Data Wrangling and Data Analysis

FDs + DI

Hakim Qahtan

Part of the slides were prepared by

Yannis Velegrakis

Department of Information and Computing Sciences

Utrecht University

Utrecht University

1

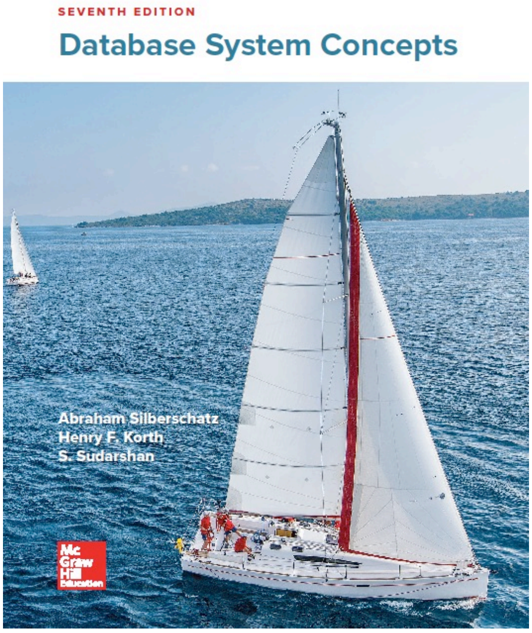
Reading Material for Today

Database System Concepts (7th Edition)

CH 7.1 - 7.4.1

Office Hours: Every Friday (10:00 – 11:45)

Location: BBG-445



Utrecht University

2

Designing Good Databases

Utrecht University

3

The Database Design Problem

- How do we know when a design is good?
- A DB schema seems to be good if it helps us to avoid redundancy and inconsistency, but are there more quality issues?
- This question can be answered using the normalization theory
- Basic rules for good DB design
  - 1 table for each entity
  - 1 table for each relationship
  - Each cell contains a single value
  - If you have BIG relations, decompose them
    - This could result in a serious problem

Utrecht University

4

Decomposing Big Relations

- Definition:

  Let R be a relation schema (with constraints).

  A decomposition of R is a set of relation schemas $R_1, R_2, \ldots, R_n$ such that

  i. each R_i consists of attributes in R and

  ii. each attribute of R occurs in at least one R_j

Utrecht University

5

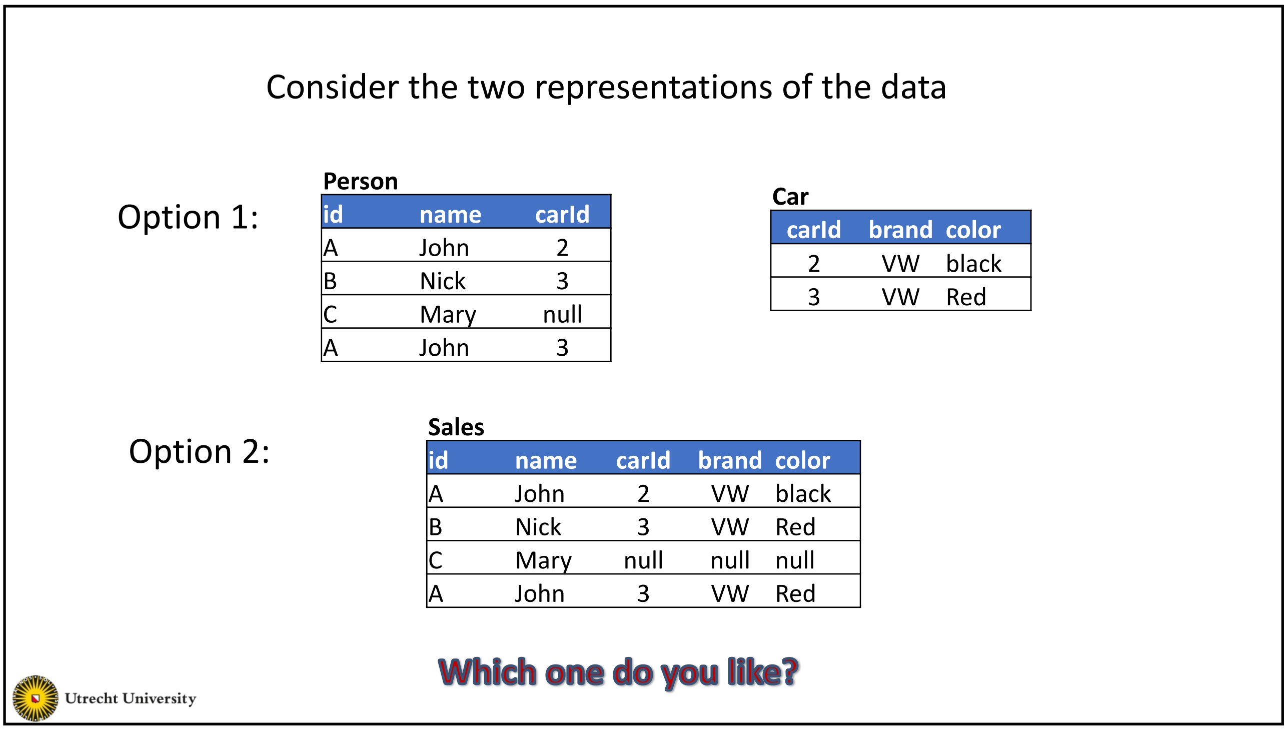
Example

Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Utrecht University

6



Consider the two representations of the data

Option 1:

Person

id	name	carId
A	John	2
B	Nick	3
C	Mary	null
A	John	3

Car

carId	brand	color
2	VW	black
3	VW	Red

Option 2:

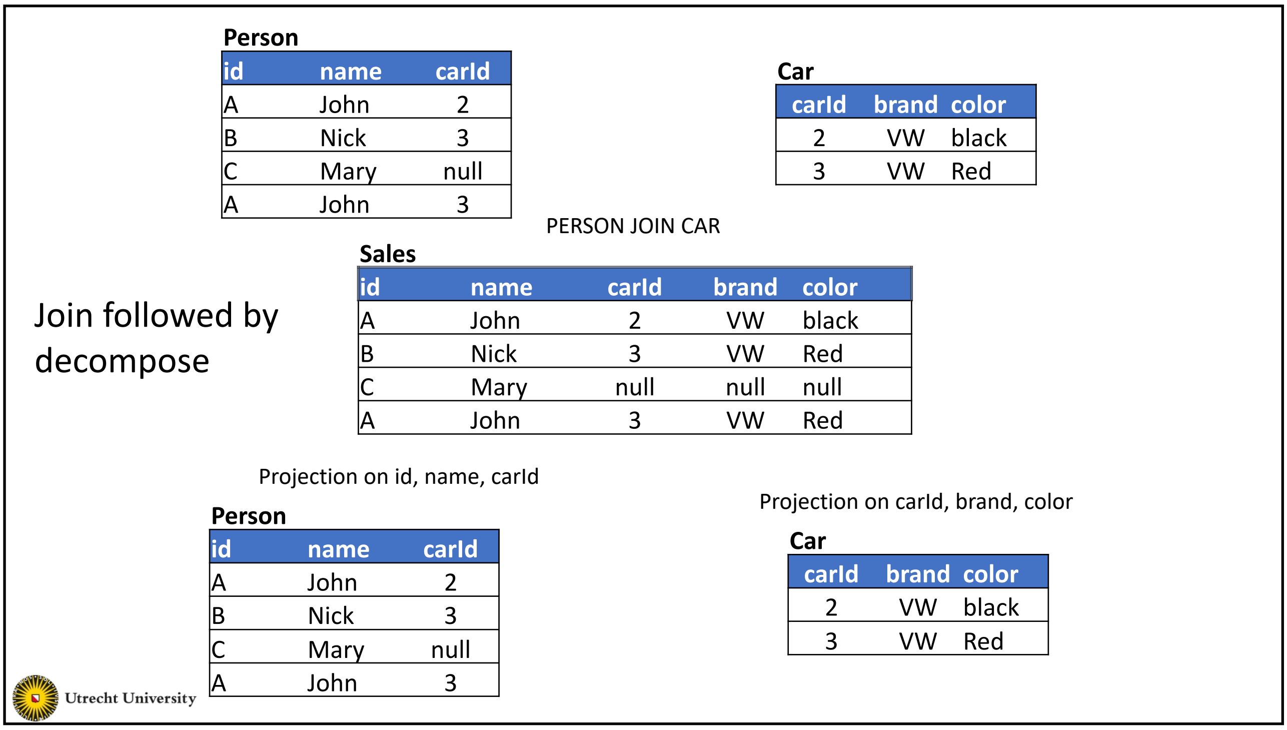
Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Which one do you like?

Utrecht University

7



Person

id	name	carId
A	John	2
B	Nick	3
C	Mary	null
A	John	3

Car

carId	brand	color
2	VW	black
3	VW	Red

PERSON JOIN CAR

Join followed by decompose

Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Projection on id, name, carId

Person

id	name	carId
A	John	2
B	Nick	3
C	Mary	null
A	John	3

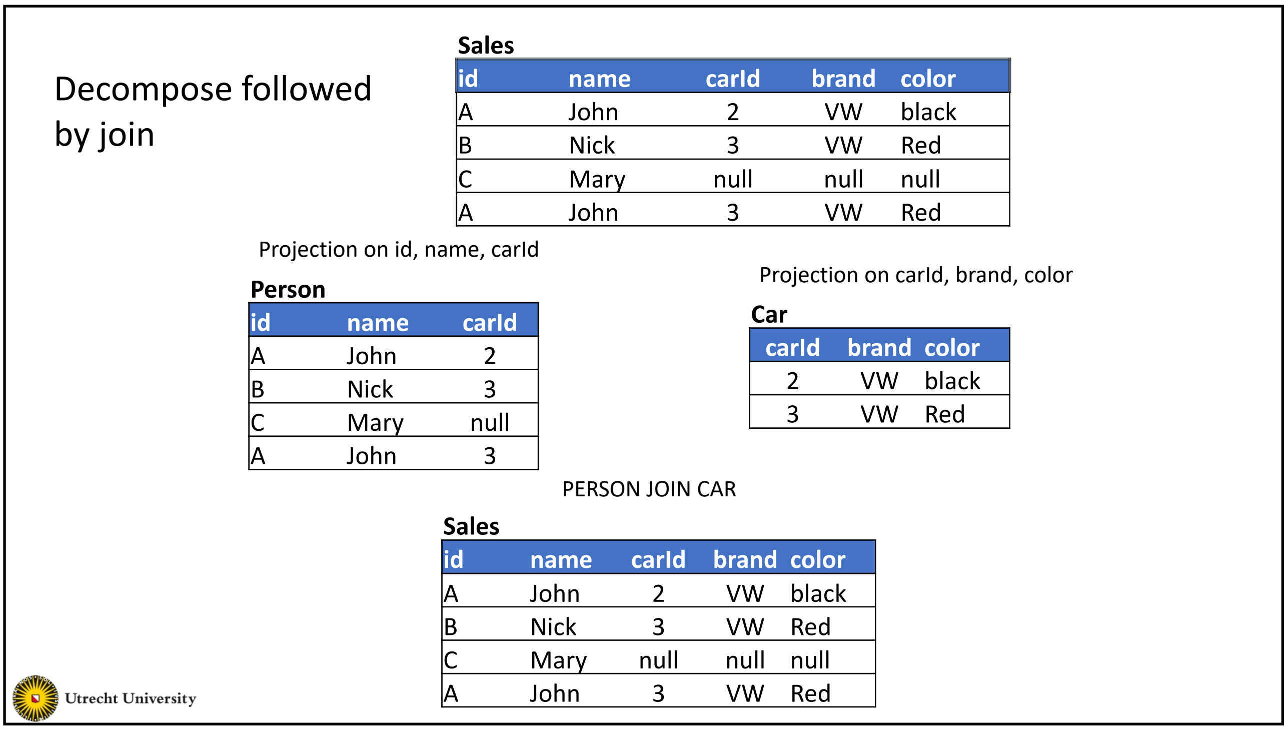
Projection on carId, brand, color

Car

carId	brand	color
2	VW	black
3	VW	Red

Utrecht University

8



Decompose followed by join

Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Projection on id, name, carId

