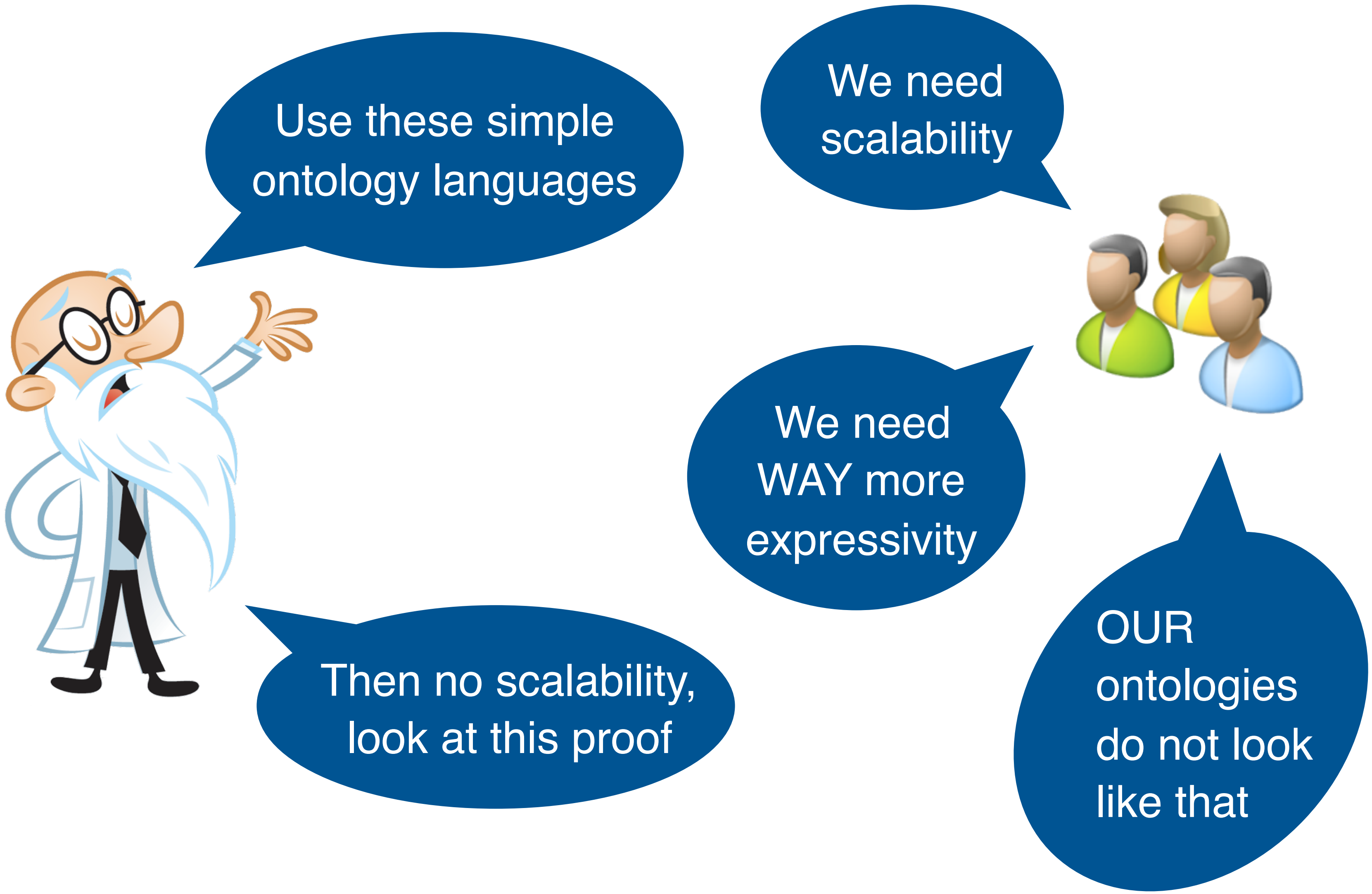
(More on)
Islands of Tractability in Ontology-Based Data Access

