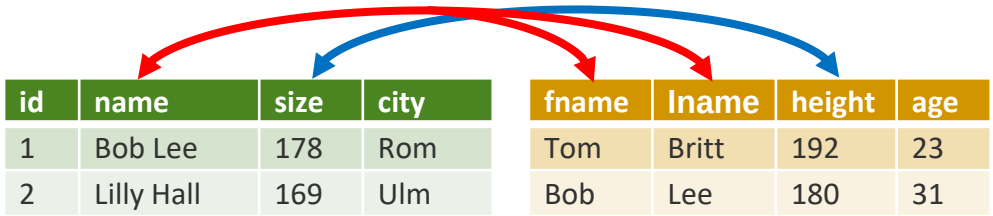


| id | name | size | city |
|---|---|---|---|
| 1 | Bob Lee | 178 | Rom |
| 2 | Lilly Hall | 169 | Ulm |

| fname | lname | height | age |
|---|---|---|---|
| Tom | Britt | 192 | 23 |
| Bob | Lee | 180 | 31 |

Matching:
*{name}* ↔ *{fname,lname}*
*{size}* ↔ *{height}*

# Constraint-based Schema Matching

Felix Naumann, Fabian Panse, Matteo Paganelli

WS 2023/2024

# Agenda

- Chair Introduction
- Organizational Information
- Schema Matching & Constraints
- Project Procedure

# Agenda

- ❑ Chair Introduction
- ❑ Organizational Information
- ❑ Schema Matching & Constraints
- ❑ Project Procedure

# Information Systems Team


Sebastian **Schmidl**


Phillip **Wenig**

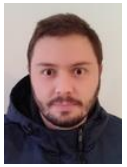

Diana **Stephan**


Prof. Felix **Naumann**


Dr. Matteo **Paganelli**


Dr. Fabian **Panse**


Tobias **Bleifuß**


Alejandro **Sierra-Múnera**


Leon **Bornemann**


Sedir **Mohammed**


Gerardo **Vitagliano**


Mazhar **Hameed**


Daniel **Lindner**


Youri **Kaminsky**

**Data Change** **Data Fusion** **Duplicate Detection**
project **AKITA** **Entity Search**
**Data Profiling** **Information Integration**
project **DataKnoller** **Web Science**
**Data Scrubbing** **Data as a Service**
project **AI4ART**
**Information Quality** **Data Cleansing** **Text Mining**
**Dependency Detection** **Linked Open Data**
**CSV parsing**
**Distributed Computing**
**Web Data** **Knowledge Management for the Arts**
project **Janus**
**Entity Recognition**
project **Metanome** **Data Preparation**
**Change Exploration**

# Agenda

- ❑ Chair Introduction
- ❑ Organizational Information
- ❑ Schema Matching & Constraints
- ❑ Project Procedure

# Course Information

- **General Information**
  - ECTS: 12
  - Total working time: 360 h (≃ 24 h per week)
  - Language: English

# Project Tasks

- **Familiarization** with the topics of
  - schema matching
  - integrity constraints
- Implementing/Reproducing **baseline schema matching** approaches
- Implementing **novel schema matching approach** using **ICs**
- **Evaluating** and **comparing** implemented approaches
- Writing **scientific paper** and final **presentation**

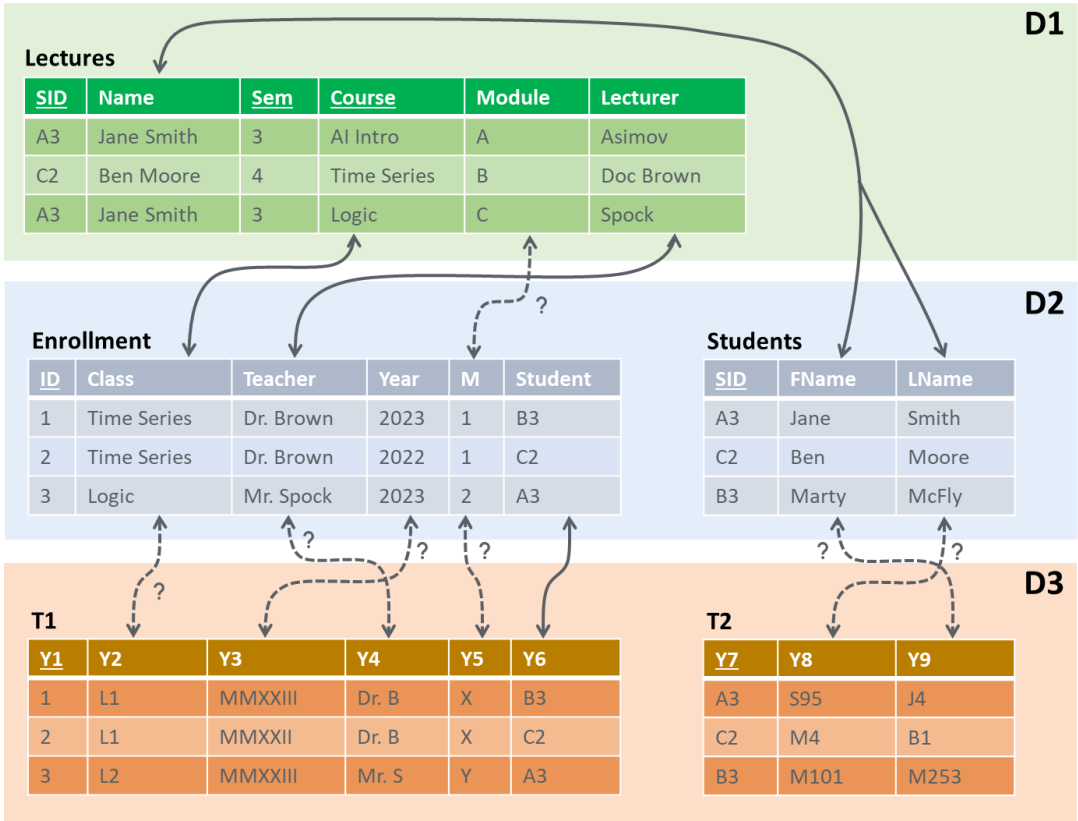
- **Project Artefacts:**
  - Schema Matching prototype (code!)
  - Scientific writeup of the results (paper!)

