# Rewriting OWL 2 QL Ontology-Mediated Queries: Succinctness

**Roman Kontchakov**

*Department of Computer Science and Inf. Systems, Birkbeck College, London*

http://www.dcs.bbk.ac.uk/~roman

# OWL 2 QL Ontology-Mediated Queries

an OMQ = an ontology + a CQ

# OWL 2 QL Ontology-Mediated Queries

an OMQ = an ontology + a CQ

an OWL 2 QL **ontology** is a finite set of axioms of the form

$$\forall x\, \big(\tau(x) \to \tau'(x)\big) \qquad \forall x\, \big(\tau(x) \land \tau'(x) \to \bot\big)$$

$$\forall x, y\, \big(\varrho(x,y) \to \varrho'(x,y)\big) \qquad \forall x, y\, \big(\varrho(x,y) \land \varrho'(x,y) \to \bot\big)$$

$$\forall x\, \varrho(x,x) \qquad \forall x\, \big(\varrho(x,x) \to \bot\big)$$

**classes** or **concepts** $\quad \tau(x) \;::=\; \top \;\mid\; A(x) \;\mid\; \exists y\, \varrho(x,y)$

**properties** or **roles** $\quad \varrho(x,y) \;::=\; \top \;\mid\; P(x,y) \;\mid\; P(y,x)$

# OWL 2 QL Ontology-Mediated Queries

an OMQ = an ontology + a CQ

an OWL 2 QL **ontology** is a finite set of axioms of the form

$$\forall x\,\big(\tau(x) \to \tau'(x)\big) \qquad \forall x\,\big(\tau(x) \land \tau'(x) \to \bot\big)$$

$$\forall x, y\,\big(\varrho(x,y) \to \varrho'(x,y)\big) \qquad \forall x, y\,\big(\varrho(x,y) \land \varrho'(x,y) \to \bot\big)$$

$$\forall x\, \varrho(x,x) \qquad \forall x\,\big(\varrho(x,x) \to \bot\big)$$

**classes** or **concepts** $\quad \tau(x) \ ::= \ \top \mid A(x) \mid \exists y\, \varrho(x,y)$

**properties** or **roles** $\quad \varrho(x,y) \ ::= \ \top \mid P(x,y) \mid P(y,x)$

an **data instance** $\mathcal{A}$ is a finite set of ground atoms $S(\vec{a})$

a **conjunctive query** (CQ) $q(\vec{x})$ is $\exists \vec{y}\, \varphi(\vec{x}, \vec{y})$,
where $\varphi$ is a conjunction of atoms $S(\vec{z})$ all of whose variables are among $\vec{x}, \vec{y}$

# OWL 2 QL Ontology-Mediated Queries

an OMQ = an ontology + a CQ

an OWL 2 QL **ontology** is a finite set of axioms of the form

$$\forall x\,\big(\tau(x) \to \tau'(x)\big) \qquad \forall x\,\big(\tau(x) \land \tau'(x) \to \bot\big)$$

$$\forall x, y\,\big(\varrho(x,y) \to \varrho'(x,y)\big) \qquad \forall x, y\,\big(\varrho(x,y) \land \varrho'(x,y) \to \bot\big)$$

$$\forall x\,\varrho(x,x) \qquad \forall x\,\big(\varrho(x,x) \to \bot\big)$$

**classes** or **concepts** $\qquad \tau(x) \ ::= \ \top \mid A(x) \mid \exists y\,\varrho(x,y)$

**properties** or **roles** $\qquad \varrho(x,y) \ ::= \ \top \mid P(x,y) \mid P(y,x)$

an **data instance** $\mathcal{A}$ is a finite set of ground atoms $S(\vec{a})$

a **conjunctive query** (CQ) $q(\vec{x})$ is $\exists \vec{y}\,\varphi(\vec{x},\vec{y})$,
where $\varphi$ is a conjunction of atoms $S(\vec{z})$ all of whose variables are among $\vec{x}, \vec{y}$

