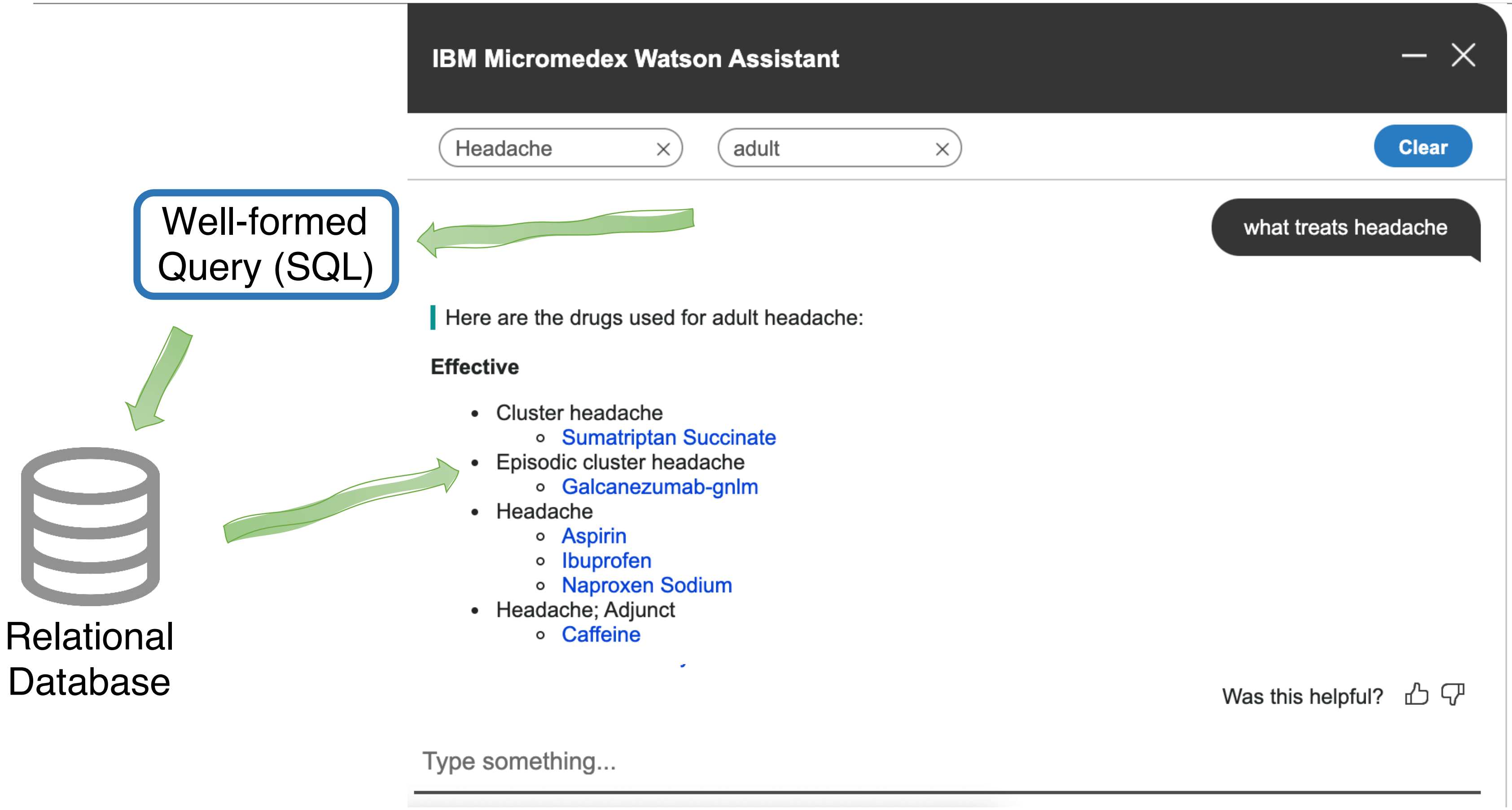
Ontology-Enriched Query Answering on Relational Databases

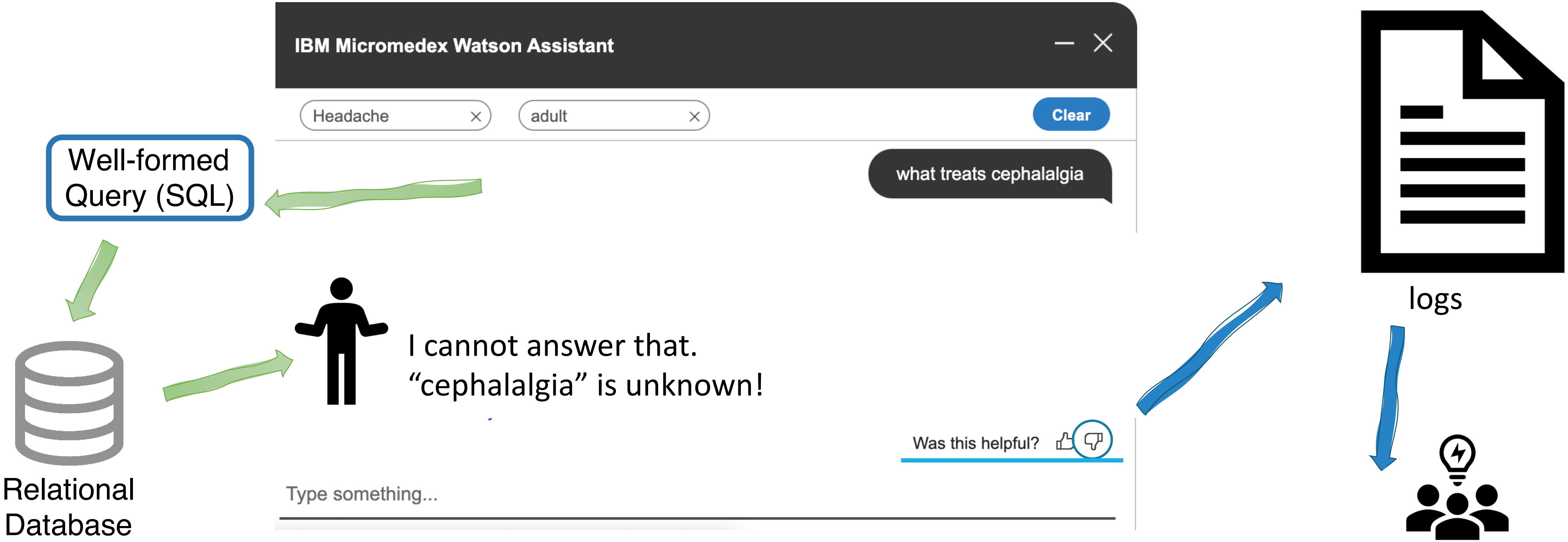
S. Ahmetaj, V. Efthymiou, R. Fagin, P.G. Kolaitis, C. Lei, F. Ozcan, L. Popa

