UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR INFORMATIONSSYSTEME

Özgür L. Özçep

Logic, Logic, and Logic

Lecture 1: Motivation and Overview
7. April 2022

Informationssysteme CS4130
(Summer 2022)

Organizational Stuff

Organization

- Lectures with self-test wake-up exercises
  - On site (IFIS 2035), but (old) lectures provided as slides and recordings in beforehand
  - Time: Thursdays, 12:15–13:45
  - Start: Thursday, 7 April 2021
- Exercise labs
  - on site (IFIS 2035)
  - Time: Thursdays 14:15–15:45
  - Start: Thursday, 7 April 2021
  - Discuss solutions presented by students
- Lecture and exercise related material in Moodle "Informationssysteme"
- Written exam at the end of the semester
  - Preliminary dates: 21 July 2022 (Second Chance: 4 October 2022)
  - Prerequisite for exam
    - Presentation of a solution for one exercise sheet
    - One-page-summary of lecture (handout)

Exercise Lab

- Weekly exercise sheets in Moodle (provided beforehand)
- Each group of (up to 2 or 3) students gets responsible for one exercise sheet and must present its solution
- Other students solve the exercise on their own: No solutions to be handed in/no corrections
- **YOUR TODO**: Sign in for an exercise sheet in Moodle

One-page summary

- Each group of students (=the same as exercise group) is responsible for summarizing one lecture (= the one associated with the exercise)
- Summary to be handed in within 2 weeks after exercise presentation (possibly revision required) via email to the lecturer
- meant as preparation for written exams for all students
- One page in any format you like / you find useful for preparing for the written exam
- Format may be as in
  - Theorem of the day (https://www.theoremoftheday.org/)
  - or a Poster format (but such that it is readable when printed in DIN A4)
  - See e.g. LaTeXformats at overleaf templates (https://www.theoremoftheday.org/)

Sometimes English Becomes Less Important

Prologue

La loi 101 (Charte de la langue française)

Principe du deux pour un□le texte français doit être écrit en caractères deux fois plus gros que ceux de la version en langue étrangère.

Two for one principle□an english (for clarity) text should be written in characters twice smaller than its french counterpart.

Exception□the english version of the text of the Law itself can be written in characters five times bigger than the french original.

Slide example by Bruno Poizat from a conference talk

- Model Theorist
- Has a wonderful (unconventional) book on model theory
  - Was not well received (for some years)
  - until he translated it into English

Lit: B. Poizat. A Course in Model Theory. Universitext. Springer Verlag, 2000.

Colour Coding and formatting for the Slides

- Formulae will be encoded in this greenish color
- Newly introduced terminology will be given in this light blue color and definitions will be given in light blue boxes
- Important results (observations, theorems) will be given in red boxes
- Emphasizing some aspects will done in red
- Examples will be given with orange boxes
- Bibliographic notes are given in this darker blue.

Agenda (overview)

- Logic, Logic, Logic (2 lectures)
- Logical Foundations of Database Systems: Finite Model Theory (2 lectures)
- Data Integration and Data Exchange (3 lectures)
- Semantic Integration with OBDA: Bridging the DB and Ontology World (2 lectures)
- Semantic Integration on Ontology Level: Ontology Integration (2 lectures)
- Database Repairs (1 lecture)
- Stream Processing (1 lecture)

First-Order Logic

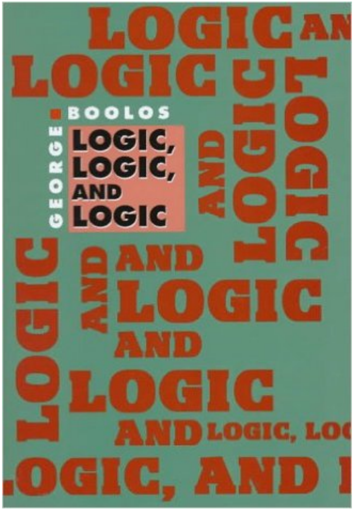
"Logic, Logic, and Logic"