Person

id	name	carId
A	John	2
B	Nick	3
C	Mary	null
A	John	3

Projection on carId, brand, color

Car

carId	brand	color
2	VW	black
3	VW	Red

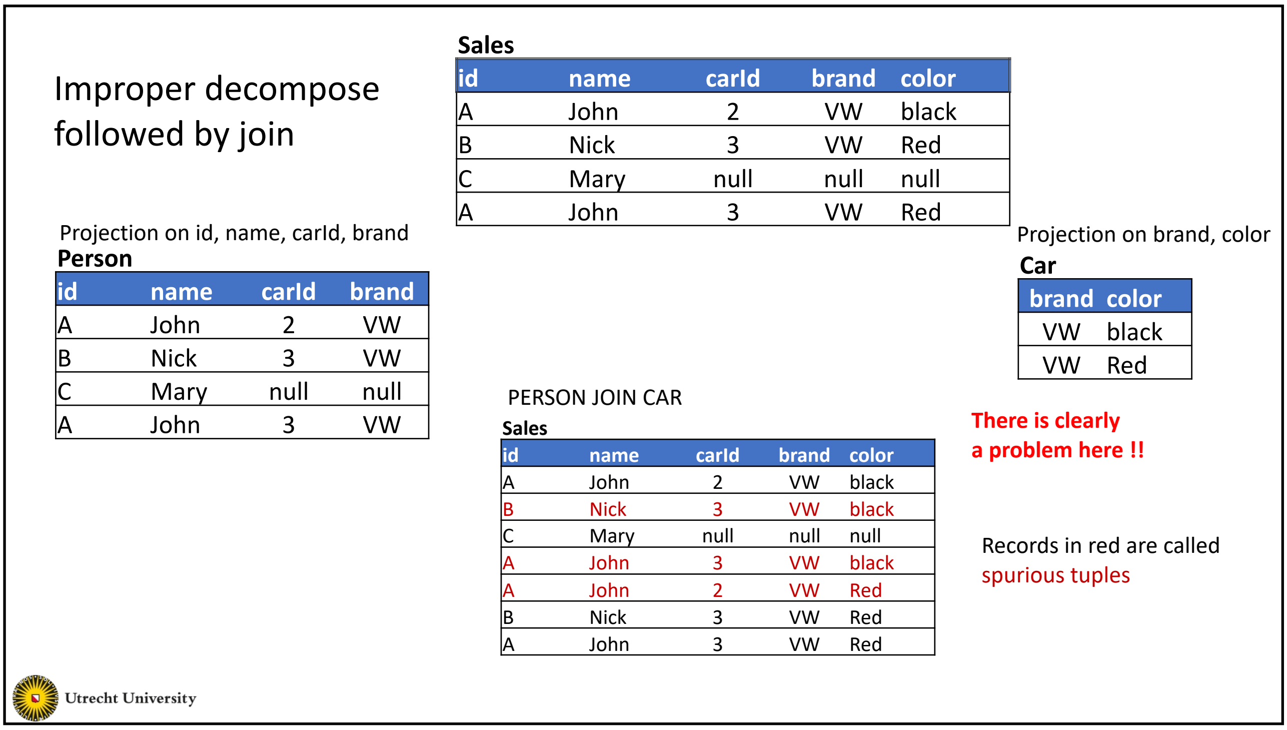
PERSON JOIN CAR

Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Utrecht University

9



Improper decompose followed by join

Sales

id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	Red
C	Mary	null	null	null
A	John	3	VW	Red

Projection on id, name, carId, brand

Person

id	name	carId	brand
A	John	2	VW
B	Nick	3	VW
C	Mary	null	null
A	John	3	VW

Projection on brand, color

Car

brand	color
VW	black
VW	Red

PERSON JOIN CAR

Sales

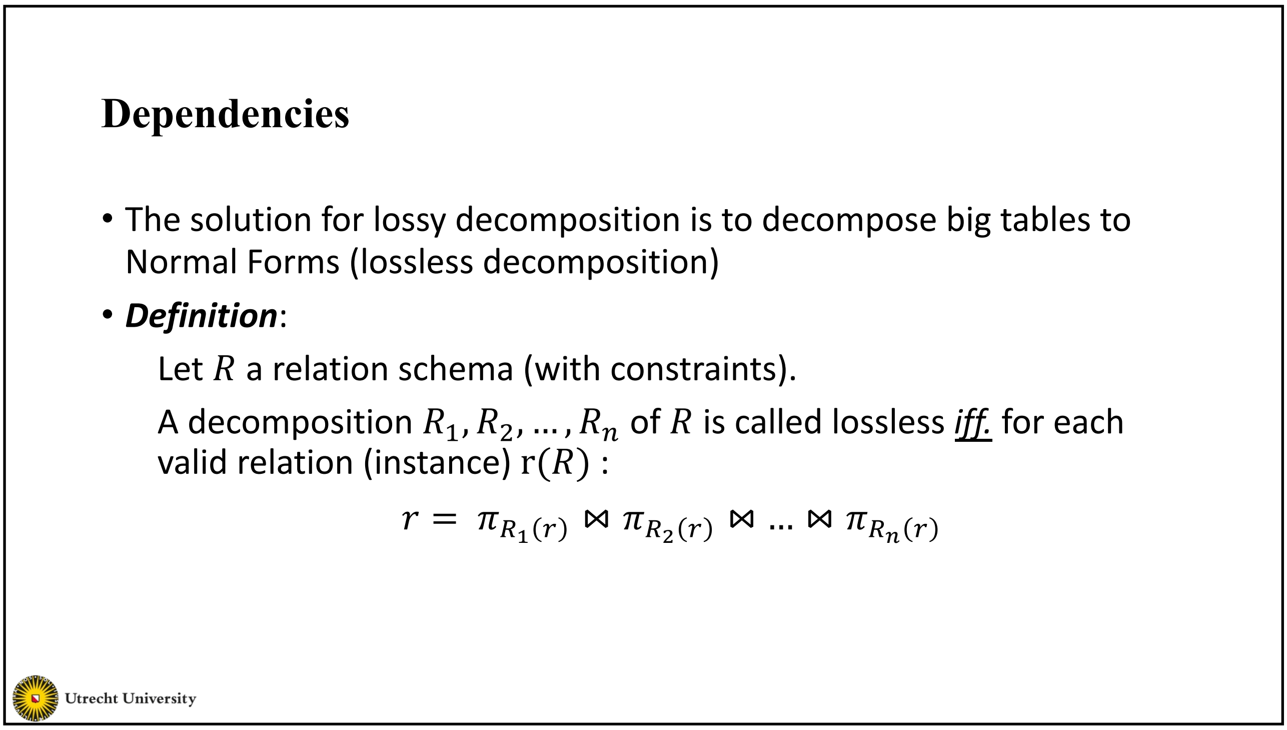
id	name	carId	brand	color
A	John	2	VW	black
B	Nick	3	VW	black
C	Mary	null	null	null
A	John	3	VW	black
A	John	2	VW	Red
B	Nick	3	VW	Red
A	John	3	VW	Red

There is clearly a problem here !!

Records in red are called spurious tuples

Utrecht University

10



Dependencies

- The solution for lossy decomposition is to decompose big tables to Normal Forms (lossless decomposition)
- ***Definition***:

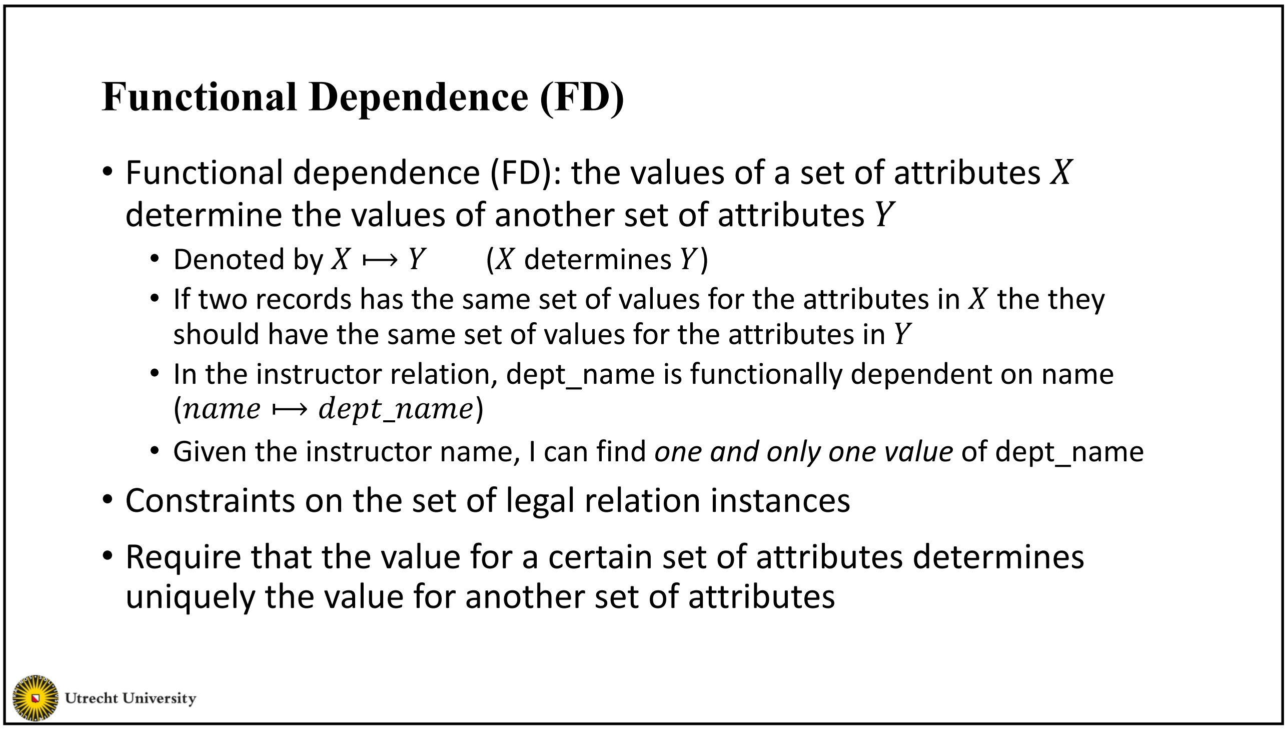
  Let R a relation schema (with constraints).

