TECHNISCHE UNIVERSITÄT DRESDEN

CPEC CENTER FOR PERSPICUOUS COMPUTING

Christian Alrabbaa, Stefan Borgwardt, Tom Friese, Patrick Koopmann, Julián Méndez, Alexej Popovič
Technische Universität Dresden, Center for Perspicuous Computing (CPEC)

Explaining Description Logic Entailments in Practice with EVEE and EVONNE

3rd Workshop on Explainable Logic-Based Knowledge Representation (XLoKR'22), 31st July, 2022

Description Logics and Ontologies

Description Logics

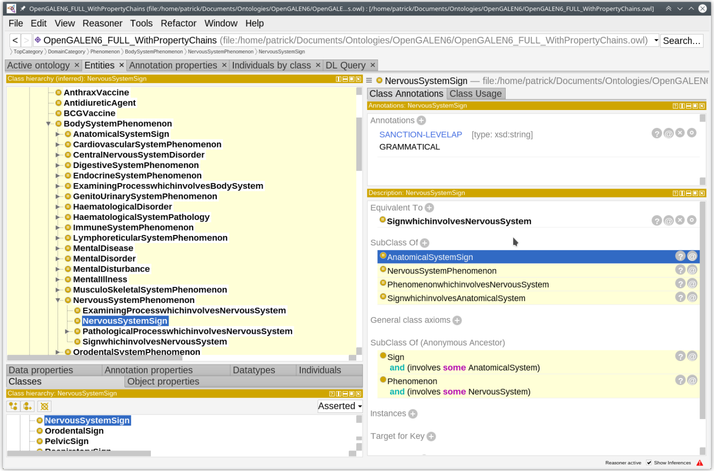
- Well-established formalism for specifying terminological knowledge in Ontologies
- Applications in biology, medicine, semantic web, and more
  - SNOMED CT: medical, over 300,000 concepts
  - BioPortal: repository of bio-medical ontologies, currently hosting 1,004 ontologies defining 14,685,604 terms
  - MOWLCorp: ontologies obtained by web-crawling, containing 20,996 ontologies

Description Logics and Ontologies

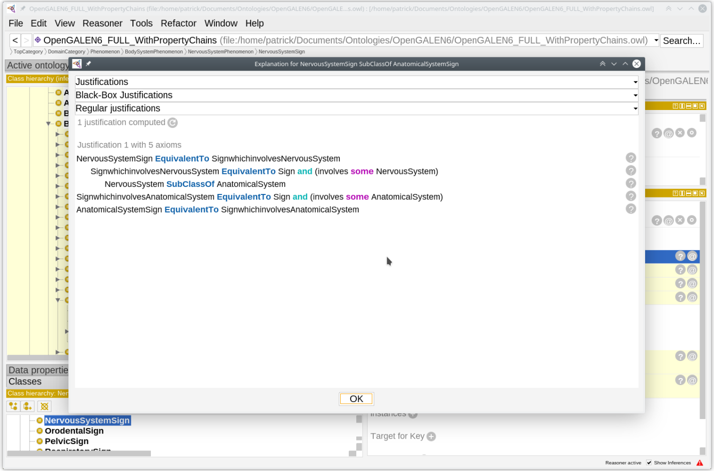
Description Logics

- Well-established formalism for specifying terminological knowledge in Ontologies
- Applications in biology, medicine, semantic web, and more
  - SNOMED CT: medical, over 300,000 concepts
  - BioPortal: repository of bio-medical ontologies, currently hosting 1,004 ontologies defining 14,685,604 terms
  - MOWLCorp: ontologies obtained by web-crawling, containing 20,996 ontologies
- With increasing complexity of the ontology, understanding entailments becomes both crucial and difficult
  - Requirement for tools to explain entailments

Current Explanation Tool of Choice: Justifications



Current Explanation Tool of Choice: Justifications



Justifications

Justifications: Minimal subsets entailing given statement

In practice often insufficient:

