An Open Question in Dynamic Descriptive Complexity

Thomas Schwentick

Venice, March 2017

technische universität dortmund

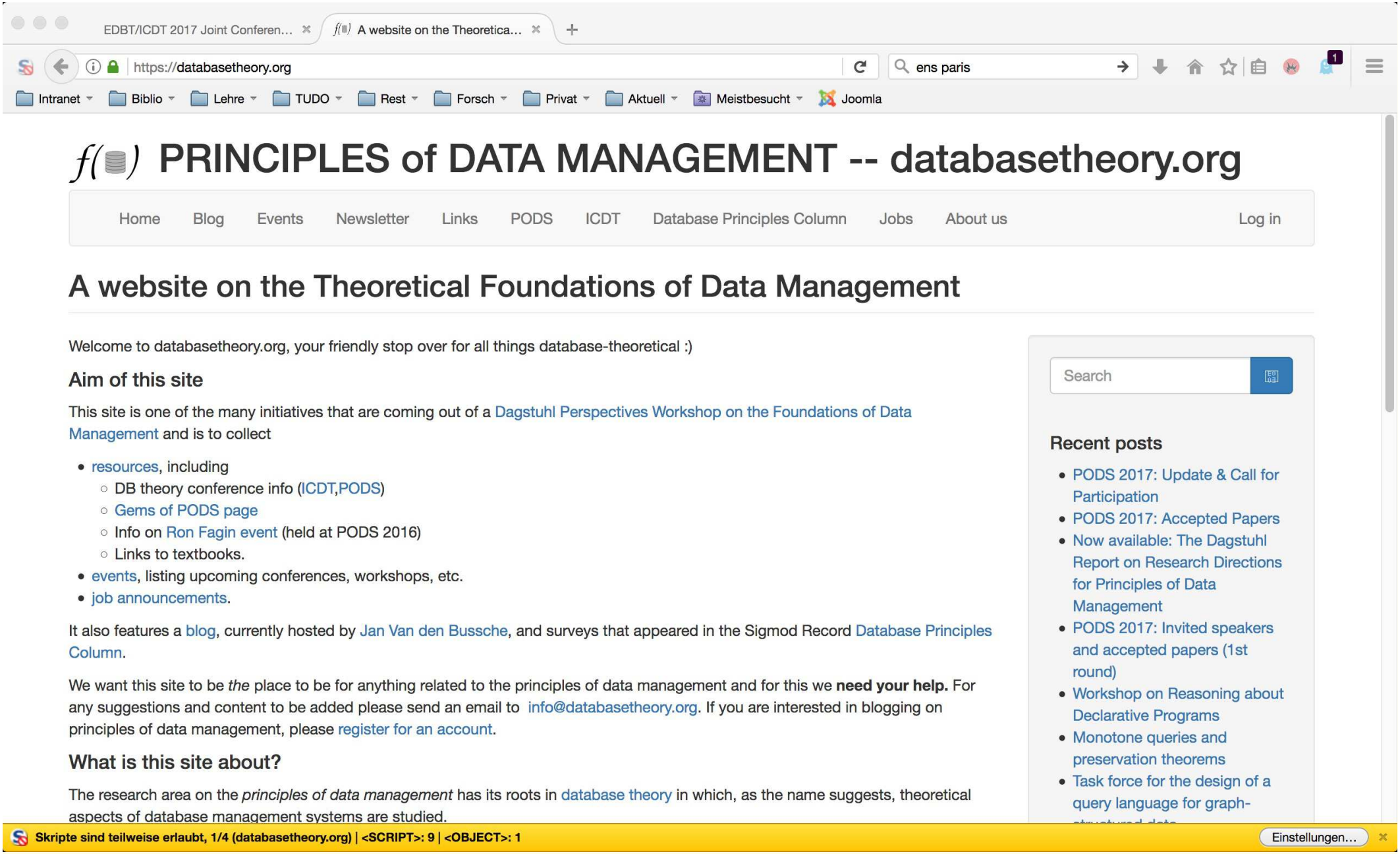
Lehrstuhl Logik in der Informatik

Short version of this talk

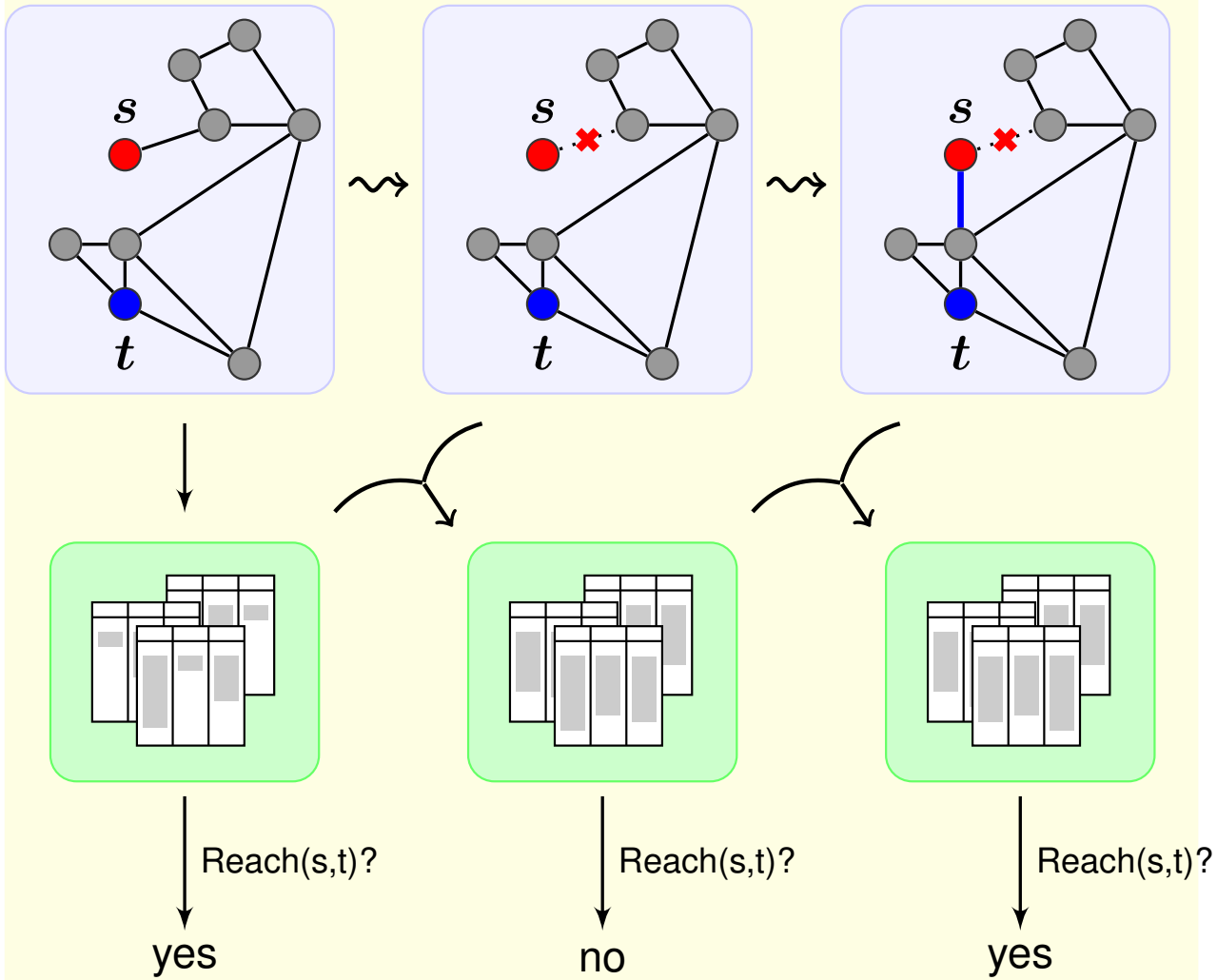
Open Question

- Does REACHABILITY $\in$ **DynProp** hold?

Commercial


EDBT/ICDT 2017 Joint Conferen…
A website on the Theoretica…
https://databasetheory.org
ens paris
Intranet
Biblio
Lehre
TUDO
Rest
Forsch
Privat
Aktuell
Meistbesucht
Joomla
PRINCIPLES of DATA MANAGEMENT -- databasetheory.org
Home
Blog
Events
Newsletter
Links
PODS
ICDT
Database Principles Column
Jobs
About us
Log in
A website on the Theoretical Foundations of Data Management
Welcome to databasetheory.org, your friendly stop over for all things database-theoretical :)
Aim of this site
This site is one of the many initiatives that are coming out of a Dagstuhl Perspectives Workshop on the Foundations of Data Management and is to collect
resources, including
DB theory conference info (ICDT,PODS)
Gems of PODS page
Info on Ron Fagin event (held at PODS 2016)
Links to textbooks.
events, listing upcoming conferences, workshops, etc.
job announcements.
It also features a blog, currently hosted by Jan Van den Bussche, and surveys that appeared in the Sigmod Record Database Principles Column.
We want this site to be the place to be for anything related to the principles of data management and for this we need your help. For any suggestions and content to be added please send an email to info@databasetheory.org. If you are interested in blogging on principles of data management, please register for an account.
What is this site about?
The research area on the principles of data management has its roots in database theory in which, as the name suggests, theoretical aspects of database management systems are studied.
Search
Recent posts
PODS 2017: Update & Call for Participation
PODS 2017: Accepted Papers
Now available: The Dagstuhl Report on Research Directions for Principles of Data Management
PODS 2017: Invited speakers and accepted papers (1st round)
Workshop on Reasoning about Declarative Programs
Monotone queries and preservation theorems
Task force for the design of a query language for graph-
Skripte sind teilweise erlaubt, 1/4 (databasetheory.org) | <SCRIPT>: 9 | <OBJECT>: 1
Einstellungen…

