

Data Integration, Data Warehousing, and Entity Resolution

Nicholas Mattei, Tulane University

CMPSS3660 – Introduction to Data Science – Fall 2019

<https://rebrand.ly/TUDataScience>



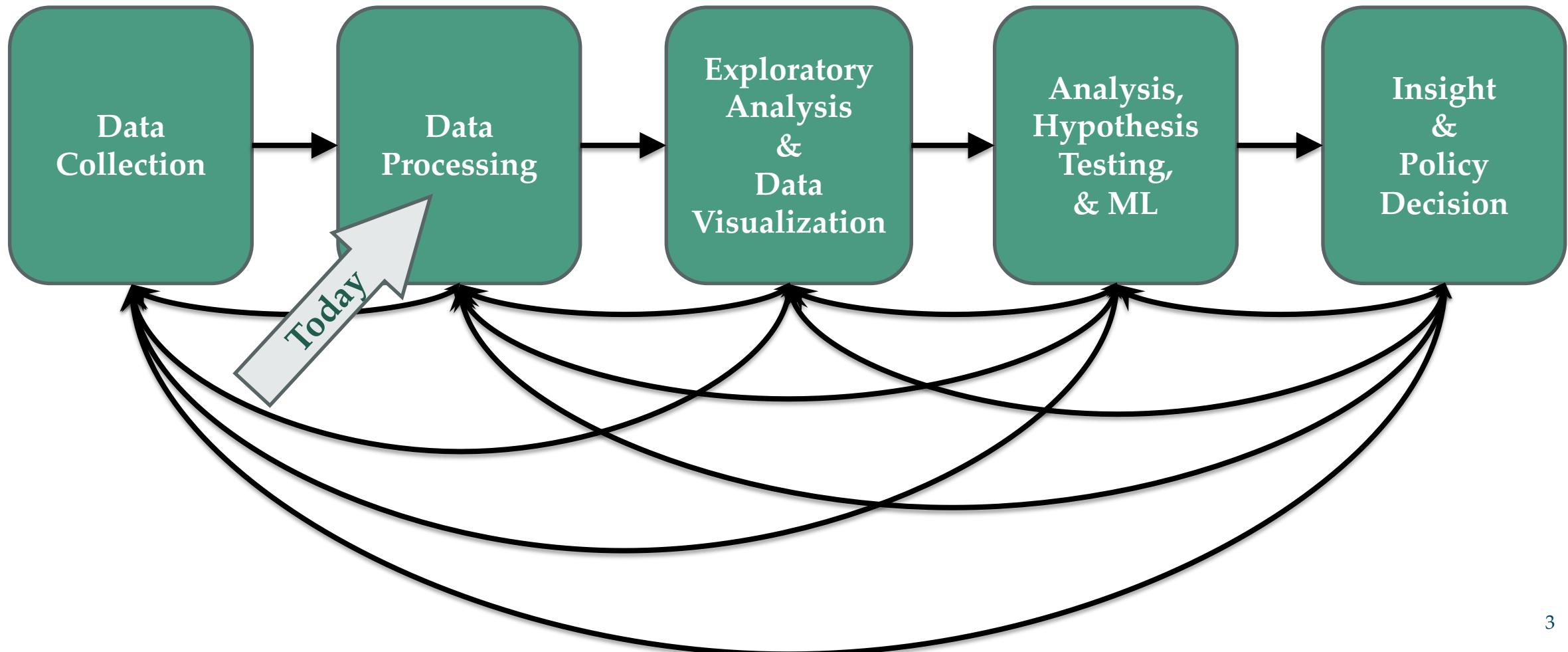
Many Thanks

Slides based off Introduction to Data Science from John P. Dickerson -
<https://cmsc320.github.io/>

Announcements

- Project 1 Due
- Go over lab...