# Agenda

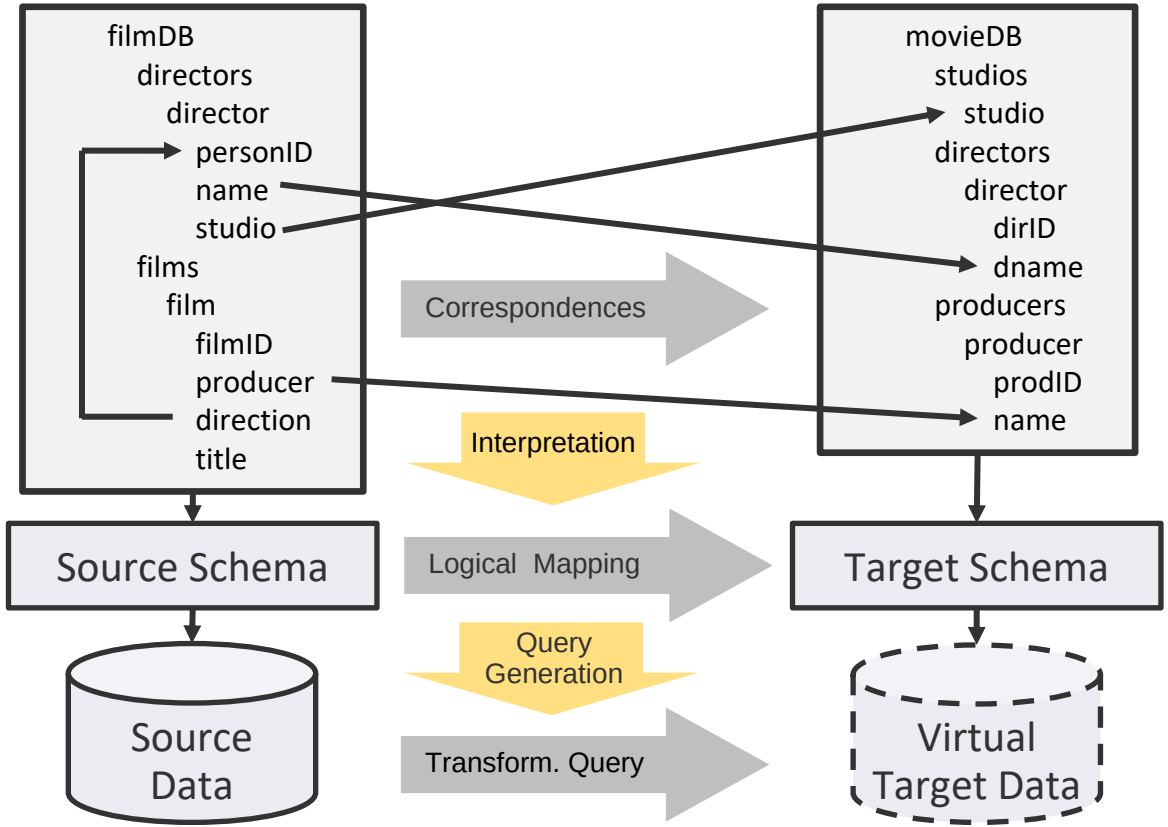
- ❑ Chair Introduction
- ❑ Organizational Information
- ❑ Schema Matching & Constraints
- ❑ Project Procedure

# Why Schema Matching?

- Data Integration to improve data quality
- Schema matching required to align data sources to be integrated