How it started...



A. Quamar, C. Lei, D. Miller, F. Özcan, J. Kreulen, R. Moore, V. Efthymiou. An Ontology-Based Conversation System for Knowledge Bases. *SIGMOD 2020*

How it started…

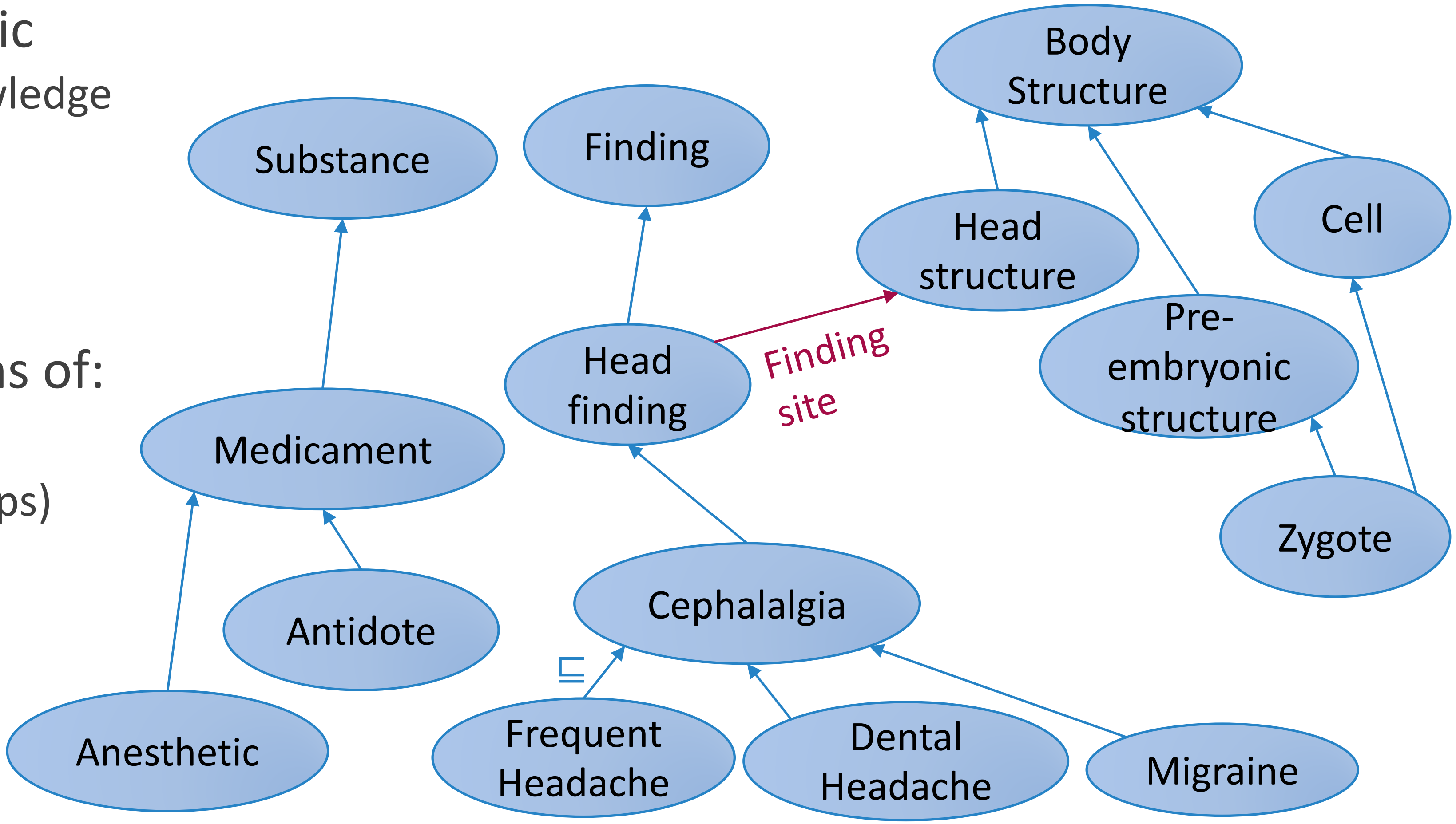


Many of the unknown terms are contained in SNOMED CT and our DB actually has the data to answer such queries

By using SNOMED, we could also provide more complete answers to some queries

DL Ontologies

- Based on Description Logic
  - a family of logic-based knowledge representation formalisms
  - decidable fragments of FOL
- Describe domains in terms of:
  - concepts (aka classes)
  - roles (aka binary relationships)