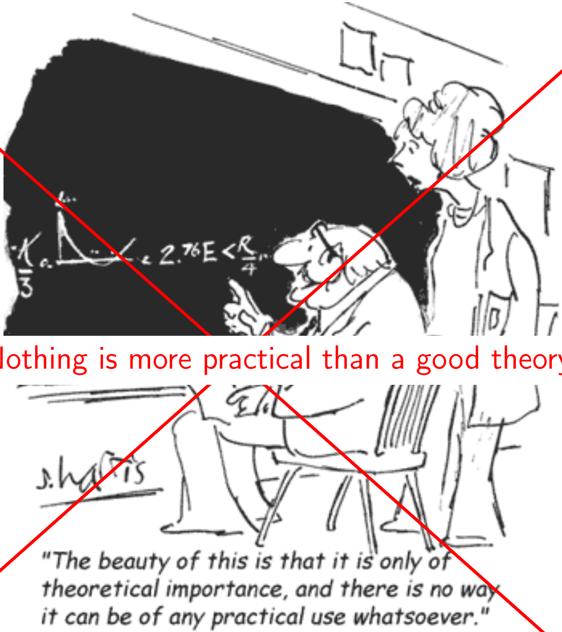
- Interesting collection of essays
- Rather "philosophical logic"

- But we adopt the motto:

  Logic everywhere !

- We are interested not only in logics per se but
- (Knowledge on) logics useful for computer science





But: "Nothing is more practical than a good theory"

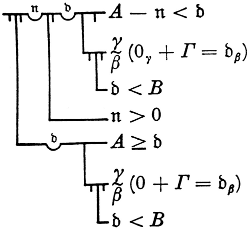
Logic and Logics

- Science of logic
  - investigates mathematical structures (static and dynamic)
  - and formal languages to describe them
  - distinguishing between syntax
  - and semantics (truth conditions for sentences)
  - providing notions of satisfaction, entailment (from semantics)
  - and of provability, inference (calculus)
- A logic: A language with syntax, semantics (and possibly calculus)
- There are many different logics (within computer science)
- But in any case somehow related to first-order logic

First-Order Logic (FOL)

- Also called predicate logic (or quantification logic)
- Aristotelian syllogisms already incorporate restricted FOL
  - All Philosophers are wise men. All wise men are nice. Hence all Philosophers are nice men.
  - Restricted to unary predicates

- Modern FOL started with Frege's "Begriffsschrift"
  - language constructs based on constants, variables, predicates, functions, boolean connectives, quantifiers
  - Formal axioms and inference rules
  - His 2-dimensional representation format aesthetic but not practical



FOL Structures

- A formalism to investigate (mathematical) **structures**

$$\mathfrak{A} = (A, R_1^{\mathfrak{A}}, \ldots R_n^{\mathfrak{A}}, f_1^{\mathfrak{A}}, \ldots, f_m^{\mathfrak{A}}, c_1^{\mathfrak{A}}, \ldots, c_l^{\mathfrak{A}})$$

- (Non-logical) Vocabulary
  - Relation symbols/predicates R_i with arities
  - Function symbols f_i (with arities)
  - Constant symbols c_i
- Components of the structure
  - Universe/Domain A
  - Interpretations/denotations of nonlogical symbols
    - Relation $R^{\mathfrak{A}} \subseteq A^n$ (for n-ary relation symbol R)
    - Function $f^{\mathfrak{A}} \in A^n \longrightarrow A$ (for n-ary function symbol f)
    - Individuals $c^{\mathfrak{A}} \in A$ (for constants c)

Example FOL Structures

- Graphs $\mathfrak{G} = (V, E^{\mathfrak{G}})$
  1. V = nodes of the graph
  2. $E^{\mathfrak{G}} \subseteq V^2$ = edges of the graph
- Undirected, loopless graphs $\mathfrak{G} = (V, E^{\mathfrak{G}})$
  1. as above
  2. as above
  3. Additionally: edge relation is symmetric and a-reflexive
- Need an appropriate language to formulate constraints such as in 3.