- can be large
- inferences often not obvious

Showing how to obtain the inference would be better

- simple reasoning steps leading to conclusion
- generally known as proof

A Proof

$$\begin{tikzcd}
(\mathsf{stefan}, \mathsf{xlokr22\text{-}9}) : \mathsf{presents} \arrow[d] & & \mathsf{xlokr22\text{-}9} : \mathsf{ScientificTalk} \arrow[dll] \\
\mathsf{stefan} : \exists \mathsf{presents}.\mathsf{ScientificTalk} \arrow[d] & & \exists \mathsf{presents}.\mathsf{ScientificTalk} \sqsubseteq \mathsf{Researcher} \arrow[dll] \\
\mathsf{stefan} : \mathsf{Researcher} & &
\end{tikzcd}$$

Meet the Family

- *data structures*
- *algorithms*
- *proof generation*

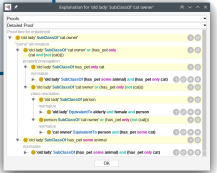
Evee-libs

Meet the Family

easy integration in Protégé

- *data structures*
- *algorithms*
- *proof generation*

Evee-libs → Evee-protégé



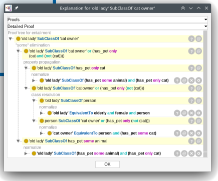
Meet the Family

easy integration in Protégé

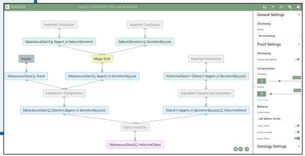
- *data structures*
- *algorithms*
- *proof generation*

Evee-libs → Evee-protégé

Evee-libs → Evonne



advanced web application



Proof Generation in Evee-libs

Three basic methods:

- Proofs based on ELK
- Elimination proofs based on LETHE and FAME
- Detailed proofs based on LETHE

In addition:

- Optimization criteria
  - Proof with lowest (weighted) size or depth
- Hide inferences of known terms

1. Proofs Based on ELK

ELK is the state-of-the-art reasoner for the lightweight logic $\mathcal{EL}$

- good performance on large-scale ontologies
- reasoning based on inference rules

$\mathcal{EL}$ inference rules

$$\mathsf{R}_0\ \frac{}{C \sqsubseteq C} \qquad \mathsf{R}_\top\ \frac{}{C \sqsubseteq \top} \qquad \mathsf{R}_\sqsubseteq\ \frac{C \sqsubseteq D \quad D \sqsubseteq E}{C \sqsubseteq E} \qquad \mathsf{R}^-_{\sqcap,1}\ \frac{C \sqsubseteq D \sqcap E}{C \sqsubseteq D} \qquad \mathsf{R}_\bot\ \frac{C \sqsubseteq \exists r.D \quad D \sqsubseteq \bot}{C \sqsubseteq \bot}$$

$$\mathsf{R}^-_{\sqcap,2}\ \frac{C \sqsubseteq D \sqcap E}{C \sqsubseteq E} \qquad \mathsf{R}^+_\sqcap\ \frac{C \sqsubseteq D \quad C \sqsubseteq E}{C \sqsubseteq D \sqcap E} \qquad \mathsf{R}_\exists\ \frac{C \sqsubseteq \exists r.D \quad D \sqsubseteq E}{C \sqsubseteq \exists r.E} \qquad \mathsf{R}_\mathcal{H}\ \frac{C \sqsubseteq \exists r.D \quad r \sqsubseteq s}{C \sqsubseteq \exists s.D}$$

- existing library and Protégé plugin (Kazakov, Klinov, Stupnikov 2017)
- we add support for optimized proofs (size, depth)

2. Elimination Proofs

- Support expressive DLs, no need of explicit inference rules
- Based on one type of inference:

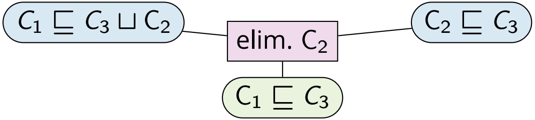
X-Elimination

$$\frac{\alpha_1, \quad \ldots, \quad \alpha_n}{\beta} \text{ eliminate } X$$

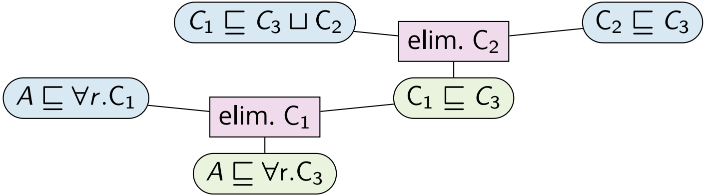
where $\alpha_1, \ldots, \alpha_n \models \beta$,
and X does not occur in β