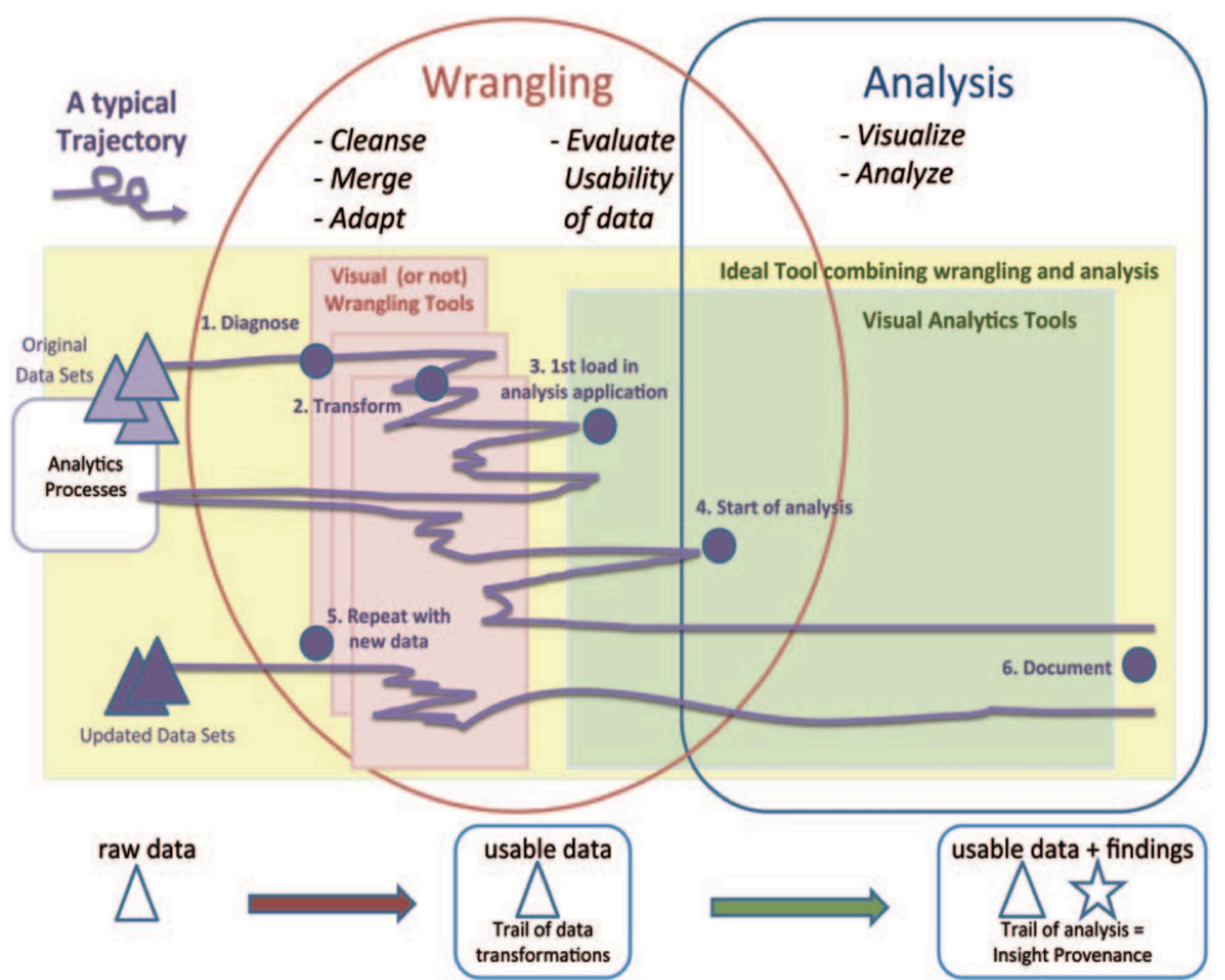
The Data LifeCycle



Overview

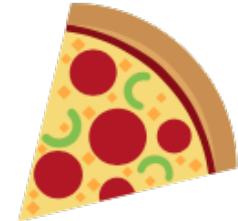
- **Goal: Get data into a structured form suitable for analysis!**
 - Variously called: data preparation, data munging, data curation.
 - Also often called ETL (Extract-Transform-Load) process.
- **Often the step where majority of time (80-90%) is spent.**
- **Key Steps:**
 - **Scraping:** extracting information from e.g., webpages, spreadsheets.
 - **Data Transformation:** to get it into the right structure.
 - **Data Integration:** combine information from multiple sources.
 - **Information Extraction:** extracting structured information from unstructured/text sources.
 - **Data Cleaning:** remove inconsistencies/errors.





A Motivating Example

- I'm back in NYC for a visit.
- Pizza in New Orleans is awful, and I want Pizza.
- A friend told me to go to Joe's Pizza, because it's the best.
- Search away...



joes pizza

yelp Find joes pizza Near New York, NY

Restaurants Home Services Auto Services More Write a Review For Businesses

joes pizza New York, NY

All Filters \$ \$\$\$ \$\$\$\$ Open Now Delivery Takeout Waitlist

Sponsored Results

Cafe Viva
227 reviews \$ - Pizza, Italian, Vegetarian

Start Order Offers takeout and delivery

All Results

1. Joe's Pizza
2417 reviews \$ - Pizza

2. Joe's Pizza
884 reviews \$ - Pizza

Search this area

Anthony & Joe's Pizza

Google

Map data ©2019 Google Terms of Use Report a problem

Sign in

Modest pit stone ovens & calzones

THROGS NECK

COLLEGE POINT

FLUSHING

Grand Central Pkwy

Central Pkwy

Forest Hills

Grand C...

Joe's Pasta and Pizza

Joe's Pizzeria

BROOKLYN

Linden Blvd

Upper Bay

6

Not Even the Same Set!

The image shows two search results for "joes pizza" side-by-side. On the left is a screenshot of the Yelp website, and on the right is a Google Maps search result.

Yelp Search Results:

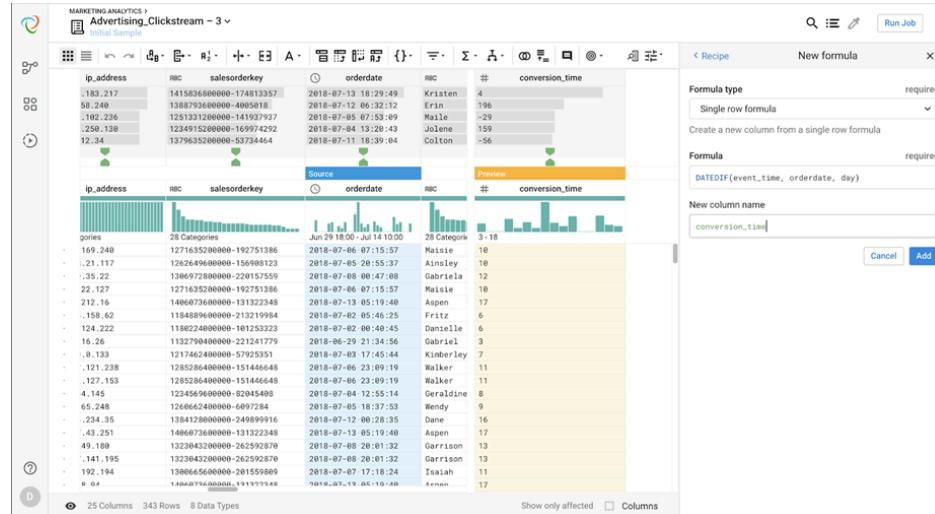
- Search Bar:** Find joes pizza | Near New York, NY
- Filters:** Restaurants, Home Services, Auto Services, More, Write a Review, For Business
- Sponsored Results:**
 - Cafe Viva:** 227 reviews, \$ • Pizza, Italian, Vegetarian. Includes a "Start Order" button and a note that it offers takeout and delivery.
- All Results:**
 - 1. Joe's Pizza:** 2417 reviews, \$ • Pizza.
 - 2. Joe's Pizza:** 884 reviews, \$ • Pizza.

Google Maps Search Results:

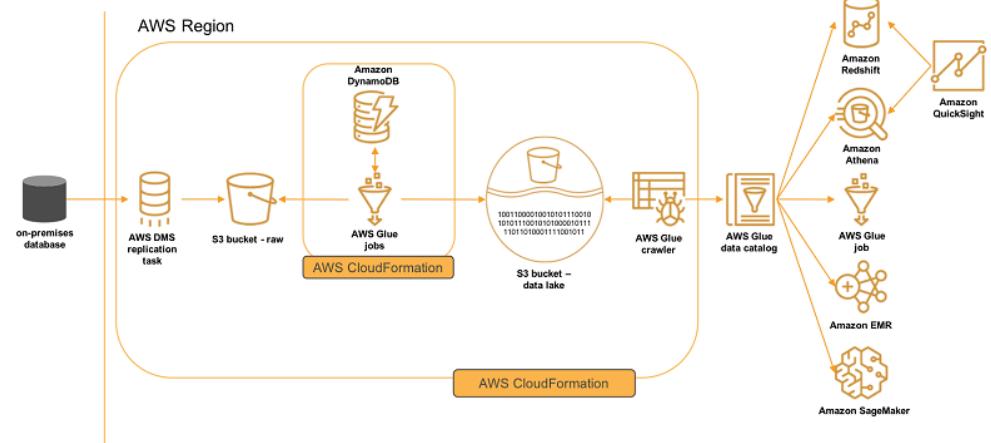
- Search Bar:** Search this area | Sign in
- Map View:** A map of New York City showing several locations of "joes pizza". Each location is marked with a red pin and a small icon of a slice of pizza. The pins are numbered 1 through 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30.
 - Pin 1: Joe's Pizza, Classic NYC slice shop
 - Pin 2: Joe's Pizza, Classic New York-style pizzeria
 - Pin 3: Joe's Pizzeria, Low-key, straightforward pizza parlor
 - Pin 4: Joe's Pizzeria
 - Pin 5: Joe's Pasta and Piz
 - Pin 6: Joe's Pizza
 - Pin 7: Joe's Pizzeria
 - Pin 8: Joe's Pizza
 - Pin 9: Joe's Pizza
 - Pin 10: Joe's Pizza
 - Pin 11: Joe's Pizza
 - Pin 12: Joe's Pizza
 - Pin 13: Joe's Pizza
 - Pin 14: Joe's Pizza
 - Pin 15: Joe's Pizza
 - Pin 16: Joe's Pizza
 - Pin 17: Joe's Pizza
 - Pin 18: Joe's Pizza
 - Pin 19: Joe's Pizza
 - Pin 20: Joe's Pizza
 - Pin 21: Joe's Pizza
 - Pin 22: Joe's Pizza
 - Pin 23: Joe's Pizza
 - Pin 24: Joe's Pizza
 - Pin 25: Joe's Pizza
 - Pin 26: Joe's Pizza
 - Pin 27: Joe's Pizza
 - Pin 28: Joe's Pizza
 - Pin 29: Joe's Pizza
 - Pin 30: Joe's Pizza
- Directions:** Buttons for directions to each of the 30 locations.
- Map Tools:** Satellite view, zoom controls, and a compass rose.