$\mathcal{C}_{\mathcal{T},\mathcal{A}}$ is the **canonical model** (**chase**) of $(\mathcal{T},\mathcal{A})$ and $\mathcal{C}_{\mathcal{T}}^{\tau(a)} = \mathcal{C}_{\mathcal{T},\{\tau(a)\}}$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$-**PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$**-PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

$\Sigma_3$**-PE** have matrix of the form $\vee\wedge\vee$ = unions of SCQs

e.g., tree-witness rewriting size $O(2^{|\Theta_Q|} \cdot |Q|^2)$ with $\underbrace{|\Theta_Q| \leq 3^{|q|}}_{\text{number of tree witnesses}}$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$-**PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

$\Sigma_3$-**PE** have matrix of the form $\vee\wedge\vee$ = unions of SCQs

e.g., tree-witness rewriting size $O(2^{|\Theta_Q|} \cdot |Q|^2)$ with $\underbrace{|\Theta_Q| \leq 3^{|q|}}_{\text{number of tree witnesses}}$

$\Pi_4$-**PE** have matrix of the form $\wedge\vee\wedge\vee$

e.g., modified tree-witness rewriting without conflicts has size $O(|\Theta_Q| \cdot |Q|^2)$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$-**PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

$\Sigma_3$-**PE** have matrix of the form $\vee\wedge\vee$ = unions of SCQs

e.g., tree-witness rewriting size $O(2^{|\Theta_Q|} \cdot |Q|^2)$ with $\underbrace{|\Theta_Q| \leq 3^{|q|}}_{\text{number of tree witnesses}}$

$\Pi_4$-**PE** have matrix of the form $\wedge\vee\wedge\vee$

e.g., modified tree-witness rewriting without conflicts has size $O(|\Theta_Q| \cdot |Q|^2)$

**NDL** (non-recursive datalog) = PE + shared subformulas = SPJU + views

e.g., Presto produced rewritings of size $|\mathcal{T}|^{O(1)} \cdot 2^{|q|}$

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$**-PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

$\Sigma_3$**-PE** have matrix of the form $\vee\wedge\vee$ = unions of SCQs

e.g., tree-witness rewriting size $O(2^{|\Theta_Q|} \cdot |Q|^2)$ with $\underbrace{|\Theta_Q| \leq 3^{|q|}}_{\text{number of tree witnesses}}$

$\Pi_4$**-PE** have matrix of the form $\wedge\vee\wedge\vee$

e.g., modified tree-witness rewriting without conflicts has size $O(|\Theta_Q| \cdot |Q|^2)$

**NDL** (non-recursive datalog) = PE + shared subformulas = SPJU + views

e.g., Presto produced rewritings of size $|\mathcal{T}|^{O(1)} \cdot 2^{|q|}$

**FO** (first-order formulas) = PE + negation ≈ RA

# Target Languages for Rewritings

**UCQ** = unions of SPJ queries

as produced, e.g., by PerfectRef size $|q|^{|\mathcal{T}|} \cdot 2^{O(|q|^2)}$

**PE** (positive existential formulas) = SPJU queries

$\Pi_2$-**PE** have matrix of the form $\wedge\vee$ = semiconjunctive queries (SCQs)

for ontologies without $\exists y$ on RHS, the size is $|q| \cdot |\mathcal{T}|$

$\Sigma_3$-**PE** have matrix of the form $\vee\wedge\vee$ = unions of SCQs

e.g., tree-witness rewriting size $O(2^{|\Theta_Q|} \cdot |Q|^2)$ with $\underbrace{|\Theta_Q| \leq 3^{|q|}}_{\text{number of tree witnesses}}$

$\Pi_4$-**PE** have matrix of the form $\wedge\vee\wedge\vee$

e.g., modified tree-witness rewriting without conflicts has size $O(|\Theta_Q| \cdot |Q|^2)$