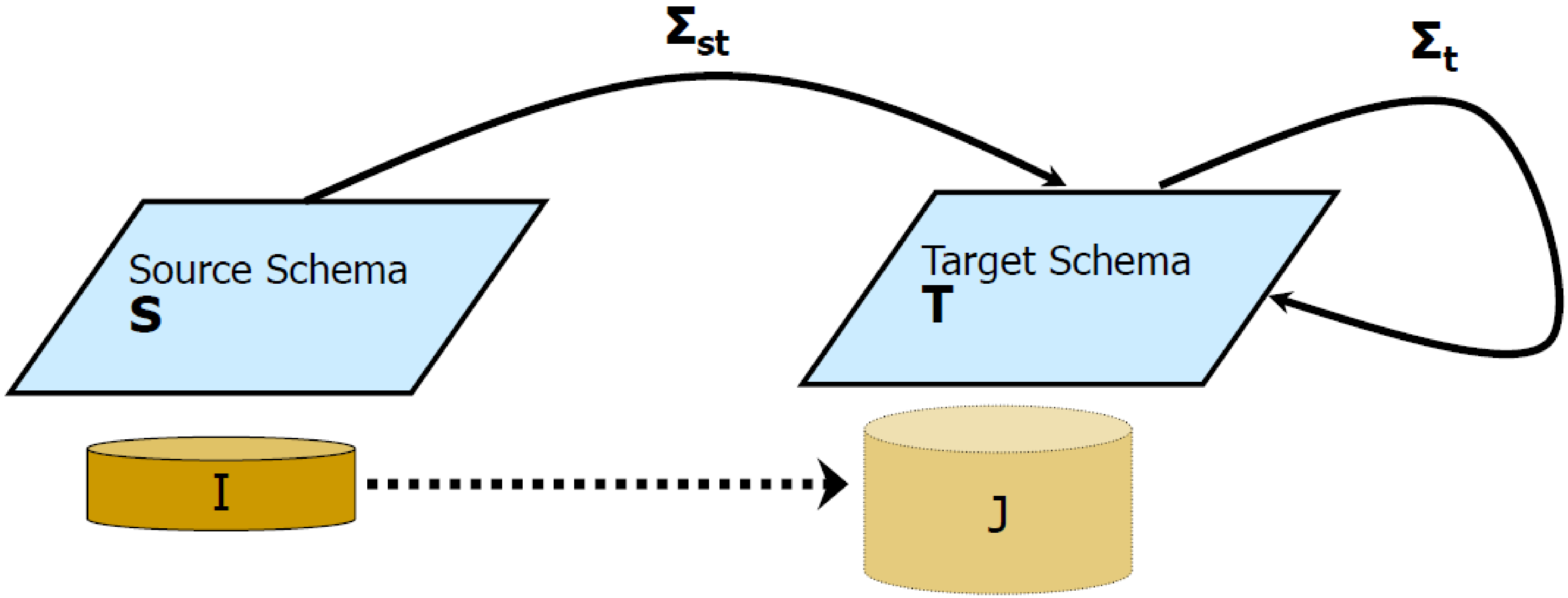


Main Challenges

- Identify and reuse only the parts of SNOMED CT that are relevant
  - we used existing tools from different AI communities
    - ontology creation from a DB, ontology matching, module extraction
  - we designed a flexible framework that goes beyond our use case

- Answer queries expressed over the vocabulary of SNOMED CT using our data
  - Two main approaches exist:
    - Materialization:
      - Materialize a universal solution (once) using the *chase* procedure from the data exchange community;
      - compute the certain answers on arbitrary conjunctive queries over the target schema using the materialized universal solution
    - Query Rewriting: Keep the original data, but rewrite every query when it comes before evaluating it

Background – Data Exchange



Data exchange setting $\mathbf{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st}, \Sigma_t)$, where

- Σ_{st} is a set of source-to-target tgds (tuple-generating dependencies)
- Σ_t is a set of target tgds and target egds