Overview

- Many of the problems ETL stack are hard to formalize, e.g., Data Cleaning.
- Others aspects have been studied in depth, e.g., schema matching and entity resolution
 - [VLDB Tutorial on Entity Resolution](#)



- A mish-mash of tools typically used:
 - Visual (e.g., Trifecta), or not (grep/sed/awk, Pandas).
 - Ad hoc programs for cleaning data, depending on the exact type of errors.
 - Different types of transformation tools.
 - Visualization and exploratory data analysis to understand and remove outliers/noise.
 - Several tools for setting up the actual pipelines, assuming the individual steps are setup (e.g., Talend, AWS Glue).



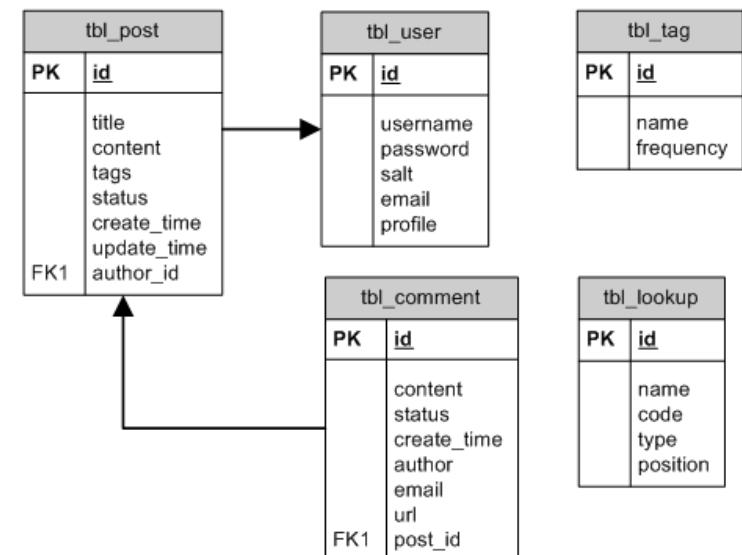
Outline

- Data Integration
- Data Quality Issues
- Data Cleaning
 - Outlier Detection
 - Constraint Based Cleaning
 - Entity Resolution

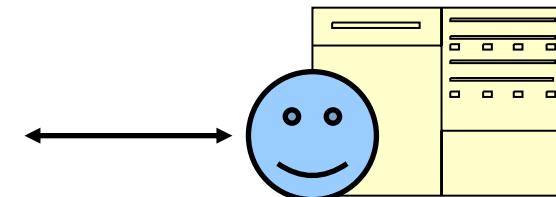
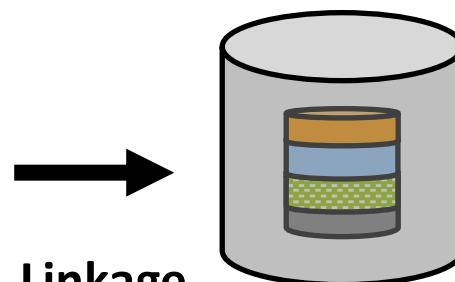
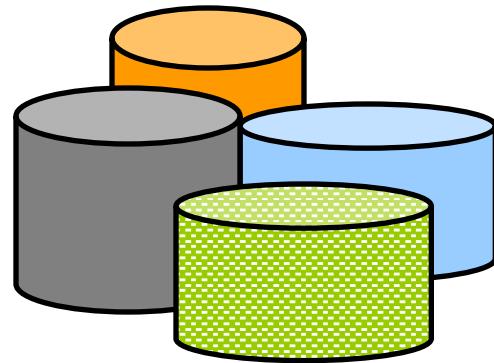
A Few Words to Remember...

- **Schema.** The organization of data within a database or table.
 - i.e., how your data is setup in either a table or RDBMS.
- **Instance.** A single record or element of your data.
 - e.g., Joe's Pizza, our 2nd athlete.
- What follows is from Data Cleaning: Problems and Current Approaches – IEEE Big Data, 2000.
 - Somewhat old: data is mostly coming from structured sources.
 - For a data scientist, the data scraping is equally important.

ID	age	wgt_kg	hgt_cm
1	12.2	42.3	145.1
2	11.0	40.8	143.8
3	15.6	65.3	165.3
4	35.1	84.2	185.8



Data Integration: The Goal!



- **Discovering** information sources (e.g. web modeling, schema learning, ...)
- **Gathering** data (e.g., wrapper learning & information extraction, federated search, ...)
- **Cleaning** data (e.g., de-duping and **linking records**) to form a single [virtual] database

- **Querying** integrated information sources (e.g. queries to views, execution of web-based queries, ...)
- **Data mining & analyzing** integrated information (e.g., collaborative filtering/classification learning using extracted data, ...)

Data Integration

- Goal: Combine data residing in different sources and provide users with a unified view of these data for querying or analysis
 - Each data source has its own schema called **local schemas**
 - Most work assumes relational schemas, but some work on XML and others.
 - The unified schema is often called **mediated schema** or **global schema**.
- Two different setups:
 1. Bring the data together into a single repository (often called data warehousing).
 2. Keep the data where it is, and send queries back and forth.