Dynamic Complexity: The Setting



- Auxiliary data can speed up query answering after change
  → Dynamic Algorithms

- Rather than dynamic algorithms, we study weak query languages as update mechanism in a dynamic setting:
  - Relational calculus (FO-logic)

  ☞ = first-order logic

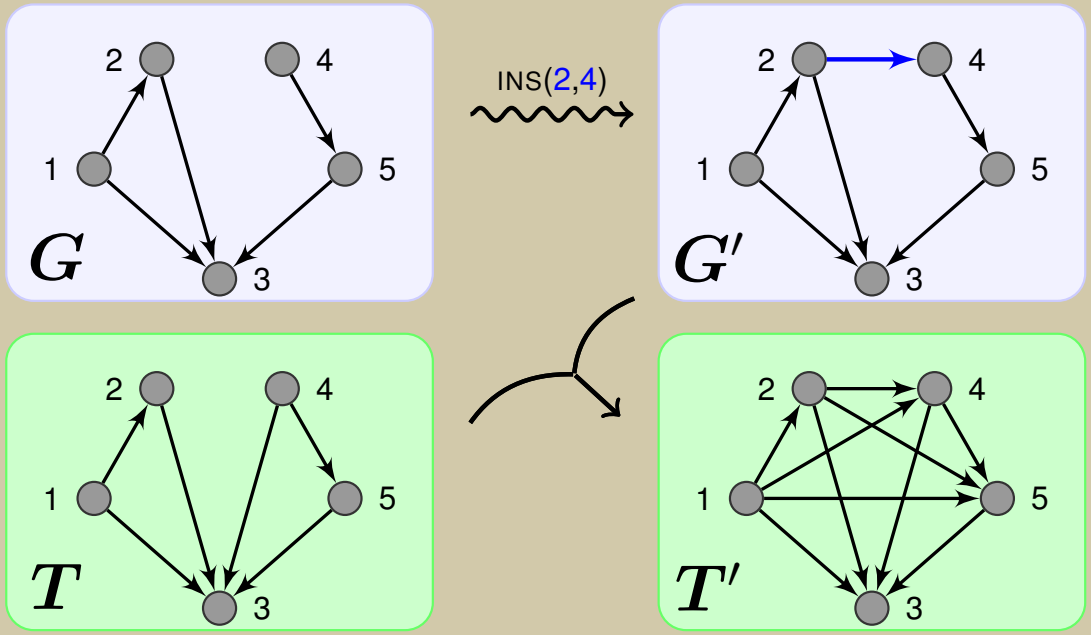
- Auxiliary data might enable FO-logic to **maintain** queries that it can not **express**

Definition

- **DynFO** $\stackrel{\text{def}}{=}$
  - queries that can be maintained by first-order logic with auxiliary relations under deletion and insertion of edges

Example

Example: Reachability under Insertion



- **Obvious idea:** store the transitive closure of the edge relation in a binary auxiliary relation T
- Update rule:
  on insert (u, v) **into** E
  update $T(x, y)$ **as** $T(x, y) \vee \big(T(x, u) \wedge T(v, y)\big)$
  - determines the pairs (x, y) in T after insertion of (u, v) to E

Motivation & Goals

- **Why DynFO?**
  - captures essentially what can be maintained in a relational database (core SQL)
  - meaningful from a complexity theoretic point of view:
    - **FO**$(+, \times) \equiv$ uniform **AC**$^{\mathbf{0}}$
      $\equiv$ circuit families of bounded depth and poly size
  - the most natural logic

- **General goal of our research:**
  - Understand the expressive power of **DynFO**

  - Which queries are in **DynFO**?
  - Which queries are **not** in **DynFO**?

