Outline

ILLINOIS INSTITUTE OF TECHNOLOGY

0) Course Info

1) Introduction

2) Data Preparation and Cleaning

3) Schema mappings and Virtual Data Integration

4) Data Exchange

5) Data Warehousing

6) Big Data Analytics

7) Data Provenance

About me

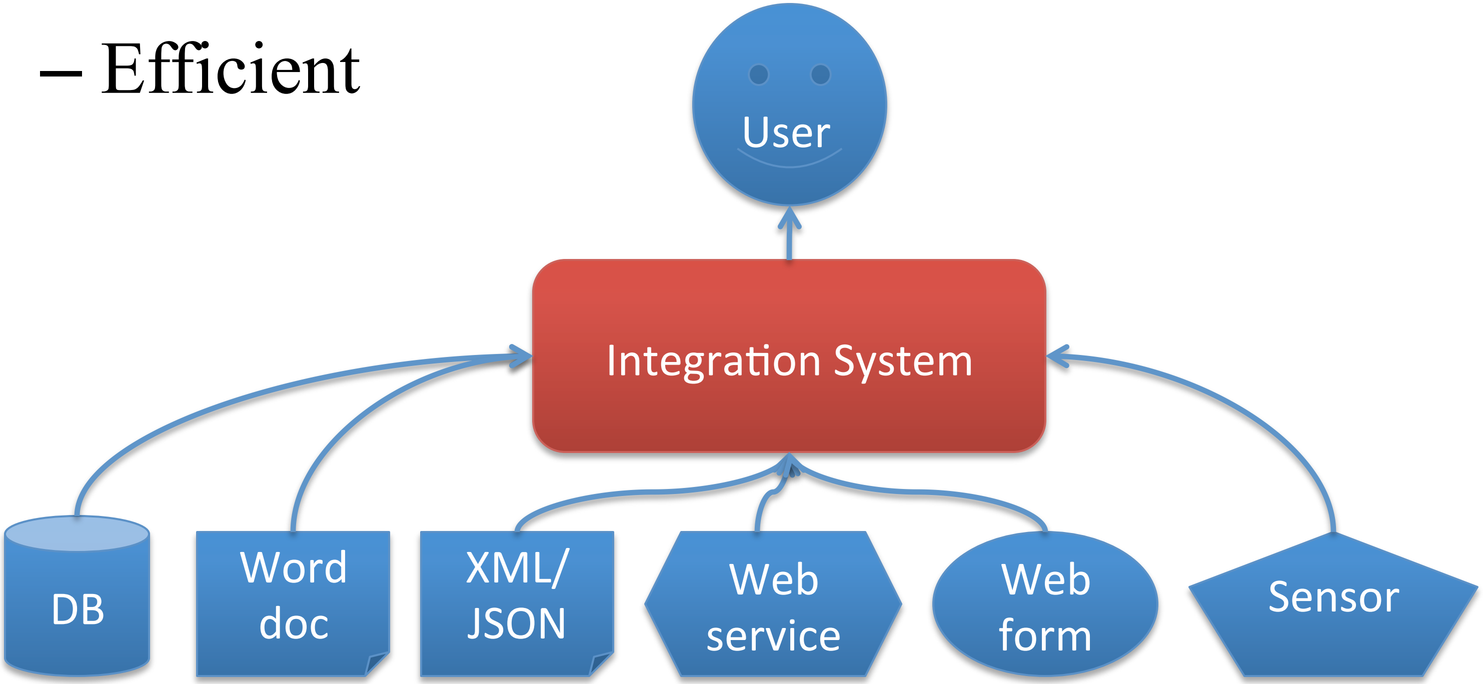
Hi, I am **Boris Glavic, Assistant Professor** in **CS**

I am a **database** guy!

I will teach you:
database stuff

What is information integration?

- Combination of data and content from multiple sources into a common format
  - Completeness
  - Correctness
  - Efficient



Why Information Integration?

