

Advanced Data Management (CSCI 640/490)

Data Integration

Dr. David Koop

Data Cleaning Types

- How can statistical techniques improve efficiency or reliability of data cleaning? (Data Cleaning **with** Statistics)
 - Example: Trifacta
 - Two tasks: Error Detection & Data Repairing
- How how can we improve the reliability of statistical analytics with data cleaning? (Data Cleaning **for** Statistics)
 - Example: SampleClean
 - Task: Do statistics and clean along the way
- Similar questions if we substitute machine learning for statistics

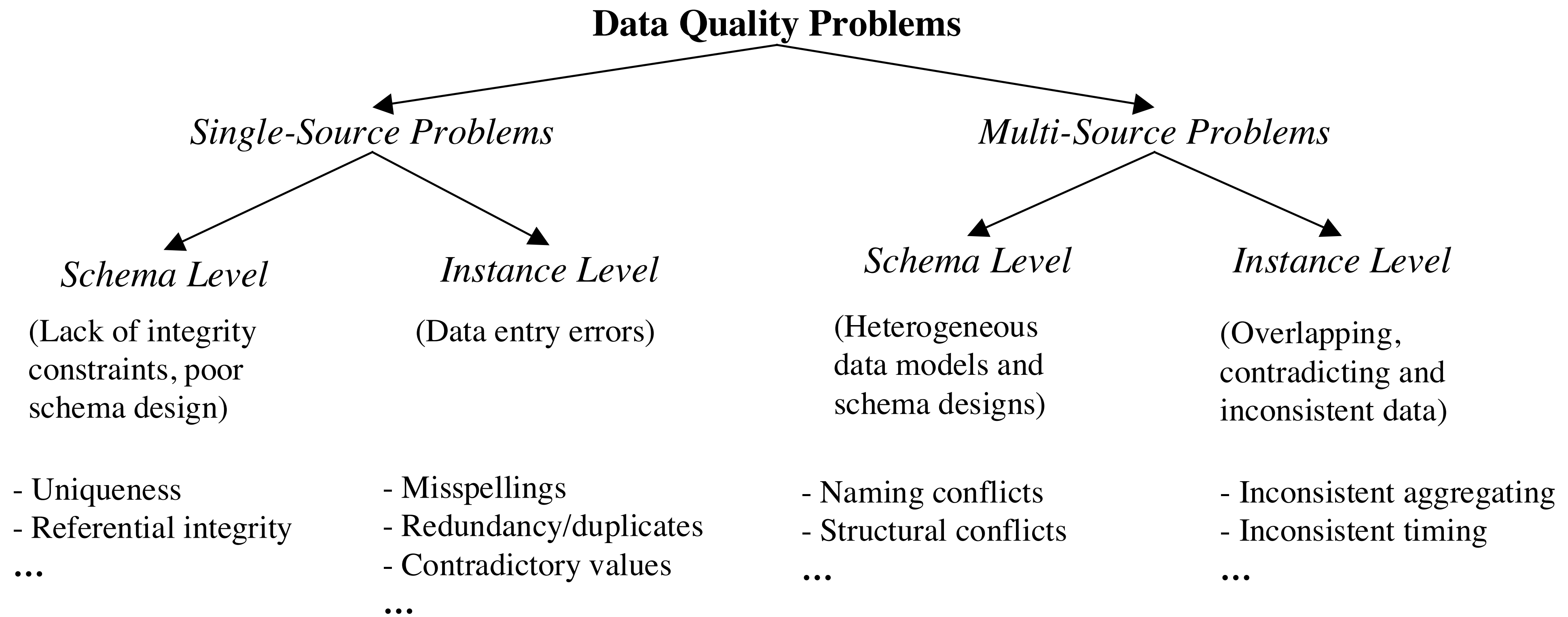
[D. Haas et al., 2016]

Misconceptions about Data Cleaning

- The end goal of data cleaning is clean data
- Data cleaning is a sequential operation
- Data cleaning is performed by one person
- Data quality is easy to evaluate

[D. Haas et al., 2016]

Classifying Data Quality Problems



[E. Rahm & H. H. Do, 2000]

