

[Full paper available at <http://korrekt.org/page/ELP>]

ELP

Tractable Rules for OWL 2

Markus Krötzsch

Sebastian Rudolph

Pascal Hitzler

Institute AIFB
Universität Karlsruhe (TH)
Germany

ISWC 2008

Copyright Markus Krötzsch

Terms of Licensing

Creative Commons
Attribution

<http://creativecommons.org/licenses/by/2.0/de/deed.en>

All redistributions and reuses of this work
must include the following:

“Based on slides by Markus Krötzsch, <http://korrekt.org>”

Image copyrights

Icons by <http://everaldo.com>, released under the terms of GPL

Where do we want to get to?



Where do we want to get to?

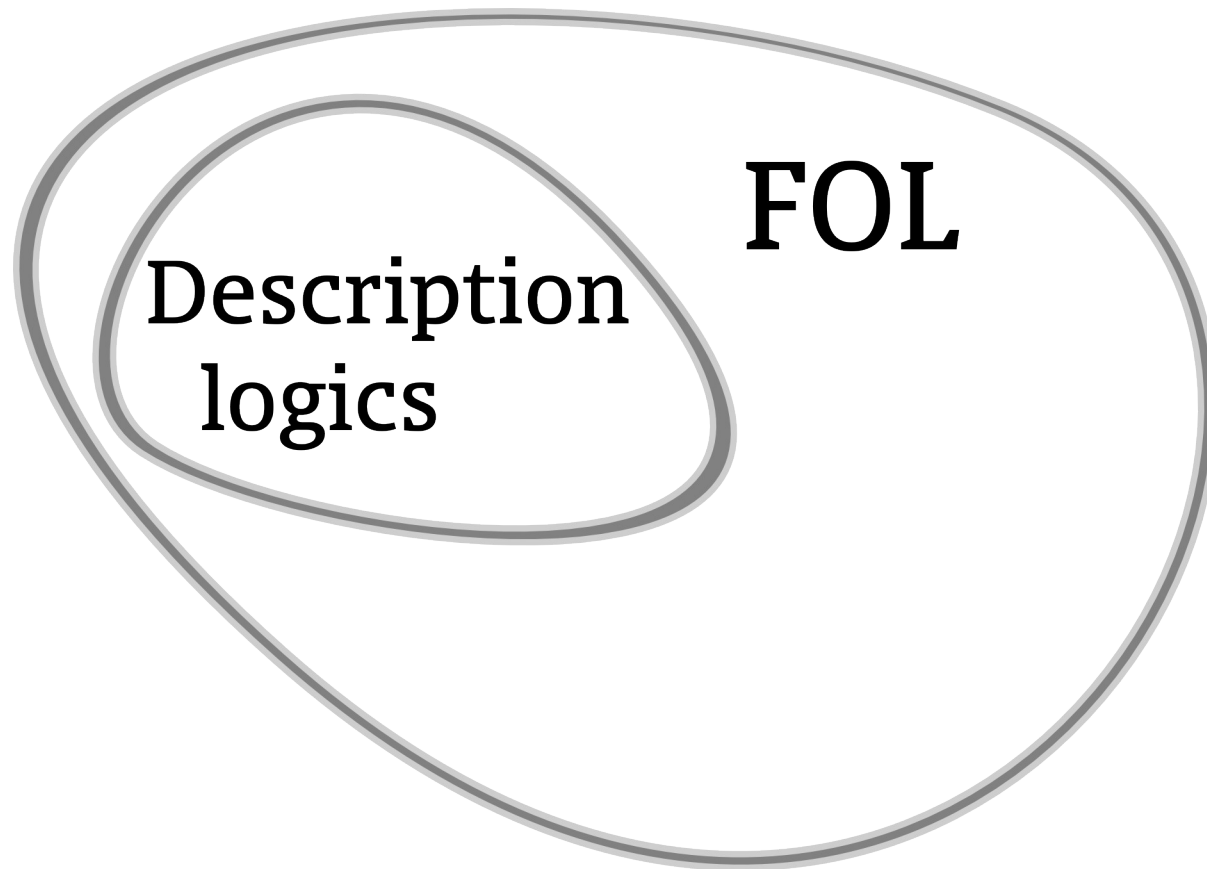


Semantic Web language that is ...

