EXTENDING LOGIC PROGRAMMING FOR LIFE SCIENCES APPLICATIONS

Despoina Magka

Department of Computer Science, University of Oxford

November 16, 2012
Bioinformatics and Semantic Technologies

- Life sciences data deluge
Bioinformatics and Semantic Technologies

- Life sciences data deluge
- Hierarchical *organisation* of biochemical knowledge
Bioinformatics and Semantic Technologies

- Life sciences data deluge
- Hierarchical organisation of biochemical knowledge
BIOINFORMATICS AND SEMANTIC TECHNOLOGIES

- Life sciences data deluge
- Hierarchical organisation of biochemical knowledge
Bioinformatics and Semantic Technologies

- Life sciences data deluge
- Hierarchical organisation of biochemical knowledge

- Fast, automatic and repeatable classification driven by Semantic technologies
Bioinformatics and Semantic Technologies

- Life sciences data deluge
- Hierarchical organisation of biochemical knowledge
- Fast, automatic and repeatable classification driven by Semantic technologies
- Web Ontology Language, a W3C standard family of logic-based formalisms
Bioinformatics and Semantic Technologies

- Life sciences data deluge
- Hierarchical organisation of biochemical knowledge

Fast, automatic and repeatable classification driven by Semantic technologies

Web Ontology Language, a W3C standard family of logic-based formalisms

OWL bio- and chemo-ontologies widely adopted
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest

- Dictionary of molecules with taxonomical information

- Pharmaceutical design and study of biological pathways

- ChEBI is manually incremented

- Currently ~30,000 chemical entities, expands at 3,500/yr

- Existing chemical databases describe millions of molecules

- Speed up growth by automating chemical classification
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information

⇝ caffeine is a cyclic molecule
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information

⇝ serotonin is an organic molecule
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information

\[ \text{\~ascorbic acid is a carboxylic ester} \]
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
- Pharmaceutical design and study of biological pathways
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
- Pharmaceutical design and study of biological pathways

ChEBI is manually incremented
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
- Pharmaceutical design and study of biological pathways

- ChEBI is manually incremented
- Currently ~30,000 chemical entities, expands at 3,500/yr
The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
- Pharmaceutical design and study of biological pathways

- ChEBI is manually incremented
- Currently ~30,000 chemical entities, expands at 3,500/yr
- Existing chemical databases describe millions of molecules
THE ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
- Dictionary of molecules with taxonomical information
- Pharmaceutical design and study of biological pathways

- ChEBI is manually incremented
- Currently ~30,000 chemical entities, expands at 3,500/yr
- Existing chemical databases describe millions of molecules
- Speed up growth by automating chemical classification
**Expressivity Limitations of OWL**

1. At least one **tree-shaped model** for each consistent OWL ontology $\leadsto$ problematic representation of cycles
Expressivity Limitations of OWL

1. At least one tree-shaped model for each consistent OWL ontology \( \iff \) problematic representation of cycles

Example

\[
\begin{array}{c}
\text{C} \\
\text{C} \\
\text{C} \\
\text{C}
\end{array}
\]

Example

1. Is cyclobutane a cyclic molecule?
2. Is cyclobutane a hydrocarbon?

Required reasoning support

1. Is cyclobutane a cyclic molecule?
2. Is cyclobutane a hydrocarbon?
**Expressivity Limitations of OWL**

1. At least one *tree-shaped model* for each consistent OWL ontology $\Rightarrow$ problematic representation of *cycles*

**Example**

\[
\text{Cyclobutane} \sqsubseteq \exists (\equiv 4) \text{hasAtom}. (\text{Carbon} \sqcap \exists (\equiv 2) \text{hasBond}. \text{Carbon})
\]

```
  C — C
 /    /
 C — C — C — C
```

1. Is cyclobutane a cyclic molecule?
2. Is cyclobutane a hydrocarbon?
3. Required reasoning support
**Expressivity Limitations of OWL**

1. At least one tree-shaped model for each consistent OWL ontology $\implies$ problematic representation of cycles

**Example**

Cyclobutane $\sqsubseteq \exists(=4)\text{hasAtom.}(\text{Carbon} \sqcap \exists(=2)\text{hasBond.}\text{Carbon})$
Expressivity Limitations of OWL

1. At least one tree-shaped model for each consistent OWL ontology \(\leadsto\) problematic representation of cycles

Example

Cyclobutane \(\subseteq\) \(\exists(=4)\) hasAtom.(Carbon \(\cap\) \(\exists(=2)\) hasBond.Carbon)

