</td>
<td>Clob2…</td>
</tr>
</tbody>
</table>

T3

<table>
<thead>
<tr>
<th>SSN</th>
<th>Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td>234234</td>
<td>Blob1…</td>
</tr>
<tr>
<td>345345</td>
<td>Blob2…</td>
</tr>
</tbody>
</table>
Applications of views

- Increased physical data independence
  - Vertical partitioning
    - Helpful for data warehousing
    - Can improve performance
      - Have large columns (e.g. photo blob)
      - Have lots of columns but each query only accesses a few

CREATE VIEW Resumes AS
SELECT T1.ssn, T1.name, T1.address,
T2.resume, T3.picture
FROM T1,T2,T3
WHERE T1.ssn=T2.ssn and T2.ssn=T3.ssn
Applications of views

- Increased physical data independence
  - Horizontal partitioning

<table>
<thead>
<tr>
<th>Customers</th>
<th>CustomersInHouston</th>
<th>CustomersInSeattle</th>
<th>CustomersInCanada</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSN</td>
<td>Name</td>
<td>City</td>
<td>Country</td>
</tr>
<tr>
<td>234234</td>
<td>Mary</td>
<td>Houston</td>
<td>USA</td>
</tr>
<tr>
<td>345345</td>
<td>Sue</td>
<td>Seattle</td>
<td>USA</td>
</tr>
<tr>
<td>345343</td>
<td>Joan</td>
<td>Seattle</td>
<td>USA</td>
</tr>
<tr>
<td>234234</td>
<td>Ann</td>
<td>Portland</td>
<td>USA</td>
</tr>
<tr>
<td>--</td>
<td>Frank</td>
<td>Calgary</td>
<td>Canada</td>
</tr>
<tr>
<td>--</td>
<td>Jean</td>
<td>Montreal</td>
<td>Canada</td>
</tr>
<tr>
<td>SSN</td>
<td>Name</td>
<td>City</td>
<td>Country</td>
</tr>
<tr>
<td>234234</td>
<td>Mary</td>
<td>Houston</td>
<td>USA</td>
</tr>
<tr>
<td>345345</td>
<td>Sue</td>
<td>Seattle</td>
<td>USA</td>
</tr>
<tr>
<td>345343</td>
<td>Joan</td>
<td>Seattle</td>
<td>USA</td>
</tr>
<tr>
<td>--</td>
<td>Frank</td>
<td>Calgary</td>
<td>Canada</td>
</tr>
<tr>
<td>--</td>
<td>Jean</td>
<td>Montreal</td>
<td>Canada</td>
</tr>
</tbody>
</table>

12
January 29, 2020
Practical Data Management
Applications of views

- Increased physical data independence
  - Horizontal partitioning
    - Helpful for data warehousing

CREATE VIEW Customers AS
CustomersInHouston UNION ALL
CustomersInSeattle UNION ALL
...
Applications of views

- Increased physical data independence
  - Partitioning
    - Mostly helpful for
Views trivia

- Virtual views
  - Computed on the fly – potentially slow
  - Always up-to-date

- Materialized views
  - Pre-computed and stored – fast to access
  - May have stale data

- Can views be updated?
  - Some SQL variants let you update the data behind *simple* views
Where is my data coming from?

- You generate the data
  - Output data that is easy to use

- External sources or preexisting data
  - Sometimes doesn’t fit your application needs
  - Need to translate the data into a usable form
"I know exactly what operations need to be done to get from data format A to data format B"

- **Extract**
  - Read relevant data

- **Transform**
  - Push data through mapping functions until done
    - Aggregations
    - Normalization
    - ...

- **Load**
  - Write to destination
Extract Transform Load (ETL)

Corporate needs you to find the differences between this picture and this picture.