  A decomposition $R_1, R_2, \dots, R_n$ of R is called lossless *iff.* for each valid relation (instance) $\mathrm{r}(R)$:

$$r = \pi_{R_1(r)} \bowtie \pi_{R_2(r)} \bowtie \dots \bowtie \pi_{R_n(r)}$$

Utrecht University

11



Functional Dependence (FD)

- Functional dependence (FD): the values of a set of attributes X determine the values of another set of attributes Y
  - Denoted by $X \longmapsto Y$ (X determines Y)
  - If two records has the same set of values for the attributes in X the they should have the same set of values for the attributes in Y
  - In the instructor relation, dept_name is functionally dependent on name ($name \longmapsto dept\_name$)
  - Given the instructor name, I can find *one and only one value* of dept_name
- Constraints on the set of legal relation instances
- Require that the value for a certain set of attributes determines uniquely the value for another set of attributes

Utrecht University

12

Functional Dependence

- Let R be a relation with attributes (A,B, C, D, E)

  $X \subseteq R,\ Y \subseteq R$

- The functional dependency

  $X \longmapsto Y$

  holds on R if and only if whenever two tuples t_1, t_2 of R agree on the attributes of X, they also agree on the attributes of Y. That is

  $$t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y]$$

- Examples:
  - The capital determines the country
  - The country determines the Internet domain

instructor table

ID	name	dept_name	salary
22322	Einstein	Physics	95000
33452	Gold	Physics	87000
21212	Wu	Finance	90000
10101	Brandt	Comp. Sci.	82000
43521	Katz	Comp. Sci.	75000
98531	Kim	Biology	78000
58763	Crick	Elec. Eng.	80000
52187	Mozart	History	65000
32343	El Said	History	86000

Does $name \longmapsto dept\_name$ hold?
Does $name \longmapsto salary$ hold?
Does dept$_name \longmapsto salary$ hold?
Does dept$_name \longmapsto name$ hold?

Note that: $ID \longmapsto A \quad \forall A \in instructor$
In this example: $name \longmapsto A \quad \forall A \in instructor$

Alternative Definition of the Keys

- K is a superkey for relation R if and only if K $\longmapsto R$
  - This is the *uniqueness* property of “key”
- K is a candidate key for R if and only if
  - K $\longmapsto R$, and
  - For any $X \subset K, X \not\twoheadrightarrow R$
    - makes sure key has minimum set of attributes (*minimality*)

Functional Dependencies

- Functional dependencies allow us to express constraints that cannot be expressed using superkeys.
- Example: Consider the instructor relation:

  We expect the following set of functional dependencies to hold:

  id $\longmapsto$ name
  id $\longmapsto$ dept_name
  name, dept_name $\longmapsto$ salary

  but would not expect the following to hold:
  salary $\rightarrow$ name

Closure of a Set of Functional Dependencies

- Given a set of functional dependencies $\mathcal{F}$, there are certain other functional dependencies that are logically implied by $\mathcal{F}$.
- The set of all functional dependencies *logically implied* by $\mathcal{F}$ is the closure of $\mathcal{F}$.
- We denote the closure of $\mathcal{F}$ by $\mathcal{F}^+$.
- We can find all of $\mathcal{F}^+$ by applying Armstrong's Axioms:
  - if $X \subseteq Y$, then $Y \longmapsto X$ (*reflexivity*)
  - if $\mathrm{X} \longmapsto Y$, then $\mathrm{AX} \longmapsto AY$ (*augmentation*)
  - if $X \longmapsto Y$ and $Y \longmapsto W$, then $X \longmapsto W$ (*transitivity*)

  these rules are sound and complete. A is a set of attributes (could be single attribute)

Functional Dependencies

- FDs can be derived from existing dependencies using Armstrong’s Axioms
- Examples:
  - If Y is a subset of X, then $\mathrm{X} \longmapsto Y$ (reflexivity)
  - If X is a key candidate, then $\mathrm{X} \longmapsto Y,\ \forall Y$
  - $\mathrm{X} \longmapsto Y \implies X \longmapsto B \qquad \forall B \in Y$
    - We can restrict the RHS to have only a single attribute

17

Examples of Armstrong’s Axioms

- if $X \subseteq Y$, then $Y \longmapsto X$ (*reflexivity*)

  name ⟼ name
  name, dept_name ⟼ name
  name, dept_name ⟼ dept_name

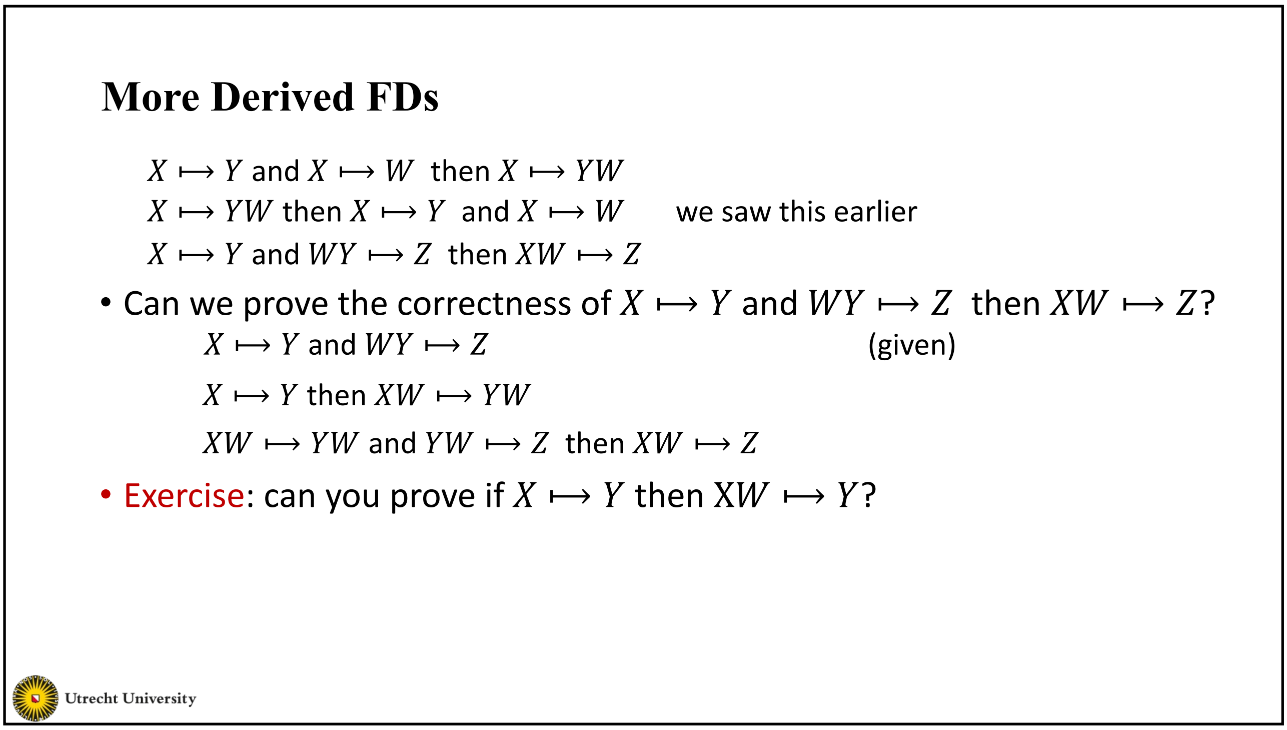
- if $\mathrm{X} \longmapsto Y$, then $\mathrm{AX} \longmapsto AY$ (*augmentation*)

  name ⟼ dept_name
  name, salary ⟼ dept_name, salary

- if $X \longmapsto Y$ and $Y \longmapsto W$, then $X \longmapsto W$ (*transitivity*)
  id ⟼ name and

  name ⟼ dept_name implies id ⟼ dept_name

18



More Derived FDs

$X \longmapsto Y$ and $X \longmapsto W$ then $X \longmapsto YW$

$X \longmapsto YW$ then $X \longmapsto Y$ and $X \longmapsto W$ we saw this earlier

$X \longmapsto Y$ and $WY \longmapsto Z$ then $XW \longmapsto Z$

- Can we prove the correctness of $X \longmapsto Y$ and $WY \longmapsto Z$ then $XW \longmapsto Z$?

  $X \longmapsto Y$ and $WY \longmapsto Z$ (given)

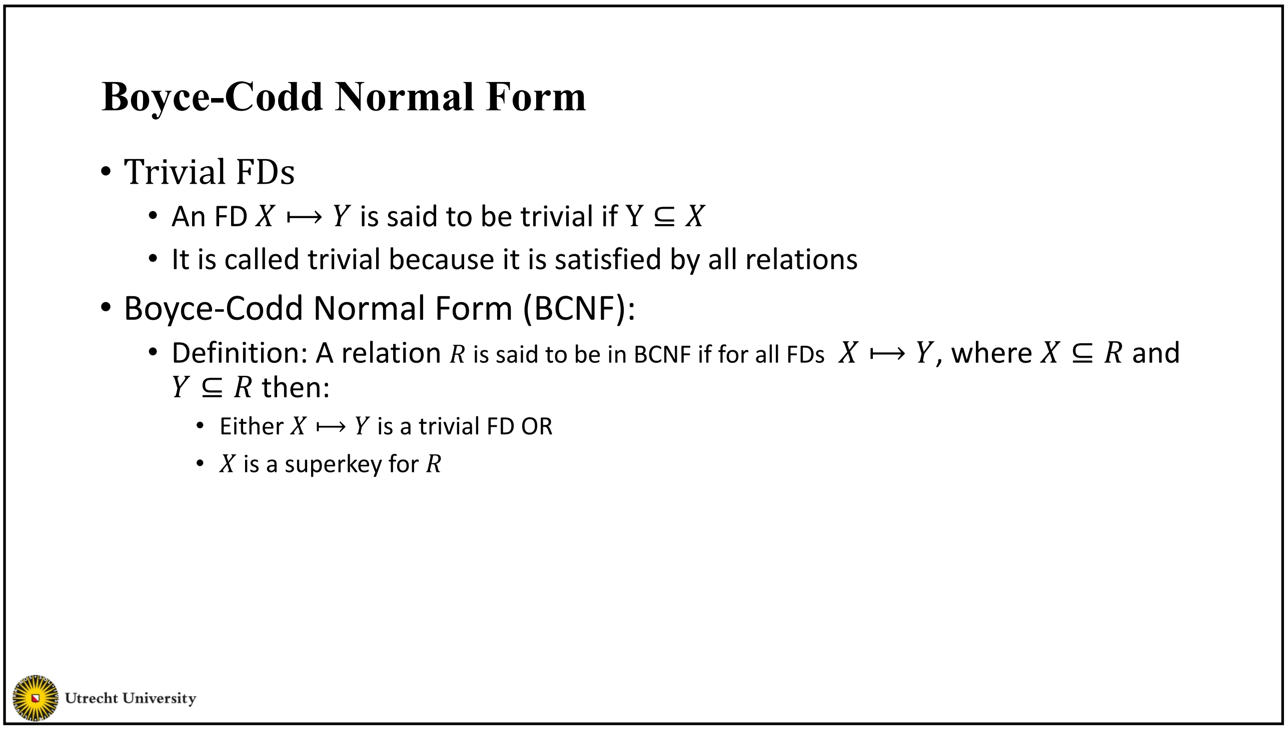
  $X \longmapsto Y$ then $XW \longmapsto YW$

  $XW \longmapsto YW$ and $YW \longmapsto Z$ then $XW \longmapsto Z$

- Exercise: can you prove if $X \longmapsto Y$ then X$W \longmapsto Y$?