FOL Syntax

- Allow variables ($x_1, x_2, \ldots$) and logical constructors

- Terms
  - variables and constants are terms
  - if $t_1, \ldots, t_n$ are terms, so is $f(t_1, \ldots, t_n)$ (for n-ary function symbol f

- Formulae
  - $t_i = t_j$ and $R(t_1, \ldots, t_n)$ (for terms t_i and n-ary relation) R
  - If ϕ is a formula, so are
    - $\neg\phi$ ("Not ϕ")
    - $\forall x\, \phi$ ("For all x it holds that ϕ")
    - $\exists x\, \phi$ ("There is an x s.t. ϕ")

  - If ϕ, ψ are formula, so are
    - $(\phi \wedge \psi)$ ("ϕ and ψ")
    - $(\phi \vee \psi)$ ("ϕ or ψ")
    - $(\phi \rightarrow \psi)$ ("If ϕ then ψ")
    - $(\phi \leftrightarrow \psi)$ ("ϕ iff ψ")

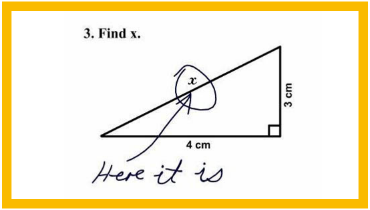
FOL Semantics

- Interpretation $\mathcal{I} = (\mathfrak{A}, \nu)$
  - ν assigns to all variables elements from domain A
  - Needed to deal with open formulae
    e.g. $\forall y\ R(y, x)$ open/free in variable x

- x-Variant $\mathcal{I}_{[x/d]}$
  same as $\mathcal{I}$ but with $d \in A$ assigned to x

- Interpretation of terms
  - $\mathcal{I}(c) = c^{\mathfrak{A}}$
  - $\mathcal{I}(x) = \nu(x)$
  - $\mathcal{I}(f(t_1, \ldots, t_n)) = f^{\mathfrak{A}}(\mathcal{I}(t_1), \ldots, \mathcal{I}(t_n))$

Because dealing with variables is non-trivial...



FOL Semantics

- Satisfaction relation $\models$
  - $\mathcal{I} \models t_1 = t_2$ iff $\mathcal{I}(t_1) = \mathcal{I}(t_2)$
  - $\mathcal{I} \models R(t_1, \ldots, t_n)$ iff $(\mathcal{I}(t_1), \ldots, \mathcal{I}(t_n)) \in R^{\mathfrak{A}}$

  - $\mathcal{I} \models \neg\phi$ iff not $\mathcal{I} \models \phi$

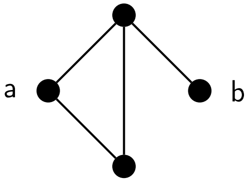
  - $\mathcal{I} \models (\phi \land \psi)$ iff $\mathcal{I} \models \phi$ and $\mathcal{I} \models \psi$
  - $\mathcal{I} \models (\phi \lor \psi)$ iff $\mathcal{I} \models \phi$ or $\mathcal{I} \models \psi$
  - $\mathcal{I} \models (\phi \rightarrow \psi)$ iff: If $\mathcal{I} \models \phi$ then $\mathcal{I} \models \psi$
  - $\mathcal{I} \models (\phi \leftrightarrow \psi)$ iff: $\mathcal{I} \models \phi$ iff $\mathcal{I} \models \psi$

  - $\mathcal{I} \models \forall x\ \phi$ iff for all $d \in A$: $\mathcal{I}_{[x/d]} \models \phi$
  - $\mathcal{I} \models \exists x\ \phi$ iff there is $d \in A$ s.t. $\mathcal{I}_{[x/d]} \models \phi$

- Known result: ν can be assumed to be defined only for the free variables in the formula.
- Terminology: $\mathcal{I}$ satisfies ϕ, $\mathcal{I}$ makes ϕ true, $\mathcal{I}$ is a model for/of ϕ
- We also write $\mathfrak{A} \models \phi(\vec{x}/\nu)$

Examples

- Consider loopless, symmetric graphs $\mathfrak{G} = (G, E^{\mathfrak{G}})$



- $\phi_1 := \exists x \, \exists y \, E(x, y)$ $\mathfrak{G} \models \phi_1$? Yes!
- $\phi_2(x) := \exists y \, \exists z \, E(x, y) \wedge E(x, z) \wedge E(y, z)$ $\mathfrak{G} \models \phi_2(x/a)$? Yes!
- $\phi_3(x, y) := E(x, y)$ $\mathfrak{G} \models \phi_3(x/a, y/b)$? NO!