Carsten Lutz, University of Bremen

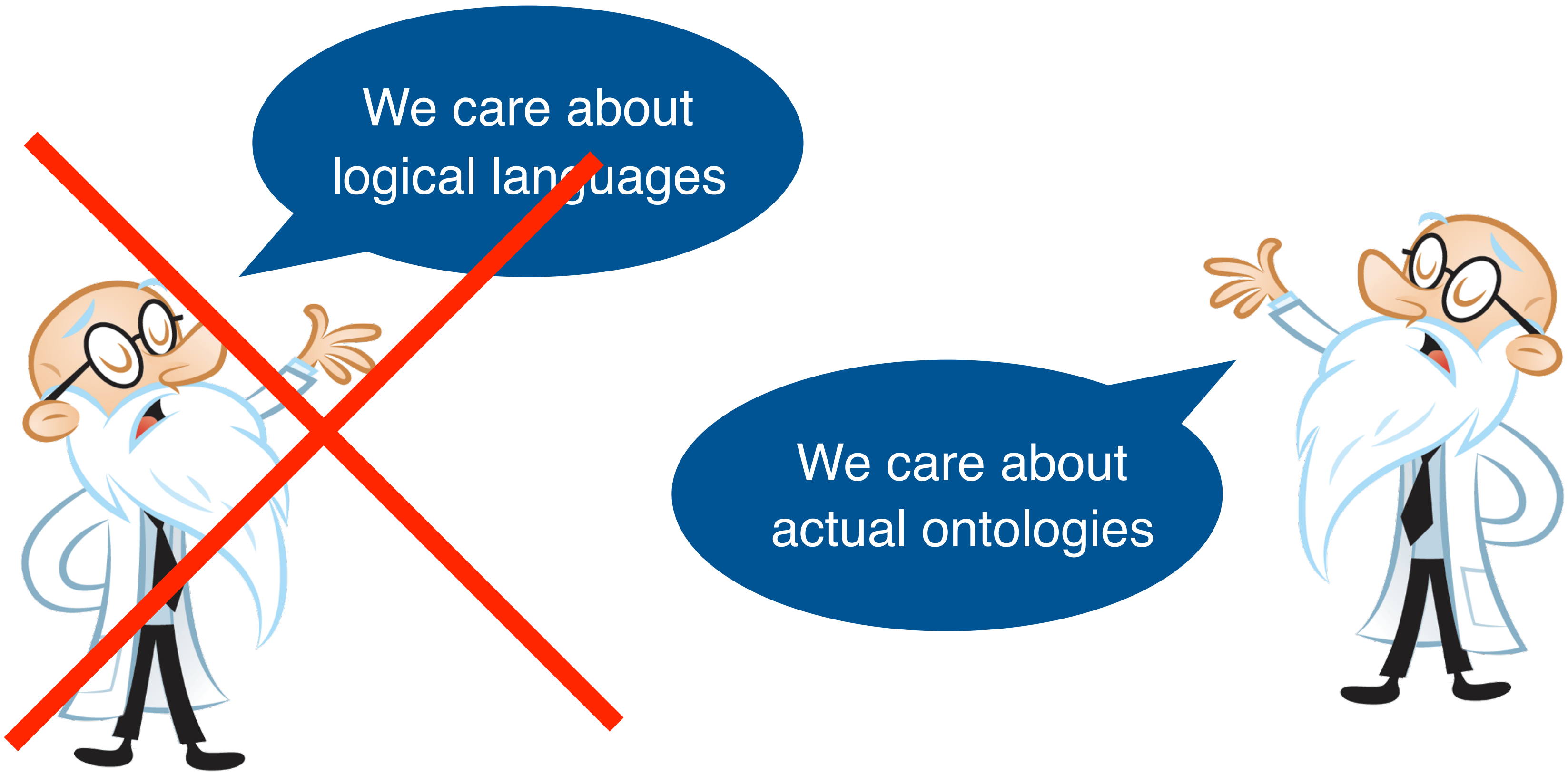
Scientists vs. Users



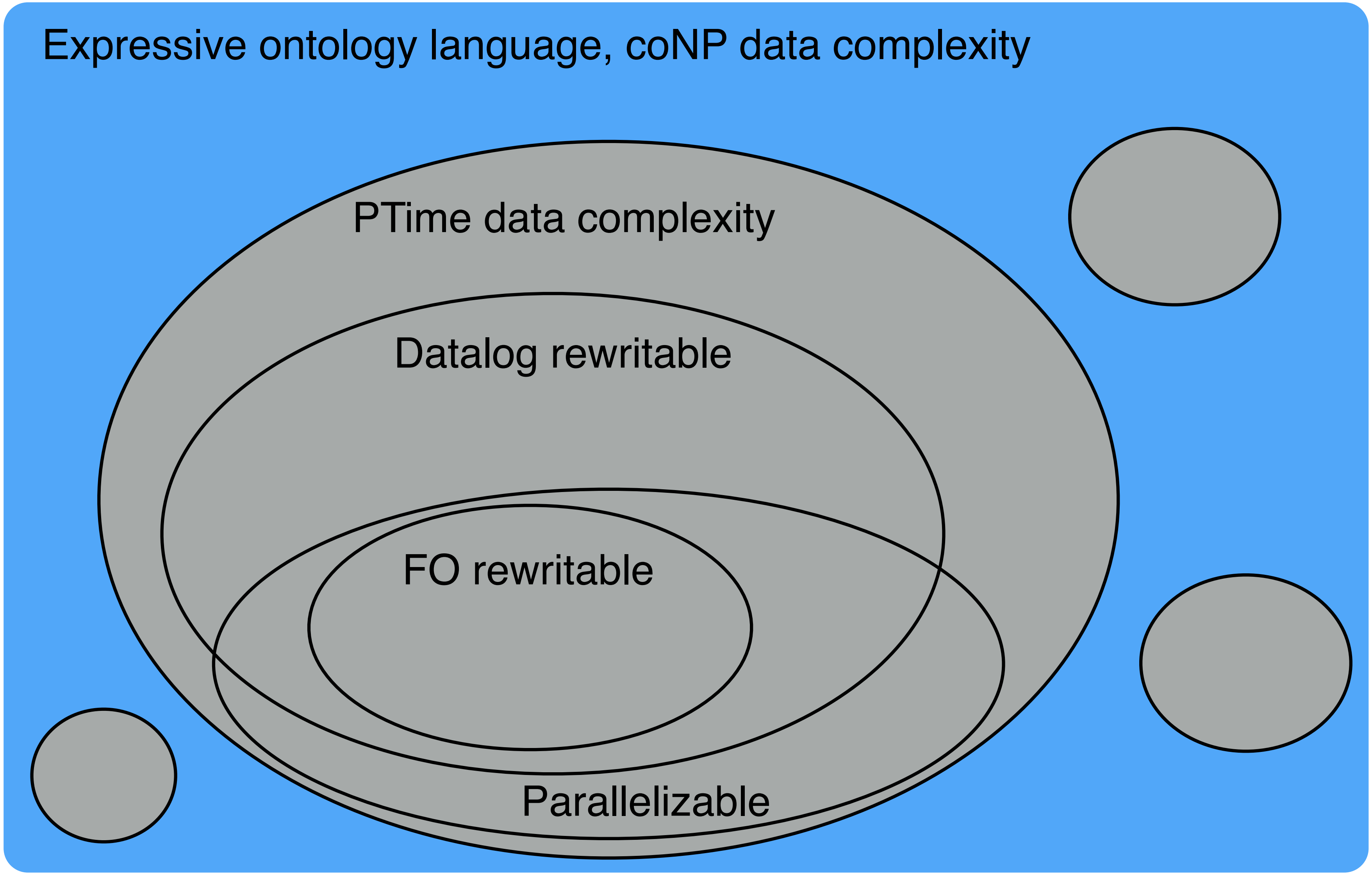
Scientists vs. Users

Observations:

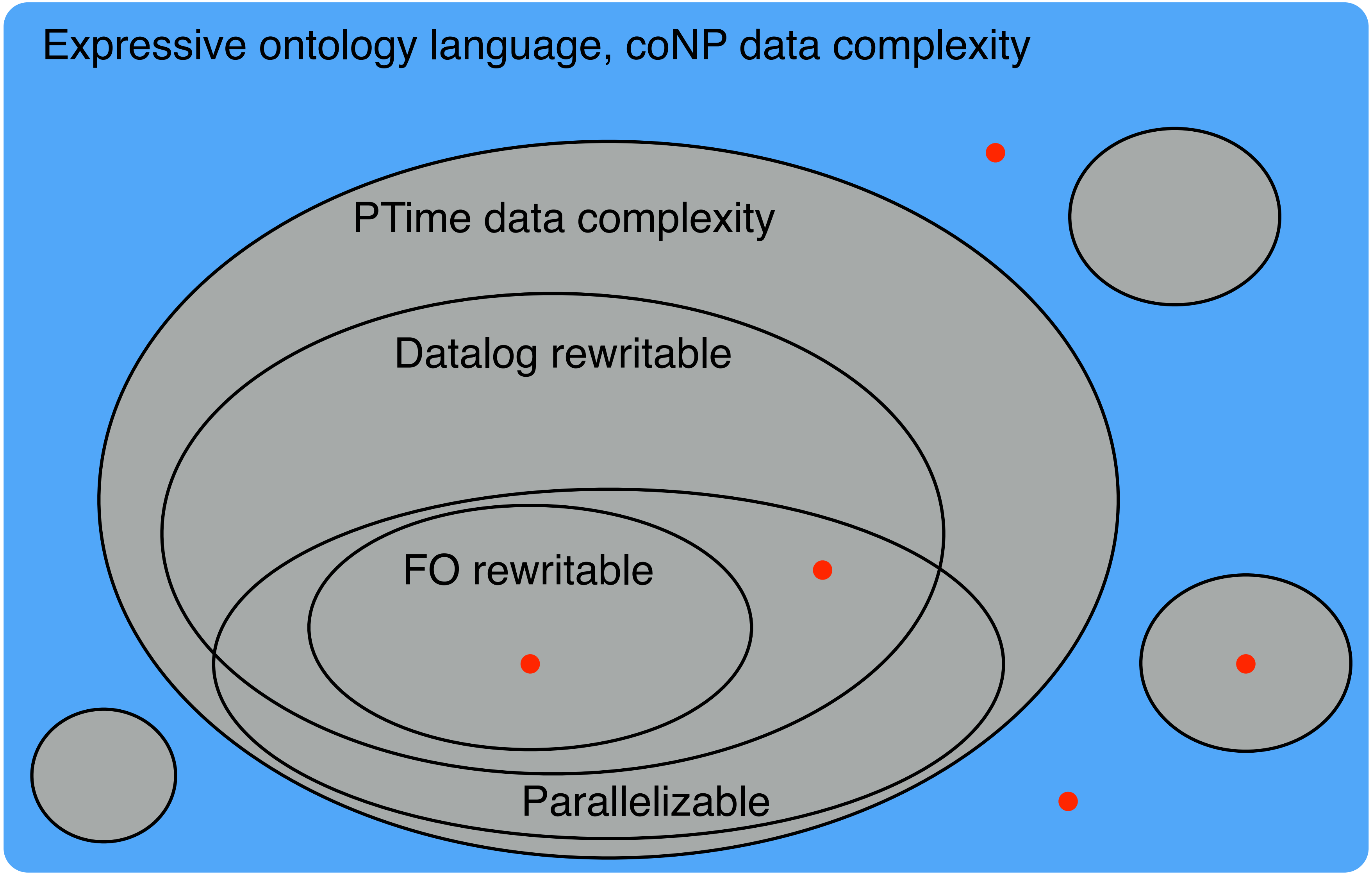
- users insist on using expressive languages with many features
- concrete ontologies from applications tend to have simple structure