- Data is already available, right?
- …, but
- Heterogeneity
  - Structural
    - Data model (relational, XML, unstructured)
    - Schema (if there)
  - Semantic
    - Naming and identity conflicts
    - Data conflicts
  - Syntactic
    - Interfaces (web form, query language, binary file)

Why Information Integration?

- Autonomy
  - Sources may not give you unlimited access
    - Web form only support a fixed format of queries
    - Does not allow access to unlimited amounts of data
  - Source may not be available all the time
    - Naming and identity conflicts
    - Data conflicts
  - Data, schema, and interfaces of sources may change
    - Potentially without notice

"Real World" Examples?

- Portal websites
  - Flight websites (e.g., Expedia) gather data from multiple airlines, hotels
- Google News
  - Integrates information from a large number of news sources
- Science:
  - Biomedical data source
- Business
  - Warehouses: integrate transactional data

Example Integration Problem [1]

- Integrate stock ticker data from two web services A and B
  - **Service A**: Web form (Company name, year)
  - **Service B**: Web form (year)

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [2]

ILLINOIS INSTITUTE
OF TECHNOLOGY

- **Service A:**

```
<Stock>
        <Company>IBM</Company>
        <DollarValue>155.8</DollarValue>
        <Month>12</Month>
</Stock>
```

- **Service B:**

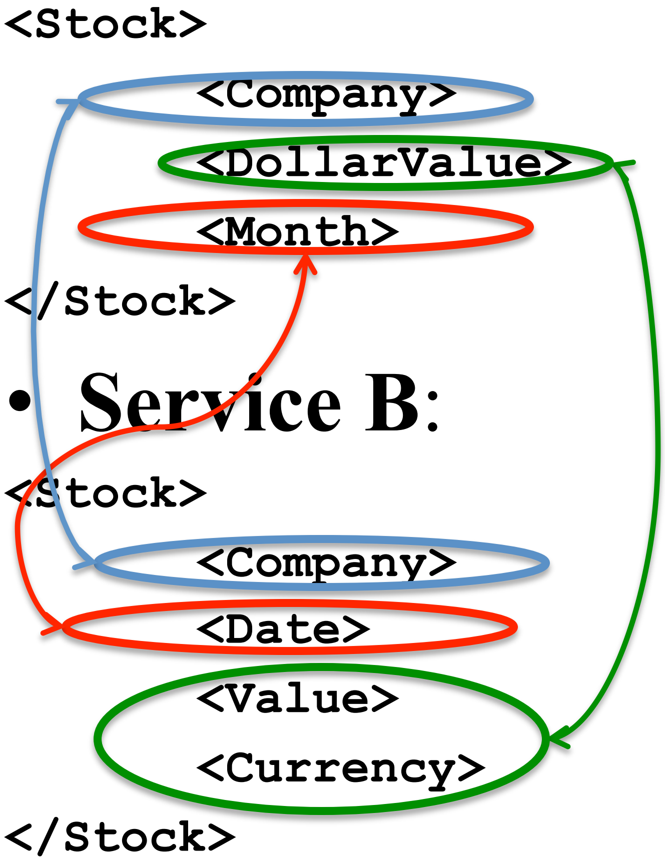
```
<Stock>
        <Company>International Business Machines</Company>
        <Date>2014-08-01</Date>
        <Value>106.8</Value>
        <Currency>Euro</Currency>
</Stock>
```

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [2]

- **Service A:**



```
<Stock>
        <Company>
        <DollarValue>
        <Month>
</Stock>
```