1. Data Warehousing

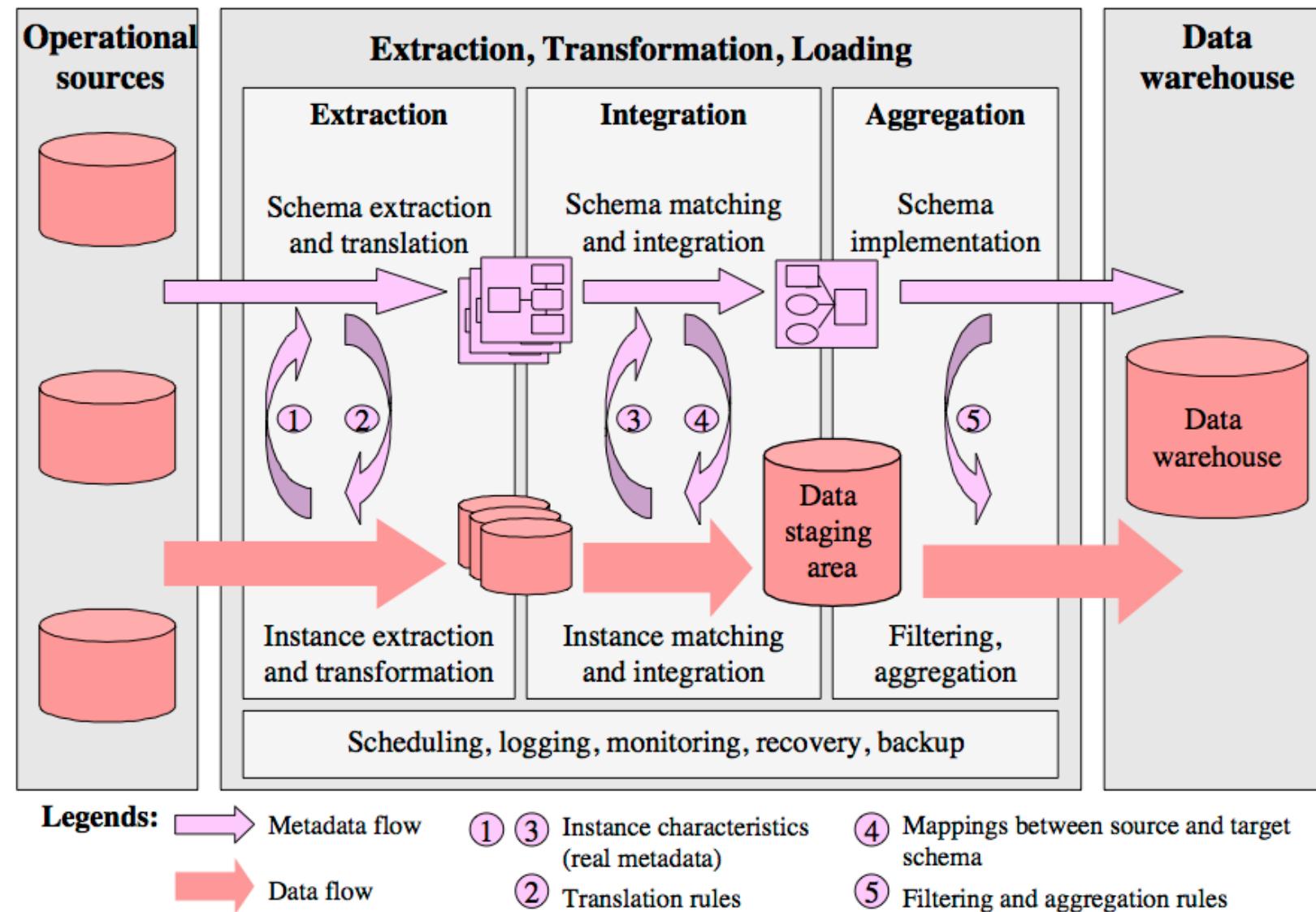
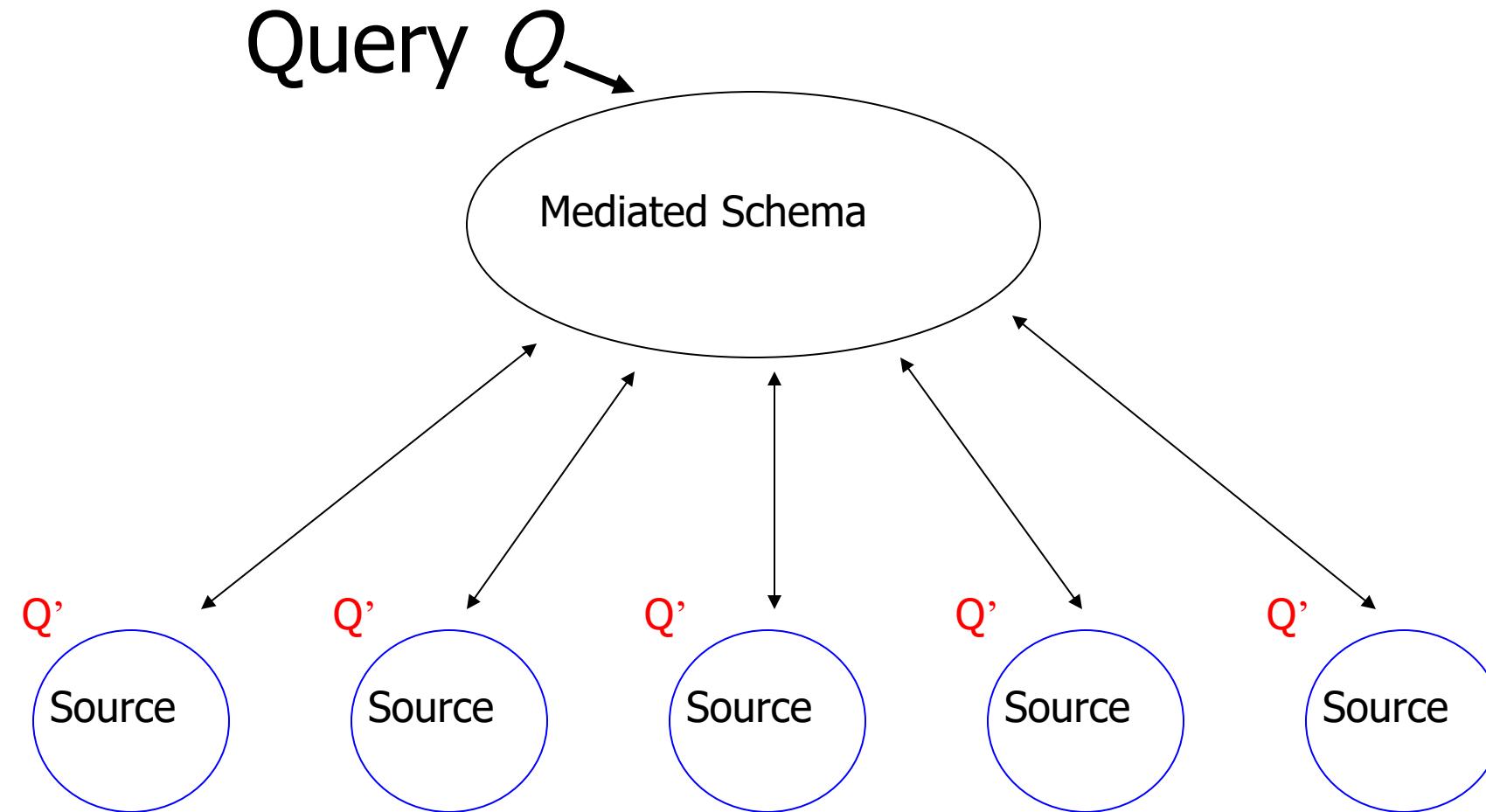