- focussed on data,
- rule-oriented,
- able to express schema knowledge,
- easy to implement,
- of polynomial worst-case complexity,
- compatible with OWL.

OWL and Description Logics

OWL and Description Logics



OWL and Description Logics

“Sebastian ordered some Thai
curry.”

sebastian: \exists orderedDish.ThaiCurry

*$\exists x. \text{orderedDish}(\text{sebastian}, x)$
 $\wedge \text{ThaiCurry}(x)$*

OWL and Description Logics

“Everything ordered as a dish is actually a dish.”

$\top \sqsubseteq \forall \text{orderedDish.Dish}$

$\forall x. \forall y. \text{orderedDish}(x, y) \rightarrow \text{Dish}(y)$

OWL and Description Logics

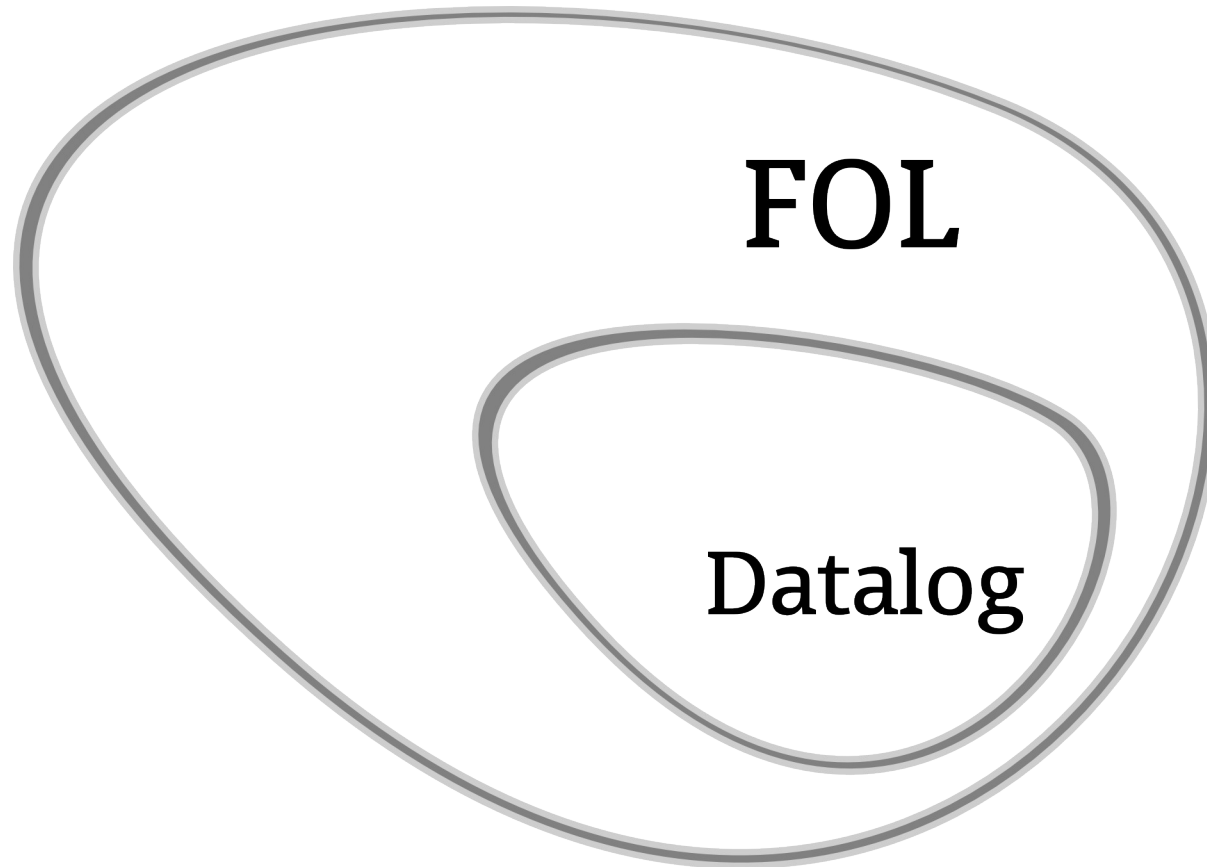
“Every Thai curry dish contains peanut oil.”