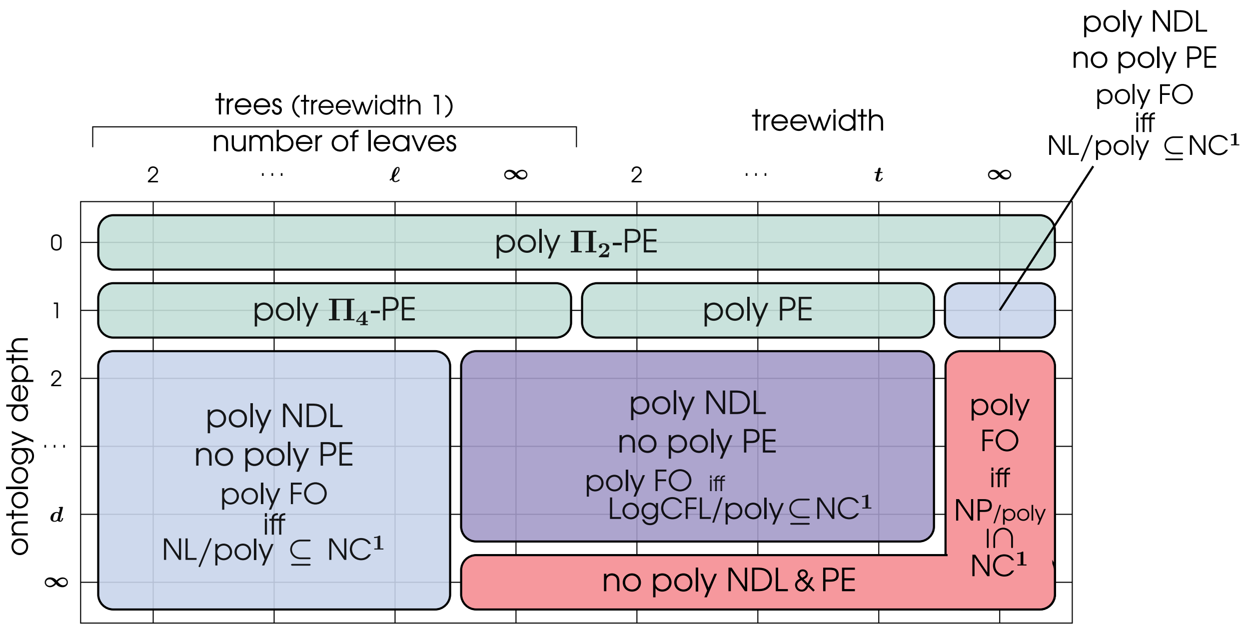
**NDL** (non-recursive datalog) = PE + shared subformulas = SPJU + views

e.g., Presto produced rewritings of size $|\mathcal{T}|^{O(1)} \cdot 2^{|q|}$

**FO** (first-order formulas) = PE + negation ≈ RA

**no additional constants, no assumptions on data**

# Succinctness Landscape




ontology depth

$0$ = no axioms with $\exists y$ on the right-hand side

$d$ $\approx$ trees $\mathcal{C}_{\mathcal{T}}^{\tau(a)}$ of labelled nulls are of depth at most $d$

# Hypergraphs for OMQs

$A_1(x) \to \exists y \big(R(x,y) \land Q(y,x)\big)$

$A_1(x) \to \exists y\, P_\zeta(x,y)$ $\quad P_\zeta(x,y) \to R(x,y)$ $\quad P_\zeta(x,y) \to Q(y,x)$

$A_2(x) \to \exists y\, S_1(x,y)$ $\quad S_1(x,y) \to S_2(x,y)$

$A_3(x) \to \exists y\, Q(x,y)$ $\quad \exists y\, Q(y,x) \to \exists y\, S_1(x,y)$