Entailment

- $X \models \phi$ iff all models of X are models of ϕ
  - We say: X entails ϕ or ϕ follows from X
  - X: set of sentences
  - ϕ: sentence

- Note: entailment definition (per se) not easy implementable $\Longrightarrow$: Notion of derivibility/inference in a calculus (see later lectures)

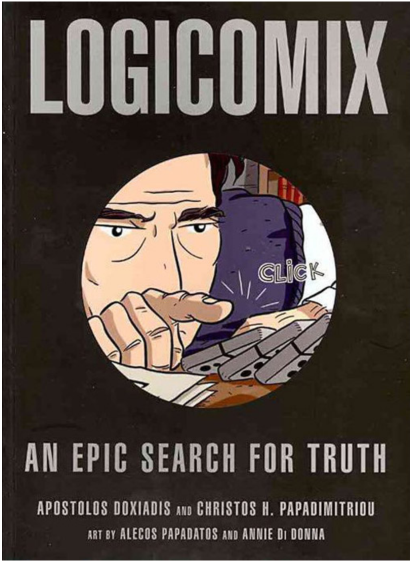
Algorithmic Problems in First-Order Logic

- Model Checking:
  - Input: graph (or generally structure) $\mathfrak{G}$, formula $\phi(x_1, \ldots, x_n)$ and assignment $[x_1/a_1, \ldots, x_n/a_n]$
  - Output: Is $\mathfrak{G} \models \phi(x_1/a_1, \ldots, x_n/a_n)$ the case?
- Satisfiability Problem
  - Input: sentence ϕ
  - Output: Does there exist a structure $\mathfrak{G}$ s.t. $\mathfrak{G} \models \phi$?
- Complexity of problems
  - Model checking problem is decidable and PSPACE complete (in combined complexity)
  - Satisfiability is undecidable

Role of Logic for/in Computer Science

The Burden of Logic in the 19-20th Century

- Role of logic as a foundation for all of mathematics
- Literature hint: Logicomix
  - fantastic graphic novel
  - Narrator: Philosopher and logician B. Russell
  - About the illusions, disillusions, and landmarking results at the end of the 19th century



Foundations of Mathematics with Mathematical Logic

- Attempts to find formal foundation for mathematical logic
- Hilberts Program (1900-1928)
  - Mathematics is consistent
  - Mathematics is (semantically) complete
  - Mathematics is decidable

Awakening

- Young Gödel proves (1931-33)
  - arithmetics not complete
  - consistency of set theory not provable
- Church/Turing (1936/37)
  - First-order logic is not decidable
  - Valid sentences not recursive
  - Sentences true in arithmetic not recursively enumerable (semi-decidable)
- Nonetheless there are the following positive insights
  - Syntactically completeness for FOL (Gödel, 1930)
  - ZFC (Zermelo-Fraenkel Set Theory) can be used to formalize all contemporary mathematics

The Unusual Effectiveness of Logic

- Logic (Research) and Computer Science had fruitful effects onto each other
- Logic even more w.r.t. CS (than w.r.t. mathematics)
- "Logic is the calculus of CS"

Lit: M. Y. Vardi. From philosophical to industrial logics. In Proceedings of the 3rd Indian Conference on Logic and Its Applications, ICLA'09, pages 89–115, Berlin, Heidelberg, 2009. Springer-Verlag.

Lit: J. Y. Halpern, R. Harper, N. Immerman, P. G. Kolaitis, M. Y. Vardi, and V. Vianu. On the unusual effectiveness of logic in computer science. Bull. Symbolic Logic, 7(2):213–236, 2001.

Why is this the Case?

- Logic is so general that it allows to
  - talk precisely about the objects within a computer/computation model
  - specify and reason about the properties of runs in the model

- Even more: One can characterize complexity classes with logics (Descriptive Complexity)

So ...

As an upcoming computer scientist (in academia or industry) you should train in formal models, in particular **logics**, because:

- you want to apply successfully for a job
- But more importantly: you want to keep your job

Computer Science Areas Effected by Logic Research

- Database Systems
- Ontology-Based Information Systems
- Semantic Integration
- Computer-Aided Verification (Model Checking)
- Computational Complexity
- High-Level Stream Processing
- Multi-Agent Systems
- Machine Learning (e.g. probabilistic graph models and logics)
- Semantic Web
- Logic Programming
- Knowledge Representation
- Semantics of Programms
- Digital Design ...