# Schema Matching and Mapping



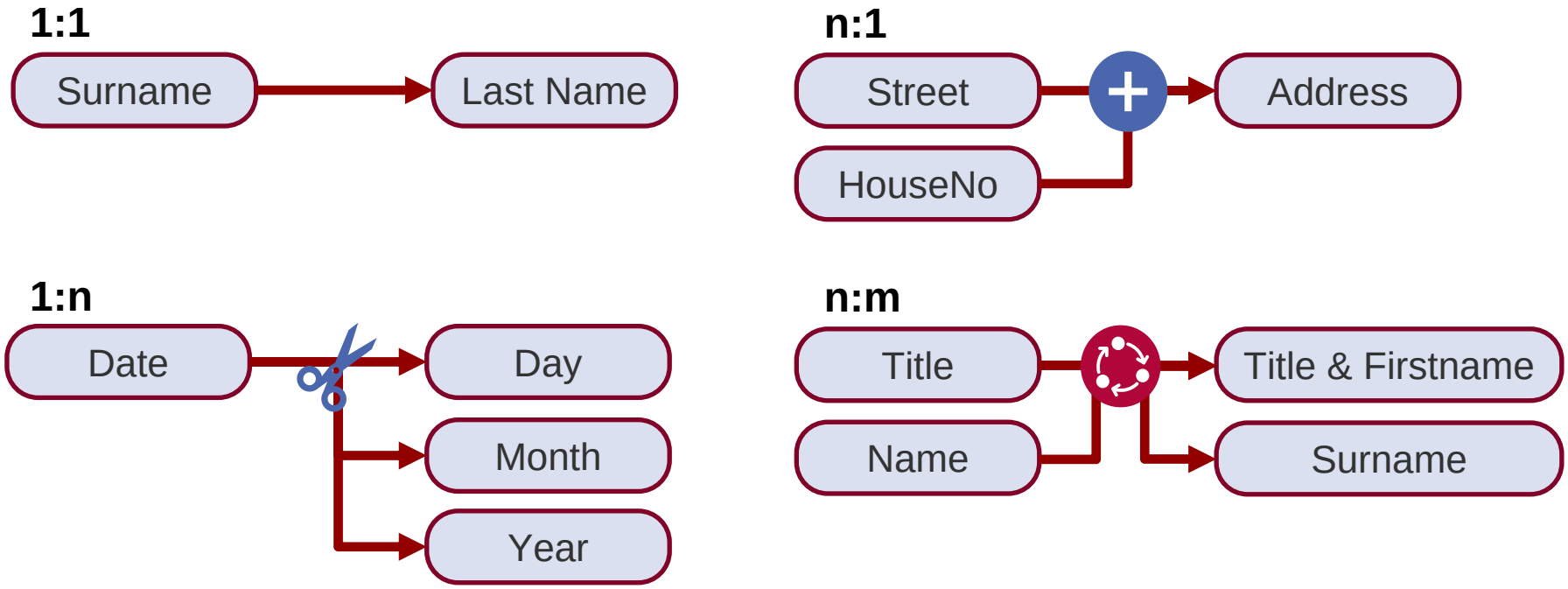

**Schema Matching:**

- Correspondencies between Schema Elements
- 1:1, 1:n, n:1, n:m

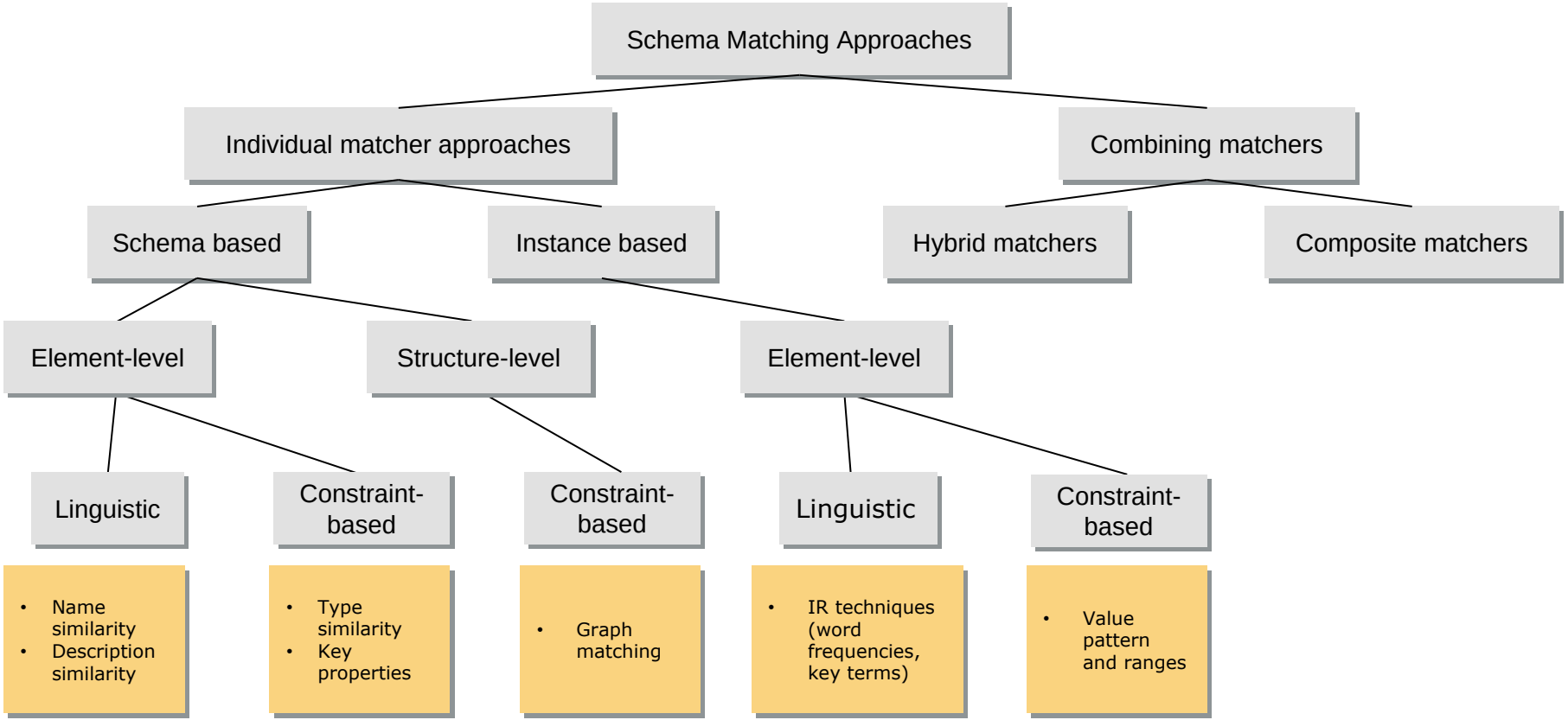
**Schema Mapping:**

- GaV, LaV, GLaV
- Tuple-Generating Dependencies (tgd)

# Schema Matching - Correspondences



# Schema Matching - Approaches



# Schema Matcher

**Source 1: Book**

| ID (char(14)) | Title | Author | Publisher |
|---|---|---|---|
| 978-0124160446 | Principles of Data Integration | Anhai Doan | Morgan Kaufmann |
| 978-1627052238 | Big Data Integration | Xin L. Dong | Morgan & Claypool |
| 978-1484222461 | SQL on Big Data | Sumit Pal | Apress |

**Source 2: Writings**

| ISBN (char(14)) | Name | Authors | Press |
|---|---|---|---|
| 978-1484222461 | SQL on Big Data | Pal | Apress |
| 978-1608459247 | Data Management in the Cloud | Agrawal, Das, Abbadi | Morgan & Claypool |
| 978-0124160446 | Principles of Data Integration | Doan | Morgan Kaufmann |

**Linguistic:**
Syntactic and semantic name similarity

# Schema Matcher

**Source 1: Book**

| ID char(14) | Title | Author | Publisher |
|---|---|---|---|
| 978-0124160446 | Principles of Data Integration | Anhai Doan | Morgan Kaufmann |
| 978-1627052238 | Big Data Integration | Xin L. Dong | Morgan & Claypool |
| 978-1484222461 | SQL on Big Data | Sumit Pal | Apress |

**Source 2: Writings**

| ISBN char(14) | Name | Authors | Press |
|---|---|---|---|
| 978-1484222461 | SQL on Big Data | Pal | Apress |
| 978-1608459247 | Data Management in the Cloud | Agrawal, Das, Abbadi | Morgan & Claypool |
| 978-0124160446 | Principles of Data Integration | Doan | Morgan Kaufmann |

**Linguistic:**
Syntactic and semantic name similarity

**Constraint:**
Similar data types, ICs (e.g., unique, FK, RegExs)

# Schema Matcher

**Source 1: Book**

| ID (char(14)) | Title | Author | Publisher |
|---|---|---|---|
| 978-0124160446 | Principles of Data Integration | Anhai Doan | Morgan Kaufmann |
| 978-1627052238 | Big Data Integration | Xin L. Dong | Morgan & Claypool |
| 978-1484222461 | SQL on Big Data | Sumit Pal | Apress |