Dirty and Cleaned Data

(a) Dirty Data

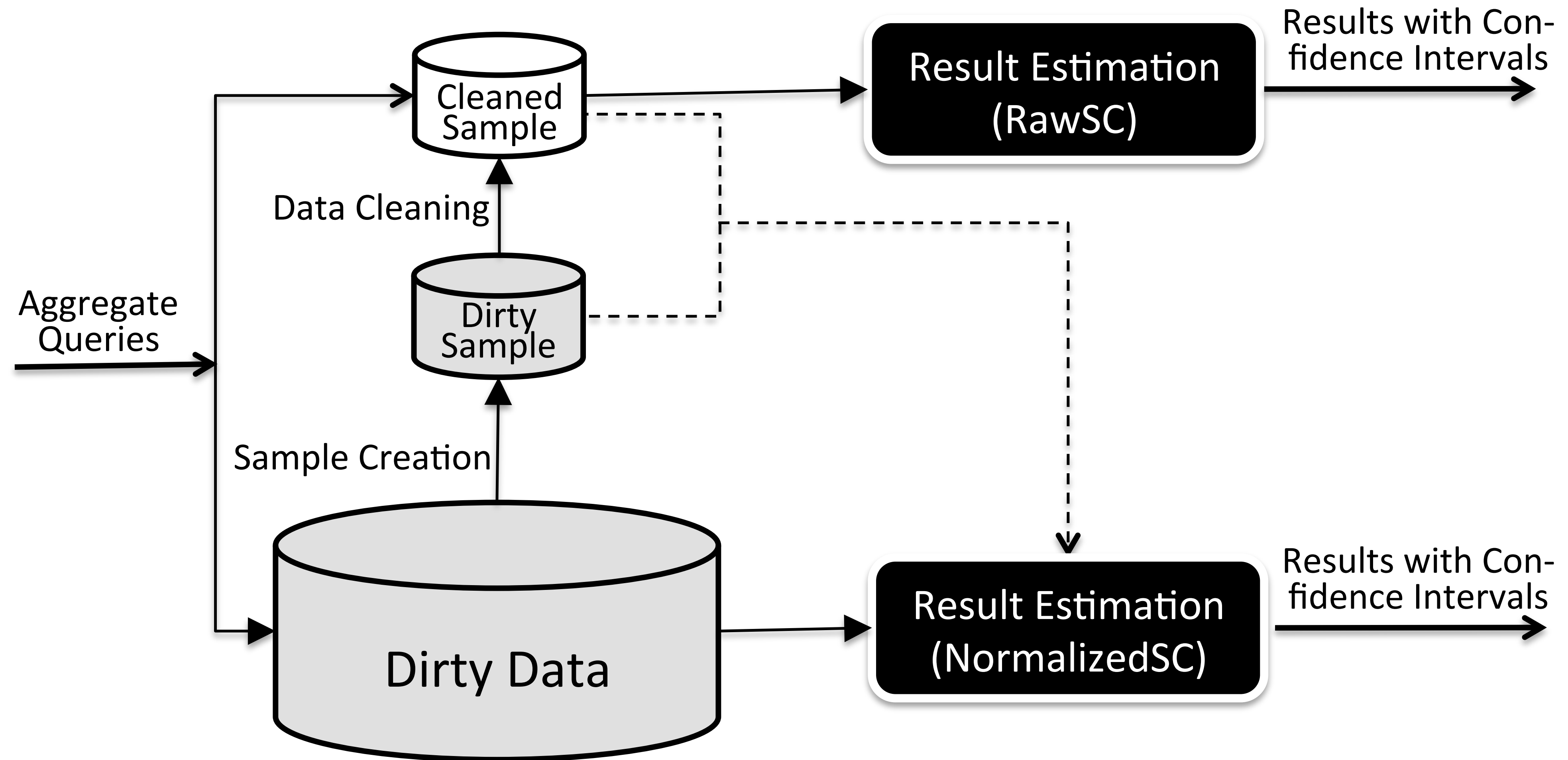
id	title	pub_year	citation_count
<i>t</i> ₁	CrowdDB	11	18
<i>t</i> ₂	TinyDB	2005	1569
<i>t</i> ₃	YFilter	Feb, 2002	298
<i>t</i> ₄	Aqua		106
<i>t</i> ₅	DataSpace	2008	107
<i>t</i> ₆	CrowdER	2012	1
<i>t</i> ₇	Online Aggr.	1997	687
...	...	...	...
<i>t</i> ₁₀₀₀₀	YFilter - ICDE	2002	298

(b) Cleaned Sample

id	title	pub_year	citation_count	#dup
<i>t</i> ₁	CrowdDB	2011	144	2
<i>t</i> ₂	TinyDB	2005	1569	1
<i>t</i> ₃	YFilter	2002	298	2
<i>t</i> ₄	Aqua	1999	106	1
<i>t</i> ₅	DataSpace	2008	107	1
<i>t</i> ₆	CrowdER	2012	34	1
<i>t</i> ₇	Online Aggr.	1997	687	3