Figure 1. Steps of building a data warehouse: the ETL process

In-place Integration



Data Integration

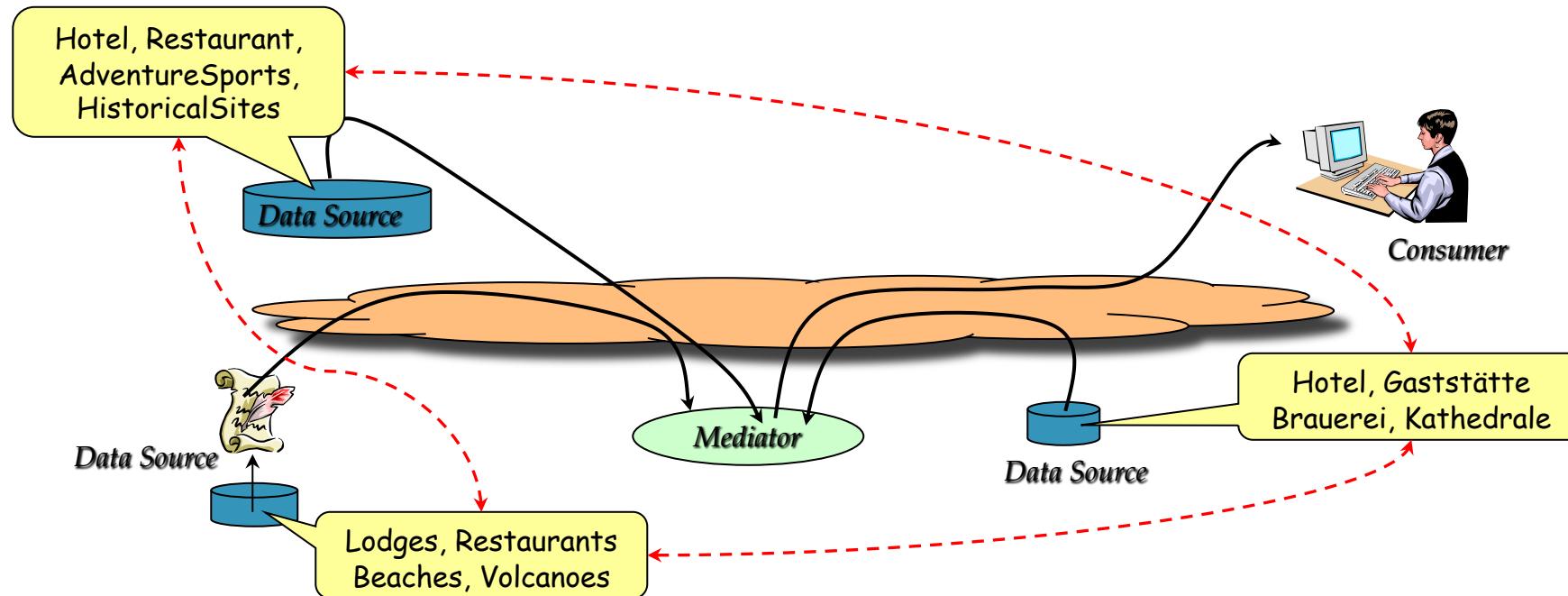
- Two different setups:
 1. Bring the data together into a single repository (often called data warehousing)
 - Relatively easier problem - only need one-way-mappings
 - Query performance predictable and under your control
 2. Keep the data where it is, and send queries back and forth
 - Need two-way mappings -- a query on the mediated schema needs to be translated into queries over data source schemas
 - Not as efficient and clean as data warehousing, but a better fit for dynamic data
 - Or when data warehousing is not feasible

Data Integration: Key Challenges

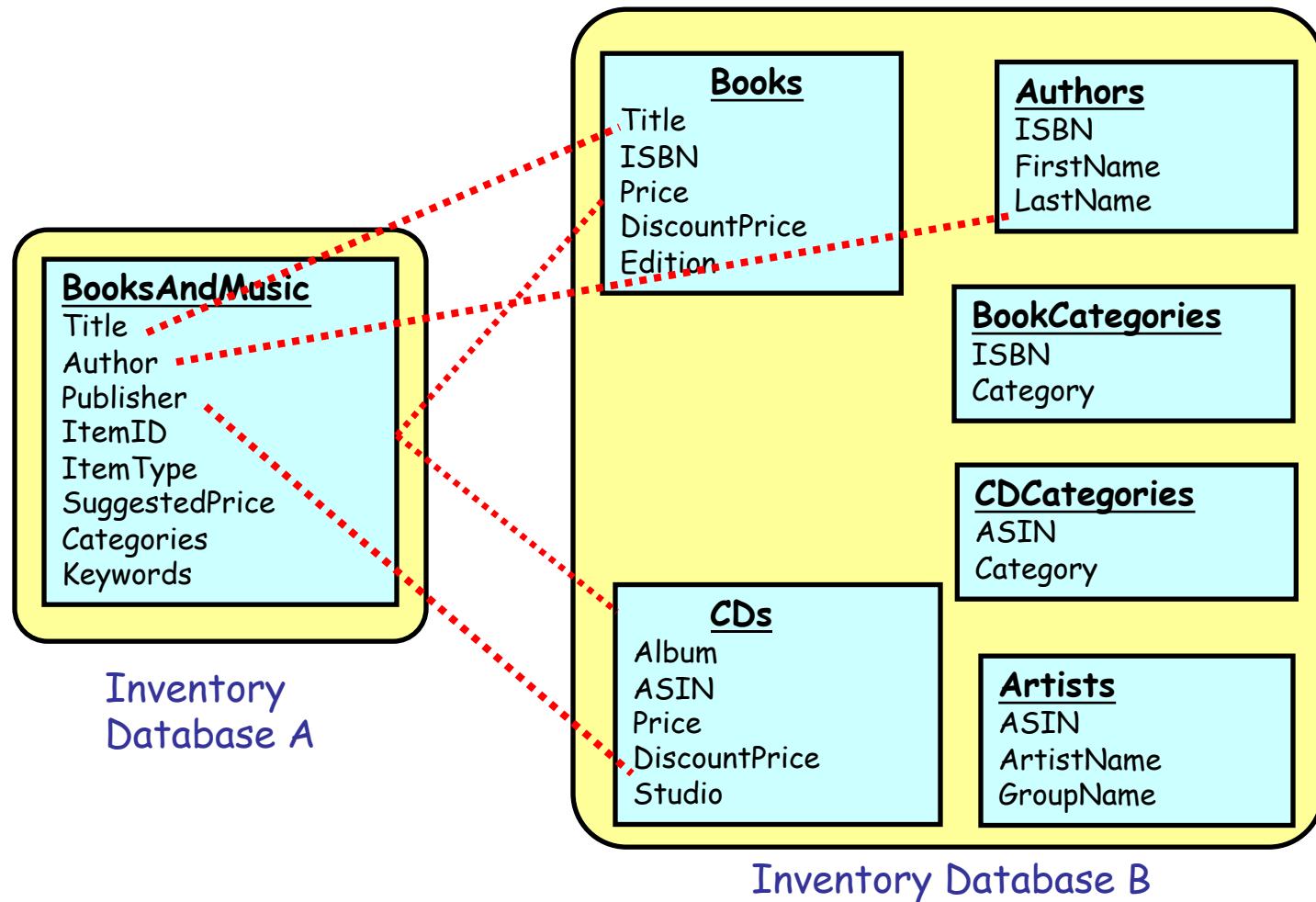
- Data extraction, reconciliation, and cleaning.
 - Get the data from each source in a structured form.
 - Need to use wrappers to extract data and define local schema.
- Schema alignment and mapping.
 - Figure out mappings / matchings between schemas.
 - Decide on the best mediated schema.
- Answer queries over the global schema.
 - Decide mapping a query on global schema onto queries over local schemas
 - Also need to decide which sources contain relevant data
- Limitations in mechanisms for accessing sources.
 - Many sources have limits on how you can access them!
 - Limits on the number of queries you can issues (say 100 per min)
 - Limits on the types of queries
 - e.g., must enter a zipcode to get information from a web source.