**Source 2: Writings**

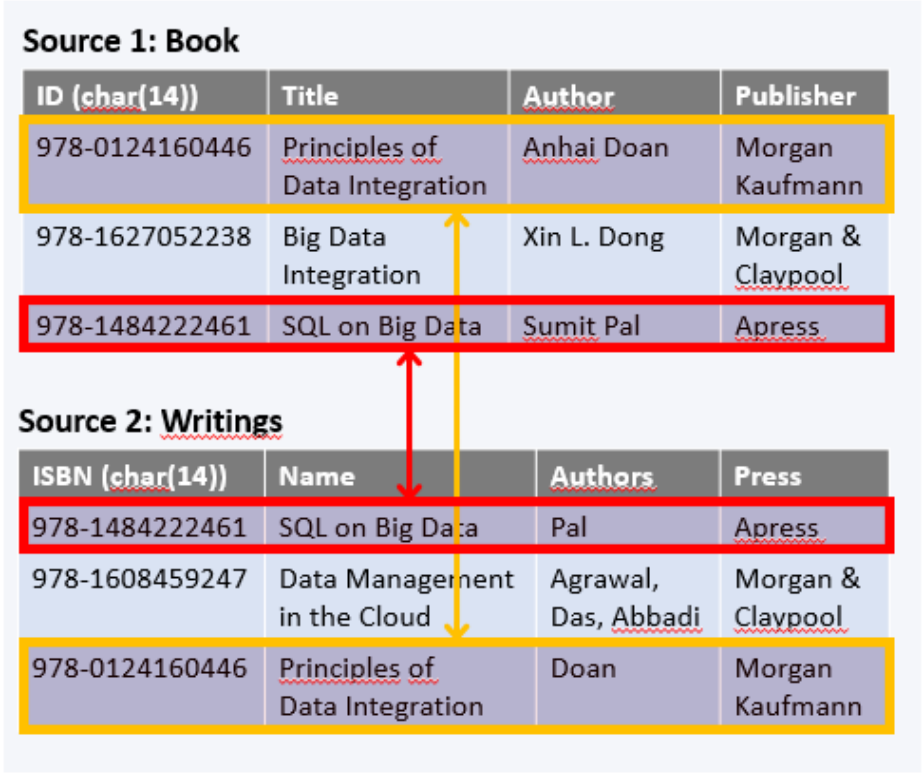
| ISBN (char(14)) | Name | Authors | Press |
|---|---|---|---|
| 978-1484222461 | SQL on Big Data | Pal | Apress |
| 978-1608459247 | Data Management in the Cloud | Agrawal, Das, Abbadi | Morgan & Claypool |
| 978-0124160446 | Principles of Data Integration | Doan | Morgan Kaufmann |

**Linguistic:**
Syntactic and semantic name similarity

**Constraint:**
Similar data types, ICs (e.g., unique, FK, RegExs)

**Instance-based (vertical):**
Similarity of values in two columns
(e.g., set-based, domain-based, classifier)

# Schema Matcher




**Linguistic:**
Syntactic and semantic name similarity

**Constraint:**
Similar data types, ICs (e.g., unique, FK, RegExs)

**Instance-based (vertical):**
Similarity of values in two columns
(e.g., set-based, domain-based, classifier)

**Instance-based (horizontal):**
Similarity of two records (i.e., duplicate detection)

# Integrity Constraints

- **Unique Column Combinations (UCCs)**
  - $A\ is\ unique \Leftrightarrow \forall r_1, r_2 \in D: r_1 \neq r_2 \Rightarrow r_1[A] \neq r_2[A]$
- **Functional Dependencies (FDs)**
  - $A \rightarrow B \Leftrightarrow \forall r_1, r_2 \in D: r_1[A] = r_2[A] \Rightarrow r_1[B] = r_2[B]$
- **Inclusion Dependencies (INDs)**
  - $A \subseteq B \Leftrightarrow \forall r_1 \in D, \exists r_2 \in D: r_1[A] = r_2[\mathrm{B}]$
- **Further Constraints:**
  - Order Dependencies (ODDs)
  - Matching Dependencies (MDs)
  - Denial Constraints (DCs)

# Agenda

- ❑ Chair Introduction
- ❑ Organizational Information
- ❑ Schema Matching & Constraints
- ❑ Project Procedure

University of Marburg

# Big Data Analytics Group

Prof. Thorsten
**Papenbrock**

Alexander
**Vielhauer**

Marcian
**Seeger**

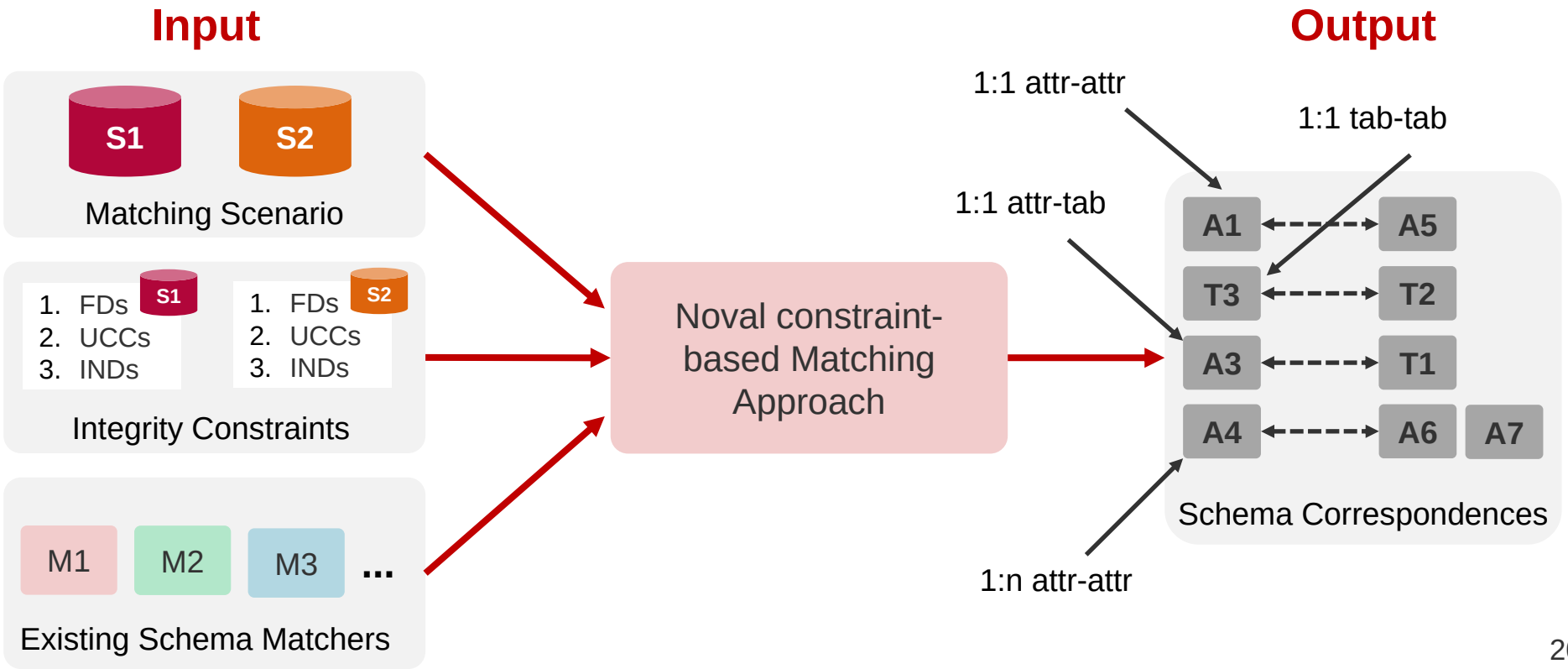
Philipps Universität Marburg

Research on:

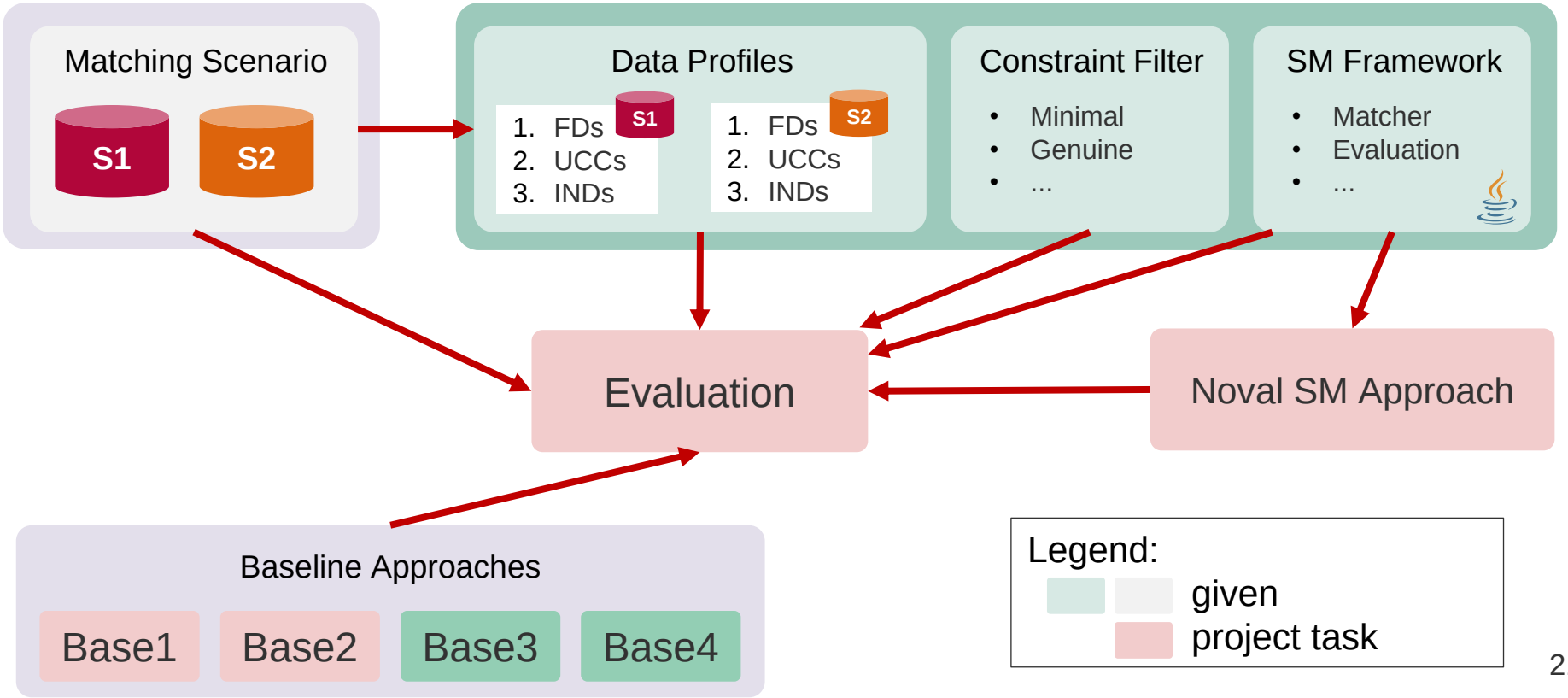
- Data Profiling
- Schema Matching

Seminar: „StructureMatch: Schema Matching with Data Profiling“

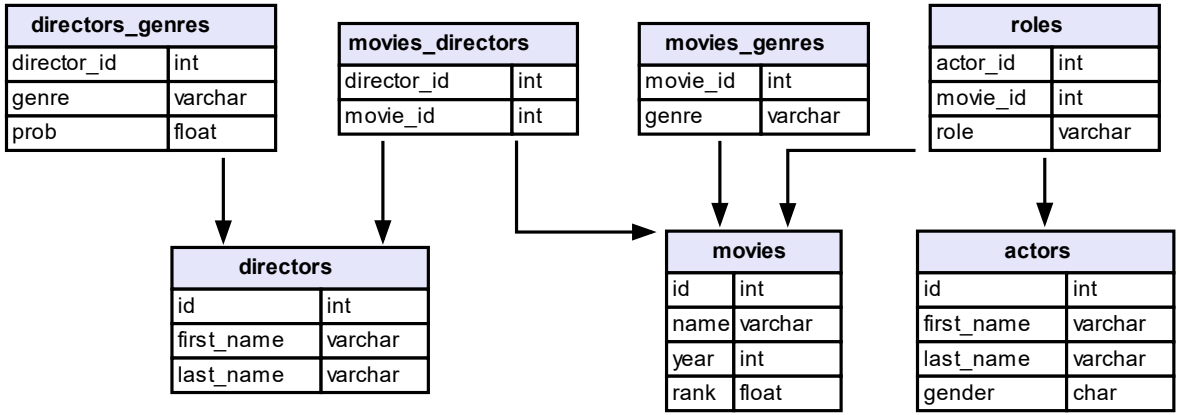
# Project Specification (1)



# Project Specification (2)



# Matching Scenarios




Different schemas
of different complexities
with different FK patterns
(https://relational.fit.cvut.cz/)

**Modifications:**

- Renaming of attributes & using different encoding of attribute values
- Denormalization (Join of tables)
- Removal of attributes
- Merging of attributes (⇒1:n and n:m correspondencies)

# Data Profiling

- **Detection of UCCs, FDs, and INDs with DPQL**
  - Based on Metanome
  - Intuitive to use

```
1  SELECT
2    X Determinant, Y AS Unique
3  FROM
4    CC(Teams,Trainers) X,
5    CC(Pokemon) Y
6  WHERE
7    UCC(Y)
8    AND (IND(X,Y) OR FD(X,Y))
```

M. Seeger, S. Schmidl, et al., DPQL: The Data Profiling Query Language, BTW 2023

# Constraint Filtering

- **Rules to reduce number of constraints**
  - Only minimal FDs
  - Only genuine FDs, UCCs, and INDs
  - No FDs where the determinante is a key candidate

- **Can be applied in arbitrary combination**

# Schema Matching Framework

- **Provides**
  - Import Functions
  - Classes for Modeling Schemas and Integrity Constraints
  - Several individual Matchers
  - Evaluation Methods

- **Implemented in Java**

# Baseline Approaches (1)

PoWareMatch: A Quality-aware Deep Learning Approach to Improve Human Schema Matching
Roee Shraga, et al., JDIQ 2022
https://dl.acm.org/doi/pdf/10.1145/3483423
https://github.com/shraga89/PoWareMatch/tree/master/DataFiles

