

Database Theory

Unit 0 — What is Database Theory?

What is the “shape” of our data?

User Side
(very minor role in this course)

Interpretability of queries.
Natural language to formal query
language.

Relational, Graph, Text, Semi-
structured XML/JSON/etc), ...

Streaming, distributed, privacy
preserving,

User

Question / Query



What language can we use
to define the question?

Logic, Automata, SQL, GQL, Regular
Path Queries, SPARQL, XPATH, GNNs,
Document Spanners, ...

Queries

Query Languages

Complexity

How difficult is it to answer queries in the language?

Expressivity

What can be asked with the language?

Equivalence

When do two queries always give the same result over every database?

Complexity

- Dependence on only the data, only the query, both, tradeoffs?
- If intractable, can we identify special cases for which there are efficient algorithms?
- Which features of the language are the cause of higher complexity?
→ Influence back on language design.

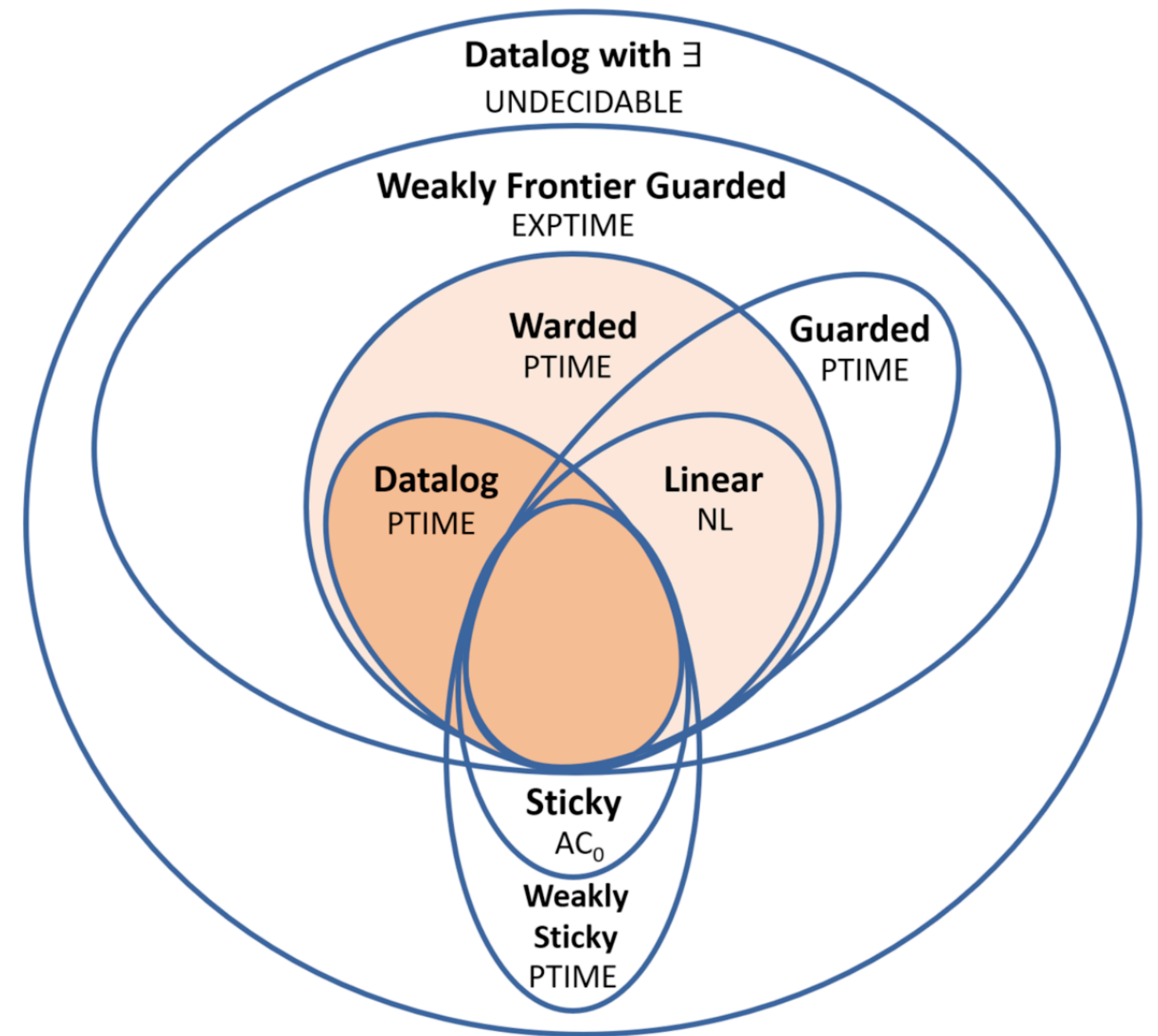


Figure 1: Syntactic containment of Datalog[±] languages. Annotations (non-bold) denote data complexity. All names that do not explicitly mention Datalog refer to the respective Datalog[±] languages. E.g., “Sticky” refers to “Sticky Datalog[±]”.

