

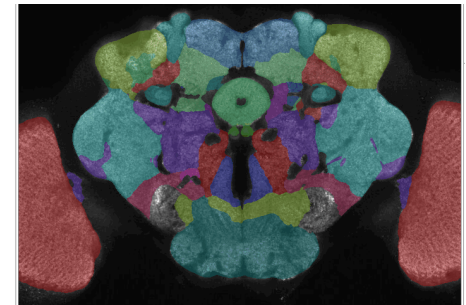


# Existential Rules in Ontological Modelling

**Markus Krötzsch**  
Dresden University of Technology

# Rules and Ontologies

- Ontologies
  - Structural models
  - Static
  - Output known at design time
- Rules
  - Computational models
  - Dynamic
  - Problem solving at runtime



1	1	3	4	4	3	1	
1	2	1	1	1	2	1	
3	1	4	3	2	1		
1	1	2					
4	3	1		1	1	2	
1	1			1	1	4	
2	2	1	1	1	2	1	

# Rules in Computation

- Logic programming
  - PROLOG, Answer Set Programming
- Data access and query languages
  - Datalog
- Information integration and data exchange
  - Database Dependencies and Constraints
- Derivation
  - Deduction Rules, Production Rules

# Two Unknowns

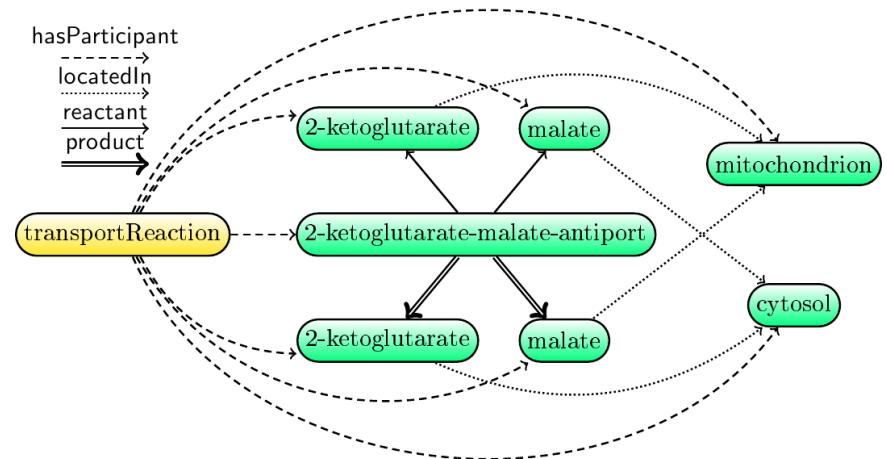
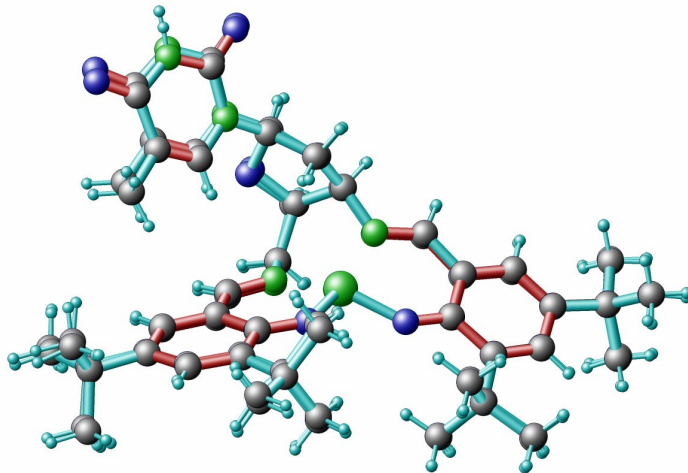
- Unknown Unknowns
  - Do we allow our specification to be incomplete?
  - Open World vs. Closed World
  - Entailment vs. Model Checking
- Known Unknowns
  - Do we allow “anonymous” elements?
  - Nulls (bnodes, existential quantifiers) vs. constants
  - No Unique Name Assumption on Nulls

# Rules and the Unknown

- Closed World & Unique Names
  - PROLOG
  - Datalog
  - ...
- Open World and Anonymous Individuals:
  - Tuple Generating Dependencies
  - ...