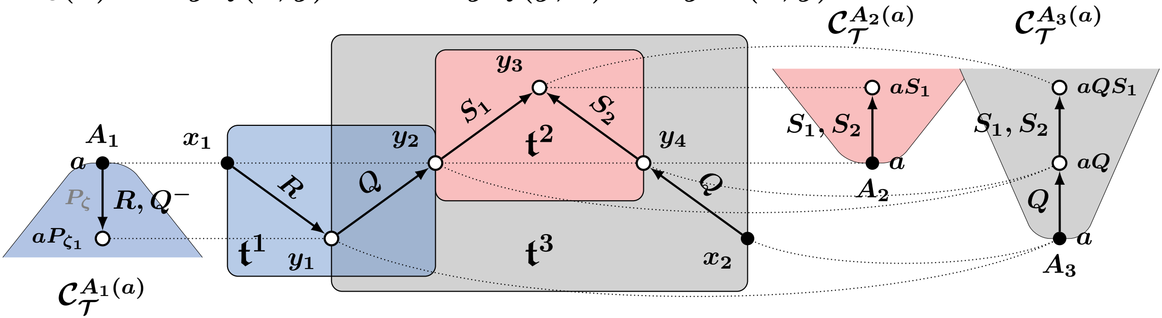
# Hypergraphs for OMQs

$$A_1(x) \to \exists y\,\big(R(x,y) \land Q(y,x)\big)$$

$$A_1(x) \to \exists y\, P_\zeta(x,y) \qquad P_\zeta(x,y) \to R(x,y) \qquad P_\zeta(x,y) \to Q(y,x)$$

$$A_2(x) \to \exists y\, S_1(x,y) \qquad S_1(x,y) \to S_2(x,y)$$

$$A_3(x) \to \exists y\, Q(x,y) \qquad \exists y\, Q(y,x) \to \exists y\, S_1(x,y)$$



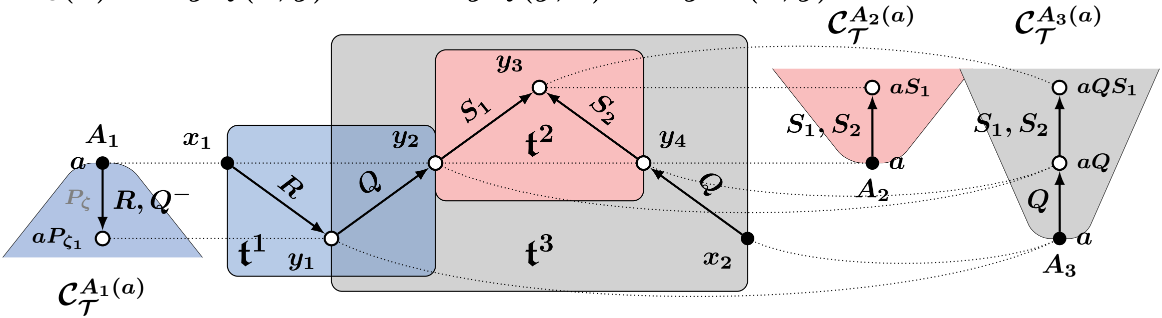
# Hypergraphs for OMQs

$A_1(x) \to \exists y\,\big(R(x,y) \land Q(y,x)\big)$

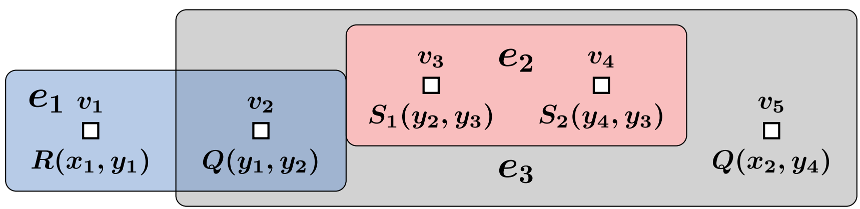
$A_1(x) \to \exists y\, P_\zeta(x,y)$ $\quad P_\zeta(x,y) \to R(x,y)$ $\quad P_\zeta(x,y) \to Q(y,x)$