This course

Other courses of module "Web and Data Science" (CS4513)

This semester: "Web-Mining-Agenten"

Effects of Computer Science to Logic Research

- Focus/Intensive research on finite structures
  - Objects of computation are finite (Finite Model Theory)
  - But: potentially infinite structures (such as infinite DBs or streams) are useful as well
- Need for extensions of FOL
  - Higher-order logics (quantification over sets/relations)
  - Recursion (Datalog)
- Feasibility of reasoning services $\Longrightarrow$ restrictions of FOL
  - Modal and temporal logics
  - Description Logics
- Connections of logic and automata models
  - Regular expressions, finite automata, sequential logics
  - Buechi automata
- Logic engineering
- Different forms of inference ...

Overview of Course With Examples

Example: Logic in DB Research (Lectures 3-4)

- Travel DB with direct connection flights
- Reachability query
- SQL allows for recursion (CONNECT key word)
- But is it really necessary?

Table Flight

Start	End
Hamburg	Berlin
Hamburg	New York
New York	Berlin
...	...

Query Q_{reach}: List all cities reachable from Hamburg!

Intuitively without recursion:

$$
\begin{aligned}
Q_{reach}(x) \quad = \quad & Flight(Hamburg, x) \lor \\
& \exists x_1 Flight(Hamburg, x_1) \land Flight(x_1, x) \lor \\
& \exists x_1, x_2 Flight(Hamburg, x_2) \land Flight(x_2, x_1) \land Flight(x_1, x) \lor \\
\ldots &
\end{aligned}
$$

Example: Logic In DB Research

- Finite Model Theory (FMT) gives a proof for the impossibility to use FOL for recursive queries
- FMT models DBs as finite relational FOL structures

Example

- Flight table becomes structure
  $\mathfrak{A} = (D, \mathit{Flight}^{\mathfrak{A}}, \mathit{Hamburg}^{\mathfrak{A}}, \mathit{Berlin}^{\mathfrak{A}}, \dots)$
- Domain D: all constants in DB
- Constants named by themselves, e.g., $\mathit{Hamburg}^{\mathfrak{A}} = \mathit{Hamburg}$
- $\mathit{Flight}^{\mathfrak{A}} = \{(\mathit{Hamburg}, \mathit{Berlin}), (\mathit{Hamburg}, \mathit{NewYork}), \dots\}$

Example: Logic In DB Research

- Investigate all relevant reasoning problems w.r.t. finite models
  - Many properties for classical FOL do not hold
  - Also w.r.t. complexity

    $\Longrightarrow$ Calls for new techniques
- In particular: Investigate properties that all FOL queries have.

Theorem

All FOL formulas are local.
(Holds even for FOL extended with aggregation)

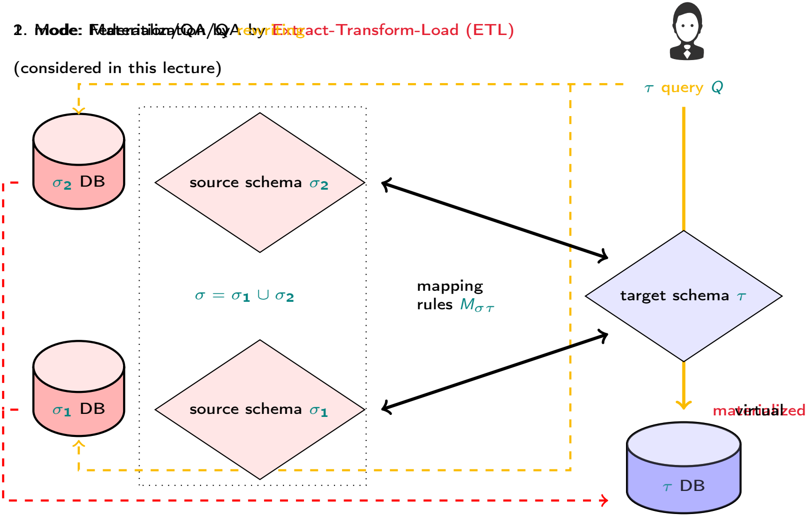
- Recursive queries are not local!