Utrecht University

19



Boyce-Codd Normal Form

- Trivial FDs
  - An FD $X \longmapsto Y$ is said to be trivial if Y $\subseteq X$
  - It is called trivial because it is satisfied by all relations
- Boyce-Codd Normal Form (BCNF):
  - Definition: A relation R is said to be in BCNF if for all FDs $X \longmapsto Y$, where $X \subseteq R$ and $Y \subseteq R$ then:
    - Either $X \longmapsto Y$ is a trivial FD OR
    - X is a superkey for R

Utrecht University

20

Use of FDs

- We use functional dependencies to:
  - Test relations to see if they are legal under a given set of functional dependencies.
    - A specific instance of a relation schema may satisfy a functional dependency even if the functional dependency does not hold on all legal instances.
    - For example, a specific instance of *instructor* may, by chance, satisfy
      name ⟼ *id*
  - Check if a relation decomposition is lossless or not
    - ***Theorem***: For relational schema $\mathrm{R}(XYZ)$, the following holds:
      If $X \longmapsto Y$ then the decomposition $R_1(XY), R_2(XZ)$ is lossless
    - In other words, a decomposition of a relation R into R_1 and R_2 is said to be lossless if at least one of the following holds:
      - $\mathrm{R}_1 \cap R_2 \longmapsto R_1$ OR
      - $\mathrm{R}_1 \cap R_2 \longmapsto R_2$

21

Use of FDs (Cont.)

- We use functional dependencies to:
  - Detect inconsistencies in the data
  - For example, if we are given that each instructors can work for only one department and:
    name ⟼ *dept_name*

  Then the highlighted records violate this FD

  When discovering an FD violation, each value can be considered as the source of violation
- Exercise: If you know that in a given relation T

  att_a ⟼ *att_b*

  Write python script to check for violations

id	name	dept_name	salary
22322	Einstein	Physics	95000
33452	Gold	Physics	87000
21212	Wu	Finance	90000
10101	Einstein	Comp. Sci.	82000
43521	Katz	Comp. Sci.	75000
98531	Kim	Biology	78000
58763	Crick	Elec. Eng.	80000

The highlighted cells reflect violations of the FD *name* → *dept_name*.

22

Conditional Functional Dependencies (CFDs)

- In the UK, zip code uniquely determines the street
- The constraint may not hold for other countries
- This constraints can be expressed as follows

  ([country = 44, zip] → street)
- It expresses a fundamental part of the semantics of the data
- It can NOT be expressed as an FD
  - It does not hold on the entire relation; instead, it holds on tuples representing UK customers only

country	area-code	phone	street	city	zip
44	131	1234567	Mayfield	Liverpool	EH4 8LE
44	131	3456789	Crichton	Manchester	EH4 8LE
01	908	3456789	Mountain Ave	NYC	07974

The highlighted cells reflect violations of the CFD *([country = 44, zip] → street)*.

Utrecht University

23

- Summarize what you learned today in 2-minutes

Utrecht University

24

Data Integration

25

Data Integration

- Organizations use databases: mainframes, workstations, servers
  - Built from scratch each time… usually
  - Many models of the same object
- Need for information sharing across databases
  - Exploit advances in distributed computing and networking

26

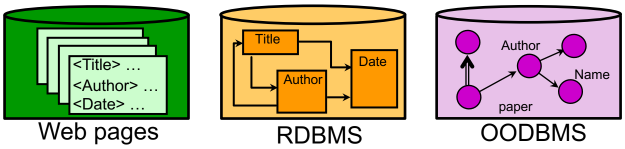
Data Integration

- Fact: Databases are managed by different persons
- Challenge 1: Provide uniform access to the users
- Challenge 2: Allow DB to interoperate but …
  - Autonomous
  - Different OS
  - Different purposes
  - Different data modeling
  - Different data formats
  - Different access and communication protocols

27

Information is Everywhere

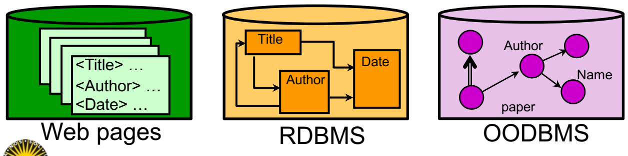
- Databases .. but not only
- Mail messages
- Web pages
- Spreadsheets
- Text documents
- Multimedia annotations
- XML Documents



28

There are many needs

- Exchanged
- Replicated
- Migrated
- Integrated
- Adapted

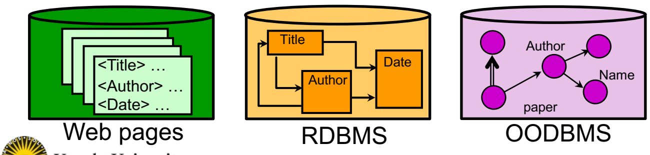


Utrecht University

29

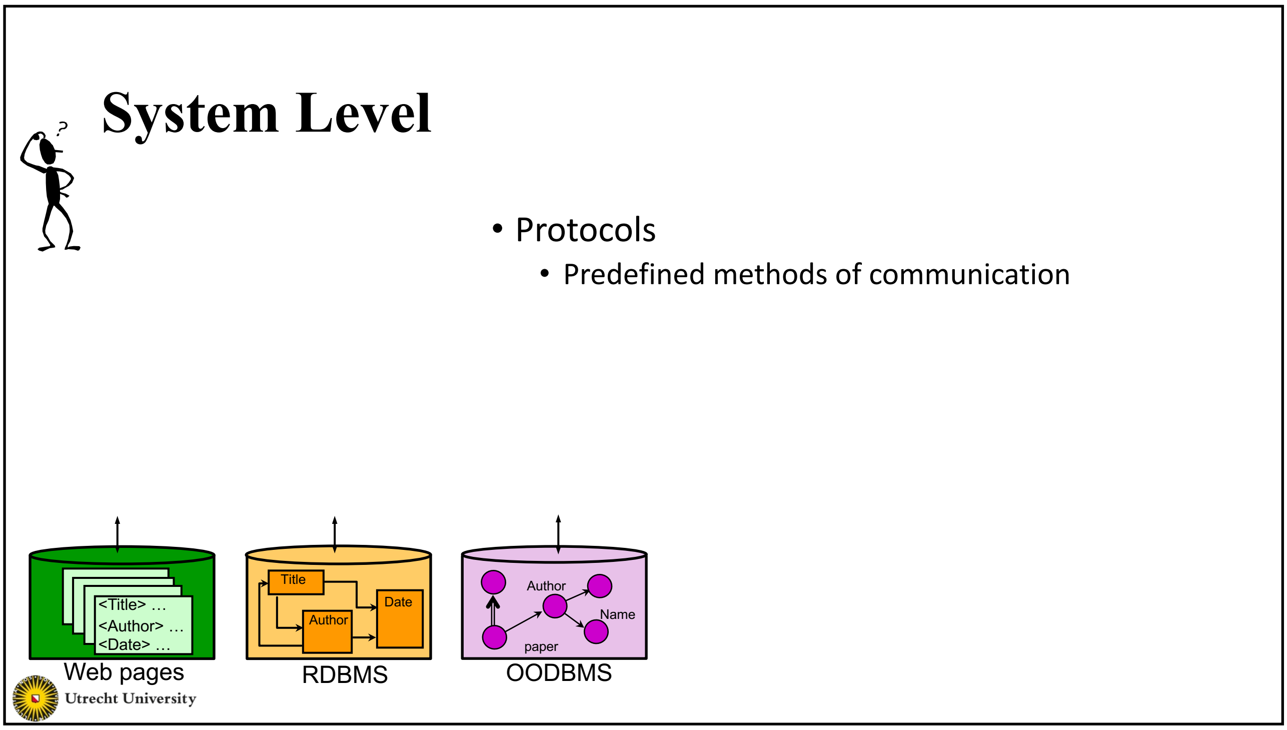
Data in different forms

- Personalization: content adapted to user
  - upon system's decision
  - upon user's request
- Customization: structure adapted to user
  - according to the user's role
  - upon user's request
- System Dependent
  - Performance Reasons
- Context dependence
  - User, Device, Network, Place, Time, Rate