They're the same picture.
Data Wrangling

“I have no clue what’s going on with my data”

- Essentially ETL but with **data exploration**
- Interactivity is important
  - Visualizations
  - Suggestions
Create a “summary table”
- Generally used for reports to draw attention to interesting values
- Able to make values into columns

“Skinny and tall” ➞ “short and wide”

<table>
<thead>
<tr>
<th>Name</th>
<th>Year</th>
<th>GDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angola</td>
<td>2015</td>
<td>100</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>2015</td>
<td>50</td>
</tr>
<tr>
<td>Angola</td>
<td>2016</td>
<td>110</td>
</tr>
<tr>
<td>Angola</td>
<td>2018</td>
<td>115</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>2017</td>
<td>55</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>2018</td>
<td>65</td>
</tr>
</tbody>
</table>
Create a “summary table”
  • Generally used for reports to draw attention to interesting values
  • Able to make values into columns

“Skinny and tall”  □  “short and wide”

<table>
<thead>
<tr>
<th>Name</th>
<th>2015</th>
<th>2016</th>
<th>2017</th>
<th>2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angola</td>
<td>100</td>
<td>110</td>
<td></td>
<td>115</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>50</td>
<td></td>
<td>55</td>
<td>65</td>
</tr>
</tbody>
</table>
Unpivot

- Usually we want to store unpivoted data
  - Easier to manage
- "Short and wide" ↔ "skinny and tall"

<table>
<thead>
<tr>
<th>Name</th>
<th>2015</th>
<th>2016</th>
<th>2017</th>
<th>2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angola</td>
<td>100</td>
<td>110</td>
<td></td>
<td>115</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>50</td>
<td></td>
<td>55</td>
<td>65</td>
</tr>
</tbody>
</table>
Data Wrangling

Quickstart - demo

Cloud Dataprep by TRIFACTA

Google-Refine → OpenRefine

TIBCO™ Clarity

alteryx
You can get data but what are you doing with it?
Existing Laws and Regulations

- **FERPA (Family Education Rights and Privacy Act)**
  - Mandatory for education institutions
    - Requires written consent to disclose academic info
    - Allows the release of directory information
Existing Laws and Regulations

- **HIPAA (Health Information Portability and Accountability Act)**
- Mandatory for healthcare and health insurance institutions
  - Privacy Rule to protect Protected Health Information
  - Security Rule to ensure administrative, physical, and technical safeguards
Existing Laws and Regulations

- SOX (Sarbanes-Oxley Act)
- Requires auditability for companies’ financial records
  - What does this have to do with data?
  - Can financial data be tampered with?
  - Can code touching financial data be tampered with?
Existing Laws and Regulations

- GDPR (European General Data Protection Regulation)
- Requires disclosure from companies about what user data they have and how they use it
  - ...but can be exploited

---

Rian Johnson 📣
@rianjohnson

---

*Episode VIII*

**THE LAST JEDI**

*We have updated our GLOBAL PRIVACY TERMS. Your trust is important to us. As part of our ongoing commitment to transparency and in preparation.*
Laws and Regulations Today

- Social Media and Politics?
- Facebook–Cambridge Analytica Scandal
  - CA uses loophole in Facebook API through an online quiz to harvest personal information data

Whistleblower Christopher Wylie

Mark Zuckerberg’s hearing
What’s at Stake?

The details from his Equifax class-action suit are BONKERS

that these weak passwords had already been compromised in previous breaches. Furthermore, Equifax employed the username “admin” and the password “admin” to protect a portal used to manage credit disputes, a password that “is a surefire way to get hacked.” This portal contained a vast trove of personal information. According to cybersecurity experts, these shortcomings...
Sensitive Information

- **PII = Personally identifying information**
  - Names
  - Student ID
  - Social security number
  - License number

- **Protected data (for legal and/or ethical reasons)**
  - Academic records (FERPA)
  - Protected Health Information (HIPAA)
  - Customer records (GLBA)

- **Passwords**
Access Control

- Block people who shouldn’t have access
  - Most large companies have a tiered-access hierarchy

- Databases usually have built-in access control:

  ```sql
  GRANT <permissions> [ON <table>] TO <user/role>
  ```

Permissions:
- Table-level operations (SELECT, DELETE, ...)
- DB-level operations (CREATE TABLE, GRANT, ...)