ADnEV: cross-domain schema matching using deep similarity matrix adjustment and evaluation
Roee Shraga, et al., VLDB 2020
https://dl.acm.org/doi/pdf/10.14778/3397230.3397237
https://github.com/shraga89/DSMA

Seeping Semantics: Linking Datasets using Word Embeddings for Data Discovery
Raul Castro Fernandez, et al., ICDE 2018
https://ieeexplore.ieee.org/document/8509314
https://github.com/mitdbg/aurum-datadiscovery

Creating Embeddings of Heterogeneous Relational Datasets for Data Integration Tasks
Paolo Papotti, et al., SIGMOD 2020
https://arxiv.org/pdf/1909.01120.pdf
https://gitlab.eurecom.fr/cappuzzo/embdi

# Baseline Approaches (2)

LEAPME: Learning-based Property Matching with Embeddings
Daniel Ayala, et al., Data & Knowledge Engineering 2021
https://www.sciencedirect.com/science/article/pii/S0169023X21000707?via%3Dihub
https://github.eii.us.es/dayala1/LEAPME

Schema Matching using Pre-Trained Language Models
Yunjia Zhang, et al., ICDE 2023
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10184612

**CODE?**

It's AI Match - A Two-Step Approach for Schema Matching Using Embeddings
Benjamin Hättasch, et al., AIDB@VLDB 2020
https://arxiv.org/abs/2203.04366

**CODE?**

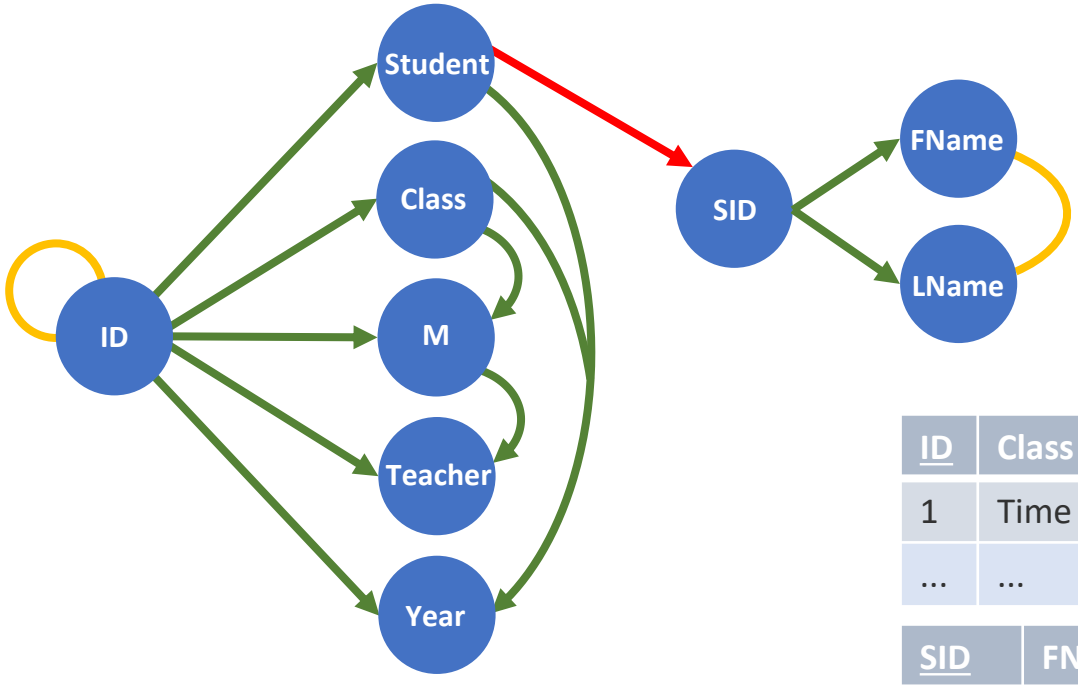
Schema Matching Using Interattribute Dependencies
Jaewoo Kang, et al., IEEE Trans. Knowl. Data Eng. 2008
https://ieeexplore.ieee.org/abstract/document/4527243

**CODE?**

# Solution Ideas

- **Build a hypergraph (HG) per schema**
  - Nodes: Attributes and Tables
  - Hyperedges: Constraints
    - Non-directed: UCCs
    - Directed: FDs & INDs
- **Prepare the hypergraphs**
- **Approach 1:** Match the HGs using existing methods
- **Approach 2:** Iterative fixing of the HG matching

# Hypergraph Building

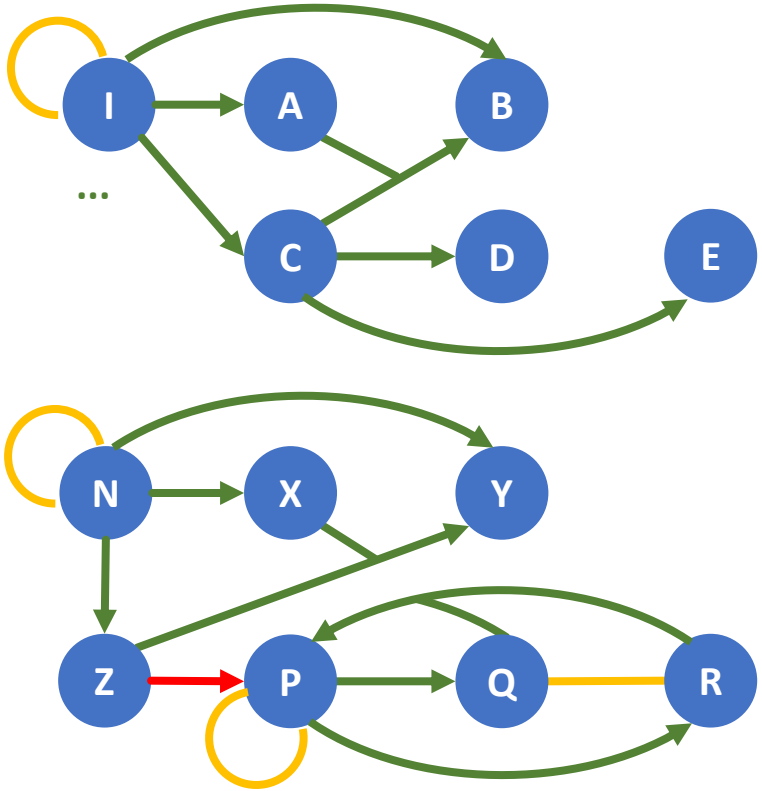



**Legend:**
- Functional dependency
- Unique constraint
- Inclusion dependency

| ID | Class | Teacher | Year | M | Student |
|---|---|---|---|---|---|
| 1 | Time Series | Dr. Brown | 2023 | 1 | B3 |
| … | … | … | … | … | … |