Utrecht University

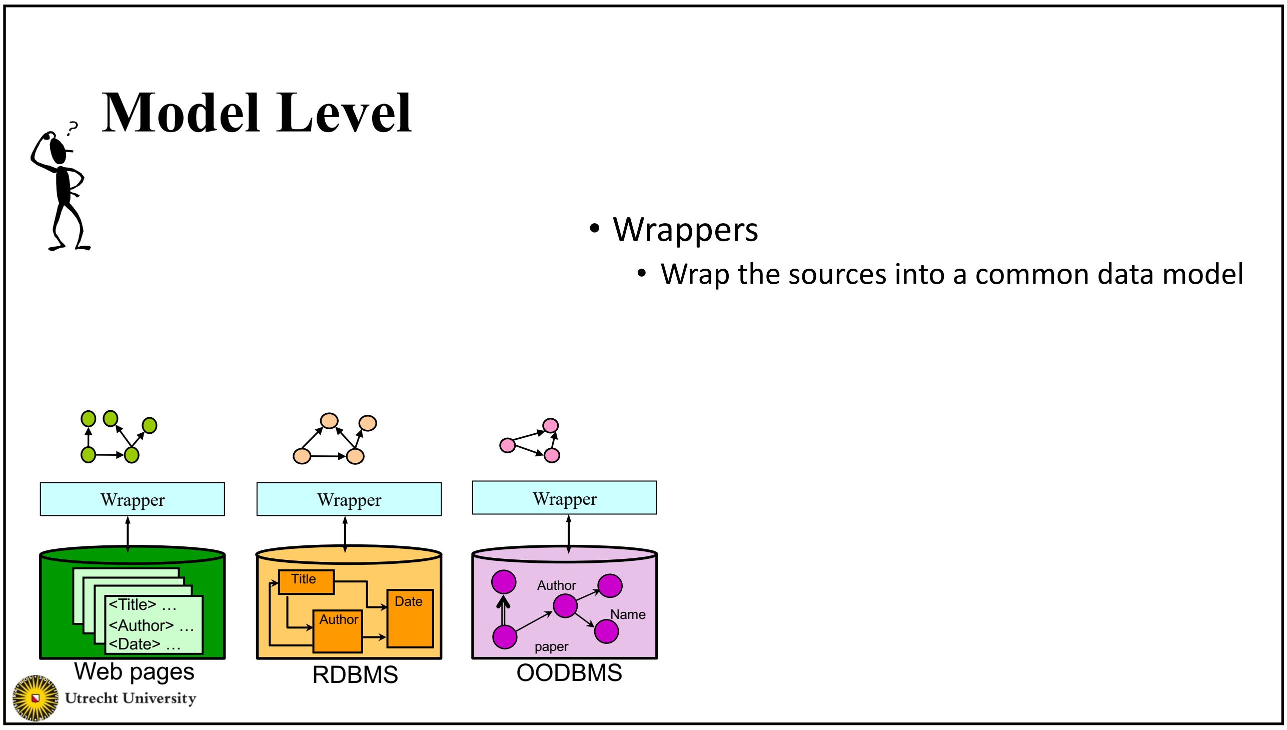
30



System Level

- Protocols
  - Predefined methods of communication

31

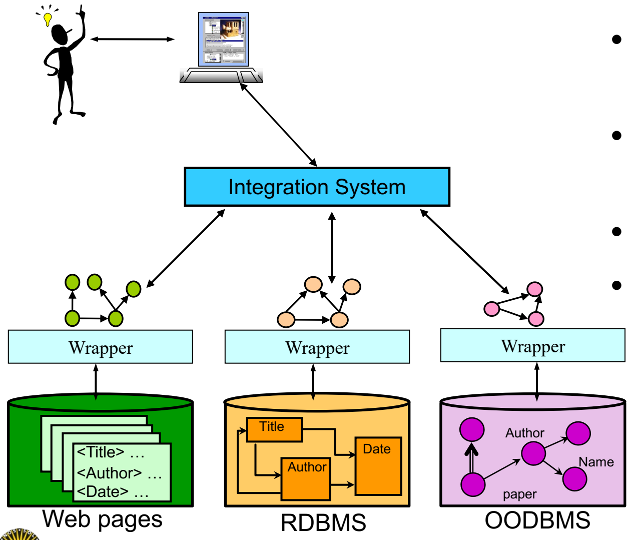


Model Level

- Wrappers
  - Wrap the sources into a common data model

32

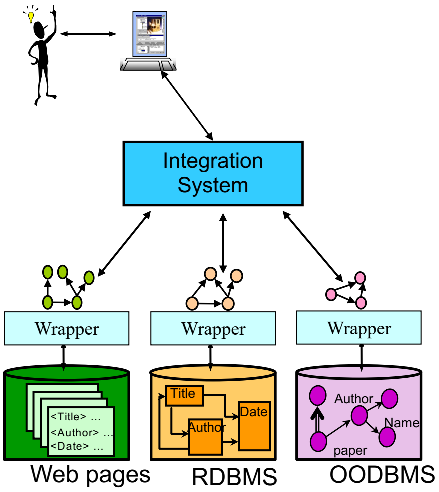
Structural and Semantic Level



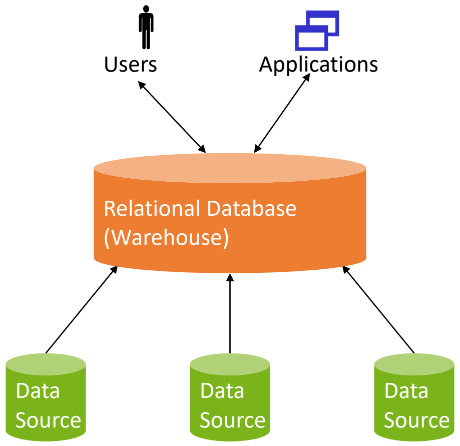
- Structural
  - Different structures for the same data
- Semantic
  - Different semantics of the same structure
- Integrated View
- Many architectures for this basic idea

An Information Integration System

- A set of Local Databases
  - Each with a Local Schema & a Local Instance
- A Global Integrated Schema
- A set of Mappings
  - Between the Global and the Local Schemas
  - Describe the relationship between the data in the sources and the data as the user "sees" them



Data Warehouse Architecture



OLAP / Decision Support
Data Cubes / Data Mining

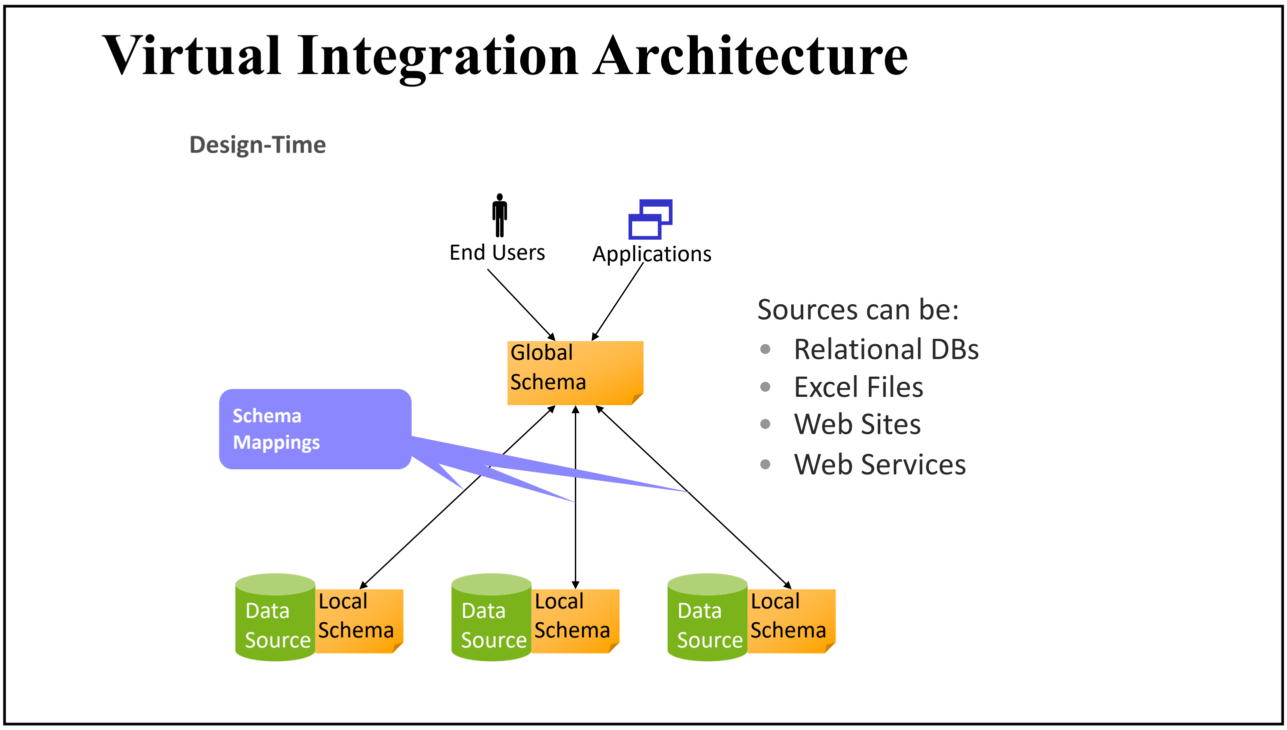
ETL Tools
(Extract-Transform-Load)

Data Cleaning

Virtual Integration Architecture

- Leave the data in the sources
- When a query comes in:
  - Determine the relevant sources to the query
  - Break down the query into sub-queries for the sources
  - Get the answers from the sources, filter them if needed and combine them appropriately
- Data is fresh
- Otherwise known as

On Demand Integration



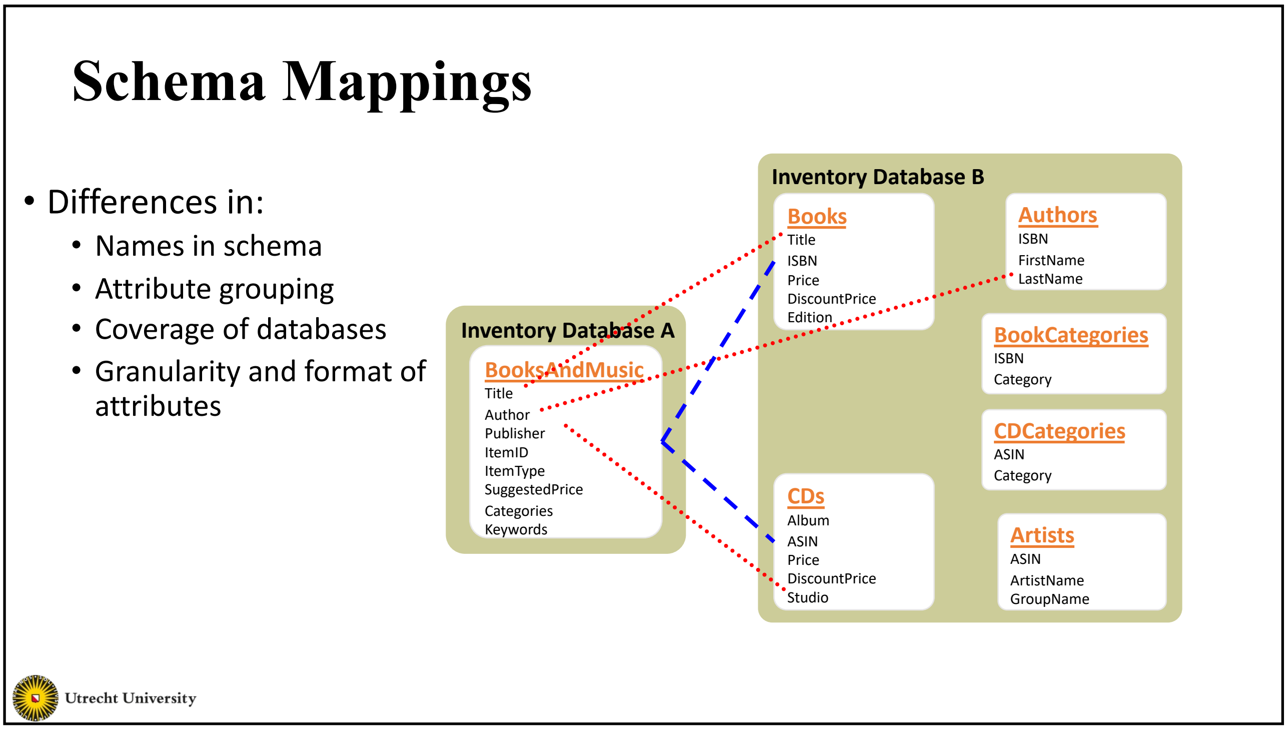
Virtual Integration Architecture

Design-Time

Sources can be:

- Relational DBs
- Excel Files
- Web Sites
- Web Services

37

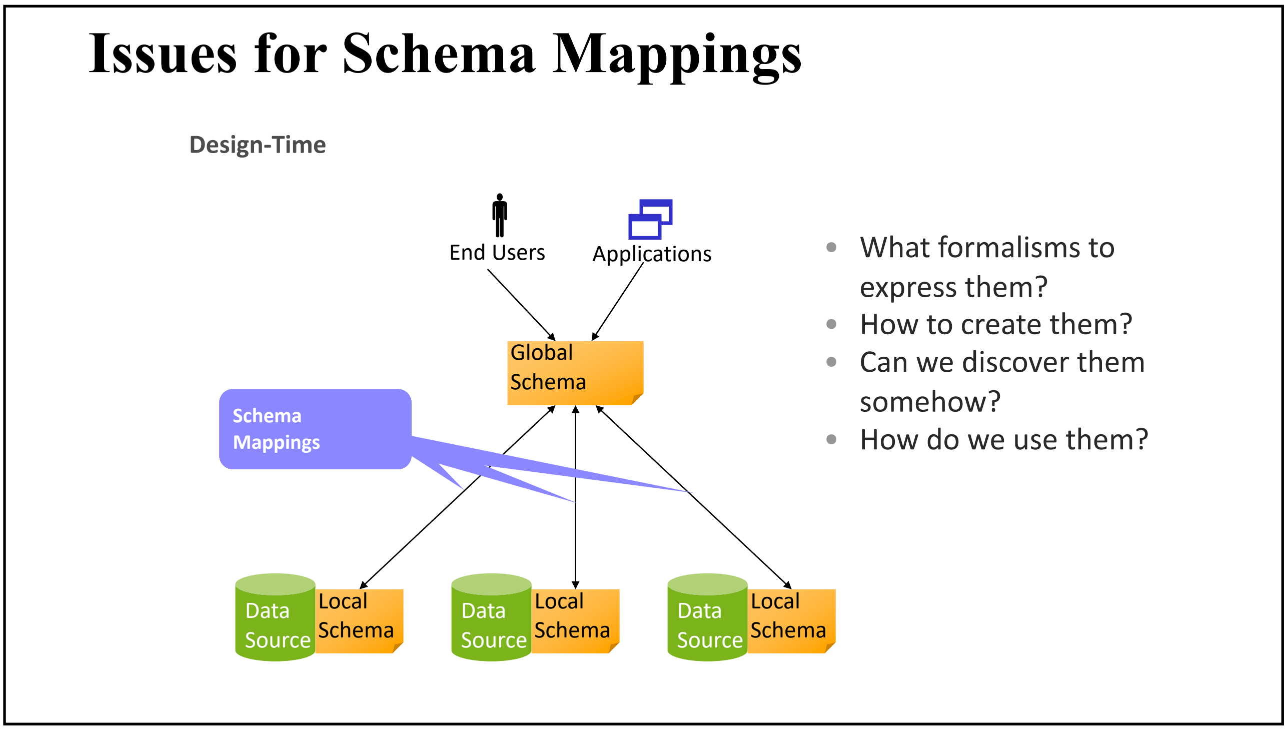


Schema Mappings

- Differences in:
  - Names in schema
  - Attribute grouping
  - Coverage of databases
  - Granularity and format of attributes