Islands of Tractability



Islands of Tractability



Basic Setup

Ontology-mediated query (OMQ): triple $(\mathcal{T}, \Sigma, q)$ where

- $\mathcal{T}$ is TBox (= ontology)
- Σ is schema for data (subset of schema of $\mathcal{T}$)
- q is query, e.g. atomic query (AQ) / conjunctive query (CQ) / UCQ

takes form $A(x)$ $\approx$ tree-shaped CQ

OMQ language:

pair $(\mathcal{L}, \mathcal{Q})$ with $\mathcal{L}$ DL (TBox language) and $\mathcal{Q}$ query language

for example $(\mathcal{EL}, \text{AQ})$, $(\mathcal{ALC}, \text{UCQ})$, etc.

Part I: Horn DLs

Horn DLs

Horn-DLs fit into the Horn fragment of FO / admit a chase procedure

Two basic Horn DLs: $\mathcal{EL}$ and $\mathcal{ELI}$ (underly OWL2 EL profile)

Concept formation rule:

$$C, D ::= A \mid \top \mid C \sqcap D \mid \exists r.C \mid \exists r^-.C$$

$\exists y\, r(y, x) \wedge C(y)$
(only $\mathcal{ELI}$)

monadic relation
(concept name)

true

should be $\wedge$

$\exists y\, r(x, y) \wedge C(y)$

TBoxes: finite sets of inclusions $C \sqsubseteq D$

Example:

$$\exists\text{manages.Project} \sqsubseteq \text{ProjectManager}$$
$$\text{ProjectManager} \sqsubseteq \exists\text{assistedBy.PersonalAssistant}$$

This is roughly: Datalog with arity <=2 and tree-shaped rule bodies
plus existential quantification in rule heads

Horn DLs, FO, Datalog

OMQs in Horn DLs can be rewritten into monadic datalog program
(though with exponential blowup)

Exploited in practice: systems such as Clipper, Rapid, Requiem

Most interesting island of tractability is FO-rewritability

In Datalog, FO-rewritability coincides with boundedness

Theorem [BenediktTenCateColcombetVandenBoomLICS15]

Monadic datalog boundedness is 2ExpTime-complete
(assuming an unpublished result on cost automata).

We thus obtain only a 3ExpTime upper bound, no practical algorithms

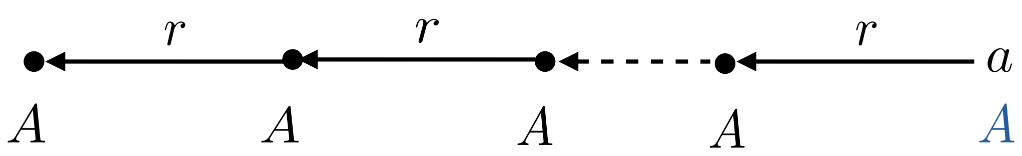
CHECK: 2ExpTime because of bounded arity?