- Eliminate symbols one by one, until only the conclusion is left
- Computed using forgetting tools LETHE and FAME

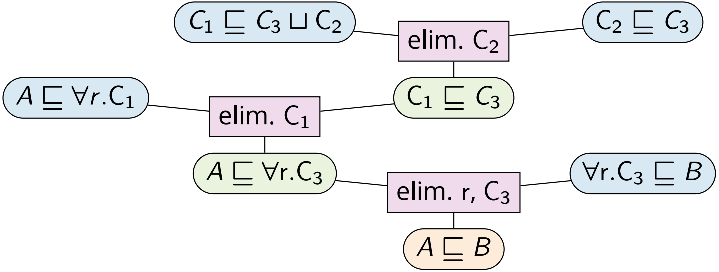
Example of an Elimination Proof



Example of an Elimination Proof



Example of an Elimination Proof



2. Elimination Proofs

The order in which we eliminate symbols affects the proof size!

How to choose a good order?

2. Elimination Proofs

The order in which we eliminate symbols affects the proof size!

How to choose a good order?

We implemented three strategies:

1. Use heuristics to pick next symbol (LPAR-20/XLoKR-20)
2. Use best-first search together with optimization criterion (IJCAR-22)
   - minimize number of eliminated names
   - optimize for given criterion, e.g. (weighted) tree size

3. Detailed Proofs using LETHE

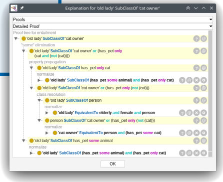
- Elimination proofs give a high-level perspective on inferences
- Detailed proofs based on LETHE's inference rules
- However, several challenges needed to be solved:
  - LETHE uses a normal form, which we have to denormalize
  - optimizations need to be deactivated
  - some inferences are performed indirectly through the algorithm, which need to be translated to rule inferences

Meet the Family

easy integration in Protégé

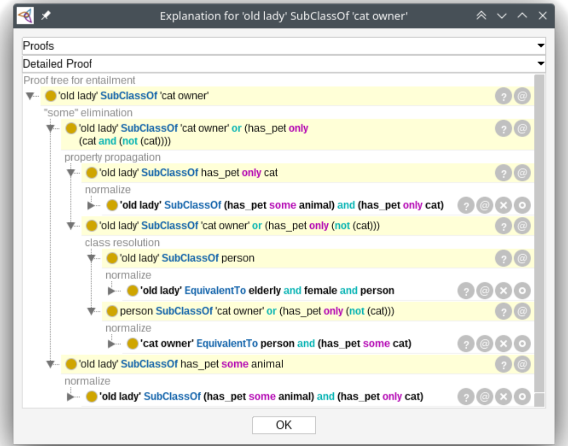
- *data structures*
- *algorithms*
- *proof generation*

Evee-libs → Evee-protégé



Evee-protege

- Protégé: standard editor for OWL ontologies
- Easy installation using plugin infrastructure



User Study

We performed a small user study on EVEE-protege

- Participants: 10 DL experts
- 5 Tasks + Questions
- Compare different proof methods

First conclusions:

- Preferred method is subjective
- Proof navigation in Protégé sometimes limited

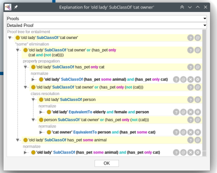
Meet the Family

easy integration in Protégé

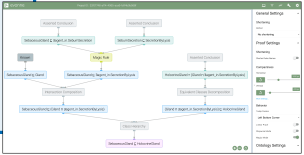
- *data structures*
- *algorithms*
- *proof generation*