Schema Matching or Alignment

- Goal: Identify corresponding elements in two schemas
 - As a first step toward constructing a global schema
 - Schema heterogeneity is a key roadblock
 - Different data sources speak their own schema



Schema Matching or Alignment



Summary

- Data integration continues to be a very active area in research and increasingly industry.
 - E.g, how do we automatically extract and query open / competitor databases.
- Solutions still somewhat ad hoc and manual, although tools beginning to emerge.
 - AWS Glue, Watson Data Pipe, etc.
- Goal: minimize the time needed to integrate a new data source!
 - Crucial opportunities may be lost otherwise.
 - Can take weeks to do it properly.
- Dealing with changes to the data sources a major headache.
 - Especially for data sources not under your control.
 - As Project 1 is showing you...

Outline

- Data Integration
- Data Quality Issues
- Data Cleaning
 - Outlier Detection
 - Constraint Based Cleaning
 - Entity Resolution

Data Quality Problems

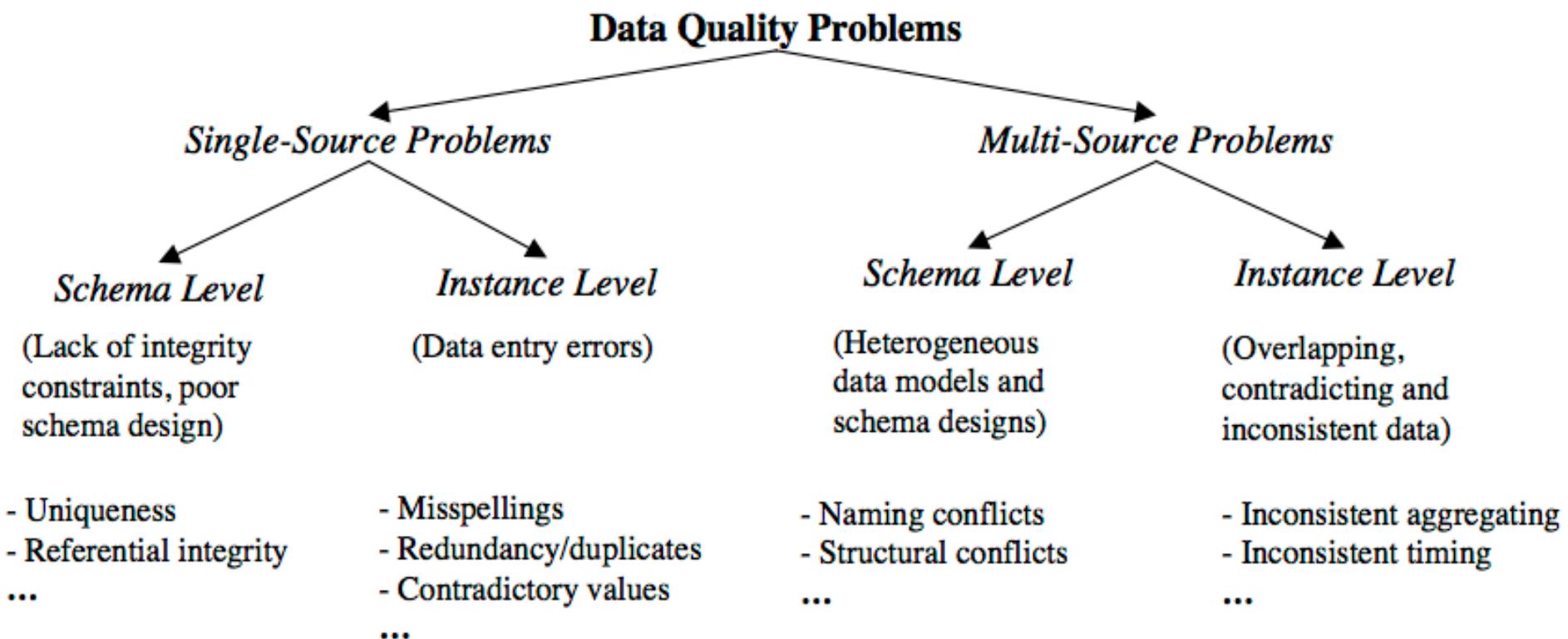


Figure 2. Classification of data quality problems in data sources

Single Source Problems

- Depends largely on the source – is it reliable or not?
- Databases can enforce constraints, whereas data extracted from files or spreadsheets, or scraped from webpages is much more messy.
- **Types of Problems:**
 - Ill-formatted data, especially from webpages or files or spreadsheets.
 - Missing or illegal values, Misspellings, Use of wrong fields, Extraction issues (not easy to separate out different fields).
 - Duplicated records, Contradicting Information, Referential Integrity Violations.
 - Unclear default values (e.g., data entry software needs something).
 - Evolving schemas or classification schemes (for categorical attributes).
 - Outliers.