Background – Examples of tgds and egds

Drug

did	dn	dc
d1	Aspirin	dc1
d2	Ibuprofen	dc5
…		

Source Schema **S**

Medicament

mid
d1
d2
…

DNames

mid	dn
d1	Aspirin
d2	Ibuprofen
…	

AIngred

sid	act.ingred.
d1	Null1
d2	Null2
…	

Target Schema **T**

Σ_{st}

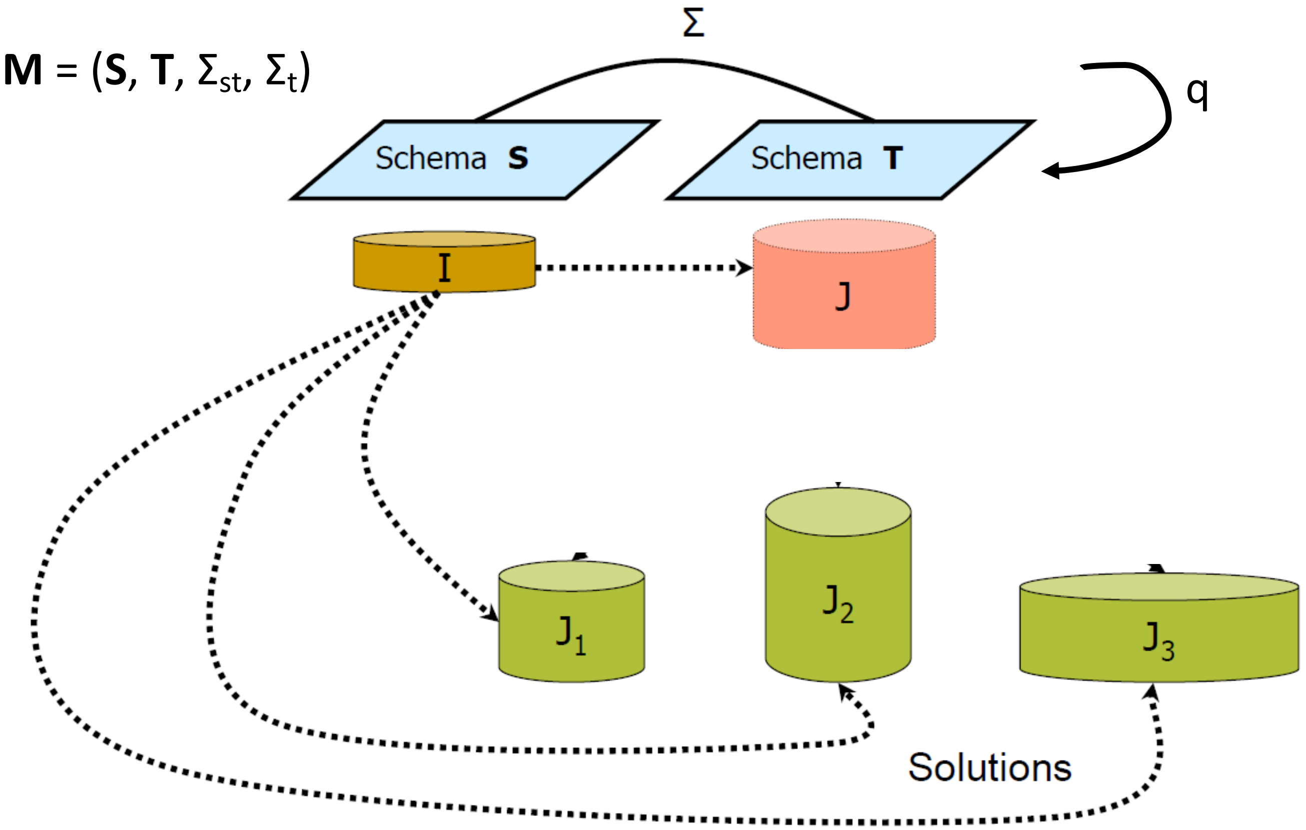
- st-tgd: $\forall did, dn, dc(Drug(did, dn, dc) \rightarrow Medicament(did) \wedge DNames(did, dn))$

Σ_t

- t-egd : $\forall mid, x, y\ (DNames(mid, x) \wedge DNames(mid, y) \rightarrow x = y)$
- t-tgd : $\forall mid, dn(DNames(mid, dn) \rightarrow \exists a\big(AIngred(mid, a)\big)$

Background – Data Exchange

M = (**S**, **T**, Σ_{st}, Σ_t)



The certain answers of a query q over **T** on I,
wrt a data exchange setting **M** are defined as:

cert(q,I,**M**) = ⋂{ q(J): J is a solution for I }

Fagin et al. 2005: if J is a universal solution for I w.r.t. **M**,
then the certain answers of every conj. query q over **T**
can be obtained by evaluating q on J
and then removing all tuples containing null values

Problem: **the chase may not always terminate**!

Background – $\mathcal{ELH}$ terminologies

- Concepts constructs: $C \coloneqq A \mid \top \mid \exists r.C \mid C \sqcap D$

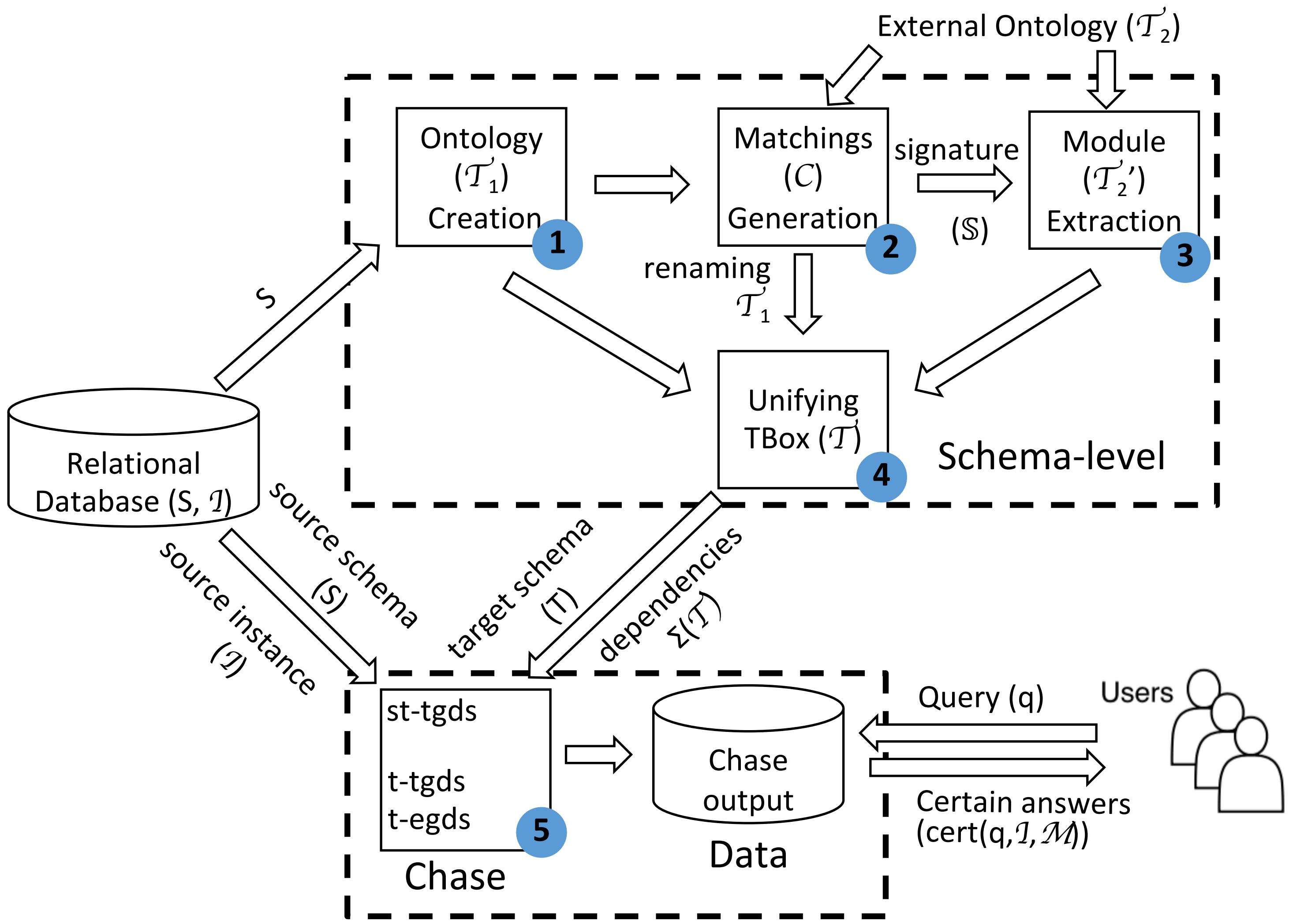
- An $\mathcal{ELH}$ terminology is a set of
  - concept definitions $A \equiv C$,
  - concept inclusions $A \sqsubseteq C$, and
  - role inclusions $r \sqsubseteq s$