Expressibility

- In recent years we learned a lot about what *can* be done in **DynFO**

- Reachability is in **DynFO** ☞ ICALP 2015
- All MSO-queries on graphs of bounded tree-width are in **DynFO** ☞ submitted

- Undirected Reachability under **FO**-defined insertions is in **DynFO** ☞ this ICDT
- $\mathbf{AC}^1$-computable queries under parameter-free definable changes are in **DynFO** ☞ this ICDT

- Bottom line: **DynFO** is much more powerful than we thought

- Expressibility results are nice (and maybe even useful)
- But one could argue that they just constitute a collection of tricks

- We do not understand the "real nature" of Dynamic Descriptive Complexity before we are able to prove inexpressibility results

Inexpressibility Results: A Sad State

- Easy observation: $q \in$ **DynFO** $\Rightarrow q \in$ **PTIME**
  - Just insert the tuples of $\mathcal{D}$ into an empty database one by one, and compute all updates
- So far there are no other general lower bound results for **DynFO**
- Most existing lower bounds apply to
  - auxiliary relations of bounded arity or
  - restricted logics or
  - both...
- k-ary **DynFO**: Update programs with at most k-ary auxiliary relations (plus the query relation)

Theorem [Dong, Su 95/98]

- REACHABILITY $\notin$ unary **DynFO**
- For every $k > 0$, k-ary **DynFO** is strictly weaker than $(k+1)$-ary **DynFO**

☞ on structures of "growing" arity

Dynamic programs with quantifier-free formulas

- Hesse initiated the study of dynamic programs with quantifier-free update formulas [Hesse 03]

Definition

- **DynProp:**
  - Queries that can be maintained in **DynFO** with quantifier-free formulas and aux **relations**
- **DynQF:**
  - Queries that can be maintained in **DynFO** with quantifier-free formulas and aux **functions** (and relations)

✎ **DynQF** formulas can use "if-then-else"-terms

- Quantifier-free update formulas? Isn't that extremely weak?

Theorem [Hesse 03]

- DET-REACH $\in$ **DynProp**
  (no quantifiers, aux relations)

Theorem [Hesse 03]

- SYM-REACH $\in$ unary **DynQF**
  (no quantifiers, unary aux functions & relations)

Theorem [Gelade, Marquardt, TS 09/12]

- **DynProp** $\subsetneq$ **DynQF**

Inexpressibility Results for Quantifier-free Fragements

Theorem [Gelade, Marquardt, TS 09/12]

- Alternating Reachability $\notin$ **DynProp**

Theorem [Gelade, Marquardt, TS 09/12]

- **FO** $\not\subseteq$ **DynProp**

Theorem [Zeume, TS 13]

- Unary, binary, and ternary **DynProp** form a strict hierarchy on graphs

Theorem [Zeume 14]

- The arity hierarchy of **DynProp** is strict on graphs under insertions

- Actually, it is also strict under insertions and deletions ☞ Vortmeier, Zeume, unpublished

Theorem [Zeume, TS 13]

- REACHABILITY $\notin$ binary **DynProp**
- REACHABILITY $\notin$ unary **DynQF**

Open Questions

- Does REACHABILITY $\in$ **DynProp** hold?
- Does REACHABILITY $\in$ **DynQF** hold?

References

- Sushant Patnaik and Neil Immerman. Dyn-FO: A parallel, dynamic complexity class. In *PODS 1994*, pages 210–221, 1994
- Sushant Patnaik and Neil Immerman. Dyn-FO: A parallel, dynamic complexity class. *J. Comput. Syst. Sci.*, 55(2):199–209, 1997
- Guozhu Dong and Jianwen Su. Arity bounds in first-order incremental evaluation and definition of polynomial time database queries. *J. Comput. Syst. Sci.*, 57(3):289–308, 1998
- Wouter Gelade, Marcel Marquardt, and Thomas Schwentick. The dynamic complexity of formal languages. *ACM Trans. Comput. Log.*, 13(3):19, 2012
- Thomas Zeume and Thomas Schwentick. On the quantifier-free dynamic complexity of reachability. In *Mathematical Foundations of Computer Science 2013*, pages 837–848, 2013
- Thomas Zeume. The dynamic descriptive complexity of k-clique. In *MFCS 2014*, 2014
- Samir Datta, Raghav Kulkarni, Anish Mukherjee, Thomas Schwentick, and Thomas Zeume. Reachability is in dynfo. In *International Colloquium on Automata, Languages, and Programming, ICALP 2015, Proceedings, Part II*, pages 159–170, 2015
- Thomas Schwentick, Nils Vortmeier, and Thomas Zeume. Dynamic complexity under definable changes. In *20th International Conference on Database Theory, ICDT 2017*, pages 19:1–19:18, 2017