Data Integration and Data Exchange (lectures 5-7)

1. Mode: Federation/QA by rewriting

2. Mode: Materialization/QA by Extract-Transform-Load (ETL)

(considered in this lecture)



Example: Querying via Ontologies (Lectures 8-9)

- Ontologies
  - Formal means to represent and reason over data
  - Specify constraints/completeness rules
  - May have many models (open world assumption)
  - May be used for access of heterogeneous data sources

- Appropriate ontology languages: Description Logics (OWL and variants)
  - Constants, concepts (unary predicates), roles (binary predicates)
  - Terminological axioms, e.g., $\mathit{Students} \sqsubseteq \mathit{Humans}$
  - Assertions axioms, e.g., $\mathit{Student}(\mathit{Frege})$
  - Description logics are feasible fragments of FOL

Example

- No university known for *Goedel*
- Completeness: $Student \sqsubseteq \exists hasUniv.University$
- Functionality constraint: $(func\ hasUniv)$

Table university

Student	*Univ*
Frege	$U - Jena$
Russell	$U - London$
Goedel	*NULL*
...	...

Example: Ontology Integration (Lectures 10-11)

- There exist many ontologies out there
- For some applications need to integrate ontologies
- Problem: Joining ontologies may lead to incoherences/inconsistencies

Example

Ontology A

- $Article \equiv \exists publ.Journal$
- $Journal \sqsubseteq \neg Proceedings$
- $(func\ publ)$

Ontology B

- $Article \equiv \exists publ.Journal \sqcup Proceedings$
- $publish(ab, procXY)$

- $O_A \cup O_B$ is inconsistent
- How to repair this?

Belief Revision

- Belief Revision deals with operators for revising theories under possible inconsistencies
- Investigates concrete revision operators
- Principles that these must fulfill (minimality etc.)
- Representation theorems

- Recent research how to adapt these for non-classical logics/ontologies

Database repairs (Lecture 12)

- Databases may become inconsistent (falsify some integrity constraint)
  - Lack of support for ICs
  - due to integration ...

- Database repair
  - repair (only) virtually
  - Allow query answering over inconsistent DB
  - Formalize repair notion (minimality of repair), complexity issues

Streams (Lecture 13)

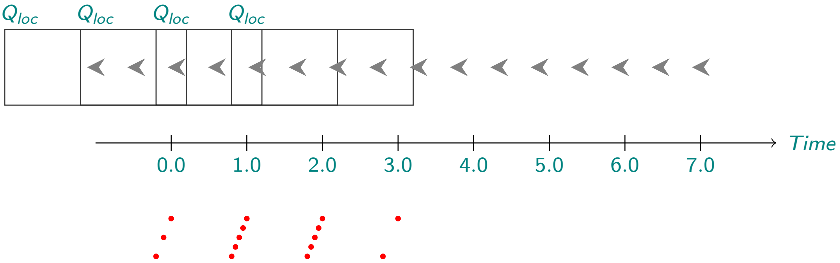
- "It's a streaming world" (Ubiquity)
  - Many data are temporal (sensor, event data)
  - Big data is mostly temporal data
- "Streams are forever" (Potential Infinity)
  - Streams are potentially infinite
  - One has to tame the infinite
  - Streams call for continuous querying (monitoring)
- "Order Matters" (Sequentiality)
  - Stream elements have an arriving order next to temporal order
  - Re-ordering or special sequencing may be needed

Lit: E. Della Valle. et al. It's a streaming world! Reasoning upon rapidly changing information. Intelligent Systems, IEEE, 24(6):83–89, nov.-dec. 2009.

Lit: J. Endrullis, D. Hendriks, and J. W. Klop. Streams are forever. Bulletin of the EATCS, 109:70–106, 2013.

Lit: E. D. Valle et al. Order matters! Harnessing a world of orderings for reasoning over massive data. Semantic Web, 4(2):219–231, 2013.

Query With Sliding Window



- The role of windows
- Stringology
- Bounded-memory computing
- High-level stream processing