Single Source Problems

Scope/Problem		Dirty Data	Reasons/Remarks
Attribute	Missing values	phone=9999-999999	unavailable values during data entry (dummy values or null)
	Misspellings	city="Liipzig"	usually typos, phonetic errors
	Cryptic values, Abbreviations	experience="B"; occupation="DB Prog."	
	Embedded values	name="J. Smith 12.02.70 New York"	multiple values entered in one attribute (e.g. in a free-form field)
	Misfielded values	city="Germany"	
Record	Violated attribute dependencies	city="Redmond", zip=77777	city and zip code should correspond
Record type	Word transpositions	name ₁ = "J. Smith", name ₂ = "Miller P."	usually in a free-form field
	Duplicated records	emp ₁ =(name="John Smith",...); emp ₂ =(name="J. Smith",...)	same employee represented twice due to some data entry errors
	Contradicting records	emp ₁ =(name="John Smith", bdate=12.02.70); emp ₂ =(name="John Smith", bdate=12.12.70)	the same real world entity is described by different values
Source	Wrong references	emp=(name="John Smith", deptno=17)	referenced department (17) is defined but wrong

Table 2. Examples for single-source problems at instance level

Multi-Source Problems

- Different sources are developed separately, and maintained by different people.
- Issue 1: Mapping information across sources (schema mapping/transformation).
 - Same issues as in data integration we saw before.
 - Naming conflicts: same name used for different objects
 - Structural conflicts: different representations across sources
- Issue 2: Entity Resolution: Matching entities across sources!
- Issue 3: Data quality issues.
 - Contradicting information, Mismatched information, etc.

Outline

- Data Integration
- Data Quality Issues
- Data Cleaning
 - Outlier Detection
 - Constraint Based Cleaning
 - Entity Resolution

Dealing with Univariate Outliers

- A set of values can be characterized by metrics such as center (e.g., mean), dispersion (e.g., standard deviation), and skew
- Can be used to identify outliers
 - Must watch out for "masking": one extreme outlier may alter the metrics sufficiently to mask other outliers
 - Should use **robust statistics**: considers effect of corrupted data values on distributions – we will talk about this in depth later
 - Robust center metrics: median, k% trimmed mean (discard lowest and highest k% values)
 - Robust dispersion:
 - Median Absolute Deviation (MAD): median distance of all the values from the median value
- A reasonable approach to find outliers: any data points 1.4826x MAD away from median
 - The above assumes that data follows a **normal** distribution
 - May need to eyeball the data (e.g., plot a histogram) to decide if this is true

Univariate Outliers

- [Wikipedia Article on Outliers](#) lists several other normality-based tests for outliers
- If data appears to be not normally distributed:
 - Distance-based methods: look for data points that do not have many neighbors
 - Density-based methods:
 - Define *density* to be average distance to k nearest neighbors
 - *Relative density* = density of node/average density of its neighbors
 - Use relative density to decide if a node is an outlier
- Most of these techniques start breaking down as the dimensionality of the data increases
 - *Curse of dimensionality* – too many different dimensions to look for outliers!
 - Can project data into lower-dimensional space and look for outliers there
 - Not as straightforward

Other Types of Outliers

- Timeseries outliers
 - Often the data is in the form of a timeseries
 - Can use the historical values/patterns in the data to flag outliers
 - Rich literature on *forecasting* in timeseries data
- Frequency-based outliers
 - An item is considered a "heavy hitter" if it is much more frequent than other items
 - In relational tables, can be found using a simple *groupby-count*
 - Often the volume of data may be too much (e.g., internet routers)
 - Approximation techniques often used
 - To be discussed sometime later in the class
- Things generally not as straightforward with other types of data
 - Outlier detection continues to be a major research area

Wrap-up

- Data wrangling/cleaning are a key component of data science pipeline
- Still largely ad hoc although much tooling in recent years
- Specifically, we covered:
 - Schema mapping and matching
 - Outliers
- Next up:
 - Constraint-based Cleaning
 - Entity Resolution/Record Linkage/Data Matching

Data Cleaning: Outlier Resolution

- Content from: [Entity Resolution Tutorial](#), VLDB 2012.
- **Goal:** Identify different manifestations of the same real world object
 - Also called: identity reconciliation, record linkage, deduplication, fuzzy matching, Object consolidation, Coreference resolution, and several others
- Motivating Examples ????
 - Postal addresses
 - Entity recognition in NLP/Information Extraction
 - Identifying companies in financial records
 - Comparison shopping
 - Author disambiguation in citation data
 - Connecting up accounts on online networks
 - Crime/Fraud Detection
 - Census
 - ...

Data Cleaning: Outlier Resolution

- Important to correctly identify references
 - Often actions taken based on extracted data
 - Cleaning up data by entity resolution can show structure that may not be apparent before
- Challenges
 - Such data is naturally ambiguous (e.g., names, postal addresses)
 - Abbreviations/data truncation
 - Data entry errors, Missing values, Data formatting issues complicate the problem
 - Heterogeneous data from many diverse sources
- No magic bullet here !!
 - Approaches fairly domain-specific
 - Be prepared to do a fair amount of manual work

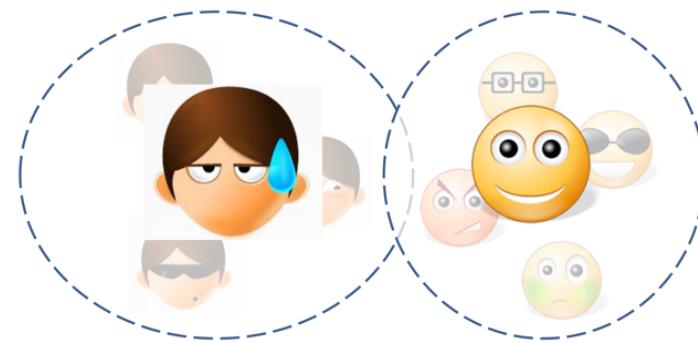
Entity Resolution: Three Slightly Different Problems

- **Setup.**

- Real world: there are entities (people, addresses, businesses).
- We have a large collection of noisy, ambiguous "references" to those entities (also called "mentions").
- Somewhat different techniques, but a lot of similarities.

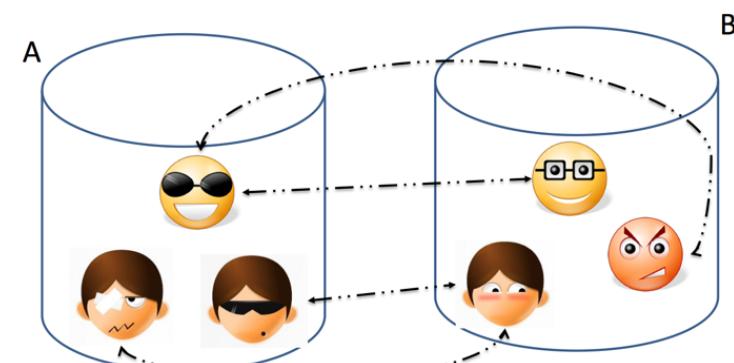
- **Deduplication.**

- Cluster records/mentions that correspond to the same entity
- Choose/construct a cluster representative
 - This is in itself a non-trivial task (e.g., averaging may work for numerical attributes, but what about string attributes?)



Entity Resolution: Three Slightly Different Problems

- **Setup.**
 - Real world: there are entities (people, addresses, businesses)
 - We have a large collection of noisy, ambiguous "references" to those entities (also called "mentions")
 - Somewhat different techniques, but a lot of similarities.
- **Record Linkage**
 - Match records across two different databases (e.g., two social networks, or financial records w/ campaign donations).
 - Typically assume that the two databases are fairly clean.