- **Service B:**

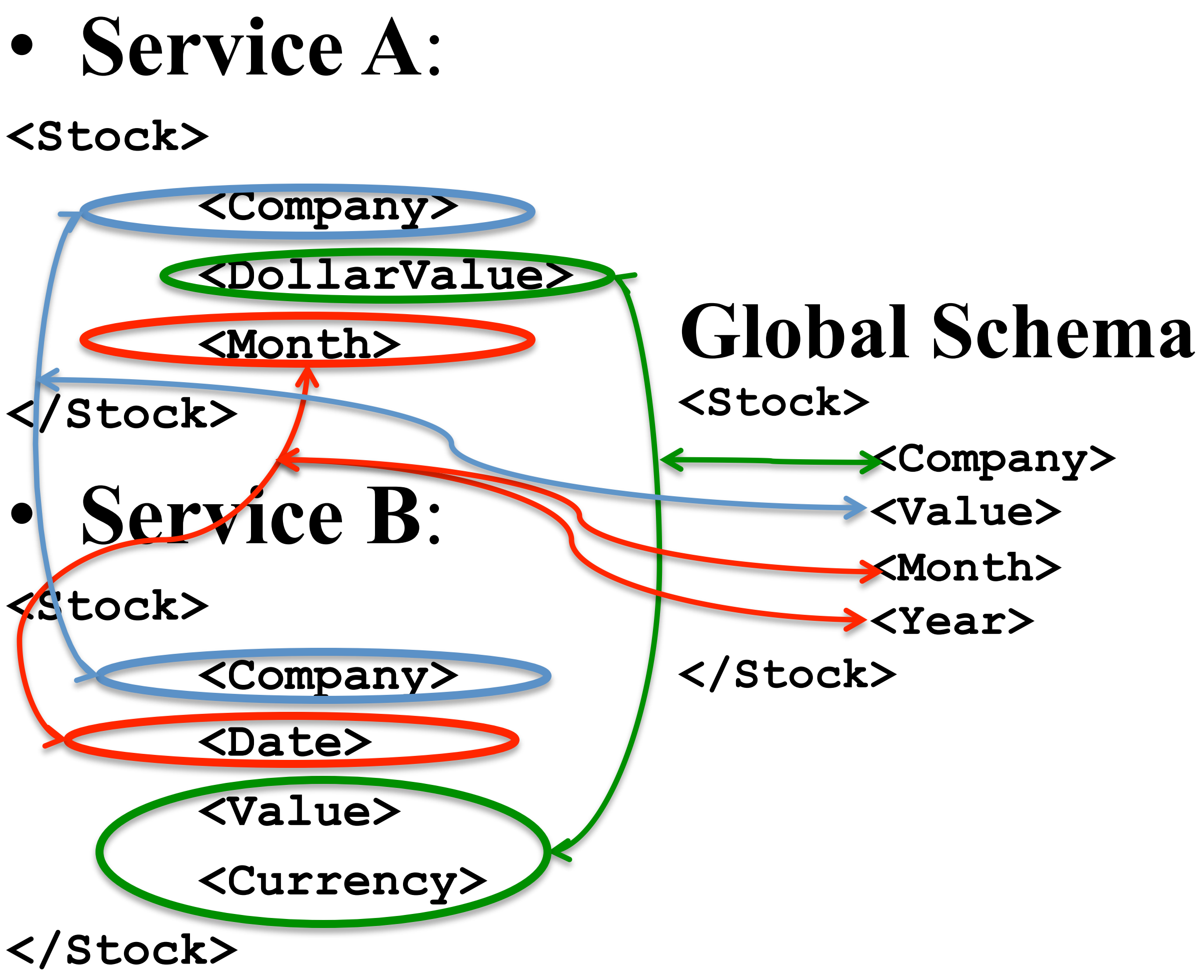
```
<Stock>
        <Company>
        <Date>
        <Value>
        <Currency>
</Stock>
```

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [2]

ILLINOIS INSTITUTE OF TECHNOLOGY



- **Service A:**

```
<Stock>
        <Company>
        <DollarValue>
        <Month>
</Stock>
```

- **Service B:**

```
<Stock>
        <Company>
        <Date>
        <Value>
        <Currency>
</Stock>
```

Global Schema

```
<Stock>
        <Company>
        <Value>
        <Month>
        <Year>
</Stock>
```