FO-rewritability

Paradigmatic OMQ in $(\mathcal{EL}, \text{AQ})$ that is not FO-rewritable:

TBox: $\exists r.A \sqsubseteq A$ Query: $A(x)$

ABox:



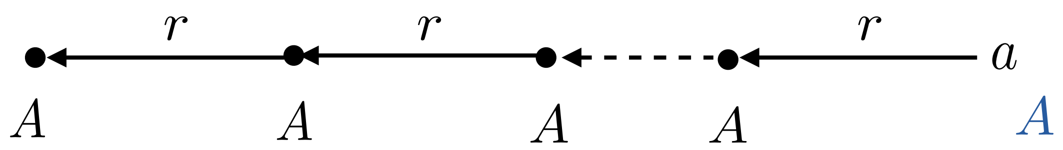
answer $A(a)$

Non-locality comes from cycles via existentials on the left-hand side.

So non-FO-rewritability = existence of certain syntactic cycles?

FO-rewritability

TBox: $\exists r.A \sqsubseteq A\,,\, \exists r.\top \sqsubseteq A$ Query: $A(x)$



FO-rewriting exists since $\exists r.\top \sqsubseteq A$ cancels non-locality:

$$A(x) \vee \exists y\, r(x, y)$$

Cancelation is main source of complexity:

- finding cycles in TBox is trivial (pure syntax)
- cycle cancelations can still occur after exponentially many steps

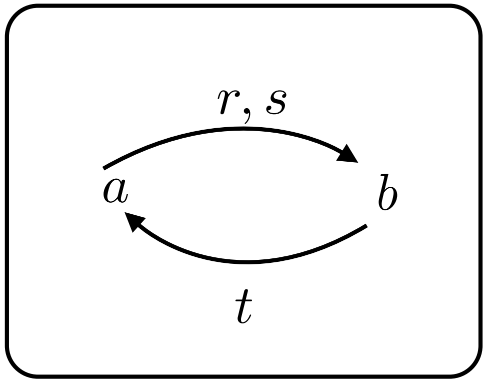
  On these steps, one can simulate a Turing machine

Unraveling Tolerance

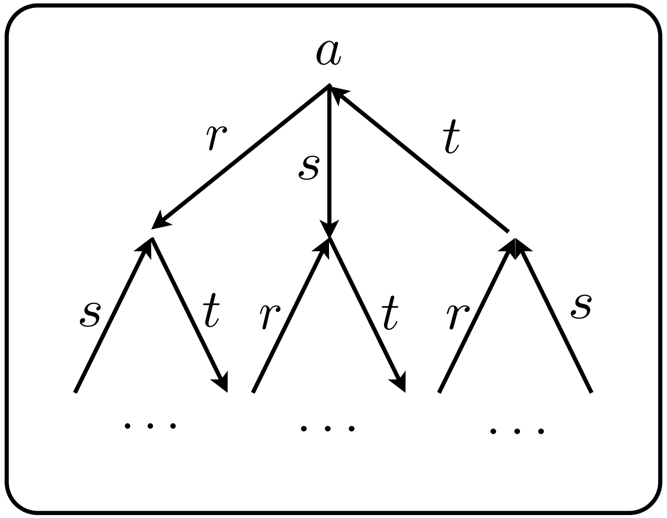
OMQ $(\mathcal{T}, \Sigma, A(x))$ is unraveling tolerant if for every Σ-ABox $\mathcal{A}$:

$$\mathcal{A}, \mathcal{T} \models A[a] \text{ iff } \mathcal{A}_a^u, \mathcal{T} \models A[a]$$

$\mathcal{A}$



$\mathcal{A}_a^u$



Theorem [L__WolterKR12]

Every OMQ from $(\mathcal{ELI}, \text{AQ})$ is *unraveling tolerant*.

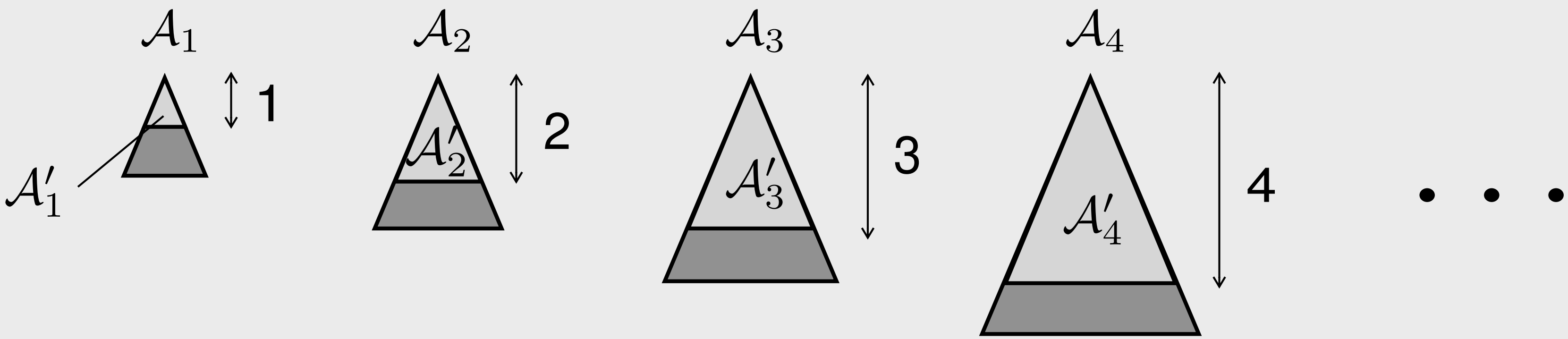
Characterizing Non-Rewritability

Unraveling tolerance enables characterization of FO-rewritability in Horn DLs.