ThaiCurry \sqsubseteq \exists *contains*.{*peanutOil*}

$\forall x. \textit{ThaiCurry}(x)$
 $\rightarrow \textit{contains}(x, \textit{peanutOil})$

Rules in First-Order Logic

Rules in First-Order Logic



Rules in First-Order Logic

“Nut allergics dislike nut products.”

$$\begin{aligned} \text{NutAllergic}(x) \wedge \text{NutProduct}(y) \\ \rightarrow \text{dislikes}(x,y) \end{aligned}$$

Rules in First-Order Logic

“People who order a dish they dislike are unhappy.”

$$\text{orderedDish}(x,y) \wedge \text{dislikes}(x,y) \rightarrow \text{Unhappy}(x)$$

Rules in First-Order Logic

“If someone dislikes an ingredient of a dish, she will also dislike the dish.”

$$\textit{dislikes}(x,z) \wedge \textit{Dish}(y) \wedge \textit{contains}(y,z) \rightarrow \textit{dislikes}(x,y)$$

Rules in First-Order Logic

“Sebastian is a nut allergic, and peanut oil is a nut product.”

→ *NutAllergic(sebastian)*

→ *NutProduct(peanutOil)*

**Can we combine
datalog rules and DL axioms?**

“

ThaiCurry $\sqsubseteq \exists \text{contains}.\{\text{peanutOil}\}$

\top $\sqsubseteq \forall \text{orderedDish.Dish}$

sebastian: $\exists \text{orderedDish.ThaiCurry}$

NutAllergic(*x*) \wedge *NutProduct*(*y*) \rightarrow *dislikes*(*x*,*y*)

orderedDish(*x*,*y*) \wedge *dislikes*(*x*,*y*) \rightarrow *Unhappy*(*x*)

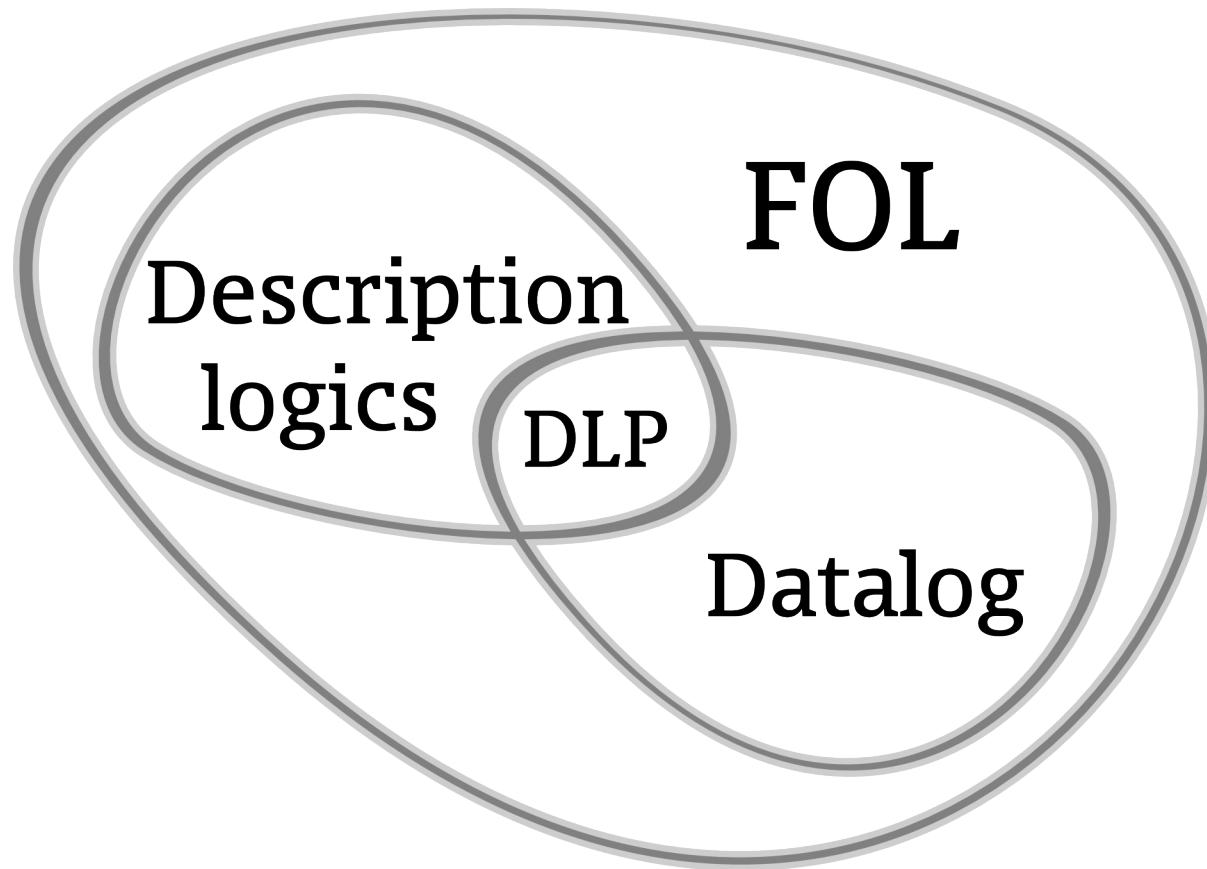
dislikes(*x*,*z*) \wedge *Dish*(*y*) \wedge *contains*(*y*,*z*) \rightarrow *dislikes*(*x*,*y*)