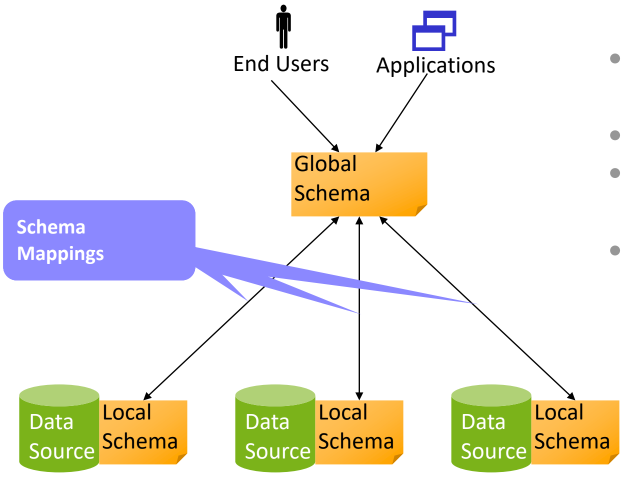
Utrecht University

38



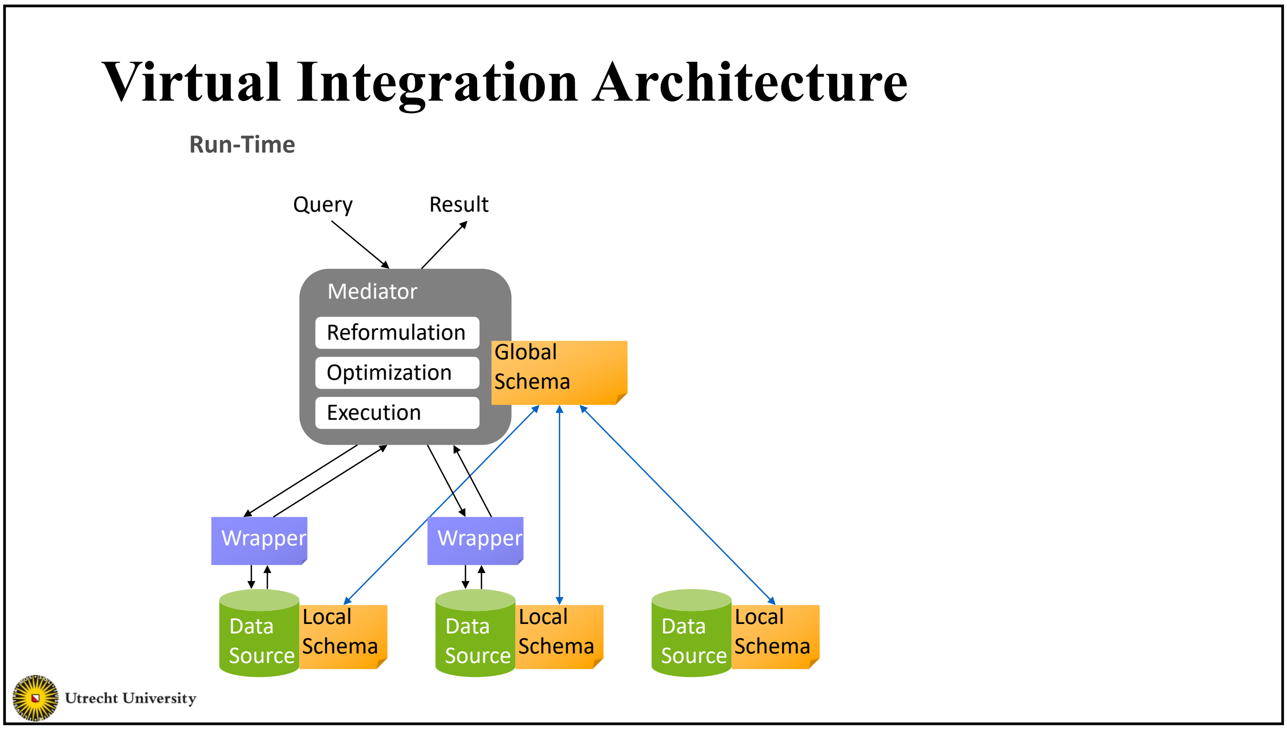
Issues for Schema Mappings

Design-Time



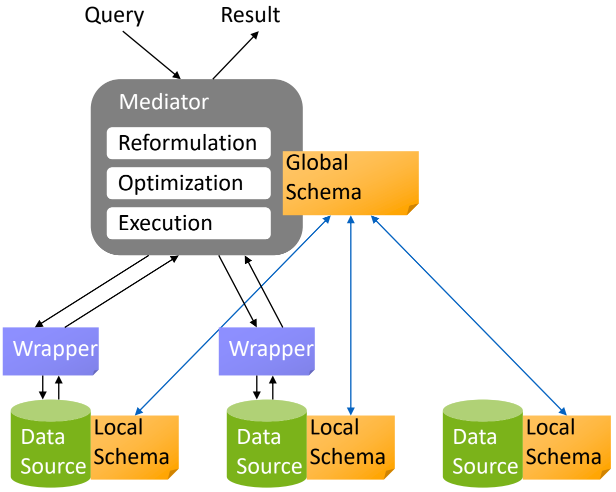
- What formalisms to express them?
- How to create them?
- Can we discover them somehow?
- How do we use them?

39



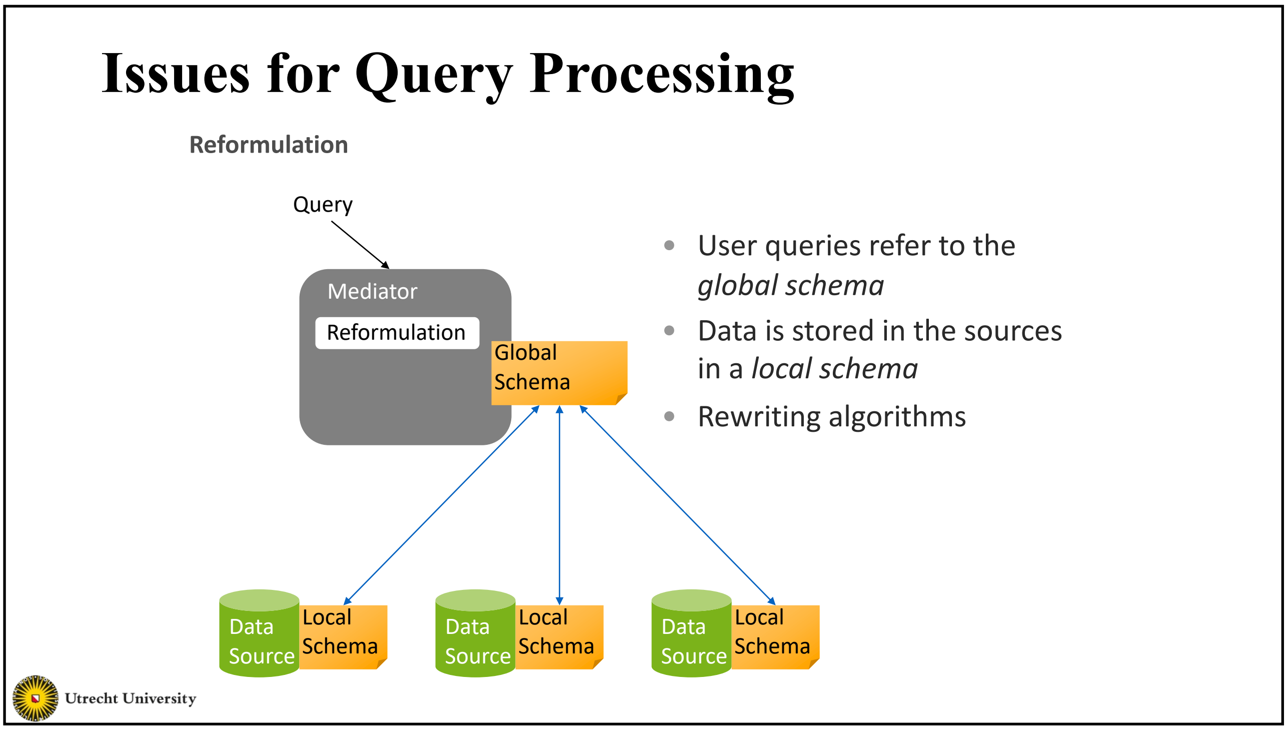
Virtual Integration Architecture

Run-Time



Utrecht University

40



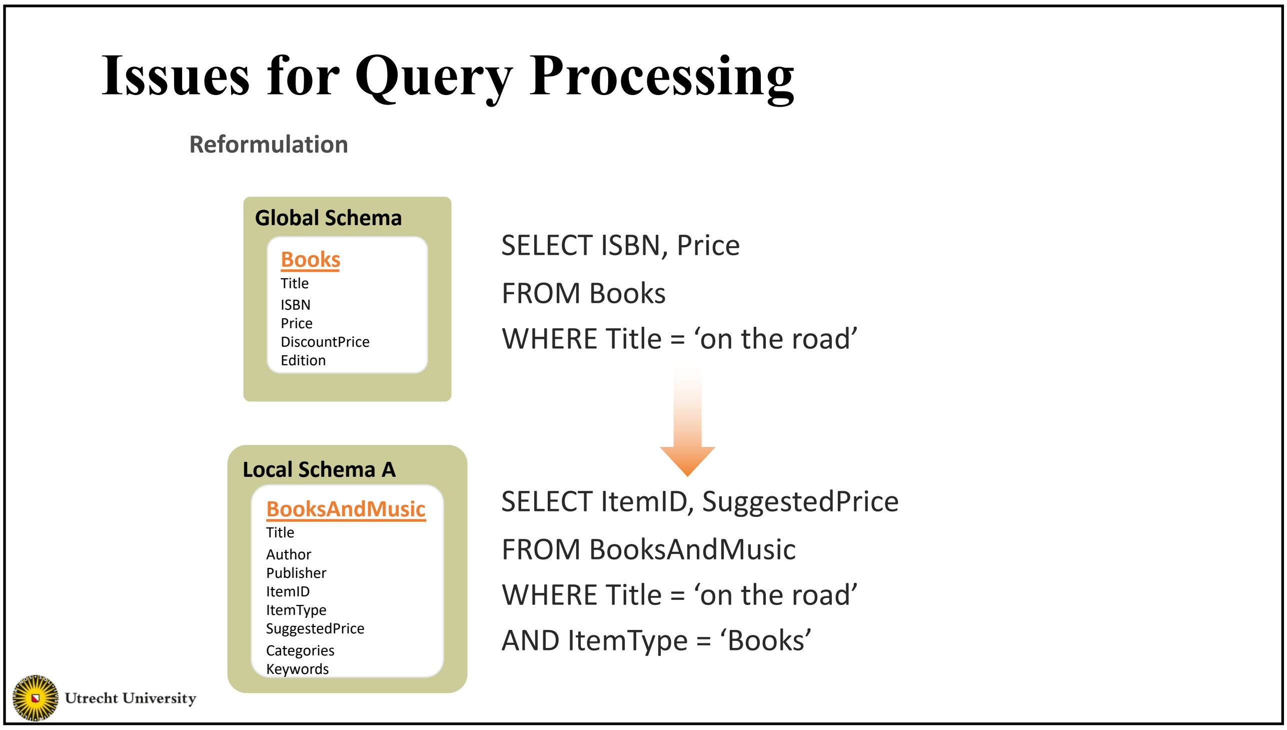
Issues for Query Processing

Reformulation

- User queries refer to the *global schema*
- Data is stored in the sources in a *local schema*
- Rewriting algorithms