Theorem [BienvenuL_WolterIJCAI13]

OMQ $(\mathcal{T}, \Sigma, A(x))$ in $(\mathcal{ELI}, \text{AQ})$ is not FO-rewritable iff

there are Σ-ABoxes



such that for all $i \geq 1$: $\mathcal{T}, \mathcal{A}_i \models A(a_0)$, but $\mathcal{T}, \mathcal{A}'_i \not\models A(a_0)$

Complexity

Via a pumping argument, we can bound the depth of the ABoxes to look at

Worst case optimal algorithms for deciding FO-rewritability can then be found via automata techniques

Theorem [BienvenuL_WolterIJCAI13]

Deciding FO-rewritability is

- PSPACE-complete in $(\mathcal{EL}, \text{AQ})$ with full ABox signature
- EXPTIME-complete in $(\mathcal{EL}, \text{AQ})$ with unrestricted ABox signature
- EXPTIME-complete in $(\mathcal{ELI}, \text{AQ})$
  (with full and unrestricted ABox signature)

Does not suggest practical approach to construct rewritings

Constructing FO-Rewritings: Preliminary

Theorem [RossmanJACM08]

If an FO-query is preserved under homomorphisms on finite structures, then it is equivalent to a UCQ.

Most OMQs Q preserved under homomorphisms on ABoxes:

if $\mathcal{A}_1 \models Q[\vec{a}]$ and $h : \mathcal{A}_1 \to \mathcal{A}_2$ homomorphism, then $\mathcal{A}_2 \models Q[h(a)]$

Corollary

In (FO-without-equality, UCQ), every FO-rewritable OMQ has a UCQ-rewriting.

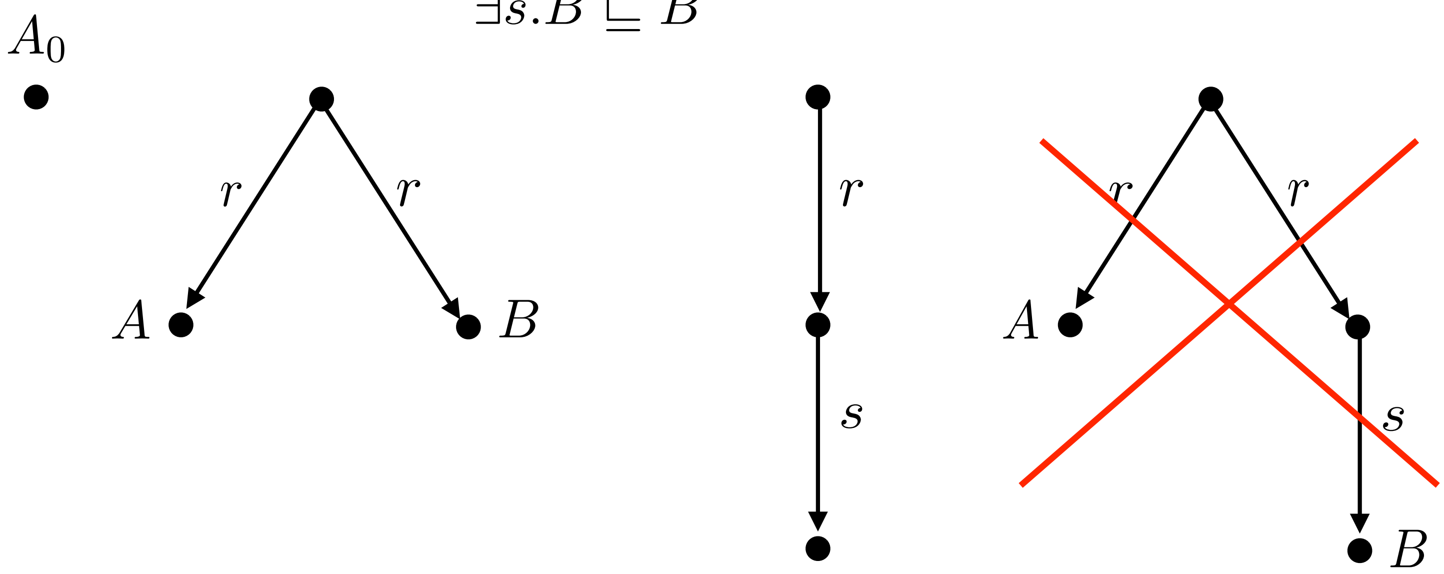
Constructing FO-Rewritings: Backwards Chaining