[Diagram of Cyclobutane with tree-shaped model]

- OWL-based reasoning support
  1. Is cyclobutane a cyclic molecule? \(\times\)
EXPRESSIVITY LIMITATIONS OF OWL

1. At least one tree-shaped model for each consistent OWL ontology \( \leadsto \) problematic representation of cycles

2. No minimality condition on the models \( \leadsto \) hard to axiomatise classes based on the absence of attributes

**Example**

Cyclobutane \( \sqsubseteq \exists (= 4) \text{hasAtom.}(\text{Carbon} \sqcap \exists (= 2) \text{hasBond.}\text{Carbon}) \)

- OWL-based reasoning support
  - 1. Is cyclobutane a cyclic molecule? ✗
**Expressivity Limitations of OWL**

1. At least one **tree-shaped model** for each consistent OWL ontology \(\leadsto\) problematic representation of **cycles**

2. **No minimality** condition on the models \(\leadsto\) hard to axiomatise classes based on the **absence** of attributes

**Example**

Cyclobutane \(\sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond}.\text{Carbon})\)

- **Is cyclobutane a cyclic molecule?** \(\times\)

- **OWL-based reasoning support**

  1. **Is cyclobutane a cyclic molecule?** \(\times\)
Expressivity Limitations of OWL

1. At least one tree-shaped model for each consistent OWL ontology \( \leadsto \) problematic representation of cycles

2. No minimality condition on the models \( \leadsto \) hard to axiomatise classes based on the absence of attributes

Example

Cyclobutane \( \sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \cap \exists(=2)\text{hasBond}.\text{Carbon}) \)

- OWL-based reasoning support
  1. Is cyclobutane a cyclic molecule? \( \times \)
  2. Is cyclobutane a hydrocarbon? \( \times \)
Expressivity Limitations of OWL

1. At least one tree-shaped model for each consistent OWL ontology \(\leadsto\) problematic representation of cycles

2. No minimality condition on the models \(\leadsto\) hard to axiomatise classes based on the absence of attributes

**Example**

Cyclobutane \(\sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond.}\text{Carbon})\)

```
C — C
|   |   |
C — C
```

```
\begin{itemize}
  \item \text{C} = \text{Carbon}
  \item \text{Oxygen}
\end{itemize}
```
**Expressivity Limitations of OWL**

1. At least one tree-shaped model for each consistent OWL ontology $\iff$ problematic representation of cycles

2. No minimality condition on the models $\iff$ hard to axiomatise classes based on the absence of attributes

**Example**

Cyclobutane $\sqsubseteq \exists (= 4) \text{hasAtom.}(\text{Carbon} \sqcap \exists (= 2) \text{hasBond.} \text{Carbon})$

Required reasoning support

1. Is cyclobutane a cyclic molecule?
2. Is cyclobutane a hydrocarbon?
**Expressivity Limitations of OWL**

1. At least one **tree-shaped model** for each consistent OWL ontology $\leadsto$ **problematic representation of cycles**

2. **No minimality** condition on the models $\leadsto$ hard to axiomatise classes based on the **absence** of attributes

**Example**

Cyclobutane $\subseteq \exists (= 4) \text{hasAtom.} (\text{Carbon} \sqcap \exists (= 2) \text{hasBond.} \text{Carbon})$

- **Required reasoning support**
  - 1. Is cyclobutane a **cyclic molecule?** ✓
  - 2. Is cyclobutane a **hydrocarbon?** ✓
RESULTS OVERVIEW

1 Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2 Acyclicity conditions for existential rules that extend previously suggested criteria
   Model-faithful acyclicity: 2EXPTIME-complete to check
   Model-summarising acyclicity: EXPTIME-complete to check

3 Implementation that draws upon DLV and performs structure-based classification with a significant speedup

4 Evaluation over part of the manually curated ChEBI ontology revealed modelling errors
   Language for representing complex objects with a favourable performance/expressivity trade-off
RESULTS OVERVIEW

1. Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2. Acyclicity conditions for existential rules that extend previously suggested criteria
RESULTS OVERVIEW

1. Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2. Acyclicity conditions for existential rules that extend previously suggested criteria
   - Model-faithful acyclicity: $2\text{EXPTIME}$-complete to check
RESULTS OVERVIEW

1. Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2. Acyclicity conditions for existential rules that extend previously suggested criteria
   - Model-faithful acyclicity: 2EXPTIME-complete to check
   - Model-summarising acyclicity: EXPTIME-complete to check
**Results Overview**

1. **Expressive and decidable formalism for modelling structured domains:** Description Graphs Logic Programs

2. **Acyclicity** conditions for existential rules that extend previously suggested criteria
   - Model-faithful acyclicity: $\text{2EXPTIME}$-complete to check
   - Model-summarising acyclicity: $\text{EXPTIME}$-complete to check

3. **Implementation** that draws upon DLV and performs structure-based classification with a significant *speedup*
RESULTS OVERVIEW

1. Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2. Acyclicity conditions for existential rules that extend previously suggested criteria
   - Model-faithful acyclicity: 2EXPTIME-complete to check
   - Model-summarising acyclicity: EXPTIME-complete to check

3. Implementation that draws upon DLV and performs structure-based classification with a significant speedup

4. Evaluation over part of the manually curated ChEBI ontology revealed modelling errors
RESULTS OVERVIEW

1. Expressive and decidable formalism for modelling structured domains: Description Graphs Logic Programs

2. Acyclicity conditions for existential rules that extend previously suggested criteria
   - Model-faithful acyclicity: $2\text{EXPTIME}$-complete to check
   - Model-summarising acyclicity: $\text{EXPTIME}$-complete to check

3. Implementation that draws upon DLV and performs structure-based classification with a significant speedup

4. Evaluation over part of the manually curated ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off
Ascorbic acid is a cyclic polyatomic entity and a carboxylic ester.
Classifying Structured Objects

Ascorbic acid is a cyclic polyatomic entity and a carboxylic ester.
ClasSifying Structured Objects

\[
\text{ascorbicAcid}(x) \rightarrow \text{hasAtom}(x, f_1(x)) \land \ldots \land \text{hasAtom}(x, f_{13}(x)) \\
\text{hasAtom}(x, f_1(x)) \land \ldots \land \text{c}(f_7(x)) \land \ldots \land \\
\text{single}(f_1(x), f_7(x)) \land \text{double}(f_7(x), f_2(x)) \land \ldots
\]

Ascorbic acid is a cyclic polyatomic entity and a carboxylic ester.
Classifying Structured Objects

\[ \text{ascorbicAcid}(x) \rightarrow \text{hasAtom}(x, f_1(x)) \land \ldots \land \text{hasAtom}(x, f_{13}(x)) \]
\[ o(f_1(x)) \land \ldots \land c(f_7(x)) \land \ldots \land \]
\[ \text{single}(f_1(x), f_7(x)) \land \text{double}(f_7(x), f_2(x)) \land \ldots \]
\[ \text{hasAtom}(x, y_1) \land \text{hasAtom}(x, y_2) \land y_1 \neq y_2 \rightarrow \text{polyatomicEntity}(x) \]
\[ \land_{i=1}^{5} \text{hasAtom}(x, y_i) \land c(y_1) \land o(y_2) \land o(y_3) \land \]
\[ c(y_4) \land \text{horc}(y_5) \land \text{double}(y_1, y_2) \land \]
\[ \text{single}(y_1, y_3) \land \text{single}(y_3, y_4) \land \text{single}(y_1, y_5) \rightarrow \text{carboxylicEster}(x) \]
**Classifying Structured Objects**

**Input fact:** ascorbicAcid(a)

**Stable model:** ascorbicAcid(a), hasAtom(a, a_1^f) for 1 ≤ i ≤ 13, o(a_i^f) for 1 ≤ i ≤ 6, c(a_i^f) for 7 ≤ i ≤ 12, h(a_{13}^f), single(a_8^f, a_3^f), single(a_9^f, a_4^f), single(a_{12}^f, a_i^f) for i ∈ {5, 11}, single(a_{11}^f, a_6^f), single(a_{10}^f, a_i^f) for i ∈ {1, 9, 11, 13}, single(a_7^f, a_i^f) for i ∈ {1, 8}, double(a_2^f, a_7^f), double(a_8^f, a_9^f), horc(a_i^f) for 7 ≤ i ≤ 13, polyatomicEntity(a), carboxylicEster(a), cyclic(a)
Classifying Structured Objects

Input fact: ascorbicAcid(a)