$A_2(x) \to \exists y\, S_1(x,y)$ $\quad S_1(x,y) \to S_2(x,y)$

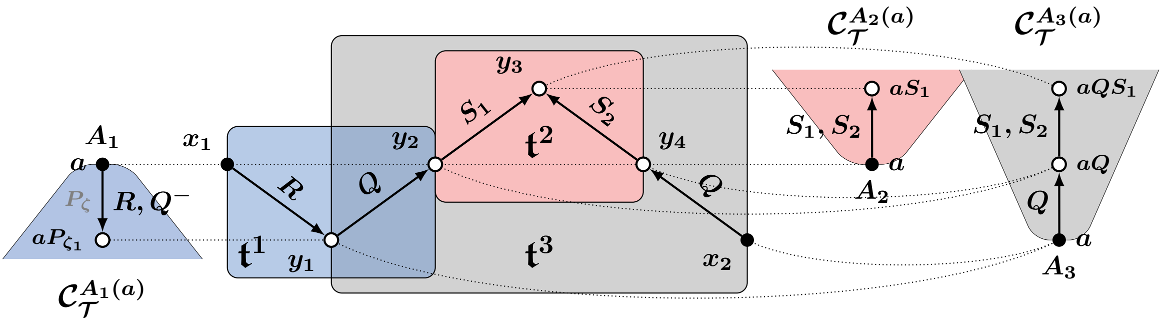
$A_3(x) \to \exists y\, Q(x,y)$ $\quad \exists y\, Q(y,x) \to \exists y\, S_1(x,y)$




**hypergraph** $\mathcal{H}(Q)$ query atoms = vertices
sets of query atoms that can be mapped to trees = hyperedges



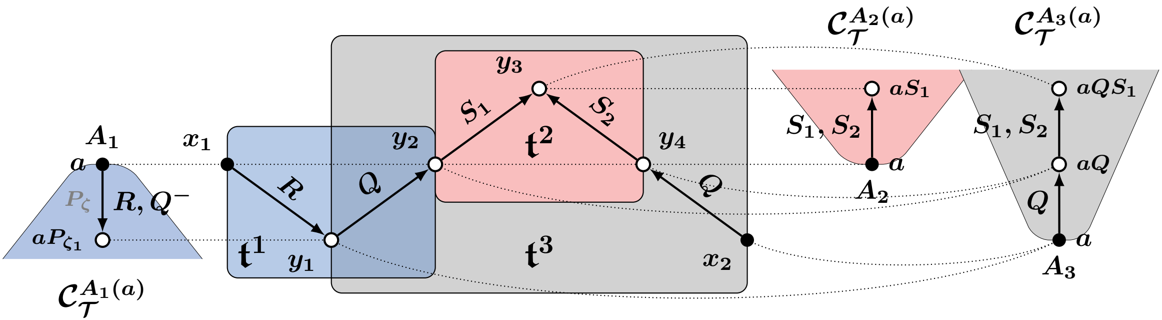
# OMQ Answering and Hypergraphs




a **homomorphism** $q \to \mathcal{C}_{\mathcal{T},\mathcal{A}}$ = a map from variables of $q$ to **ind**$(\mathcal{A})$ + an **independent subset** of $\mathcal{H}(Q)$ such that

1. each hyperedge is 'generated' by the data
2. each vertex outside hyperedges is 'present' in the data

# OMQ Answering and Hypergraphs




a **homomorphism** $q \to \mathcal{C}_{\mathcal{T},\mathcal{A}}$ **=** a map from variables of $q$ to **ind**$(\mathcal{A})$ **+** an **independent subset** of $\mathcal{H}(Q)$ such that

1. each hyperedge is 'generated' by the data
2. each vertex outside hyperedges is 'present' in the data

**hypergraph function**

$$f_H = \bigvee_{E' \text{ independent}} \Big( \bigwedge_{v \in V \setminus V_{E'}} p_v \; \wedge \; \bigwedge_{e \in E'} p_e \Big)$$

# From OMQs to Hypergraph Programs (HGPs)

a **HGP** $P$ is a hypergraph $H$ whose vertices are labelled by $0$, $1$, or a literal over $p_1, \dots, p_n$

$P$ returns $1$ on an assignment $\alpha\colon \{p_1, \dots, p_n\} \to \{0, 1\}$ if there is an independent subset in $H$ that covers all zeroes under $\alpha$

# From OMQs to Hypergraph Programs (HGPs)

a **HGP** $P$ is a hypergraph $H$ whose vertices are labelled by
$0$, $1$, or a literal over $p_1, \dots, p_n$

$P$ returns $1$ on an assignment $\alpha\colon \{p_1, \dots, p_n\} \to \{0, 1\}$
if there is an independent subset in $H$ that covers all zeroes under $\alpha$

a HGP is **monotone** if none of the labels is negative

# From OMQs to Hypergraph Programs (HGPs)

a **HGP** $P$ is a hypergraph $H$ whose vertices are labelled by $0$, $1$, or a literal over $p_1, \dots, p_n$