Proposed in [KönigLeclereMugnierThomazoRR12] for existential rules, here adapted to $(\mathcal{EL}, \text{AQ})$:

TBox:

$$\exists r.A \sqcap \exists r.B \sqsubseteq A_0$$
$$\exists r.\exists s.\top \sqsubseteq A_0$$
$$\exists s.B \sqsubseteq B$$

Query: $A_0(x)$



Termination for positive cases guaranteed, general termination achievable via tree characterization [HansenL_SeylanWolterIJCAI15]

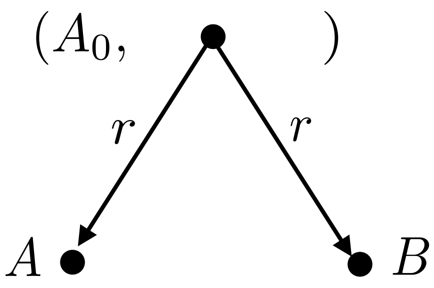
Problem: UCQ representation of rewriting quickly grows out of bounds

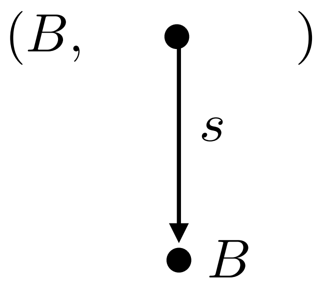
Constructing FO-Rewritings II

Backwards chaining can be realized in decomposed calculus so that

- structure sharing helps to avoid thrashing

TBox: $\exists r.A \sqcap \exists r.B \sqsubseteq A_0, \;\; \exists s.B \sqsubseteq B$ Query: $A_0(x)$





- a (succinct) non-recursive datalog rewriting is produced
- optimal ExpTime complexity is achieved

[HansenL_SeylanWolterIJCAI15]

Experiments

TBox	CI	CN	RN	no	stop	time	RQ stop	RQ time
ENVO	1942	1558	7	7	100%	2s	92.6%	2m52s
FBbi	567	517	1	0	100%	3s	86.1%	19m25s
MOHSE	3665	2203	71	1	99.6%	6m35s	58.7%	7h17m
NBO	1468	962	8	6	100%	3s	61.5%	3h05m
not-galen	4636	2748	159	44	95.9%	1h15m	48.9%	11h43m
SO	3160	2095	12	15	99.8%	4m28s	77.9%	3h53m
XP	1046	906	27	1	100%	27s	0.0%	7h33m

The actual rewritings are small (≤ 10 rules) in almost all cases

Confirms that almost all OMQs from practice fall within island!

CQs can be handled similarly, but complexity goes up (sometimes)

Part II: Non-Horn DLs

Expressive DLs

Two basic expressive DLs: $\mathcal{ALC}$ and $\mathcal{ALCI}$ (core of OWL2 DL profile)

Concept formation rule:

$$C, D ::= A \mid \top \mid \neg A \mid C \sqcap D \mid \exists r.C \mid \exists r^{-}.C$$ — (only $\mathcal{ALCI}$)

Standard first-order semantics of negation

Can also express:

disjunction $C \sqcup D$

universal restriction $\forall r.C$ $\qquad \forall y\,(r(x,y) \rightarrow C(y))$

and $\forall r^{-}.C$ $\qquad \forall y\,(r(y,x) \rightarrow C(y))$

This is roughly: traditional modal logic or

a slight restriction of the two-variable guarded fragment

Expressive DLs: Example

Schema for data: single binary relation r (data=graphs)

Ontology:

$$\top \sqsubseteq R \sqcup G \sqcup B$$

$$R \sqcap G \sqsubseteq D \qquad R \sqcap B \sqsubseteq D \qquad G \sqcap B \sqsubseteq D$$

$$R \sqcap \exists r.R \sqsubseteq D \quad G \sqcap \exists r.G \sqsubseteq D \quad B \sqcap \exists r.B \sqsubseteq D$$

Query:

$$q() = \exists x \, D(x)$$

Expresses non-3-colorability,

thus coNP-hard and provably not Datalog-rewritable [AfratiEtAl91]