Stable model: ascorbicAcid(a), hasAtom(a, a_i^f) for 1 ≤ i ≤ 13, o(a_i^f) for 1 ≤ i ≤ 6, c(a_i^f) for 7 ≤ i ≤ 12, h(a_{13}^f), single(a_8^f, a_3^f), single(a_9^f, a_4^f), single(a_{12}^f, a_i^f) for i ∈ {5, 11}, single(a_{11}^f, a_6^f), single(a_{10}^f, a_i^f) for i ∈ {1, 9, 11, 13}, single(a_7^f, a_i^f) for i ∈ {1, 8}, double(a_2^f, a_7^f), double(a_8^f, a_9^f), horc(a_i^f) for 7 ≤ i ≤ 13, polyatomicEntity(a), carboxylicEster(a), cyclic(a)

⇒ Ascorbic acid is a cyclic polyatomic entity and a carboxylic ester
Rules with function symbols in the head can axiomatise infinitely large structures.
Acycliclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures.
- Reasoning with unrestricted DGLP ontologies is undecidable.
Acyclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures
- Reasoning with unrestricted DGLP ontologies is undecidable
- Acyclicity checks are sufficient but not necessary conditions for chase termination
Acyclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures
- Reasoning with unrestricted DGLP ontologies is undecidable
- Acyclicity checks are sufficient but not necessary conditions for chase termination
- Model-faithful and model-summarising acyclicity (MFA and MSA): capture as generally as possible class of programs with models of finite size
Acyclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures.
- Reasoning with unrestricted DGLP ontologies is undecidable.
- Acyclicity checks are sufficient but *not* necessary conditions for chase termination.
- Model-faithful and model-summarising acyclicity (MFA and MSA): capture as generally as possible class of programs with models of finite size.
CyClicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures
- Reasoning with unrestricted DGLP ontologies is undecidable
- Acyclicity checks are sufficient but not necessary conditions for chase termination
- Model-faithful and model-summarising acyclicity (MFA and MSA): capture as generally as possible class of programs with models of finite size
- Cost for checking MFA and MSA
Acyclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures
- Reasoning with unrestricted DGLP ontologies is undecidable
- Acyclicity checks are sufficient but not necessary conditions for chase termination
- Model-faithful and model-summarising acyclicity (MFA and MSA): capture as generally as possible class of programs with models of finite size

- Cost for checking MFA and MSA

<table>
<thead>
<tr>
<th></th>
<th>bounded arity</th>
<th>no restriction</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFA</td>
<td>2EXPTIME-complete</td>
<td>2EXPTIME-complete</td>
</tr>
<tr>
<td>MSA</td>
<td>coNP-complete</td>
<td>EXPTIME-complete</td>
</tr>
</tbody>
</table>
Acyclicity Conditions

- Rules with function symbols in the head can axiomatise infinitely large structures
- Reasoning with unrestricted DGLP ontologies is undecidable
- Acyclicity checks are sufficient but not necessary conditions for chase termination
- Model-faithful and model-summarising acyclicity (MFA and MSA): capture as generally as possible class of programs with models of finite size
- Cost for checking MFA and MSA

<table>
<thead>
<tr>
<th></th>
<th>bounded arity</th>
<th>no restriction</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFA</td>
<td>2EXPTIME-complete</td>
<td>2EXPTIME-complete</td>
</tr>
<tr>
<td>MSA</td>
<td>coNP-complete</td>
<td>EXPTIME-complete</td>
</tr>
</tbody>
</table>

- Both subsume previously suggested polynomial conditions
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs

Quicker than other approaches:
- [Hastings et al., 2010] 140 molecules in 4 hours
- [Magka et al., 2012] 70 molecules in 450 secs

Subsumptions exposed by our prototype:
- Ascorbic acid is a polyatomic entity, a carboxylic ester and a cyclic molecule

Contradictory subclass relation from ChEBI:
- Ascorbic acid is asserted to be a carboxylic acid (release 95)
- Not listed among the subsumptions derived by our prototype
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- Quicker than other approaches:
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- **Quicker** than other approaches:
  - [Hastings et al., 2010] 140 molecules in 4 hours
  - [Magka et al., 2012] 70 molecules in 450 secs
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- **Quicker** than other approaches:
  - [Hastings et al., 2010] 140 molecules in 4 hours
  - [Magka et al., 2012] 70 molecules in 450 secs
- Subsumptions exposed by our prototype:
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- Quicker than other approaches:
  - [Hastings et al., 2010] 140 molecules in 4 hours
  - [Magka et al., 2012] 70 molecules in 450 secs
- Subsumptions exposed by our prototype:
  - ascorbic acid is a polyatomic entity, a carboxylic ester and a cyclic molecule
  - missing from the ChEBI OWL ontology
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- **Quicker** than other approaches:
  - [Hastings et al., 2010] 140 molecules in 4 hours
  - [Magka et al., 2012] 70 molecules in 450 secs
- Subsumptions exposed by our prototype:
  - ascorbic acid is a polyatomic entity, a carboxylic ester and a cyclic molecule
  - missing from the ChEBI OWL ontology
- **Contradictory** subclass relation from ChEBI:
IMPLEMENTATION