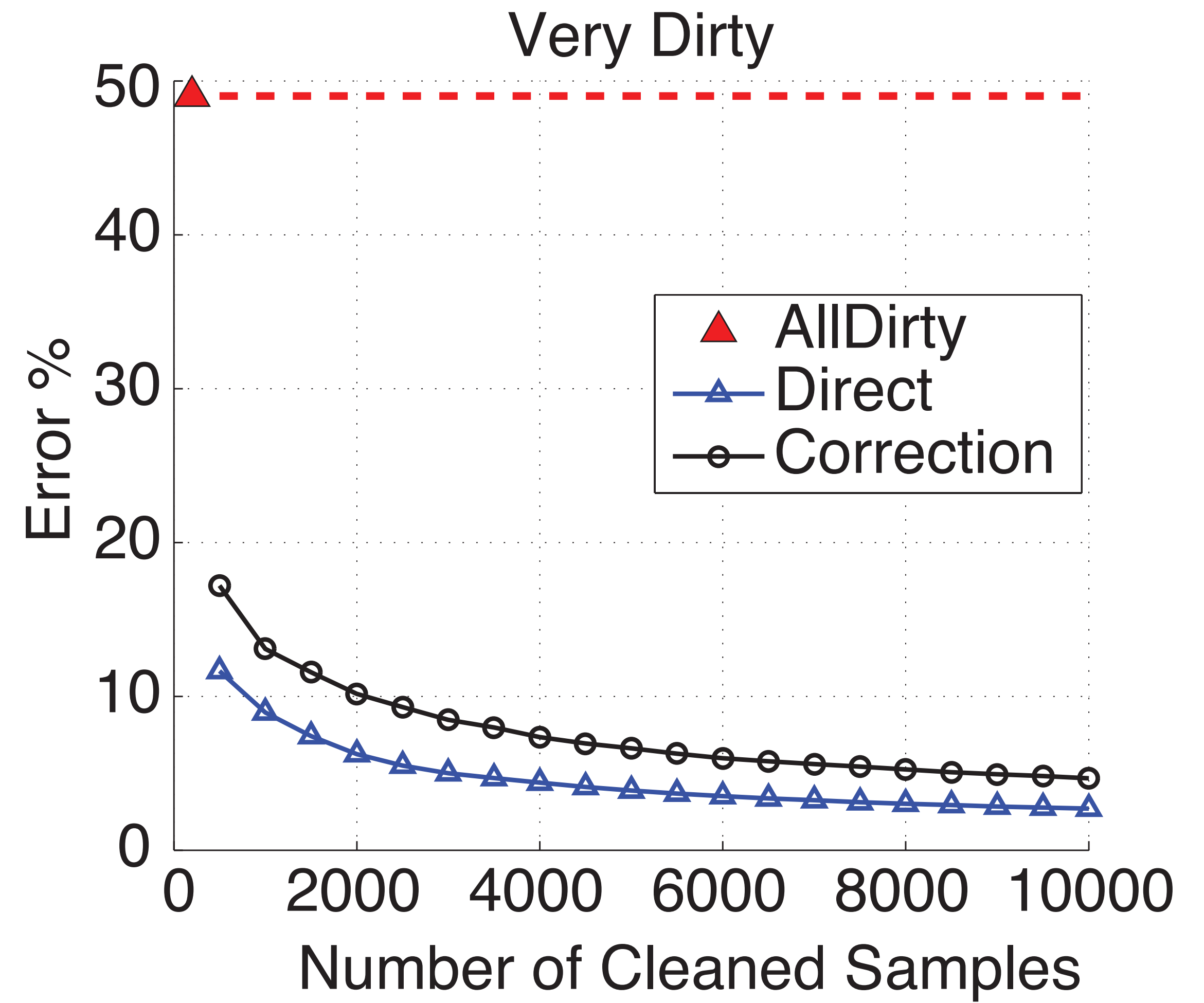
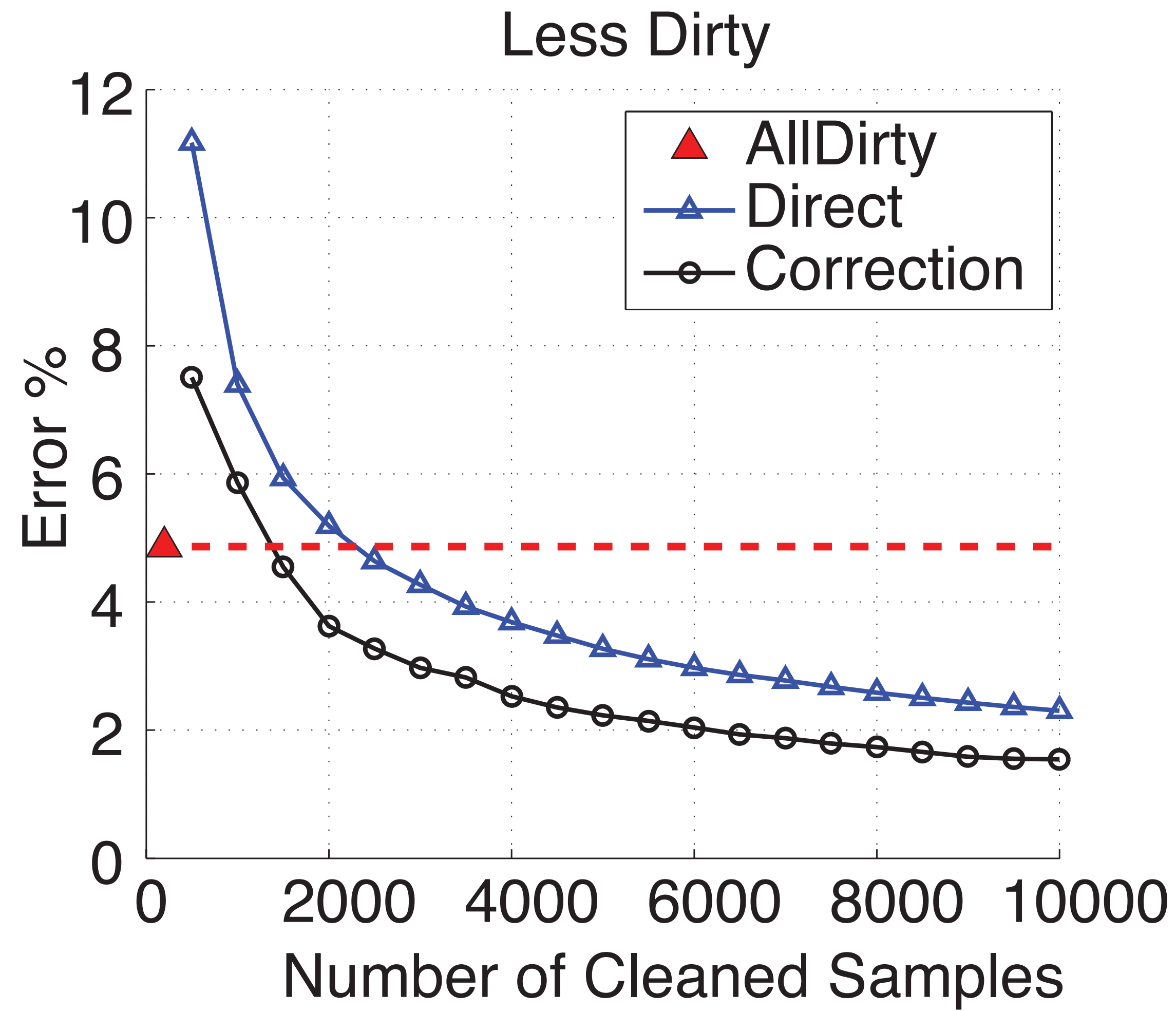
[J. Wang et al., 2014]