$P$ returns $1$ on an assignment $\alpha\colon \{p_1, \dots, p_n\} \to \{0, 1\}$ if there is an independent subset in $H$ that covers all zeroes under $\alpha$

a HGP is **monotone** if none of the labels is negative

degree of a hypergraph = the maximum number of hyperedges that contain a vertex

# From OMQs to Hypergraph Programs (HGPs)

a **HGP** $P$ is a hypergraph $H$ whose vertices are labelled by
$0$, $1$, or a literal over $p_1, \dots, p_n$

$P$ returns $1$ on an assignment $\alpha\colon \{p_1, \dots, p_n\} \to \{0, 1\}$
if there is an independent subset in $H$ that covers all zeroes under $\alpha$

a HGP is **monotone** if none of the labels is negative

degree of a hypergraph = the maximum number of hyperedges that contain a vertex

**Proposition** For every hypergraph $H$ of degree $d$, there is monotone HGP that computes $f_H$ and is of degree $\leq \max(2, d)$ and size $O(|H|)$

# From OMQs to Hypergraph Programs (HGPs)

a **HGP** $P$ is a hypergraph $H$ whose vertices are labelled by
$0$, $1$, or a literal over $p_1, \dots, p_n$

$P$ returns $1$ on an assignment $\alpha\colon \{p_1, \dots, p_n\} \to \{0, 1\}$
if there is an independent subset in $H$ that covers all zeroes under $\alpha$

a HGP is **monotone** if none of the labels is negative

degree of a hypergraph = the maximum number of hyperedges that contain a vertex

**Proposition** For every hypergraph $H$ of degree $d$, there is monotone HGP that computes $f_H$ and is of degree $\leq \max(2, d)$ and size $O(|H|)$

| OMQ $Q = (\mathcal{T}, q)$ | monotone HGP | of size | computes |
|---|---|---|---|
| $\mathcal{T}$ of depth 1 | HGP of degree 2 | $O(\lvert q\rvert)$ | $f_Q^\triangledown$ |
| tree-shaped $q$ with $\ell$ leaves | linear THGP | $\lvert q\rvert^{O(\ell^2)}$ | $f_Q^\triangledown$ |
| $q$ of treewidth $t$ and $\mathcal{T}$ of depth $d$ | THGP | $\lvert q\rvert^{O(1)} \cdot \lvert 2^{O(dt)}$ | $f_Q^\blacktriangledown$ |
| $q$ of treewidth $t$ and $\mathcal{T}$ of depth 1 | THGP of degree $2^{O(t)}$ | $\lvert q\rvert^{O(1)} \cdot 2^{O(t)}$ | $f_Q^\triangledown$ |
| tree-shaped $q$ and $\mathcal{T}$ of depth 1 | THGP of degree $2$ | $\lvert q\rvert^{O(1)}$ | $f_Q^\triangledown$ |

$f_Q^\triangledown = f_{\mathcal{H}(Q)}$ and $f_Q^\blacktriangledown$ is its modification for exponential $\mathcal{H}(Q)$

# Complexity of Boolean Functions v Size of Rewritings

**Theorem** If either $f_Q^{\triangledown}$ or $f_Q^{\blacktriangledown}$ is in

- $\mathsf{NC}^1$, then $Q$ has a polynomial FO-rewriting — Boolean formulas
- mP/poly, then $Q$ has a polynomial NDL-rewriting — monotone Boolean circuits
- $\mathsf{mNC}^1$, then $Q$ has a polynomial PE-rewriting — Boolean formulas

# Complexity of Boolean Functions v Size of Rewritings

**Theorem** If either $f_{Q}^{\triangledown}$ or $f_{Q}^{\blacktriangledown}$ is in

- $\mathsf{NC}^1$, then $Q$ has a polynomial FO-rewriting — Boolean formulas
- mP/poly, then $Q$ has a polynomial NDL-rewriting — monotone Boolean circuits
- $\mathsf{mNC}^1$, then $Q$ has a polynomial PE-rewriting — Boolean formulas