# Using Rules to Model Ontologies

- Expressive advantage:
  - Rules can express complex relational structures
  - Natural compatibility with conjunctive queries
  - Well-understood non-monotonic semantics



# Existential Rules

- Formulae of the form:

$$\forall \mathbf{x}. B1 \wedge \dots \wedge Bn \rightarrow \exists \mathbf{y}. H1 \wedge \dots \wedge Hk$$

where  $B1, \dots, Bn, H1, \dots, Hk$  are logical atoms  
(we often keep the  $\forall$  implicit)

- Also known as:
  - Tuple Generating Dependencies (TGDs)
  - Datalog+/-
  - $\forall\exists$ -rules

# Reasoning with Existential Rules

- Considered as first-order logic formulae
  - Open World Semantics
  - Sometimes Unique Name Assumption on constants
- Main practical entailment problem:
  - Conjunctive query answering
- Entailment is hard:
  - Even fact entailment is undecidable in general



# Existential Rules: Decidable Fragments

- Approach 1: Acyclicity
  - Limit amount of derived Nulls
  - Finite least models
- Approach 2: Guardedness
  - Limit non-local interactions in derivation
  - Tree-like least models
- Approach 3: Boundedness
  - Limit recursion to allow full expansion of rule sets
  - Rules can be rewritten to conjunctive queries

# Nonmonotonic Existential Rules

- Formulae of the form:

$$\forall \mathbf{x}. B1 \wedge \dots \wedge Bn \wedge \mathbf{not} C1 \wedge \dots \wedge \mathbf{not} Cm \rightarrow \exists \mathbf{y}. H1 \wedge \dots \wedge Hk$$

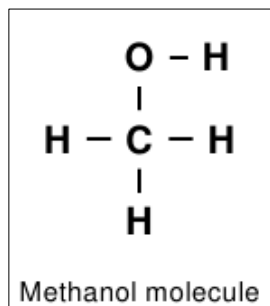
where  $B1, \dots, Bn, C1, \dots, Cm, H1, \dots, Hk$  are logical atoms  
(we often keep the  $\forall$  implicit)

- Possible Semantics of Negation:
  - Stable Model Semantics (as in ASP)
  - Well-Founded Semantics

# Nonmonotonic Existential Rules: Example

[Magka, K, Horrocks: Computing Stable Models for Nonmonotonic Existential Rules. IJCAI'13]

- Modelling molecular structures with rules:



$$\text{methanol}(x) \rightarrow \exists_{i=1}^6 y_i \cdot \bigwedge_{i=1}^6 \text{hasAtom}(x, y_i) \wedge c(y_1) \wedge o(y_2) \wedge \bigwedge_{i=3}^6 h(y_i) \wedge \bigwedge_{i=2}^5 \text{bond}(y_1, y_i) \wedge \text{bond}(y_2, y_6)$$

- Nonmonotonic negation used to control structure:

$$\bigwedge_{i=1}^3 \text{hasAtom}(x, z_i) \wedge \dots \wedge \text{bond}(z_2, z_3) \wedge \text{not } g_h(z_1) \wedge \text{not } g_h(z_2) \wedge \text{not } g_h(z_3) \rightarrow \text{organicHydroxy}(x) \wedge r_h(x)$$

$$\text{organicHydroxy}(x) \wedge \text{not } r_h(x) \rightarrow \exists_{i=1}^3 y_i \cdot \bigwedge_{i=1}^3 \text{hasAtom}(x, y_i) \wedge \dots \wedge \text{bond}(y_2, y_3) \wedge \bigwedge_{i=1}^3 g_h(y_i)$$

# Nonmonotonic Existential Rules: Experiments

[Magka, K, Horrocks: Computing Stable Models for Nonmonotonic Existential Rules. IJCAI'13]

- Data modelled in rules:
  - 500 molecule models (from ChEBI database)
  - 30 functional groups
  - 50 chemical classes→ 78,957 rules in total
- Query answering for classifying molecules:
  - DLV (Answer Set Solver): time out after 5min
  - Static analysis to partition rules + DLV: 13.5 sec→ 8,639 subsumptions of chemical classes

# Conclusions

- Rules can be used for ontological modelling
- Existential quantifiers to derive new structures (Nulls)
- Closed world? Open World? Mix and match!
  
- TODOs:
  - No unified syntax or even file format (other than RIF)
  - No tool and libraries to process and represent rules
  - No tools to model with rules
  - No reasoners optimized for ontological needs