Contributions

- Adoption of AI and data exchange methods and tools in real medical use case
- Backing our use case with concrete theoretical guarantees
  - we define acyclic $\mathcal{ELH}^{fdr}_{\!\!f}$ and show it is C-stratified
    - the standard chase always terminates in polynomial time
- A reference framework architecture for ontology-enriched query answering
  - available on github (https://github.com/IBM/ontology-enriched-query-answering)
- Experimental evaluation showing the benefits of our framework
  - more query answers by exploiting SNOMED CT as an external reference ontology

Framework Architecture

- Step 1: Ontology Creation
- Step 2: Matchings Generation
- Step 3: Module Extraction
- Step 4: Unifying the TBox
- Step 5: Query Answering via the Chase



Ontology Creation [Lei et al. 2018]

Drug

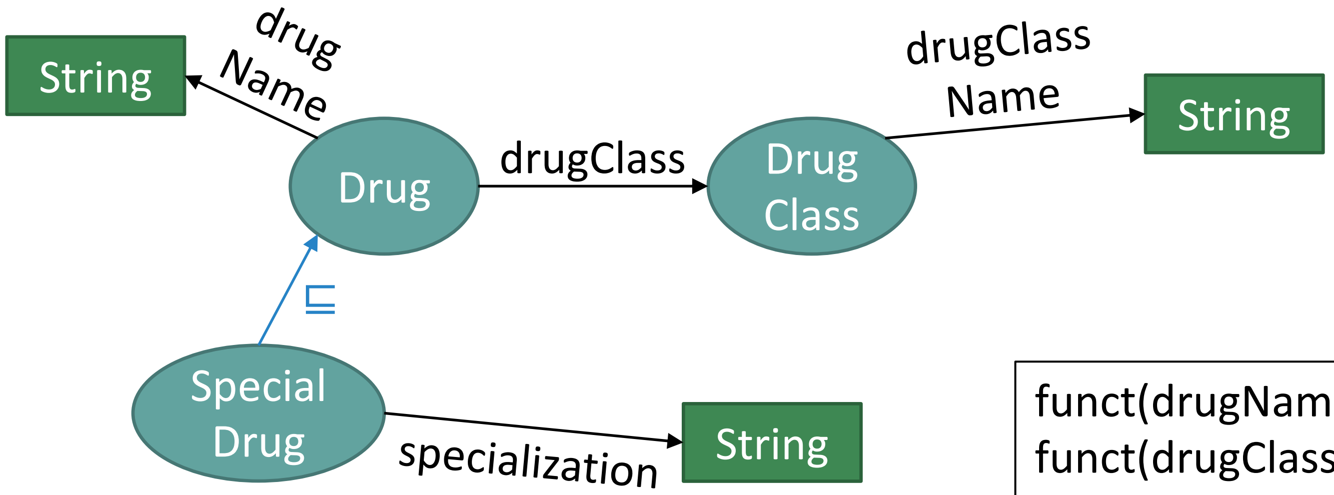
drugId	drugName	drugClass
d1	Aspirin	dc1
…		