Hint: replace each $p_v$ by the respective atom
and each $p_e$ by a formula for 'generating' the tree-witness

# Complexity of Boolean Functions v Size of Rewritings

**Theorem** If either $f_Q^{\triangledown}$ or $f_Q^{\blacktriangledown}$ is in

- $\mathsf{NC}^1$, then $Q$ has a polynomial FO-rewriting — Boolean formulas
- mP/poly, then $Q$ has a polynomial NDL-rewriting — monotone Boolean circuits
- $\mathsf{mNC}^1$, then $Q$ has a polynomial PE-rewriting — Boolean formulas

Hint: replace each $p_v$ by the respective atom
and each $p_e$ by a formula for 'generating' the tree-witness

$f_Q^{\vartriangle}$ is the **primitive evaluation function** of $Q$
— evaluating $Q$ on a single-object ABoxes

**Theorem** If $Q$ has a polynomial

- FO-rewriting, then $f_Q^{\vartriangle} \in \mathsf{NC}^1$
- NDL-rewriting, then $f_Q^{\vartriangle} \in$ mP/poly
- PE-rewriting, then $f_Q^{\vartriangle} \in \mathsf{mNC}^1$

## Complexity of Boolean Functions v Size of Rewritings

**Theorem** If either $f_Q^{\triangledown}$ or $f_Q^{\blacktriangledown}$ is in

- $\mathsf{NC}^1$, then $Q$ has a polynomial FO-rewriting — Boolean formulas
- mP/poly, then $Q$ has a polynomial NDL-rewriting — monotone Boolean circuits
- $\mathsf{mNC}^1$, then $Q$ has a polynomial PE-rewriting — Boolean formulas

Hint: replace each $p_v$ by the respective atom
and each $p_e$ by a formula for 'generating' the tree-witness

$f_Q^{\vartriangle}$ is the **primitive evaluation function** of $Q$
— evaluating $Q$ on a single-object ABoxes

**Theorem** If $Q$ has a polynomial

- FO-rewriting, then $f_Q^{\vartriangle} \in \mathsf{NC}^1$
- NDL-rewriting, then $f_Q^{\vartriangle} \in$ mP/poly
- PE-rewriting, then $f_Q^{\vartriangle} \in \mathsf{mNC}^1$

Hint: on a single-object ABox, quantifiers are 'meaningless'
and the rewriting boils down to a propositional f-la, etc.

# Representation Results for Hypergraphs

**Theorem** (*i*) Any hypergraph $\boldsymbol{H}$ is isomorphic to
a subgraph of $\mathcal{H}(\boldsymbol{Q_H})$ for a polynomial-size $\boldsymbol{Q_H}$ with an ontology of **depth 2**

(*ii*) And any HGP based on $\boldsymbol{H}$ computes a subfunction of $\boldsymbol{f^{\vartriangle}_{Q_H}}$

# Representation Results for Hypergraphs

**Theorem** (*i*) Any hypergraph $\boldsymbol{H}$ is isomorphic to
a subgraph of $\mathcal{H}(\boldsymbol{Q}_{\boldsymbol{H}})$ for a polynomial-size $\boldsymbol{Q}_{\boldsymbol{H}}$ with an ontology of **depth 2**

(*ii*) And any HGP based on $\boldsymbol{H}$ computes a subfunction of $\boldsymbol{f}^{\vartriangle}_{\boldsymbol{Q}_{\boldsymbol{H}}}$

**Theorem** (*i*) Any hypergraph $\boldsymbol{H}$ of **degree 2** is isomorphic to
$\mathcal{H}(\boldsymbol{S}_{\boldsymbol{H}})$ for a polynomial-size $\boldsymbol{S}_{\boldsymbol{H}}$ with an ontology of **depth 1**

(*ii*) . . .