| SID | FName | LName |
|---|---|---|
| A3 | Jane | Smith |
| … | … | … |

# Hypergraph Preparation




**S1:**

| I | A | B | C | D | E |
|---|---|---|---|---|---|
| ... | ... | ... | ... | ... | ... |

**FDs:** I → A,B,C,D,E; C → D,E; A,C → B
**UCCs:** {I}

**S2:**

| N | X | Y | Z |
|---|---|---|---|
| ... | ... | ... | ... |

| P | Q | R |
|---|---|---|
| ... | ... | ... |

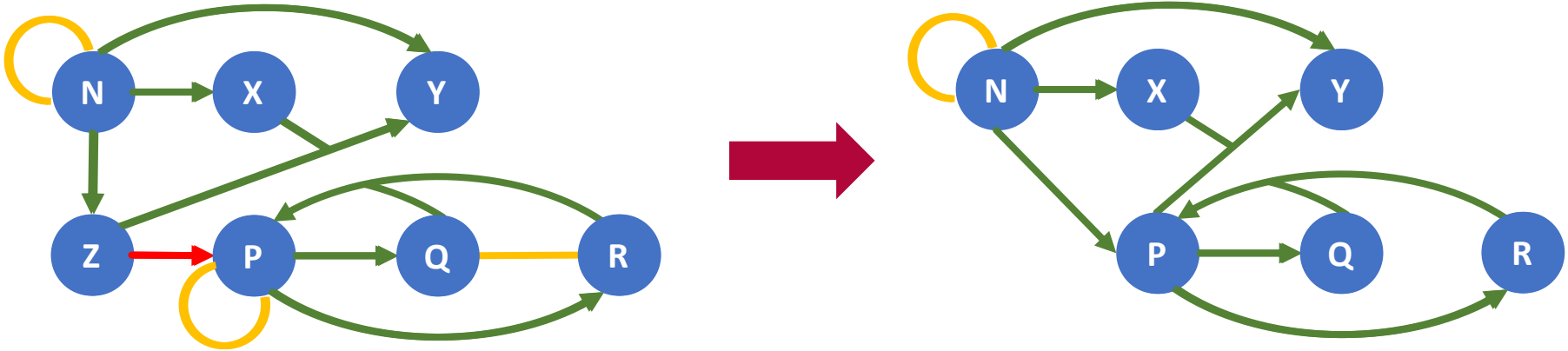
**FDs:** N → X,Y,Z; P → Q,R; Q,R → P
**UCCs:** {N}, {P}, {Q,R}
**INDs:** Z ⊆ P

# Hypergraph Preparation

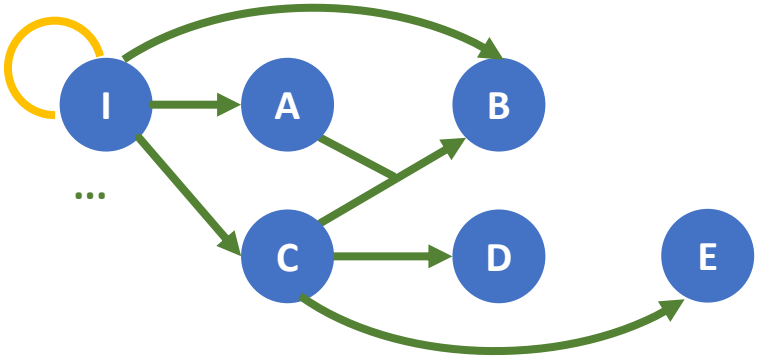
**Rule:** Remove INDs
(logical denormalization)