SampleClean Framework



[J. Wang et al., 2014]

Comparing the Two Approaches



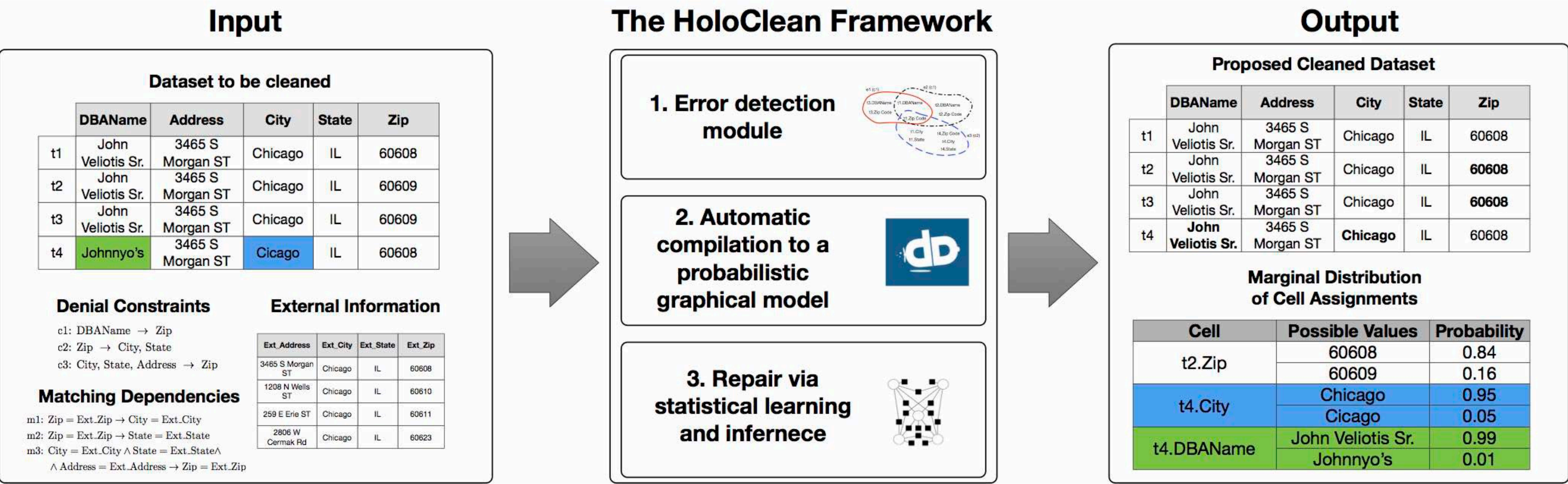
[S. Krishnan et al., 2015]

HoloClean

- A holistic data cleaning framework that combines qualitative methods with quantitative methods:
 - Qualitative: use integrity constraints or external data sources
 - Quantitative: use statistics of the data
- Driven by probabilistic inference. Users only need to provide a dataset to be cleaned and describe high-level domain specific signals.
- Can scale to large real-world dirty datasets and perform automatic repairs with high accuracy

[T. Rekatsinas et al., 2017]

HoloClean



Pandas Operations

- Filtering out missing data (`dropna`):
 - Can choose rows or columns
- Filling in missing data (`fillna`):
- Finding problems in data (statistics, special values)
- Filtering duplicates: (`drop_duplicates`, `unique`)
- Cleaning data: (`map`, `replace`, clamping data)

Assignment 3

- Salary Data
- Use Pandas (not loops)
- Part 2: CSCI 640 students need to do (b), CSCI 490 students can choose
- Part 5: use melt/pivot or a similar high-level operation

Outline

- Combining Data
- Data Integration
- Data Matching (Entity Resolution)
- Data Fusion
- Data Fusion Techniques
 - Integrating Conflicting Data: The Role of Source Dependence, X. L. Dong et al., 2009
 - **Quiz** at the beginning of class

Example: Football Game Data

- Data about football games, teams, & players
 - Game is between two Teams
 - Each Team has Players
- For each game, we could specify every player and all of their information... why is this bad?

Example: Football Game Data

- Data about football games, teams, & players
 - Game is between two Teams
 - Each Team has Players
- For each game, we could specify every player and all of their information... why is this bad?
- Normalization: reduce redundancy, keep information that doesn't change separate
- 3 Relations: Team, Player, Game
- Each relation only encodes the data specific to what it represents

Player

Id	Name	Height	Weight
----	------	--------	--------

Team

Id	Name	Wins	Losses
----	------	------	--------

Game

Id	Location	Date
----	----------	------

Example: Football Game Data

- Have each game store the id of the home team and the id of the away team (one-to-one)
- Have each player store the id of the team he plays on (many-to-one)
- What happens if a player plays on 2+ teams?