Expressivity

- Given two query languages \mathcal{L}_1 and \mathcal{L}_2 . For every query q in \mathcal{L}_1 , can we write a query in \mathcal{L}_2 that always produces the same outputs as q ?
- What kinds of concepts can a query language not ask about?
For instance, with first-order logic we cannot ask about reachability in a graph.

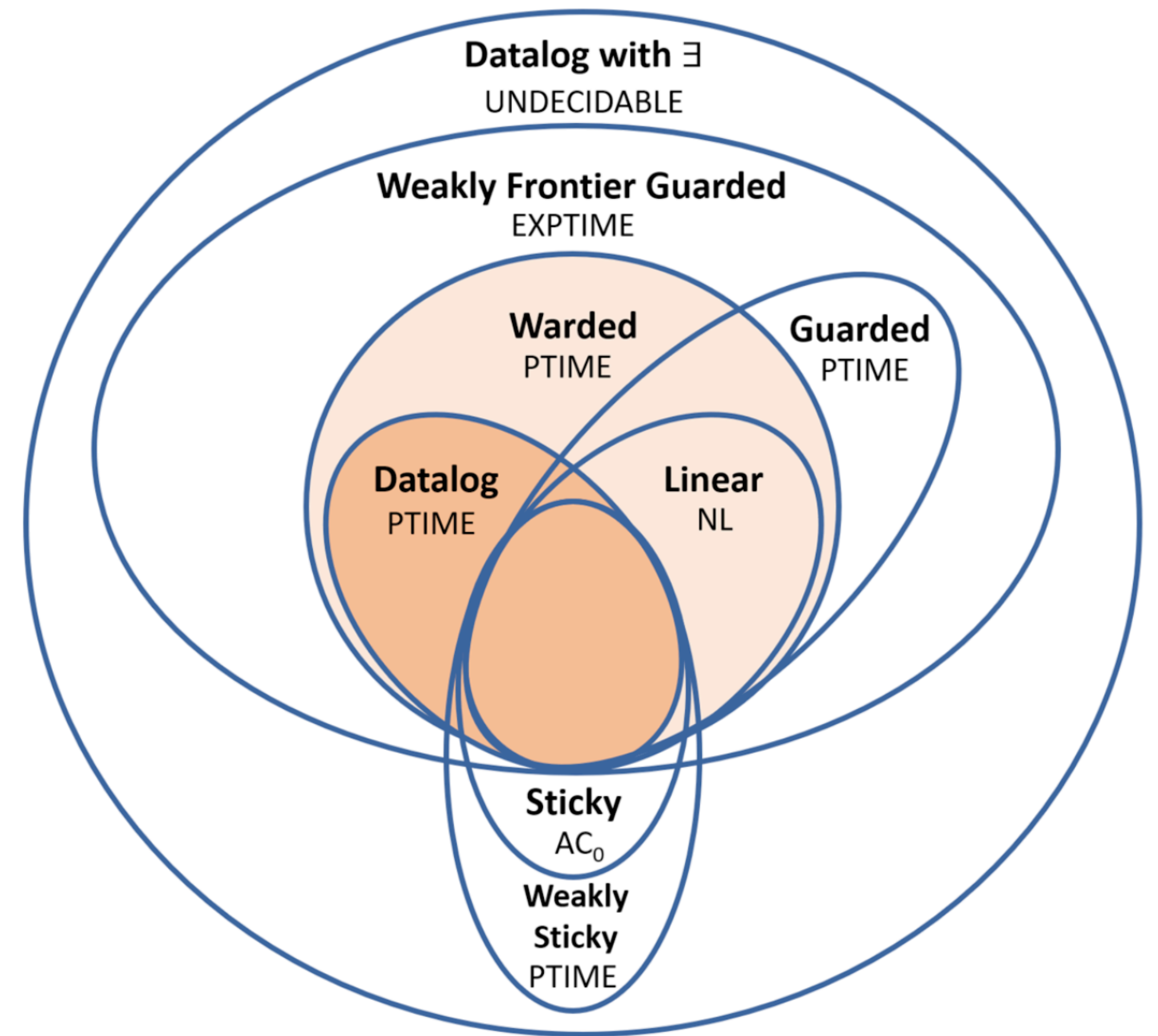


Figure 1: Syntactic containment of Datalog[±] languages. Annotations (non-bold) denote data complexity. All names that do not explicitly mention Datalog refer to the respective Datalog[±] languages. E.g., “Sticky” refers to “Sticky Datalog[±]”.