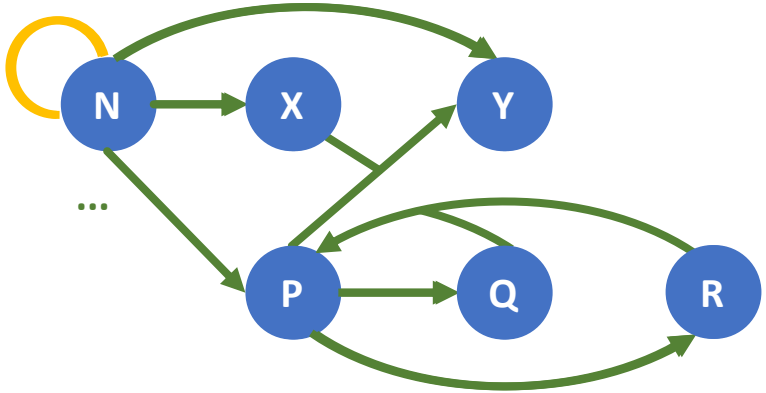
**Effects:**

- P inherits constraints from Z
- {P} and {Q,R} are not unique anymore



# Hypergraph Preparation




**S1:**

| I | A | B | C | D | E |
|---|---|---|---|---|---|
| ... | ... | ... | ... | ... | ... |

**FDs:** I → A,B,C,D,E; C → D,E; A,C → B
**UCCs:** {I}

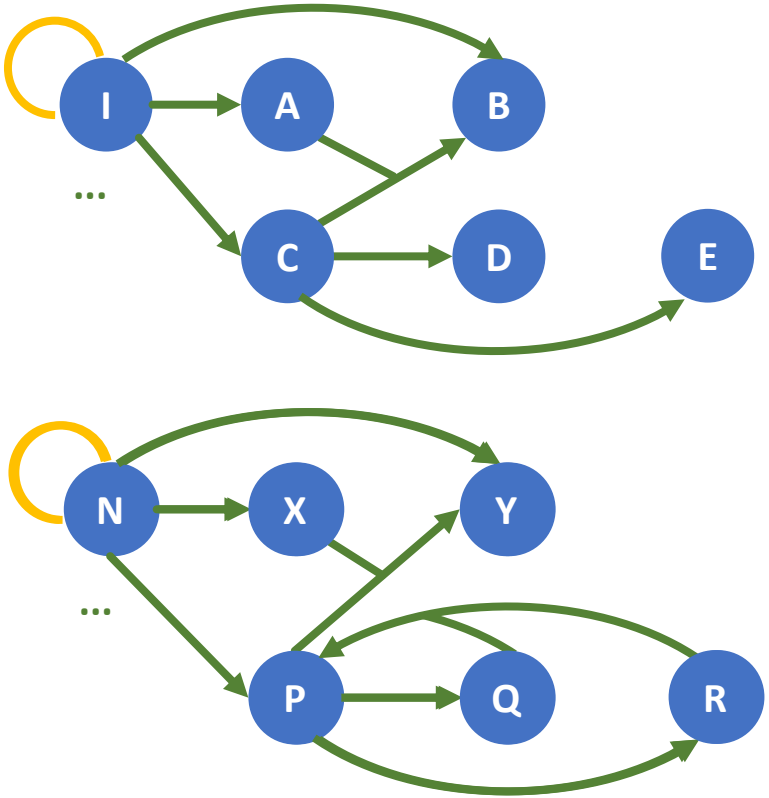
**S2:**

| N | X | Y | Z |
|---|---|---|---|
| ... | ... | ... | ... |

| P | Q | R |
|---|---|---|
| ... | ... | ... |

**FDs:** N → X,Y,Z; P → Q,R; Q,R → P
**UCCs:** {N}, {P}, {Q,R}
**INDs:** Z ⊆ P

# Approach 1: Hypergraph Matching

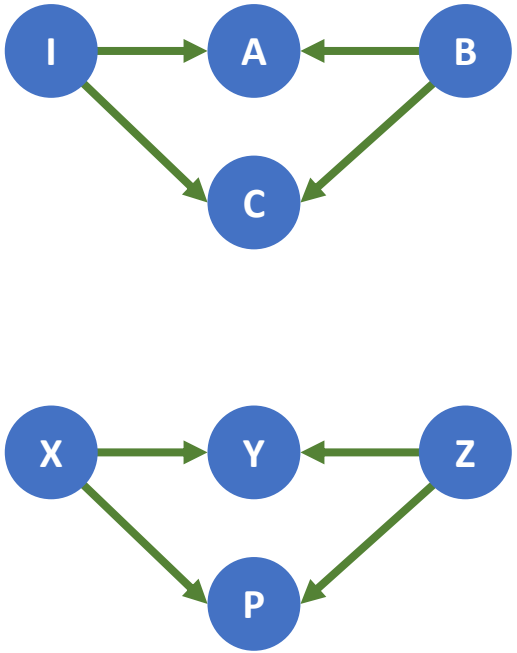



**Approach:**

- Matching of both graphs using existing algorithms
- Extension for inter-graph similarities

Hypergraph Neural Networks for Hypergraph Matching
Xiaowei Liao, et al., ICCV 2021
https://github.com/xwliao/HNN-HM

# Approach 2: Iterative Graph Fixing



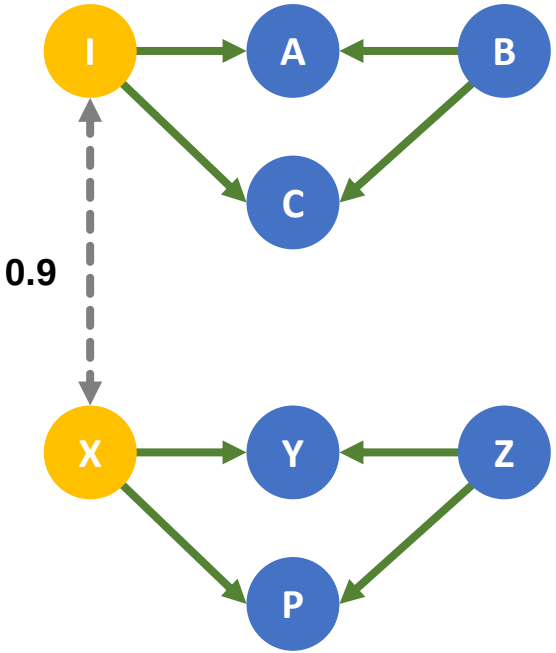

**Approach:**

- Apply simple matchers
- Until matching has high confidence:
  - Select node pair with highest similarity
  - Use hyperedges to fix the remaining node pairs

**Uncertainty:**

- I ↔ X and B ↔ Z or I ↔ Z and B ↔ X
- A ↔ Y and C ↔ P or A ↔ P and C ↔ Y

# Approach 2: Iterative Graph Fixing



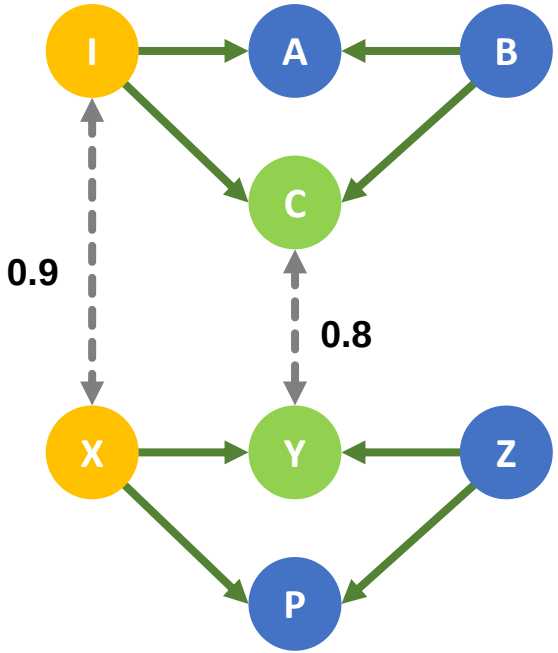

**Approach:**

- Apply matchers
- Until matching has high confidence:
  - Select node pair with highest similarity
  - Use hyperedges to fix the remaining node pairs

**Uncertainty:**

- A ↔ Y and C ↔ P or A ↔ P and C ↔ Y

# Approach 2: Iterative Graph Fixing




**Approach:**

- Apply matchers
- Until matching has high confidence:
  - Select node pair with highest similarity
  - Use hyperedges to fix the remaining node pairs

**Uncertainty: None**

# Potential Extensions

- Integration of further constraint types
  (e.g., MDs, ODDs, DCs)
- Integration of constraints across different schemas
  (e.g., inclusion dependencies)
- Detection of 1:n and n:m correspondences
  (e.g., by merging nodes within the hypergraph)
- Consideration of approximate and conditional constraints
  (i.e., constraints that are not always valid)

# Further Procedure

- **Meeting every (two) weeks to discuss …**
  - Problems
  - Solution ideas / strategies
  - Results

- **Joint meeting with Uni Marburg every month to**
  - discuss problems
  - share solutions
  - compare results
  - coordinate joint work (to avoid duplicate work)
  - communicate requirements

# Next Steps

- Familiarization with Schema Matching

  A survey of approaches to automatic schema matching
  Erhard Rahm, et al., VLDB Journal 2001
  https://link.springer.com/article/10.1007/s007780100057

- Look at program code of Baseline Approaches and corresponding papers
- Familiarization with Schema Matching Framework

  (after Uni Marburg has given you the required rights)

  https://github.com/avielhauer/schematch