Player

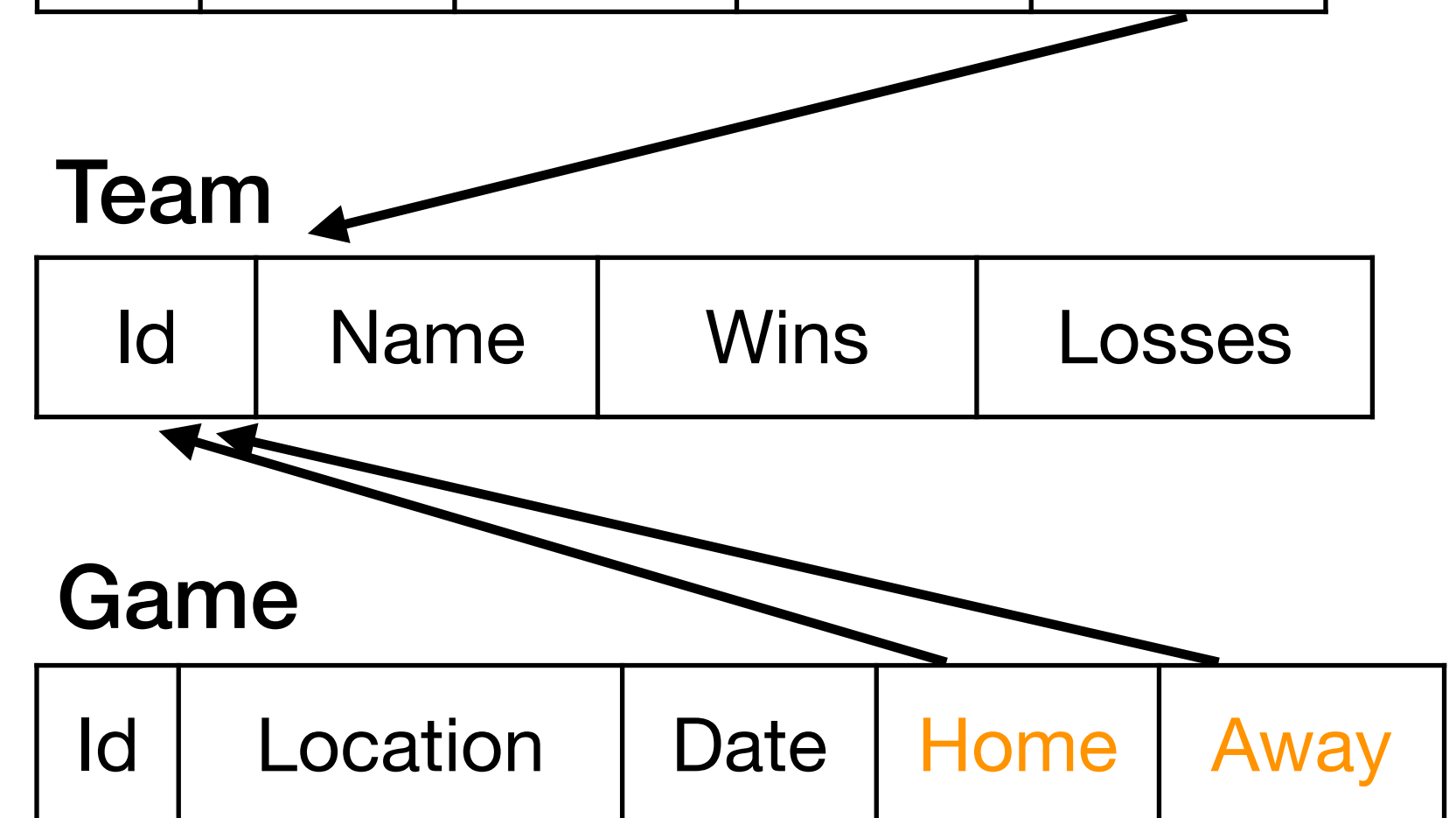
Id	Name	Height	Weight	TeamId
----	------	--------	--------	--------

Team

Id	Name	Wins	Losses
----	------	------	--------

Game

Id	Location	Date	Home	Away
----	----------	------	------	------



How does this relate to pandas?

- DataFrames in pandas are ~relations (tables)
- We may wish to normalize data in a similar manner in pandas
- However, operating on 2+ DataFrames at the same time can be unwieldy, can we merge them together?
- Two potential operations:
 - Have football game data (just the Game table) from 2013, 2014, and 2015 and wish to merge the data into one data frame
 - Have football game data and wish to find the average temperature of the cities where the games were played

Concatenation

- Take two data frames with the same columns and add more rows
- `pd.concat([data-frame-1, data-frame-2, ...])`
- Default is to add rows (`axis=0`), but can also add columns (`axis=1`)
- Can also concatenate Series into a data frame.
- `concat` preserves the index so this can be confusing if you have two default indices (0,1,2,3...)—they will appear twice
 - Use `ignore_index=True` to get a 0,1,2...