User/Role:
- Users like a user on your computer
- Roles (groups) can be predefined or created

```sql
GRANT SELECT, INSERT ON MySecureTable TO PUBLIC
```

Allow anyone who can connect to read and add data to MySecureTable
Access Control

- SQL Injection – application input acts as code
  - Union attack, tautology attack, illegal queries
  - Only possible if there is a place to inject code
  - Consistently one of the top web-based attacks
    - People simply don’t realize its an issue or...
    - People know it’s an issue and never get around to fixing it

- Considered a “solved” problem
  - Parameterize queries with prepared statements
Access Control

Other common techniques to limit access:

- Limit the number of rows that can be seen
  - Leaking a few tuples is better than leaking all of them
- Only allow aggregations
  - Grouping implicitly eliminates identification info
- Don’t store data you don’t need!
Anonymize Data

FERPA Deidentification

- ID to anonymous ID mapping should be secret

- Aggregate data (minimum n-size)
  - **Suppression** □ Don’t provide data 😞
    - Necessary for very small groups
  - **Rounding** □ Bucket data or introduce noise 😊
    - More people means you can be more specific
Implicit Disclosure

- FERPA allows institutions to disclose “directory information” without consent (institution policies can be stronger)
  - Name
  - Email
  - Photographs
  - Phone Number

- If users can derive sensitive information like grades, it violates FERPA
Implicit Disclosure

▪ “Hey, can you give me the directory information for students with a GPA of 3.5?”
Implicit Disclosure

- “Hey, can you give me the directory information for students with a GPA of 3.5?”

  Reveals sensitive information by context

  SELECT D.*
  FROM Directory AS D, Grades AS G
  WHERE D.id = G.id AND G.gpa = 3.5
Implicit Disclosure

**Re-identification of Mass. Governor William Weld**

- **Public voter data**
  - Name
  - ZIP code
  - Sex
  - Birth date
  - ...

- **Anonymous insurance data**
  - ZIP code
  - Sex
  - Birth date
  - Prescription
  - Diagnosis
  - ...
### Implicit Disclosure

#### Cambridge, MA Voter Data ($20)

<table>
<thead>
<tr>
<th>Name</th>
<th>ZIP</th>
<th>Sex</th>
<th>Bday</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>W. Weld</td>
<td>12345</td>
<td>M</td>
<td>Feb 30</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

#### Anon. Insurance Data for Researchers

<table>
<thead>
<tr>
<th>ZIP</th>
<th>Sex</th>
<th>Bday</th>
<th>MedInfo</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>12345</td>
<td>M</td>
<td>Feb 30</td>
<td>Affluenza</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- 6 matches on ZIP
- 3 matches on Sex
- 1 match on Bday

#### MedInfo

<table>
<thead>
<tr>
<th>Name</th>
<th>MedInfo</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>W. Weld</td>
<td>Affluenza</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
### Implicit Disclosure

#### Cambridge, MA Voter Data ($20)

<table>
<thead>
<tr>
<th>Name</th>
<th>ZIP</th>
<th>Sex</th>
<th>Bday</th>
</tr>
</thead>
<tbody>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>W. Weld</td>
<td>12345</td>
<td>M</td>
<td>Feb 30</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>

6 matches on ZIP
3 matches on Sex
1 match on Bday

#### Anon. Insurance Data for Researchers

<table>
<thead>
<tr>
<th>ZIP</th>
<th>Sex</th>
<th>Bday</th>
<th>MedInfo</th>
</tr>
</thead>
<tbody>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>12345</td>
<td>M</td>
<td>Feb 30</td>
<td>Afluenza</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>

Legal in 1997
Illegal since 2003

#### Name  | MedInfo
--------|---------
…      | …       
W. Weld | Afluenza
…      | …       |
Storing Passwords

- Passwords are special
  - High potential for additional security compromises
  - Only operation that should be done is equality comparison
Storing Passwords

(bobtheninja246, password)

If you do this, Ted Codd will start rolling in his grave.