DrugClass

drugClassId	drugClassName
dc1	NSAID
…	

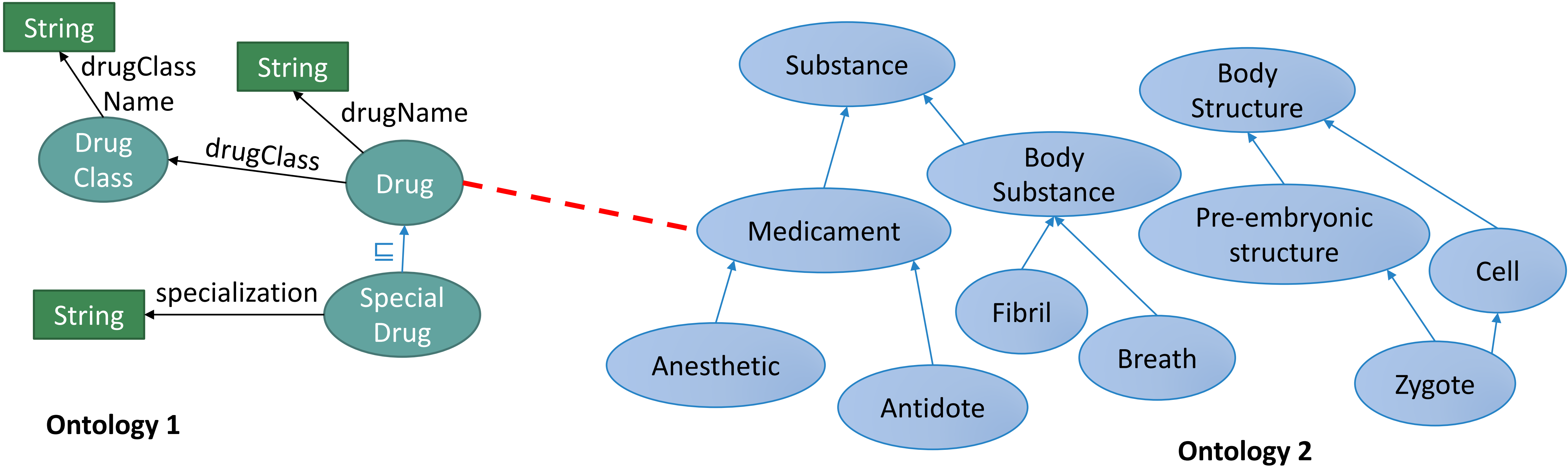
SpecialDrug

sdrugId	specialization
d8	COVID-19
…	



funct(drugName)
funct(drugClass)
funct(drugClassName)
funct(specialization)

Matchings Generation

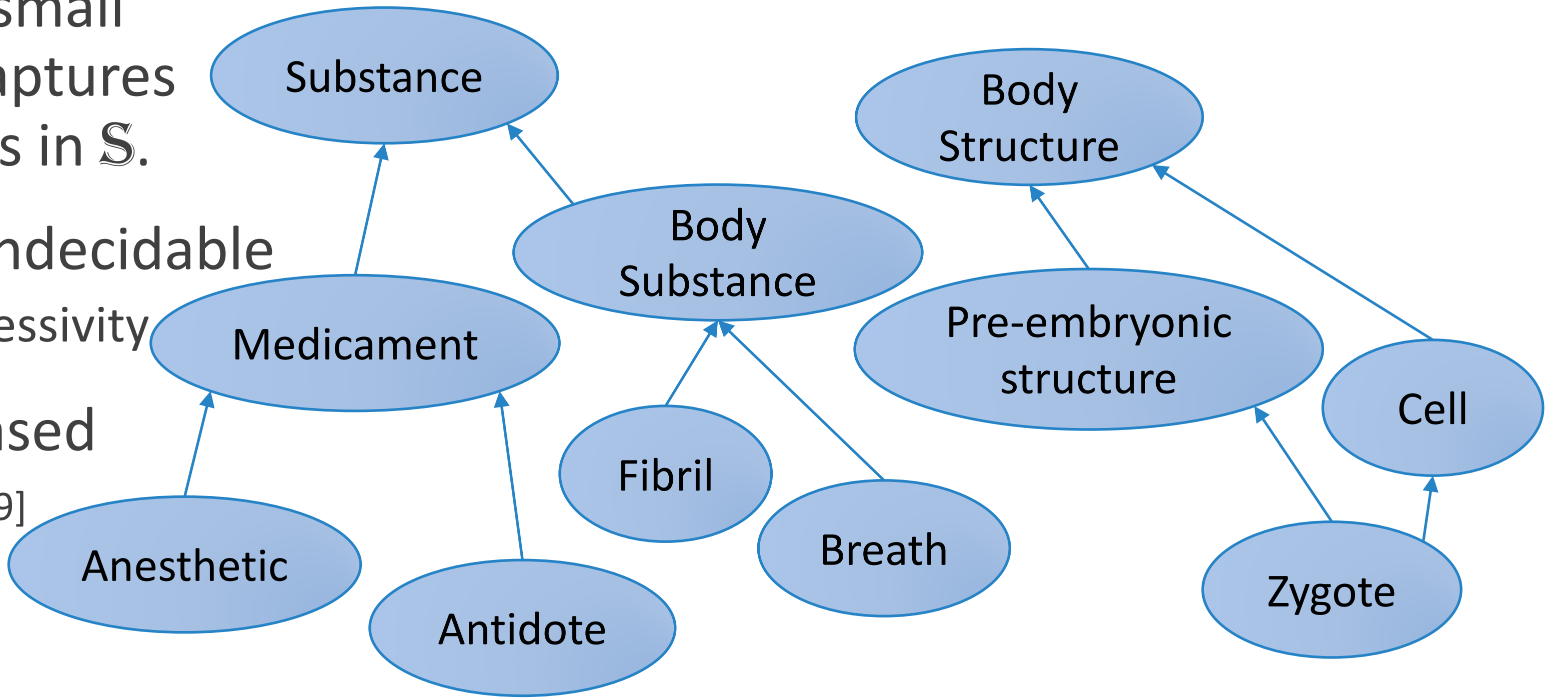


- Several methods tested, including SOTA in ontology matching (LogMap, AML) with unsatisfactory results
- Ended up providing matchings with manual inspection