Utrecht University

41



Issues for Query Processing

Reformulation

Global Schema

Books
Title
ISBN
Price
DiscountPrice
Edition

Local Schema A

BooksAndMusic
Title
Author
Publisher
ItemID
ItemType
SuggestedPrice
Categories
Keywords

```
SELECT ISBN, Price
FROM Books
WHERE Title = ‘on the road’
```

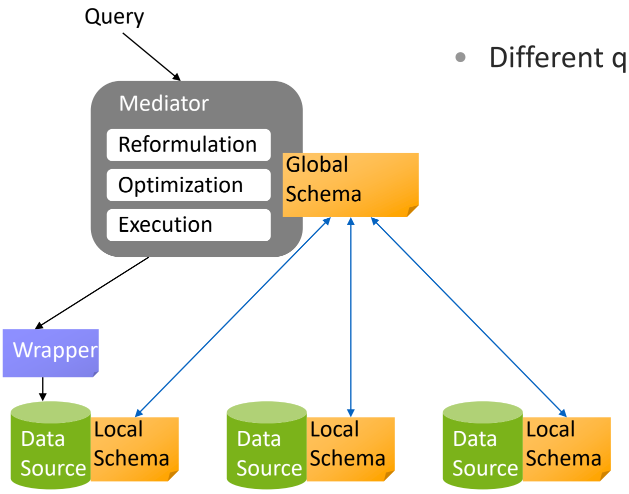
```
SELECT ItemID, SuggestedPrice
FROM BooksAndMusic
WHERE Title = ‘on the road’
AND ItemType = ‘Books’
```

Utrecht University

42

Issues for Query Processing

Query Translation



- Different query languages

Utrecht University

43

Issues for Query Processing

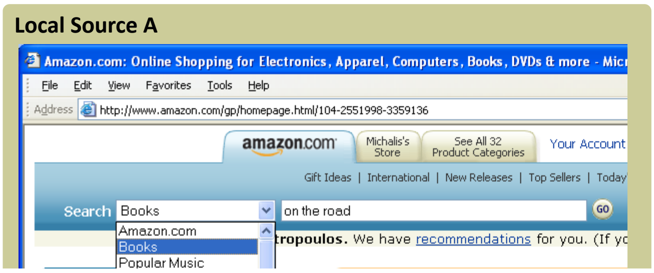
Query Translation

Global Schema

Books
Title
ISBN
Price
DiscountPrice
Edition

SELECT ISBN, Price
FROM Books
WHERE Title = ‘on the road’

Local Source A


Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more - Micr
File Edit View Favorites Tools Help
Address
amazon.com
Michalis's Store
See All 32 Product Categories
Your Account
Gift Ideas | International | New Releases | Top Sellers | Today
GO
Amazon.com
Books
Popular Music

Address: http://www.amazon.com/gp/homepage.html/104-2551998-3359136

Search: Books — on the road

tropoulos. We have recommendations for you. (If yo

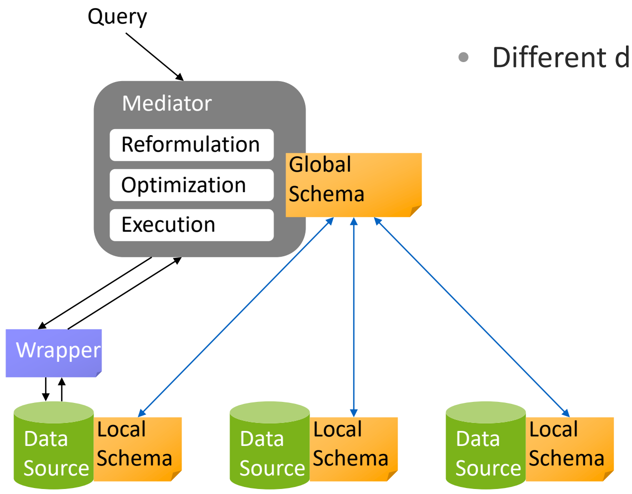
http://www.amazon.com/homepage.html?ItemType=Books&Title=on+the+road

Utrecht University

44

Issues for Query Processing

Data Translation



- Different data models

Issues for Query Processing

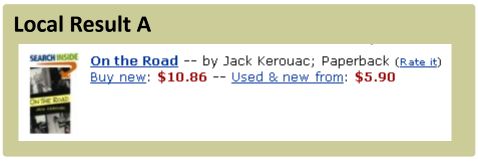
Data Translation

Global Schema

Books
Title
ISBN
Price
DiscountPrice
Edition

Title	ISBN	Price	…	…
On the Road	123	10.86	…	…

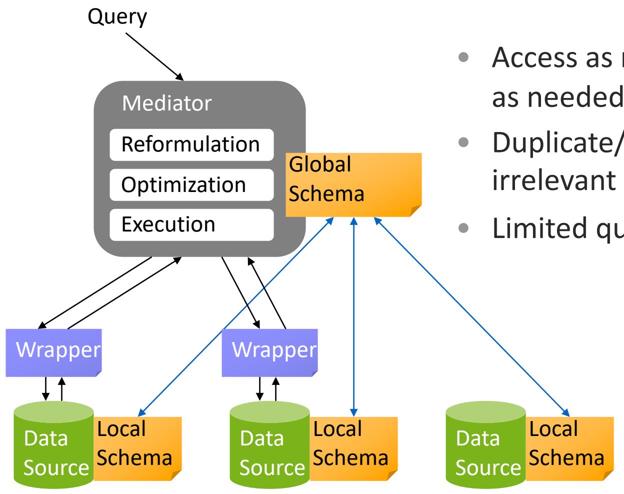
Local Result A


On the Road -- by Jack Kerouac; Paperback (Rate it)
Buy new: $10.86 -- Used & new from: $5.90

```
<table>
  <tr>
    <td>
      <a href=/details?isbn=123>
        <b>On the Road</b>
      </a>
      -- by Jack Kerouac; Paperback
      <br>
      <a href=/details?isbn=123>
        Buy new
      </a>
      :<b class=price>$10.86</b>
    </td>
  </tr>
</table>
```

Issues for Query Processing

Query Execution

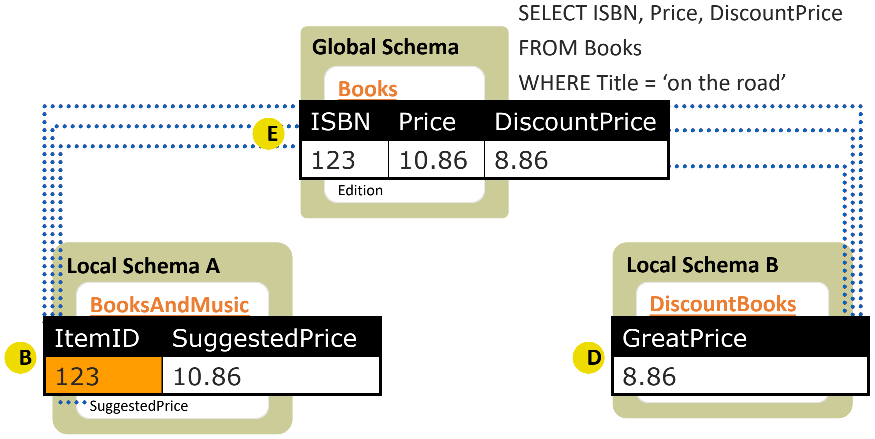


- Access as many data sources as needed
- Duplicate/redundant and irrelevant data
- Limited query capabilities

Utrecht University

Issues for Query Processing

Limited Query Capabilities



```
SELECT ISBN, Price, DiscountPrice
FROM Books
WHERE Title = ‘on the road’
```

ISBN	Price	DiscountPrice
123	10.86	8.86

ItemID	SuggestedPrice
123	10.86

GreatPrice
8.86

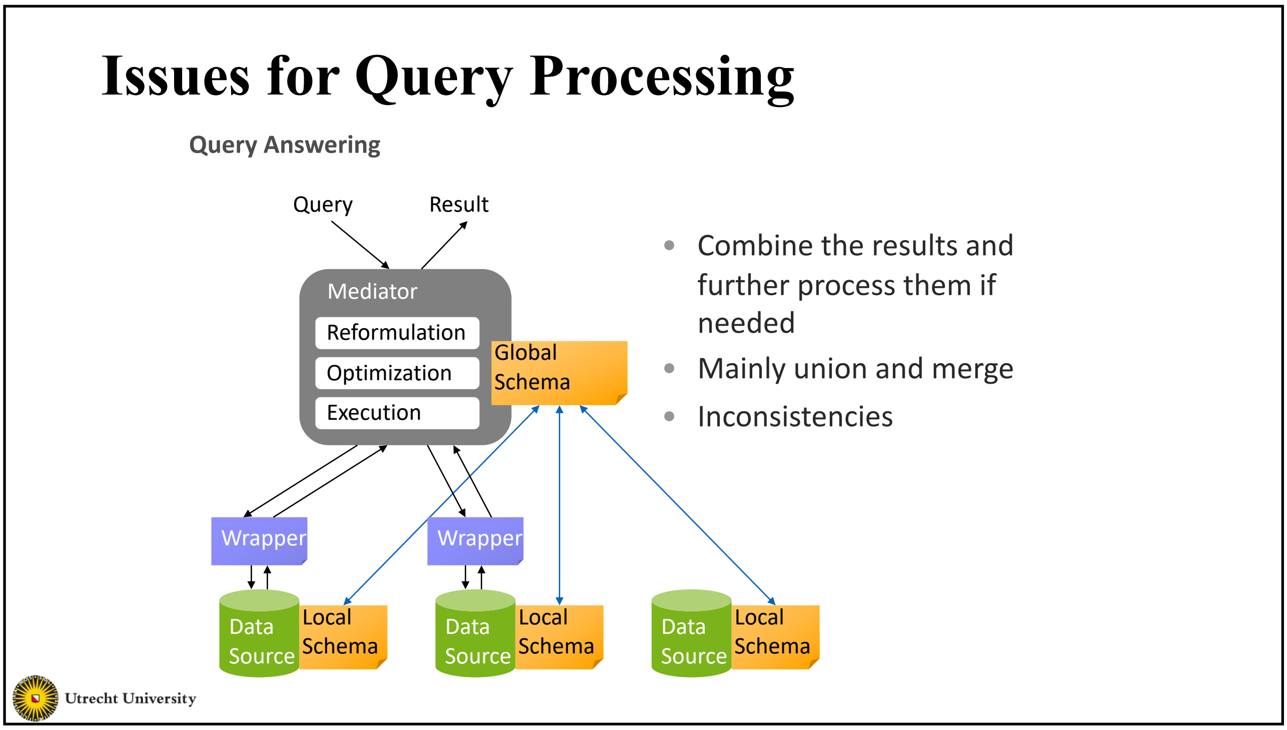
A

```
SELECT ItemID, SuggestedPrice
FROM BooksAndMusic
WHERE Title = ?on the road’
```