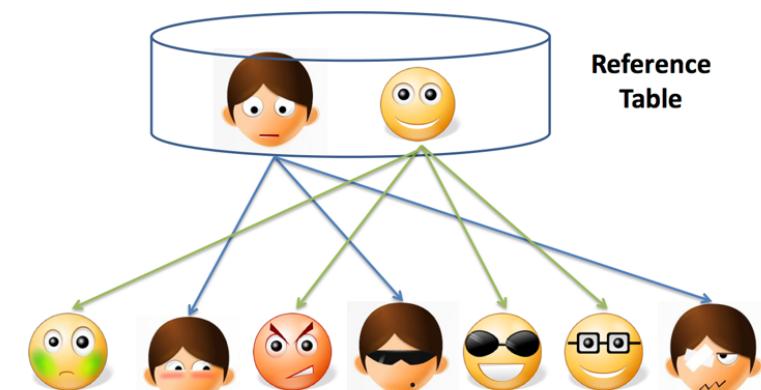
Entity Resolution: Three Slightly Different Problems

- **Setup.**

- Real world: there are entities (people, addresses, businesses)
- We have a large collection of noisy, ambiguous "references" to those entities (also called "mentions")
- Somewhat different techniques, but a lot of similarities.

- **Reference Matching.**

- Match "references" to clean records in a reference table.
- Commonly comes up in "entity recognition" (e.g., matching newspaper article mentions to names of people).



Entity Resolution: Data Matching

- Comprehensive treatment: Data Matching; P. Christen; 2012 (Springer).
- One of the key issues is finding similarities between two references
 - What similarity function to use?
- Edit Distance Functions
 - Levenshtein Distance: min number of changes to go from one reference to another.
 - Many adjustments to the basic idea proposed (e.g., higher weights to changes at the start)
 - Not cheap to compute, especially for millions of pairs
- Set Similarity
 - Some function of intersection size and union size
 - E.g., Jaccard distance = size of intersection / size of union
 - Much faster to compute
- Vector Similarity
 - Cosine similarity – we'll talk about this much more in NLP lectures

Entity Resolution: Data Matching

- Q-Grams
 - Find all length-q substrings in each string
 - Use set/vector similarity on the resulting set
- Several approaches that combine the above (especially q-grams and edit distance, e.g., Jaro-Winkler)
- Soundex: Phonetic Similarity Metric
 - Homophones should be encoded to the same representation so spelling errors can be handled
 - Robert and Rupert get assigned the same code (R163), but Rubin yields R150
- May need to use Translation Tables
 - To handle abbreviations, nicknames, other synonyms
- Different types of data requires more domain-specific functions
 - E.g., geographical locations, postal addresses
 - Also much work on computing distances between XML documents etc.

Entity Resolution: Algorithms

- Simple threshold method
 - If the distance below some number, the two references are assumed to be equal
 - May review borderline matches manually
- Can be generalized to rule-based:
 - Example from Christen, 2012

$$\begin{aligned}
 & (s(\text{GivenName})[r_i, r_j] \geq 0.9) \wedge (s(\text{Surname})[r_i, r_j] = 1.0) \\
 & \quad \wedge (s(\text{BMonth})[r_i, r_j] = 1.0) \wedge (s(\text{BYear})[r_i, r_j] = 1.0) \Rightarrow [r_i, r_j] \rightarrow \text{Match} \\
 & (s(\text{GivenName})[r_i, r_j] \geq 0.7) \wedge (s(\text{Surname})[r_i, r_j] \geq 0.8) \\
 & \quad \wedge (s(\text{BDay})[r_i, r_j] = 1.0) \wedge s(\text{BMonth})[r_i, r_j] = 1.0 \\
 & \quad \quad \wedge (s(\text{BYear})[r_i, r_j] = 1.0) \Rightarrow [r_i, r_j] \rightarrow \text{Match} \\
 & (s(\text{GivenName})[r_i, r_j] \geq 0.7) \wedge (s(\text{Surname})[r_i, r_j] \geq 0.8) \\
 & \quad \wedge (s(\text{StrName})[r_i, r_j] \geq 0.8) \wedge (s(\text{Suburb})[r_i, r_j] \geq 0.8) \Rightarrow [r_i, r_j] \rightarrow \text{Match} \\
 & (s(\text{GivenName})[r_i, r_j] \geq 0.7) \wedge (s(\text{Surname})[r_i, r_j] \geq 0.8) \\
 & \quad \wedge (s(\text{BDay})[r_i, r_j] \leq 0.5) \wedge (s(\text{BMonth})[r_i, r_j] \leq 0.5) \\
 & \quad \quad \wedge (s(\text{BYear})[r_i, r_j] \leq 0.5) \Rightarrow [r_i, r_j] \rightarrow \text{Non-Match} \\
 & (s(\text{GivenName})[r_i, r_j] \geq 0.7) \wedge (s(\text{Surname})[r_i, r_j] \geq 0.8) \\
 & \quad \wedge (s(\text{StrName})[r_i, r_j] \leq 0.6) \wedge (s(\text{Suburb})[r_i, r_j] \leq 0.6) \Rightarrow [r_i, r_j] \rightarrow \text{Non-Match}
 \end{aligned}$$

Entity Resolution: Algorithms

- May want to give more weight to matches involving rarer words
 - More naturally applicable to record linkage problem
 - If two records match on a rare name like "Machanavajjhala", they are likely to be a match
 - Can formalize this as "probabilistic record linkage"
- Constraints: May need to be satisfied, but can also be used to find matches
 - Often have constraints on the matching possibilities
 - Transitivity: M1 and M2 match, and M2 and M3 match, and M1 and M3 must match
 - Exclusivity: M1 and M2 match \rightarrow M3 cannot match with M2
 - Other types of constraints:
 - E.g., if two papers match, their venues must match

Entity Resolution: Algorithms

- Clustering-based ER Techniques:
 - Deduplication is basically a clustering problem
 - Can use clustering algorithms for this purpose
 - But most clusters are very small (in fact of size = 1)
 - Some clustering algorithms are better suited for this, especially Agglomerative Clustering
 - Unlike K-Means would work here

Entity Resolution: Algorithms

- Crowdsourcing
 - Humans are often better at this task
 - Can use one of the crowdsourcing mechanisms (e.g., Mechanical Turk) for getting human input on the difficult pairs
 - Quite heavily used commercially (e.g., to disambiguate products, restaurants, etc.)

Entity Resolution: Scaling to Big Data

- One immediate problem
 - There are $O(N^2)$ possible matches
 - Must reduce the search space
- Use some easy-to-evaluate criterion to restrict the pairs considered further
 - May lead to false negative (i.e., missed matches) depending on how noisy the data is
- Much work on this problem as well, but domain-specific knowledge likely to be more useful in practice
- One useful technique to know: min-hash signatures
 - Can quickly find potentially overlapping sets
 - Turns up to be very useful in many domains (beyond ER)