Matchings C = {(Drug, Medicament)}

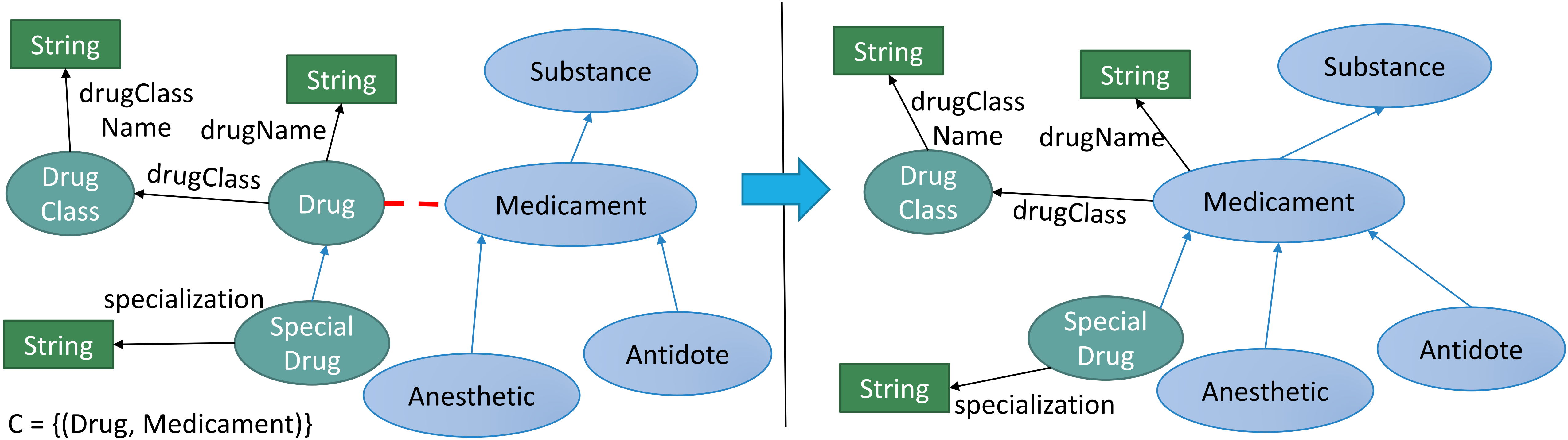
Module Extraction [Cuenca Grau et al. 2009]

- Given a signature $\mathbb{S}$, retain a small subset of the ontology that captures only the meaning of the terms in $\mathbb{S}$.
- ExpTime-complete or even undecidable
  - depending on the ontology expressivity
- We use a syntactic locality-based module extraction [Grau et al. 2009]
  - ⊥T*-syntactic locality
- $\mathbb{S} = \{N_2 \mid (N_1, N_2) \in C\}$
  - Running example: C = {(Drug, Medicament)}
    - $\mathbb{S}$ = {Medicament}



Unifying the TBox

- To produce the unified TBox T:
  - for every matching $(N_1,N_2) \in C$ rename every occurrence of N_1 in Ontology 1 with N_2
  - return the union of Ontology 1 (after renaming) and the $\mathbb{S}$-module from Ontology 2



C = {(Drug, Medicament)}

Expressivity of the Unified TBox (use case)

- SNOMED CT belongs to the acyclic $\mathcal{ELH}$ fragment of Description Logics
- Ontology generated from the DB falls under acyclic $\mathcal{EL}$, extended with domain and range restrictions, as well as functionality assertions

➢ The unified TBox can be expressed in **acyclic $\mathcal{ELH}^{fdr}$**

- $\mathcal{ELH}^{fdr}$ is $\mathcal{ELH}$ extended with domain and range restrictions, and limited functionality
  - (simplified): no functional roles are allowed on the right-hand side of axioms
- Acyclicity intuition *(proper definitions in the paper)*
  - $\mathcal{ELH}$ acyclicity: prevents a concept from directly or indirectly referring to (aka using) itself
  - $\mathcal{ELH}^{fdr}$ acyclicity: need additional conditions to take care of domain & range restrictions and functionality
    - Example: $A \sqsubseteq \exists r$ $\quad rng(r) \sqsubseteq A$ $\quad$ (acyclic under the $\mathcal{ELH}$ acyclicity conditions, but results in infinite chase)

Chase – st-tgds and t-egds

- Our schema exchange setting **M** = (**S**, **T**, Σ_{st}, Σ_t):
  - use the relational schema S of the input DB as the source schema **S**
  - use the unified TBox T as the target schema **T**
- Use the following rule to generate st-tgds from every relation R of S:

$$\boxed{R(\underline{x_1}, \ldots, x_n) \rightarrow R'(x_1) \wedge R'^{1,2}(x_1, x_2) \wedge \cdots \wedge R'^{1,n}(x_1, x_n),}$$

where x_1 is the primary key of R, and R', $R'^{1,j}$ are fresh relation names.
If (R, R'') ∈ C, we replace $R'(x_1)$ above with $R''(x_1)$, i.e., we rename R as R''

- Every functional role r gives rise to the t-egd:

$$\boxed{r(x, y) \wedge r(x, z) \rightarrow y = z}$$

Chase – t-tgds

- Fact: for every $\mathcal{EL}$ concept C, there is a conj. query $q_C(x)$ with a free variable x, s.t. $C(x) \equiv q_C(x)$
  - **Case 1:** $q_C(x) \coloneqq \exists \bar{y} \varphi_C(\bar{y}, x)$, where $\bar{y}$ is a non-empty tuple of variables
  - **Case 2:** $q_C(x) \coloneqq A_1(x) \wedge \cdots \wedge A_n(x)$, where $A_1(x), \ldots, A_n(x)$ are concept names

- The tgds arising from an $\mathcal{ELH}^{fdr}$ terminology have one of the following seven types*:
  1) $A(x) \rightarrow \exists \bar{y} \varphi_C(\bar{y}, x)$ *(arises from $A \sqsubseteq C$, where C is of Case 1)*
  2) $A(x) \rightarrow A_1(x) \wedge \cdots \wedge A_n(x)$ *(arises from $A \sqsubseteq C$, where C is of Case 2)*
  3) $\varphi_C(\bar{y}, x) \rightarrow A(x)$ *(arises from $C \sqsubseteq A$, where C is of Case 1)*
  4) $A_1(x) \wedge \cdots \wedge A_n(x) \rightarrow A(x)$ *(arises from $C \sqsubseteq A$, where C is of Case 2)*
  5) $r_1(x, y) \rightarrow r_2(x, y)$ *(arises from $r_1 \sqsubseteq r_2$)*
  6) $r(x, y) \rightarrow A(x)$ *(arises from $dom(r) \sqsubseteq A$)*
  7) $r(x, y) \rightarrow A(y)$ *(arises from $rng(r) \sqsubseteq A$)*