Merges (aka Joins)

- Need to merge data from one DataFrame with data from another DataFrame
- Example: Football game data merged with temperature data

Game

Id	Location	Date	Home	Away
0	Boston	9/2	1	15
1	Boston	9/9	1	7
2	Cleveland	9/16	12	1
3	San Diego	9/23	21	1

Weather

wld	City	Date	Temp
0	Boston	9/2	72
1	Boston	9/3	68
...	...	...	...
7	Boston	9/9	75
...	...	...	...
21	Boston	9/23	54
...	...	...	...
36	Cleveland	9/16	81

No data for San Diego



Merges (aka Joins)

- Want to join the two tables based on the location and date
- Location and date are the **keys** for the join
- What happens when we have missing data?
- Merges are **ordered**: there is a left and a right side
- Four types of joins:
 - Inner: intersection of keys (match on both sides)
 - Outer: union of keys (if there is no match on other side, still include with NaN to indicate missing data)
 - Left: always have rows from left table (no unmatched right data)
 - Right: like left, but with no unmatched left data

Inner Strategy

Merged

Id	Location	Date	Home	Away	Temp	wld
0	Boston	9/2	1	15	72	0
1	Boston	9/9	1	7	75	7
2	Cleveland	9/16	12	1	81	36

No San Diego entry

Outer Strategy

Merged

Id	Location	Date	Home	Away	Temp	wld
0	Boston	9/2	1	15	72	0
NaN	Boston	9/3	NaN	NaN	68	1
...	...	...	...	...	...	...
1	Boston	9/9	1	7	75	7
NaN	Boston	9/10	NaN	NaN	76	8
...	...	...	...	...	...	...
NaN	Cleveland	9/2	NaN	NaN	61	22
...	...	...	...	...	...	...
2	Cleveland	9/16	12	1	81	36
...	...	...	...	...	...	...
3	San Diego	9/23	21	1	NaN	NaN

Left Strategy

Merged

Id	Location	Date	Home	Away	Temp	wld
0	Boston	9/2	1	15	72	0
1	Boston	9/9	1	7	75	7
2	Cleveland	9/16	12	1	81	36
3	San Diego	9/23	21	1	NaN	NaN

Right Strategy

Merged

Id	Location	Date	Home	Away	Temp	wld
0	Boston	9/2	1	15	72	0
NaN	Boston	9/3	NaN	NaN	68	1
...	...	...	...	...	...	...
1	Boston	9/9	1	7	75	7
NaN	Boston	9/10	NaN	NaN	76	8
...	...	...	...	...	...	...
NaN	Cleveland	9/2	NaN	NaN	61	22
...	...	...	...	...	...	...
2	Cleveland	9/16	12	1	81	36
...	...	...	...	...	...	...

No San Diego entry

Data Merging in Pandas

- `pd.merge(left, right, ...)` or `left.merge(right, ...)`
- Default merge: join on matching column names
- Better: specify the column name(s) to join on via `on` kwarg
 - If column names differ, use `left_on` and `right_on`
 - Multiple keys: use a list
- `how` kwarg specifies type of join (`"inner"`, `"outer"`, `"left"`, `"right"`)
- Can add suffixes to column names when they appear in both tables, but are not being joined on
- Can also merge using the index by setting `left_index` or `right_index` to `True`

Merge Arguments

Argument	Description
<code>left</code>	DataFrame to be merged on the left side.
<code>right</code>	DataFrame to be merged on the right side.
<code>how</code>	One of 'inner', 'outer', 'left', or 'right'; defaults to 'inner'.
<code>on</code>	Column names to join on. Must be found in both DataFrame objects. If not specified and no other join keys given, will use the intersection of the column names in <code>left</code> and <code>right</code> as the join keys.
<code>left_on</code>	Columns in <code>left</code> DataFrame to use as join keys.
<code>right_on</code>	Analogous to <code>left_on</code> for <code>left</code> DataFrame.
<code>left_index</code>	Use row index in <code>left</code> as its join key (or keys, if a MultiIndex).
<code>right_index</code>	Analogous to <code>left_index</code> .
<code>sort</code>	Sort merged data lexicographically by join keys; <code>True</code> by default (disable to get better performance in some cases on large datasets).
<code>suffixes</code>	Tuple of string values to append to column names in case of overlap; defaults to ('_x', '_y') (e.g., if 'data' in both DataFrame objects, would appear as 'data_x' and 'data_y' in result).
<code>copy</code>	If <code>False</code> , avoid copying data into resulting data structure in some exceptional cases; by default always copies.
<code>indicator</code>	Adds a special column <code>_merge</code> that indicates the source of each row; values will be 'left_only', 'right_only', or 'both' based on the origin of the joined data in each row. [W. McKinney, Python for Data Analysis]

Outline

- ~~Combining Data~~
- Data Integration
- Data Matching (Entity Resolution)
- Data Fusion
- Data Fusion Techniques
 - Integrating Conflicting Data: The Role of Source Dependence,
X. L. Dong et al., 2009
 - **Quiz** at the beginning of class

Introduction to Data Integration

A. Doan, A. Halevy, and Z. Ives

Data Integration

```
select title, startTime
from Movie, Plays
where Movie.title=Plays.movie AND
        location="New York" AND
        director="Woody Allen"
```