Evee-libs → Evee-protégé

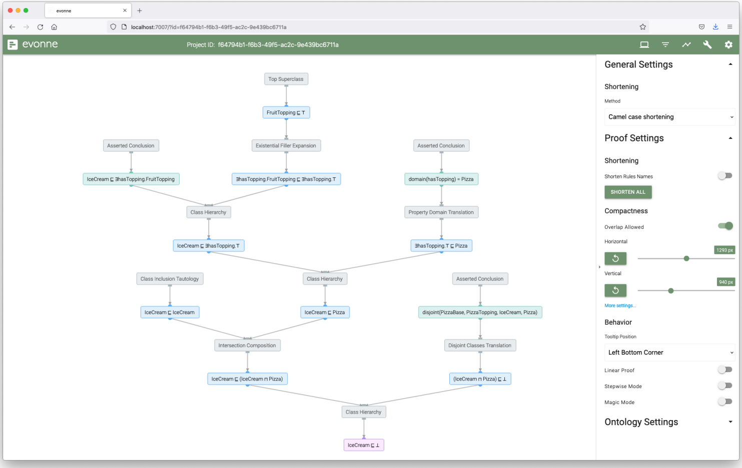
Evee-libs → Evonne



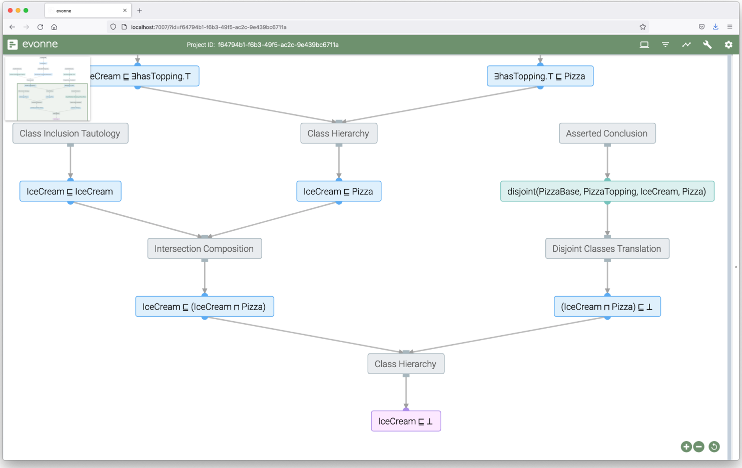
advanced web application



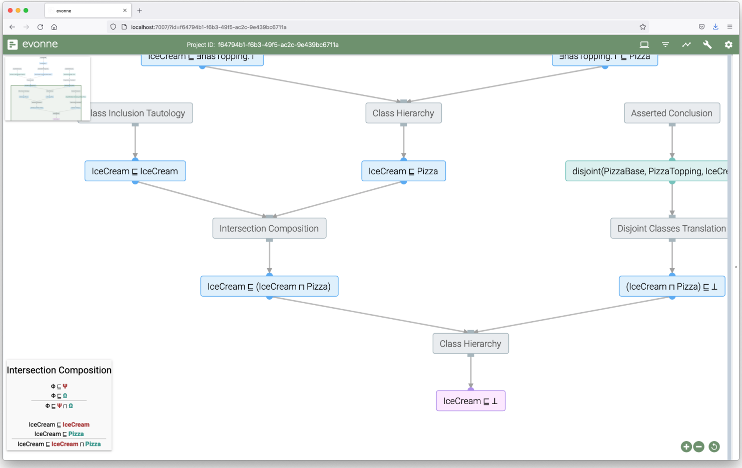
Advanced Proof Navigation with Evonne



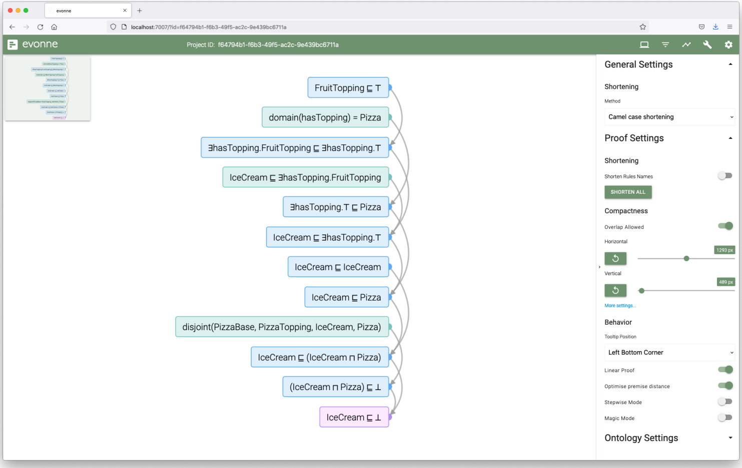
Advanced Proof Navigation with Evonne



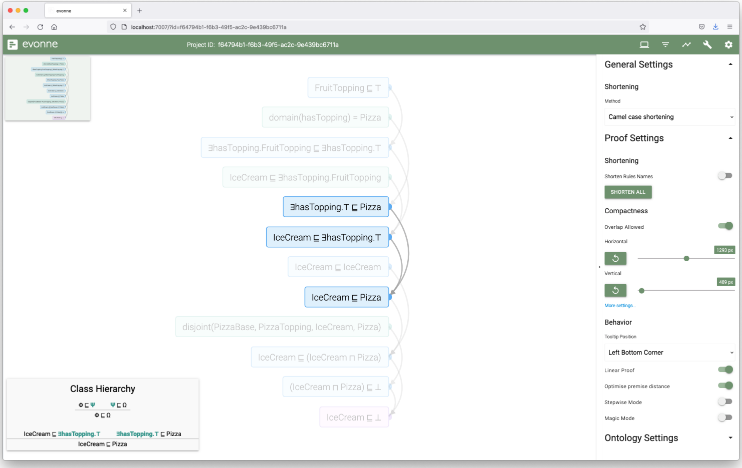
Advanced Proof Navigation with Evonne



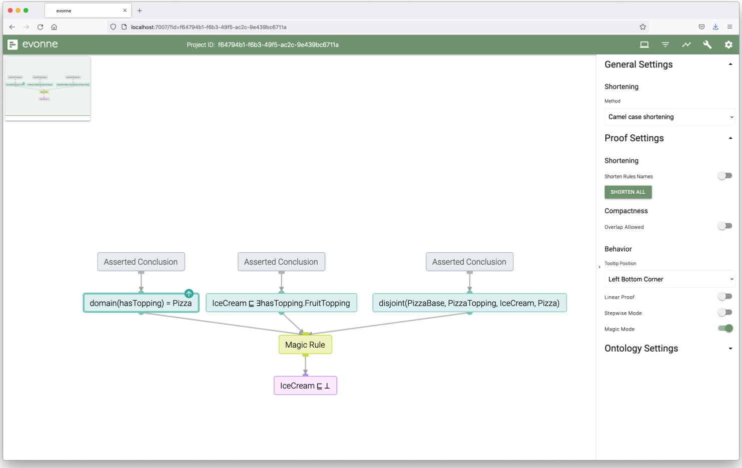
Advanced Proof Navigation with Evonne



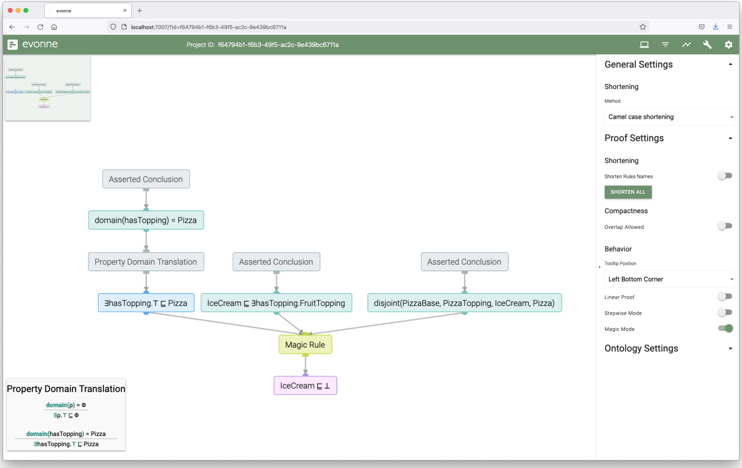
Advanced Proof Navigation with Evonne



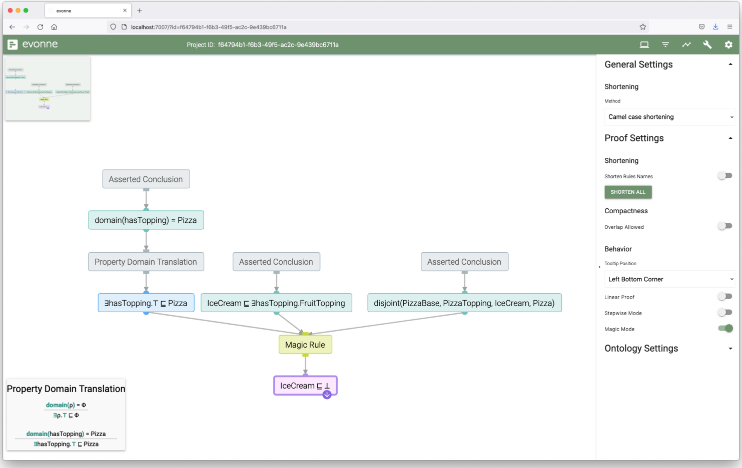
Advanced Proof Navigation with Evonne



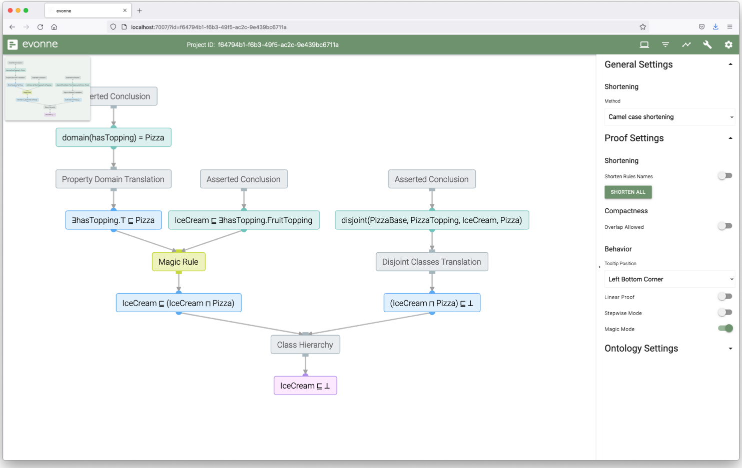
Advanced Proof Navigation with Evonne