C

```
SELECT GreatPrice
FROM DiscountBooks
WHERE ISBN = ?23
```

Utrecht University

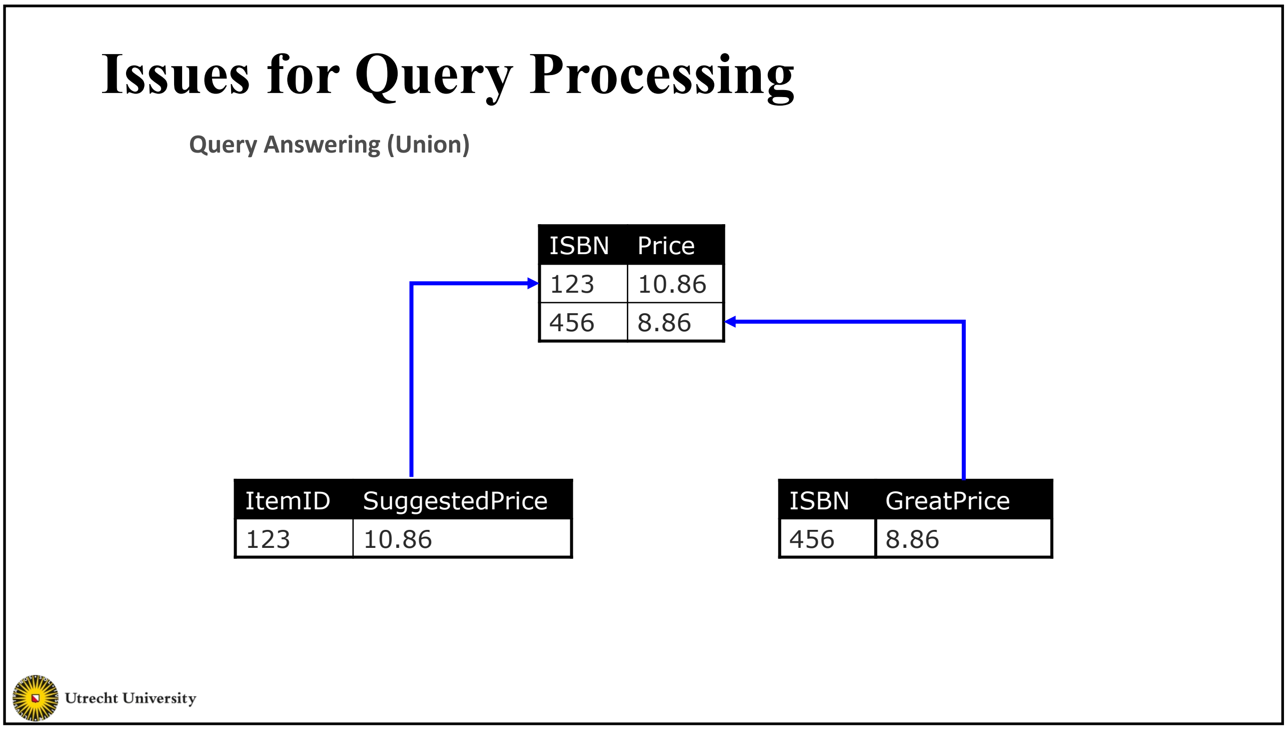


Issues for Query Processing

Query Answering

- Combine the results and further process them if needed
- Mainly union and merge
- Inconsistencies

49



Issues for Query Processing

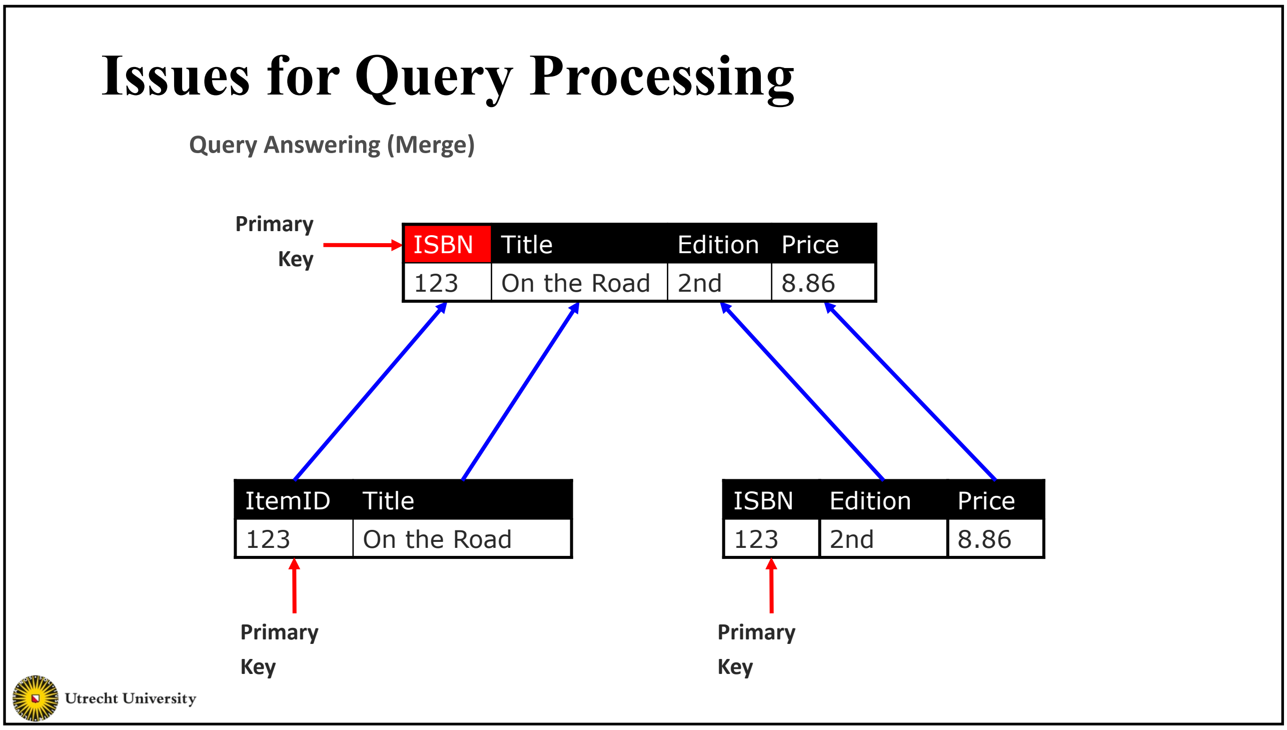
Query Answering (Union)

ISBN	Price
123	10.86
456	8.86

ItemID	SuggestedPrice
123	10.86

ISBN	GreatPrice
456	8.86

50



Issues for Query Processing

Query Answering (Merge)

Primary Key

ISBN	Title	Edition	Price
123	On the Road	2nd	8.86

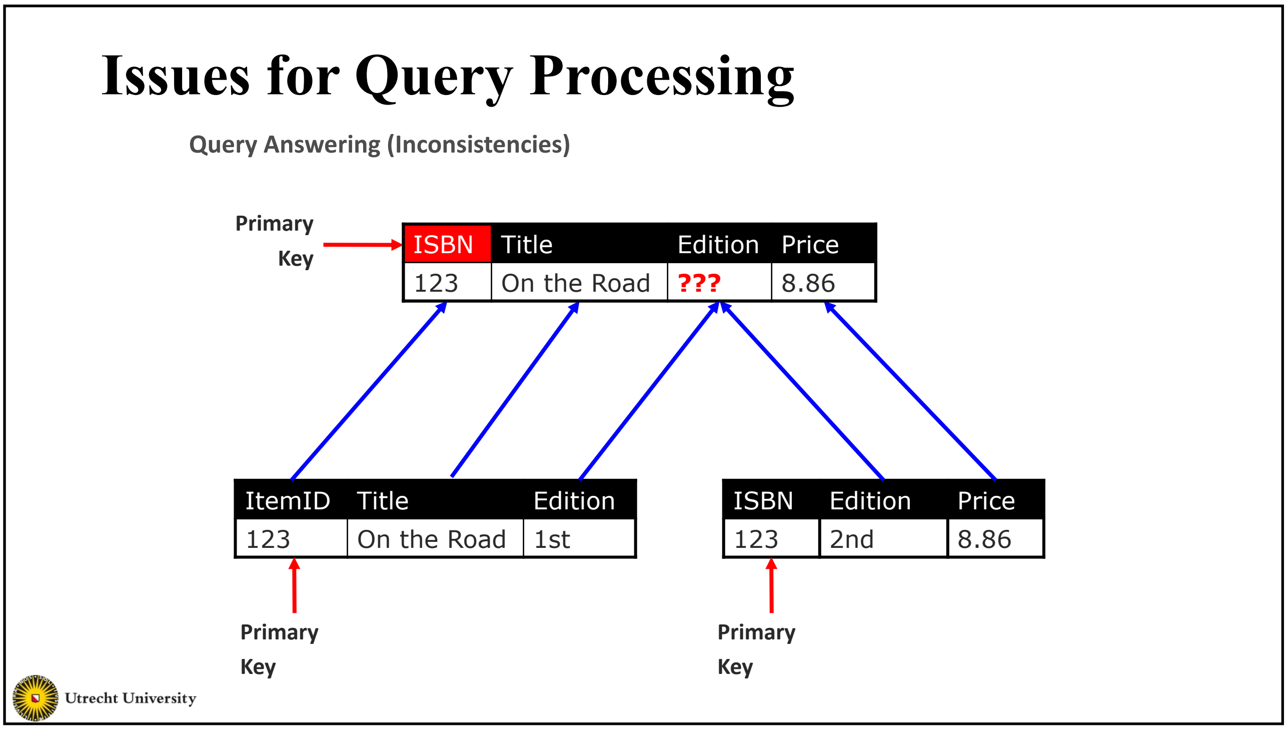
ItemID	Title
123	On the Road

Primary Key

ISBN	Edition	Price
123	2nd	8.86

Primary Key

51



Issues for Query Processing

Query Answering (Inconsistencies)

Primary Key

ISBN	Title	Edition	Price
123	On the Road	???	8.86

ItemID	Title	Edition
123	On the Road	1st

Primary Key

ISBN	Edition	Price
123	2nd	8.86

Primary Key

52