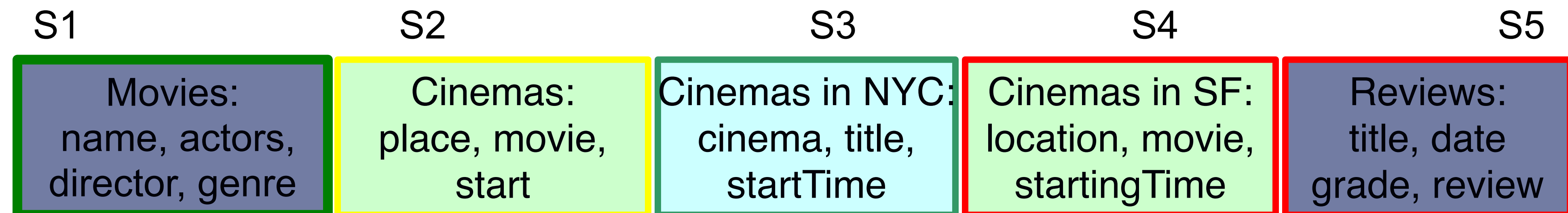
Movie: Title, director, year, genre

Actors: title, actor

Plays: movie, location, startTime

Reviews: title, rating, description

Sources S1 and S3 are relevant, sources S4 and S5 are irrelevant, and source S2 is relevant but possibly redundant.



[AH Doan et al., 2012]

Data Matching & Data Fusion

- Google Thinks I'm Dead
(I know otherwise.) [R. Abrams, NYTimes, 2017]
- Not only Google, but also Alexa:
 - "Alexa replies that Rachel Abrams is a sprinter from the Northern Mariana Islands (which is true of someone else)."
 - "He asks if Rachel Abrams is deceased, and Alexa responds yes, citing information in the Knowledge Graph panel."

Me ↓

could be me...? →

Rachel Abrams
American writer

Rachel Abrams was an American writer, editor, and artist. She was the wife of Elliott Abrams. [Wikipedia](#)

Born: January 2, 1951

Died: June 7, 2013

Spouse: Elliott Abrams (m. 1980–2013)

Parents: Midge Decter

Children: Sarah Abrams, Jacob Abrams, Joseph Abrams

Not me {

Definitely not me ←

People also search for



Data Integration, Data Matching, & Data Fusion

- Data Integration: focus on integrating data from different sources
- Data Matching (aka Entity Resolution aka Record Linkage):
want to know that two entities (often in different sources) are the same "real" entity
- When sources are orthogonal, no problems
- What happens when two sources provide the same type of information and they **conflict**?
- Data Fusion: create a single object while resolving conflicting values