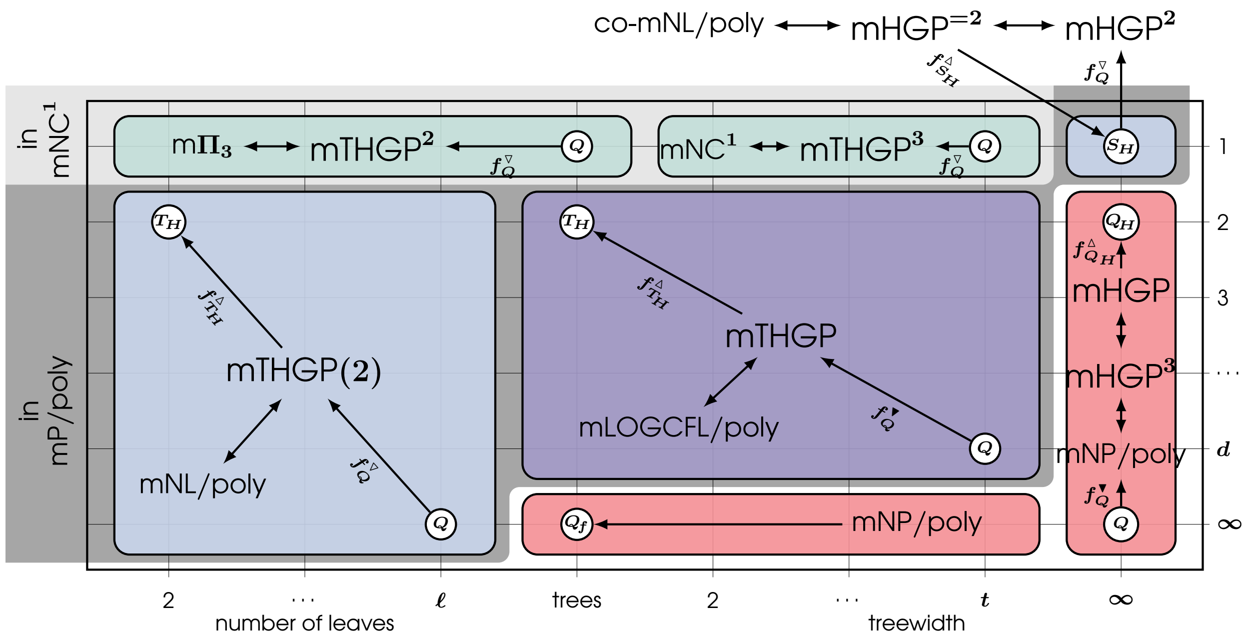
# Representation Results for Hypergraphs

**Theorem** (*i*) Any hypergraph $H$ is isomorphic to
a subgraph of $\mathcal{H}(Q_H)$ for a polynomial-size $Q_H$ with an ontology of **depth 2**
(*ii*) And any HGP based on $H$ computes a subfunction of $f^{\vartriangle}_{Q_H}$

**Theorem** (*i*) Any hypergraph $H$ of **degree 2** is isomorphic to
$\mathcal{H}(S_H)$ for a polynomial-size $S_H$ with an ontology of **depth 1**
(*ii*) . . .

**Theorem** Any **tree** hypergraph $H$ with $\ell$ leaves is isomorphic to
a subgraph of $\mathcal{H}(T_H)$ for a polynomial-size $T_H$ with an of ontology of **depth 2**
and a **tree-shaped CQ** with $\ell$ leaves
(*ii*) . . .

# Roadmap for Succinctness Proofs




$$\mathsf{m}\Pi_3 \subsetneq \mathsf{mAC}^0 \subsetneq \mathsf{mNC}^1 \subsetneq \mathsf{mNL/poly} \subseteq \mathsf{mLOGCFL/poly} \subsetneq \mathsf{mP/poly} \subsetneq \mathsf{mNP/poly}$$

**Boolean formulas** **nondeterministic Boolean circuits**

# Conclusions

- HGPs provide a natural link between complexity classes and hypergraph functions

- polynomial **PE-rewritings** exists only in for very **restricted** cases of ontologies of depth 0 & 1 (except unbounded treewidth)

- polynomial **NDL-rewritings** exists for most cases where query answering is **tractable**

optimal NDL-rewritings in Stas's talk

- existence of polynomial PE-, NDL- and FO-rewritings is closely related to **circuit complexity**