<table>
<thead>
<tr>
<th>Username</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobtheninja246</td>
<td>password</td>
</tr>
<tr>
<td>xXxDragonSlayerxXx</td>
<td>password</td>
</tr>
<tr>
<td>420_E-Sports_Masta</td>
<td>qwertyuiop</td>
</tr>
</tbody>
</table>
Quick overview of hashing

- Hash(input) ➔ hash value
- Hashing is deterministic
- Ideally hashing is noninvertible
- Ideally hash values are uniformly spread out
# Storing Passwords

**Hash it!**

(bobtheninja246, hash(password))

(bobtheninja246, FCgJFI9ryz)

<table>
<thead>
<tr>
<th>Username</th>
<th>Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobtheninja246</td>
<td>FCgJFI9ryz</td>
</tr>
<tr>
<td>xXxDragonSlayerxXx</td>
<td>FCgJFI9ryz</td>
</tr>
<tr>
<td>420_E-Sports_Masta</td>
<td>p8mel6usIF</td>
</tr>
</tbody>
</table>
Storing Passwords

Hash it!

(bobtheninja246, hash(password))

(bobtheninja246, FCgJFI9ryz)

Issues/pitfalls:
• Hashing functions have precomputed “rainbow tables”
• Some hashing functions are fast so brute forcing attacks can happen
• Patterns can occur for the same passwords

<table>
<thead>
<tr>
<th>Username</th>
<th>Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobtheninja246</td>
<td>FCgJFI9ryz</td>
</tr>
<tr>
<td>xXxDragonSlayerxXx</td>
<td>FCgJFI9ryz</td>
</tr>
<tr>
<td>420_E-Sports_Masta</td>
<td>p8mel6usIF</td>
</tr>
</tbody>
</table>
Storing Passwords

Salt it and hash it!

(bobtheninja246, slowhash(password * random salt), random salt)

<table>
<thead>
<tr>
<th>Username</th>
<th>Hash</th>
<th>Salt</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobtheninja246</td>
<td>HHxrd5o7Cn</td>
<td>WUKhhIFBLc</td>
</tr>
<tr>
<td>xXxDragonSlayerxXx</td>
<td>7rYFQIowpW</td>
<td>mq5rFL6JzF</td>
</tr>
<tr>
<td>420_E-Sports_Masta</td>
<td>cQF4DdSFfn</td>
<td>S8e0zpATNR</td>
</tr>
</tbody>
</table>
Storing Passwords

*Salt* it and hash it!

\[(\text{bobtheninja246}, \text{slowhash(password} \times \text{random salt}), \text{random salt})\]

These are just the fundamentals! Many companies outsource password management because it can get very complicated.

<table>
<thead>
<tr>
<th>Username</th>
<th>Hash</th>
<th>Salt</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobtheninja246</td>
<td>HHxrd5o7Cn</td>
<td>WUKhhIFBLc</td>
</tr>
<tr>
<td>xXxDragonSlayerxXx</td>
<td>7rYFQIowpW</td>
<td>mq5rFL6JzF</td>
</tr>
<tr>
<td>420_E-Sports_Masta</td>
<td>cQF4DdSFfn</td>
<td>S8e0zpATNR</td>
</tr>
</tbody>
</table>
Data Quality

- Quality is not only about cleanliness
- Quality may also involve significance
  - Are certain groups large enough to draw meaningful aggregates?
  - If my data is a sample of a population, does it accurately depict that population?
  - Did I ask the right kinds of questions to get good data?
Even Affects Machine Learning

- Training data □ Prediction program
  - Prediction program believes that the training data is representative of a population and covers all cases
  - If there’s bias in the training data, it will affect the model

Google Photos Tags Two African-Americans As Gorillas Through Facial Recognition Software
Takeaways

- Be good stewards of the data you have
- There’s more to data management than the technical bits