Equivalence

- Say we have two queries q and q' in the same query language. Are the outputs of the two queries the same on every database?
- If there are many equivalent ways to write the same query, which way is the most efficient with respect to evaluation?

Data Models

Data Model Dimensions

“Shape”

What is the **formal description** of how the data is stored?

Common examples:

- Relational
- Graphs
- Semi-structured
- Documents

Availability

How can data be accessed?

In **distributed settings**, access to data that is not local induces additional cost.

Streaming data only gives a small window into the whole data at any given time.

Further Complexities

Various additional notions have been added on top of the other two.

Data can be stored in some way that inherently preserves privacy.

Data can be probabilistic.

Further Topics

Data Quality

Cleaning up data in a formally justified way.

Discovering new data / expanding on the data currently in the database

Incompleteness & Uncertainty

Parts of data is missing (e.g. NULLs)

Data might not be 100% certain, e.g., data from ML models or crowdsourcing.

Data Access

Managing data under concurrency / transactions.

Indexing structures.

Connections to Other Areas

Complexity Theory

Complexity of Query Evaluation

Logic Programming

More advanced query languages fall into logic programming.

Constraint Satisfaction

CSP = Conjunctive Queries.

Logic / Model Theory

Expressiveness of languages.

Graph Theory

Structure of queries and data matters. Homomorphisms.

Automata

XML, Document Spanners,
Stream processing, Graph
Queries

This Lecture

- The lecture will focus primarily on the relational model.
Inherent connections to logic.
Most commonly studied and compatible with most query languages.
Common in practice (SQL, Logic, Graphs, Datalog, ...).
- Opportunity to look into other models via presentations.
Semi-structured data (XML/JSON).
Plain text as a data model.
Probabilistic data.

Database Theory

Unit 0.5 — General Information

Classes

Language:

All parts of the class are held in English

Time:

Every week of term: Fridays, 09:15 -

11:00

(see TISS in case of uncertainty)

Place:

All classes will be in person in room

Prerequisites

Level

This course is designed for master's and doctoral students.

Research-oriented /
independent study /
interactivity

Courses

It is highly recommended to take [Formal Methods in CS \(185.291\)](#) before this course.

[Complexity Theory \(181.142\)](#) can be helpful in parallel to this course.

Prerequisites

Level

This course is designed for master's and doctoral students.

Courses

It is highly recommended to take [Formal Methods in CS \(185.291\)](#) before this course.

[Complexity Theory \(181.142\)](#) can be helpful in parallel to this course.

Helpful Knowledge

- Database basics
- Formal logic
- Intro complexity theory
[In particular, reductions!](#)

Self Assessment

- Quiz on TUWEL course for you to judge your own knowledge on course prerequisites.
- Not graded, entirely **optional**.
- If you are still unsure, talk to me.

181.140-2024W / Self-Assessment / Self Assessment Quiz / Vorschau

TEST Self Assessment Quiz

Test [Einstellungen](#) [Fragen](#) [Ergebnisse](#) [Fragensammlung](#) [Mehr](#) ▼


Zurück

Frage 1

Bisher nicht beantwortet

Erreichbare Punkte: 1,00

 Frage markieren

 Frage bearbeiten

v1 (neueste)

Which of the following best describes the complexity class NP?

- a. Problems that can be solved in polynomial time by a deterministic Turing machine.
- b. Problems for which a solution can be verified in polynomial time by a deterministic Turing machine.
- c. Problems that cannot be solved by any Turing machine.
- d. Problems that can be solved in exponential time by a deterministic Turing machine.

Communication

- During & after classes
- TUWEL
- TISS
(room and time information)
- Course homepage
<https://dbai.tuwien.ac.at/staff/mlanzing/dbt/>



Assessment

Presentation

Presentation on one database theory research article.

Sufficient for mark 3.

Oral Exam

Only possible if presentation part was done. Optional if happy with lower mark.

Necessary for marks 1 and 2.

Presentation

- **Choose a paper**
Preselected list of around 20 database theory research papers on various topics that go beyond the lecture material.
- **Read and present**
Understand the main contributions and summarise them in a ~20min presentation for your colleagues.
- More information on the papers to choose from in a later class.

A Good Presentation

- Basic understanding of the article.
- Honestly identify parts which you did not understand.
E.g.: which prerequisites were missing or were not available to look up?
- Relate the paper to the contents of the course.
- Provide context of important background articles / topics.
- Think about how to communicate these theoretical ideas to the audience.
E.g., proof ideas and intuitions can be interesting but full proofs are often problematic in 20minutes.
- Be ready to answer questions about the article.