- Draws upon DLV, a deductive databases engine
- Evaluation with data extracted from ChEBI
- 500 molecules under 51 chemical classes in 40 secs
- Quicker than other approaches:
  - [Hastings et al., 2010] 140 molecules in 4 hours
  - [Magka et al., 2012] 70 molecules in 450 secs
- Subsumptions exposed by our prototype:
  - ascorbic acid is a polyatomic entity, a carboxylic ester and a cyclic molecule
  - missing from the ChEBI OWL ontology
- Contradictory subclass relation from ChEBI:
  - Ascorbic acid is asserted to be a carboxylic acid (release 95)
  - Not listed among the subsumptions derived by our prototype
CONCLUSIONS

Results

1. Expressive and decidable formalism for structured domains

Future directions

- SMILES-based surface syntax
- Detect subsumptions between classes
- Extensions with numerical datatypes
- Define a mapping of DGLPs to RDF

Thank you! Questions?!?
# Conclusions

## Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules

---

Future directions

- SMILES-based surface syntax
- Detect subsumptions between classes
- Extensions with numerical datatypes
- Define a mapping of DGLPs to RDF

Thank you! Questions?!?
**CONCLUSIONS**

- **Results**
  1. Expressive and decidable formalism for structured domains
  2. Novel acyclicity conditions for existential rules
  3. DLV-based implementation exhibits a significant speedup
CONCLUSIONS

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors
Conclusions

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off
Conclusions

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

Future directions

- SMILES-based surface syntax
**Conclusions**

- **Results**
  1. Expressive and decidable formalism for structured domains
  2. Novel acyclicity conditions for existential rules
  3. DLV-based implementation exhibits a significant speedup
  4. Evaluation over ChEBI ontology revealed modelling errors

  Language for representing complex objects with a favourable performance/expressivity trade-off

- **Future directions**
  - SMILES-based surface syntax

  \[ \land_{i=1}^{5} \text{hasAtom}(x, y_i) \land \text{c}(y_1) \land \text{o}(y_2) \land \text{o}(y_3) \land \text{c}(y_4) \land \text{double}(y_1, y_2) \land \text{single}(y_1, y_3) \land \text{single}(y_3, y_4) \land \text{single}(y_1, y_5) \rightarrow \text{carboxylicEster}(x) \]
CONCLUSIONS

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

Future directions

- SMILES-based surface syntax

  define carboxylicEster
  some hasAtom SMILES(C − O − C(＝ O) − *)
  end.
Conclusions

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

Future directions

- SMILES-based surface syntax
- Detect subsumptions between classes
**Conclusions**

- **Results**
  1. Expressive and decidable formalism for structured domains
  2. Novel *acyclicity* conditions for existential rules
  3. DLV-based *implementation* exhibits a significant speedup
  4. Evaluation over ChEBI ontology *revealed* modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

- **Future directions**
  - SMILES-based *surface syntax*
  - Detect subsumptions *between classes*
    - E.g., *Carboxylic ester* is an *organic molecular entity*
**Conclusions**

**Results**

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

**Future directions**

- SMILES-based surface syntax
- Detect subsumptions between classes
- Extensions with numerical datatypes

Thank you! Questions?!?
Conclusions

Results
1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

Future directions
- SMILES-based surface syntax
- Detect subsumptions between classes
- Extensions with numerical datatypes
- Define a mapping of DGLPs to RDF
CONCLUSIONS

Results

1. Expressive and decidable formalism for structured domains
2. Novel acyclicity conditions for existential rules
3. DLV-based implementation exhibits a significant speedup
4. Evaluation over ChEBI ontology revealed modelling errors

Language for representing complex objects with a favourable performance/expressivity trade-off

Future directions

- SMILES-based surface syntax
- Detect subsumptions between classes
- Extensions with numerical datatypes
- Define a mapping of DGLPs to RDF

Thank you! Questions?!?