Advanced Proof Navigation with Evonne



Advanced Proof Navigation with Evonne



Conclusion

Proofs to explain DL entailments

- Library EVEE-libs used by frontends EVEE-protege and EVONNE
- ELK proofs, elimination proofs, detailed LETHE proofs
- Optimization w.r.t. various measures

Conclusion

Proofs to explain DL entailments

- Library EVEE-libs used by frontends EVEE-protege and EVONNE
- ELK proofs, elimination proofs, detailed LETHE proofs
- Optimization w.r.t. various measures

Future work:

- User study of EVONNE
- Explain also non-entailments using interpretations or abduction

Conclusion

Proofs to explain DL entailments

- Library Evee-libs used by frontends Evee-protege and Evonne
- Elk proofs, elimination proofs, detailed Lethe proofs
- Optimization w.r.t. various measures

Future work:

- User study of Evonne
- Explain also non-entailments using interpretations or abduction

Try it out!

- You can try Evonne online
- Demo at DL (9.8., 16:00-16:30; 10.8., 10:30-11:00)

Conclusion

Proofs to explain DL entailments

- Library Evee-libs used by frontends Evee-protege and Evonne
- Elk proofs, elimination proofs, detailed Lethe proofs
- Optimization w.r.t. various measures

Future work:

- User study of Evonne
- Explain also non-entailments using interpretations or abduction

Try it out!

- You can try Evonne online
- Demo at DL (9.8., 16:00-16:30; 10.8., 10:30-11:00)

Thank you!