<u>Steps</u>

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [3]

- SQL interface for integrated service

```
SELECT month, value
FROM ticker
WHERE year = 2014
        AND cmp = ‘IBM’
```

- Service A: **(IBM, 2014)**
- Service B: **(2014)**

<u>Steps</u>

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [4]

- For web service A we can either
  - Get stocks for **IBM** in **all years**
  - Get stocks for **all companies** in **2014**
  - Get stocks for **IBM** in **2014**
- Trade-off between amount of processing that we have to do locally, amount of data that is shipped, …

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [5]

- **Service A**: (IBM, 2014)
- **Service B**: (2014)

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [6]

- Service A:

```
<Stock>
        <Company>IBM</Company>
        <DollarValue>155.8</DollarValue>
        <Month>12</Month>
…
```

- Service B:

```
<Stock>
        <Company>International Business Machines</Company>
        <Date>2014-12-01</Date>
        <Value>106.8</Value>
        <Currency>Euro</Currency>
…
```

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [7]

- IBM vs. Integrated Business Machines

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [8]

- Granularity of time attribute
  - Month vs. data
- What if both services return different values (after adapting granularity)
  - Average?
  - Median?
  - Trust-based?

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Example Integration Problem [9]

ILLINOIS INSTITUTE OF TECHNOLOGY

- Return final results:

```
<Stock>
        <Month>01</Month>
        <Value>105</Value>
</Stock>
…
<Stock>
        <Month>12</Month>
        <Value>107</Value>
</Stock>
```

Steps

1) Interfaces
2) Schema integration
3) Translate queries
4) Optimization
5) Send queries to sources
6) Gather query results
7) Entity resolution
8) Fusion
9) Return final results

Why hard?

- System challenges
  - Different platforms (OS/Software)
  - Efficient query processing over multiple heterogeneous systems
- Social challenges
  - Find relevant data
  - Convince people to share their data
- Heterogeneity of data and schemas
  - A problem that even exists if we use same system

Why hard? Cont.

- Often called **AI-complete**
  - Meaning: “It requires human intelligence to solve the problem”
  - Unlikely that general completely automated solutions will exit
- So why do we still sit here
  - There exist automated solutions for relevant less general problems
  - Semi-automated solutions can reduce user effort (and may be less error prone)

AI completeness

- Yes, but still why is this problem really so hard?
  - **Lack of information**: e.g., the attributes of a database schema have only names and data types, but no computer interpretable information on what type of information is stored in the attribute
  - **Undecidable computational problems**: to decide whether a user query can be answered from a set of sources that provide different views on the data requires **query containment** checks which are undecidable for certain query types

Relevant less general problems

- **Data cleaning**:
  - Clean dirty data before integration
  - Conformance with a set of constraints
  - Deal with missing and outlier values
- **Entity resolution**
  - Determine which objects from multiple dataset represent the same real world entity
- **Data fusion**
  - Merge (potentially conflicting) data for the same entity

Relevant less general problems

- **Schema matching**
  - Given two schemas determine which elements store the same type of information
- **Schema mapping**
  - Describe the relationships between schemas
    - Allows us to rewrite queries written against one schema into queries of another schema
    - Allows us to translate data from one schema into

Relevant less general problems

- **Virtual data integration**
  - Answer queries written against a **global mediated schema** by running queries over **local sources**
- **Data exchange**
  - Map data from one schema into another
- **Warehousing: Extract, Transform, Load**
  - Clean, transform, fuse data and load it into a data warehouse to make it available for analysis

Relevant less general problems

- **Integration in Big Data Analytics**
  - Often “pay-as-you-go”:
    - No or limited schema
    - Engines support wide variety of data formats
- **Provenance**
  - Information about the origin and creation process of data
  - Very important for integrated data
    - E.g., “from which data source is this part of my query result”