Relevant islands of tractability include FO- and Datalog-rewritability

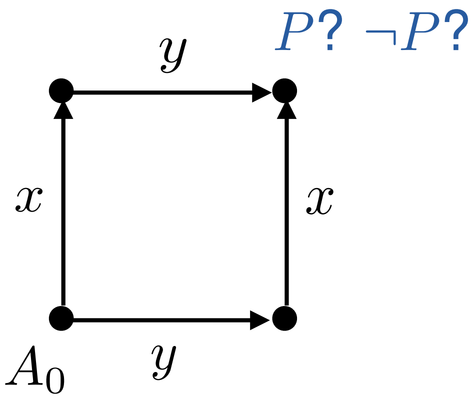
No Unraveling Tolerance

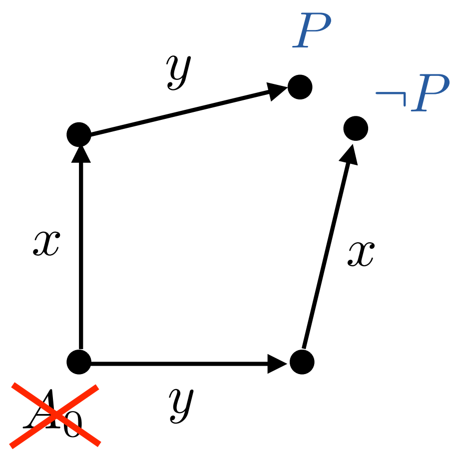
Non-Horn DLs are NOT unraveling tolerant:

TBox: $\exists x.\exists y.P \sqcap \exists y.\exists x.P \sqsubseteq A_0$

$\exists x.\exists y.\neg P \sqcap \exists y.\exists x.\neg P \sqsubseteq A_0$

Query: $A_0(x)$





Tree-based approaches not likely to be successful. What can we do?

Valuable resource: CSP-connection

OBDA and CSP

A template is a finite relational structure T. CSP(T) is:

Given: finite relational structure S

Question: $T \leftarrow S$?

We concentrate on binary CSPs: only unary and binary relations

BAQs: Boolean atomic queries $\exists x\, A(x)$

Theorem [BienvenuTenCateL_WolterPODS13]

Every OMQ from ($\mathcal{ALCI}$,BAQ) is equivalent to the complement of a CSP and vice versa.

More On Expressive Power

$(\mathcal{ALC}, \mathrm{BAQ})$	coCSP	Boolean MDDLog w. single EDB in rule body
$(\mathcal{ALC}, \mathrm{AQ})$	multi-template coCSP w. single constant	MDDLog w. single EDB in rule body
$(\mathcal{ALC}, \mathrm{UCQ})$	coMMSNP w free FO-variables	MDDLog
$(\mathrm{GF}, \mathrm{UCQ})$	coGMSNP w free FO-variables	Frontier-guarded disjunctive Datalog

1exp / 2exp

poly

poly

[BienvenuTenCateL_WolterPODS13]

On Complexity / Rewritings

Thus studying islands of tractability for OMQs and CSPs is equivalent

For example, $(\mathcal{ALC}, \text{AQ})$ has dichotomy between PTime and coNP iff the Feder-Vardi conjecture holds (a problem for algebraists, it seems)

Two caveats:

- For every CSP, there is a binary CSP of the same complexity, up to polytime reductions

  But classification below PTime not known to be equivalent!

- There are important OMQ languages such as $(\mathcal{ALCF}, \text{AQ})$ for which CSP connection breaks

  counting quantifiers

Theorem [L_WolterKR12]

$(\mathcal{ALCF}, \text{AQ})$ contains queries that are coNP-intermediate (unless P=NP)

Rewritings: Decidability

Theorem

1. FO-definability of coCSPs is NP-complete. [LaroseLotenTardiffLMCS07]
2. Datalog-definability of coCSPs is NP-complete. [BartoKozikFOCS09, KozikKrokhinValerioteWillardAU14]

Can be lifted to multi-template CSPs with single constant

Exponential blowup in translation OMQ => CSP “materializes”

Theorem [BienvenuTenCateL_WolterPODS13]

FO-rewritability and Datalog-rewritability in $(\mathcal{ALCI}, \text{BAQ})$ and $(\mathcal{ALCI}, \text{AQ})$ is NExpTime-complete.

Constructing Rewritings (in Theory)

FO-Rewritings:

- From CSP-connection and results on homomorphism dualities: if there is an FO-rewriting, then there is a tree-UCQ-rewriting
- Pumping argument: depth and outdegree of tree-CQs can be bounded double exponentially
- Enumerate all CQs of these dimensions, check whether they are rewriting (red. to query answering)

Datalog-Rewritings:

- If there is a rewriting, then there is one of width at most three [BartoKozikEtAl]
- Canonical width-3 Datalog program of Feder and Vardi is a rewriting iff there is one [SiamJComp98]

More practical / pragmatic approaches (even incomplete) needed!

Thank You!

This and related research carried out under ERC Consolidator Grant

CODA - Custom-Made Ontology Based Data Access

August 2015 - July 2020, University of Bremen