*We treat each axiom $A \equiv C$ as two inclusions $A \sqsubseteq C$ and $C \sqsubseteq A$

Chase Termination

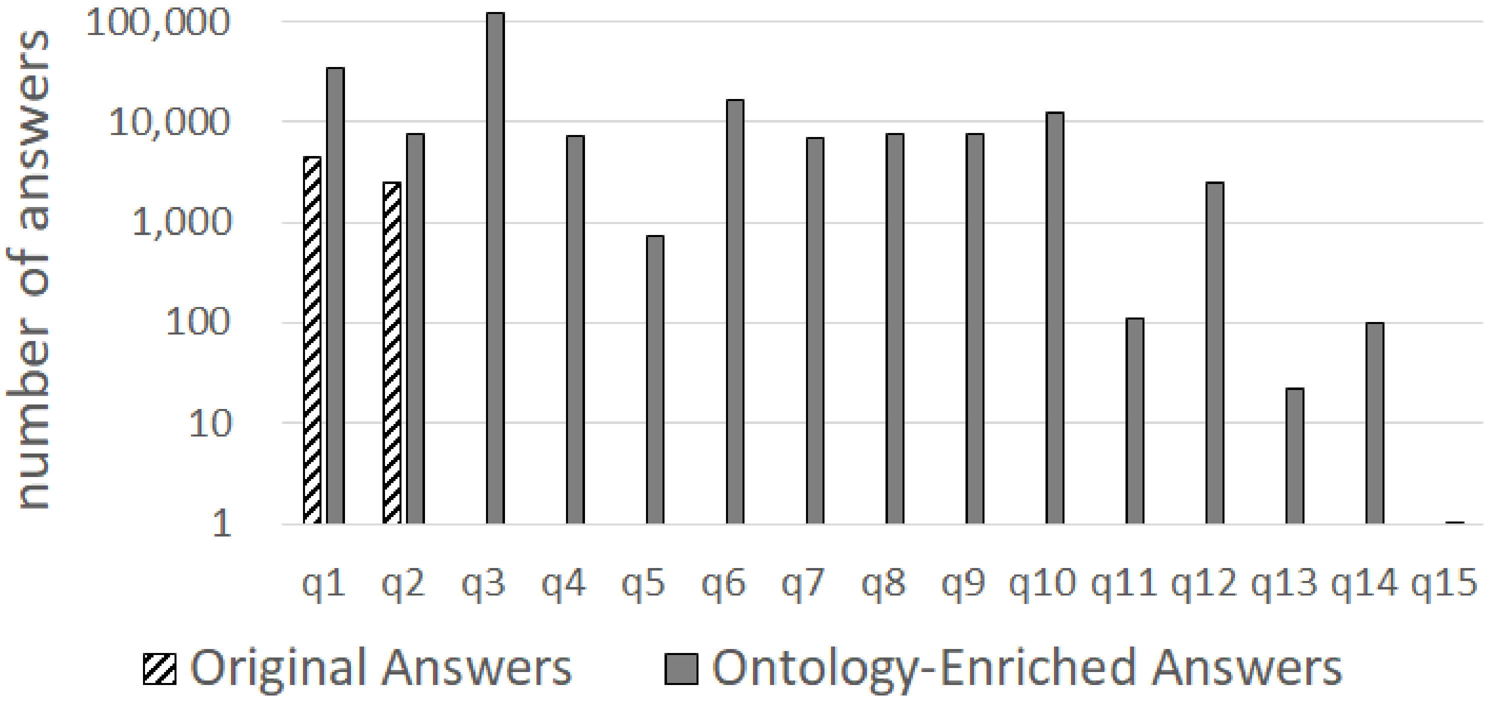
Theorem: Let T be an acyclic $\mathcal{ELH}^{fdr}$ terminology and let $\Sigma(T)$ be the associated set of tgds and egds. Then $\Sigma(T)$ is C-stratified.

[Meier, Schmidt, and Lausen 2009]: if a set Σ of tgds and egds is C-stratified, then, on every input database instance J, the standard chase w.r.t. Σ terminates in time bounded by a polynomial in the size of J.

Evaluation

- **Input DB (MDB):** 62 relations of arities from 2 to 11, 158 FKs, 500k+ tuples, 62.3MB
- **Ontologies**:
  - MDB ontology (Step 1): 49 concepts, 170 roles (all with domain & range), 156 functional
  - SNOMED CT: 356k concepts, 119 roles (none is functional or with domain/range restrictions)
  - 12 matchings identified (with manual inspection, after running LogMap, AML)
    - signature $\mathbb{S}$ given for module extraction contains 12 elements
  - $\mathbb{S}$-module in SNOMED CT: 35 concepts, 7 roles
  - Unified TBox: 72 concepts, 177 roles, 156 functional, 170 with domain & range restrictions
- **Chase:**
  - Number of tgds: 62 st-tgds, 154 t-tgds, 156 t-egds
  - Chase execution time: 1,676ms (870ms for st-tgds, 806ms for t-tgds and t-egds)
  - Chase space overhead: 24% (62.3 MB used for the source instance; 77.5 MB used for the chase output)

Evaluation Results



Queries selected from logs Jan-June 2019

- Original Answers: just renaming
- Ontology-Enriched Answers: using our framework

Beneficial for two types of queries:

- CQs whose conjuncts all appear in MDB, but we learned something new about them from SNOMED CT (q_1-q_5)
- CQs with some conjuncts unknown (q_6-q_{15})
  - could not be answered originally

Query-answering times ranged from 1ms (for q_{13}-q_{15}) to 576ms (for q_3), averaging **64ms**.

Thank you!

The source code of this work is publicly available:

https://github.com/IBM/ontology-enriched-query-answering