Webpage and Faculty

- **Course Info**
  - **Course Webpage**: http://cs.iit.edu/~cs520
  - **Google Group**: https://groups.google.com/d/forum/cs520-2015-spring-group
    - Used for announcements
    - Use it to discuss with me, TA, and fellow students
  - **Syllabus:** http://cs.iit.edu/~cs520/files/syllabus.pdf
- **Faculty**
  - **Boris Glavic** (http://cs.iit.edu/~glavic)
  - **Email:** bglavic@iit.edu
  - **Phone**: 312.567.5205
  - **Office**: Stuart Building, room 226C
  - **Office Hours**: Mondays, 12pm-1pm
    (and by appointment)

TAs

- TAs
  - TBA

Workload and Grading

- **Exams (60%)**
  - Final
- **Homework Assignments** (preparation for exams!)
  - Practice theory for final exam
  - Practice the tools we discuss in class
- **Literature Review (40%)**
  - In groups of 2 students
  - Topics will be announced soon
  - You have to read a research paper
  - Papers will be assigned in the first few weeks of the course
  - You will give a short presentation (15min) on the topic in class
  - You will write a report summarizing and criticizing the paper (up to 4 pages)

Course Objectives

- Understand the problems that arise with querying heterogeneous and autonomous data sources
- Understand the differences and similarities between the data integration/exchange, data warehouse, and Big Data analytics approaches
- Be able to build parts of a small data integration pipeline by “glueing” existing systems with new code

Course Objectives cont.

- Have learned formal languages for expressing schema mappings
- Understand the difference between virtual and materialized integration (data integration vs. data exchange)
- Understand the concept of data provenance and know how to compute provenance

Fraud Policies

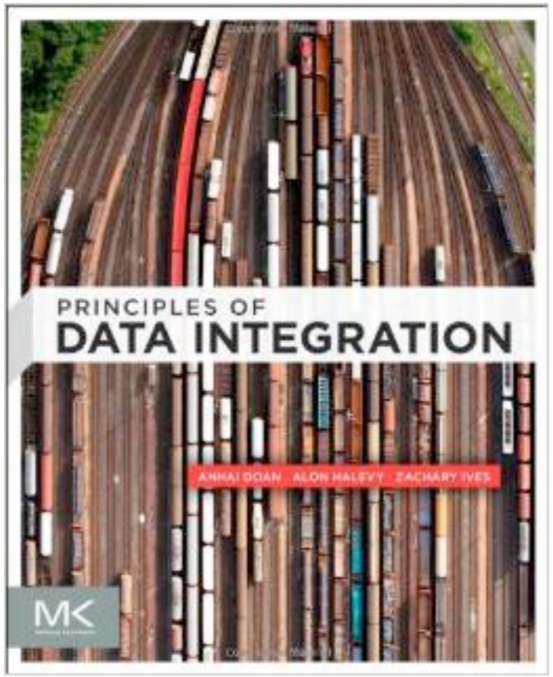
- All work has to be original!
  - Cheating = 0 points for review/exam
  - Possibly E in course and further administrative sanctions
  - Every dishonesty will be reported to office of academic honesty
- Late policy:
  - -20% per day
  - You have to give your presentation to pass the course!
  - No exceptions!

Fraud Policies cont.

- Literature Review:
  - – Every student has to contribute in both the presentation and report!
  - – **Don’t let others freeload on you hard work!**
    - Inform me or TA immediately

Reading and Prerequisites

- **Textbook:** Doan, Halevy, and Ives.
  - **Principles of Data Integration**, 1st Edition
  - Morgan Kaufmann
  - Publication date: 2012
  - ISBN-13: 978-0124160446
  - Prerequisites:
    - CS 425



Additional Reading

- Papers assigned for literature review
- Optional: Standard database textbook

Outline

0) Course Info

1) Introduction
2) Data Preparation and Cleaning
3) Schema mappings and Virtual Data Integration
4) Data Exchange
5) Data Warehousing
6) Big Data Analytics
7) Data Provenance