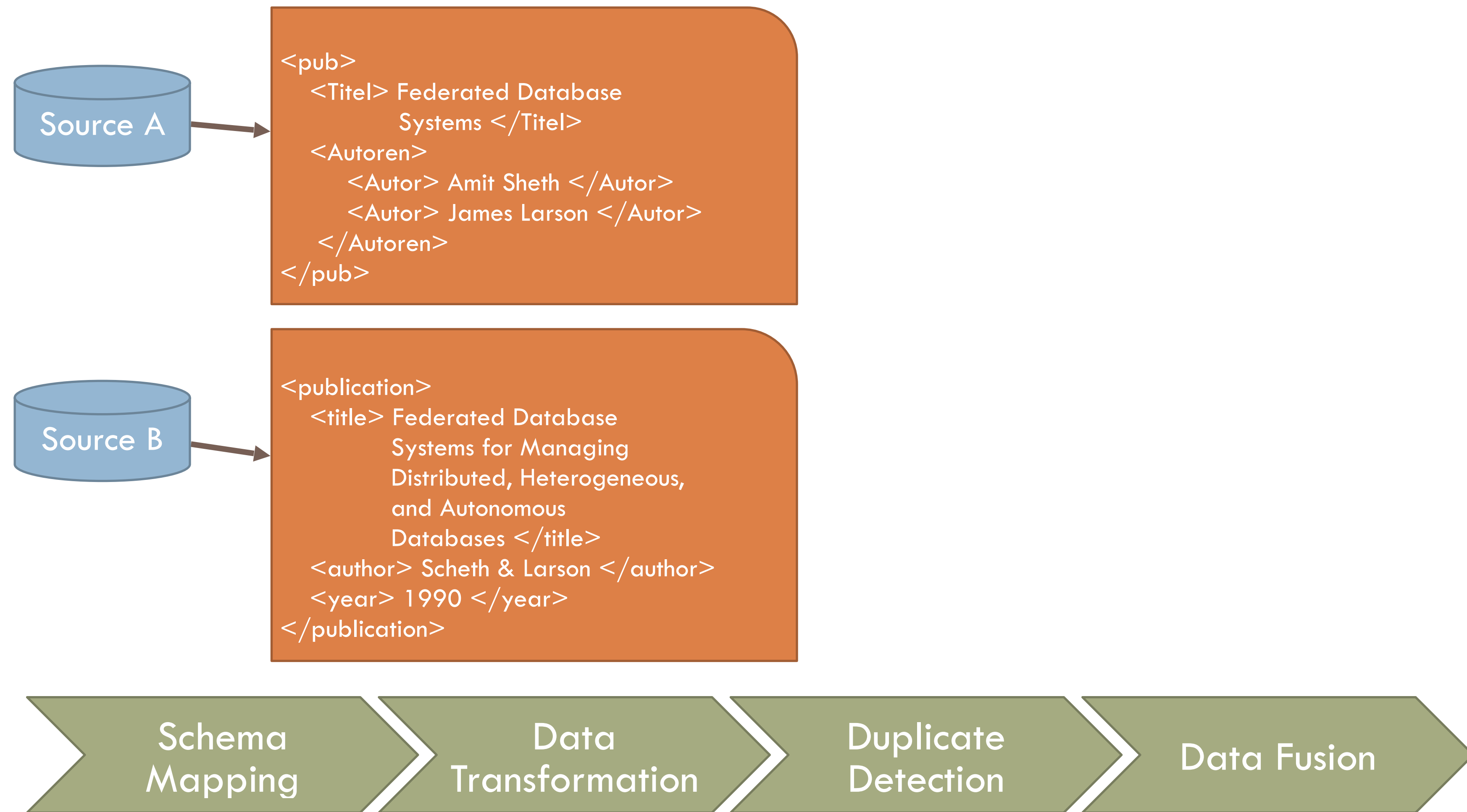
Outline

- ~~Combining Data~~
- ~~Data Integration~~
- ~~Data Matching (Entity Resolution)~~
- Data Fusion
- Data Fusion Techniques
 - Integrating Conflicting Data: The Role of Source Dependence, X. L. Dong et al., 2009
 - **Quiz** at the beginning of class

Data Fusion— Resolving Data Conflicts in Integration

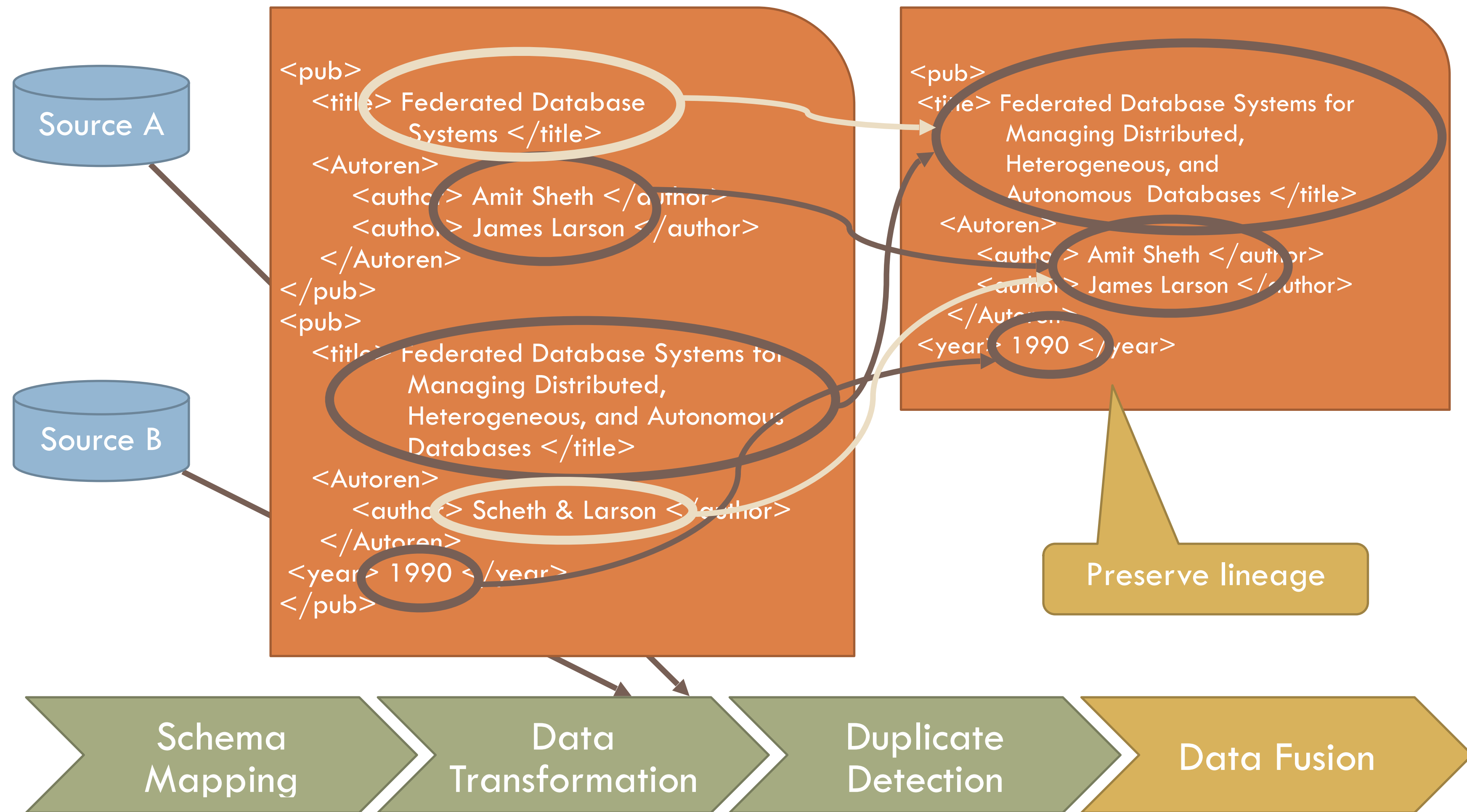
X. L. Dong and F. Naumann

Information Integration



[L. Dong and F. Naumann, 2009]

Information Integration



[L. Dong and F. Naumann, 2009]

Outline

- ~~Combining Data~~
- ~~Data Integration~~
- ~~Data Matching (Entity Resolution)~~
- ~~Data Fusion~~
- Data Fusion Techniques
 - Integrating Conflicting Data: The Role of Source Dependence,
X. L. Dong et al., 2009
 - **Quiz** at the beginning of class