\rightarrow *NutAllergic*(*sebastian*)

\rightarrow *NutProduct*(*peanutOil*)

”

Combining OWL and Rules



DLP: “OWL \cap datalog”

ThaiCurry $\sqsubseteq \exists \text{contains}.\{\text{peanutOil}\}$ ✓

\top $\sqsubseteq \forall \text{orderedDish.Dish}$ ✓

sebastian: $\exists \text{orderedDish.ThaiCurry}$ ✗

NutAllergic(*x*) \wedge *NutProduct*(*y*) \rightarrow *dislikes*(*x*,*y*) ✗

orderedDish(*x*,*y*) \wedge *dislikes*(*x*,*y*) \rightarrow *Unhappy*(*x*) ✗

dislikes(*x*,*z*) \wedge *Dish*(*y*) \wedge *contains*(*y*,*z*) \rightarrow *dislikes*(*x*,*y*) ✗

\rightarrow *NutAllergic*(*sebastian*) ✓

\rightarrow *NutProduct*(*peanutOil*) ✓

SWRL: “OWL \cup datalog”

ThaiCurry $\sqsubseteq \exists \text{contains}.\{\text{peanutOil}\}$ ✓

\top $\sqsubseteq \forall \text{orderedDish}.\text{Dish}$ ✓

sebastian: $\exists \text{orderedDish}.\text{ThaiCurry}$ ✓

NutAllergic(x) \wedge *NutProduct*(y) \rightarrow *dislikes*(x,y) ✓

orderedDish(x,y) \wedge *dislikes*(x,y) \rightarrow *Unhappy*(x) ✓

dislikes(x,z) \wedge *Dish*(y) \wedge *contains*(y,z) \rightarrow *dislikes*(x,y) ✓

\rightarrow *NutAllergic*(*sebastian*) ✓

\rightarrow *NutProduct*(*peanutOil*) ✓



SWRL is undecidable.

DL-safe Rules

DL-safe Rules

**Restrict rules to apply only to
named individuals.**

DL-safe Rules

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$ ✓

$\top \sqsubseteq \forall orderedDish.Dish$ ✓

$sebastian: \exists orderedDish.ThaiCurry$ ✓

$NutAllergic(x) \wedge NutProduct(y) \rightarrow dislikes(x,y)$ 🔒

$orderedDish(x,y) \wedge dislikes(x,y) \rightarrow Unhappy(x)$ 🔒

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow dislikes(x,y)$ 🔒

$\rightarrow NutAllergic(sebastian)$ ✓

$\rightarrow NutProduct(peanutOil)$ ✓

DL Rules

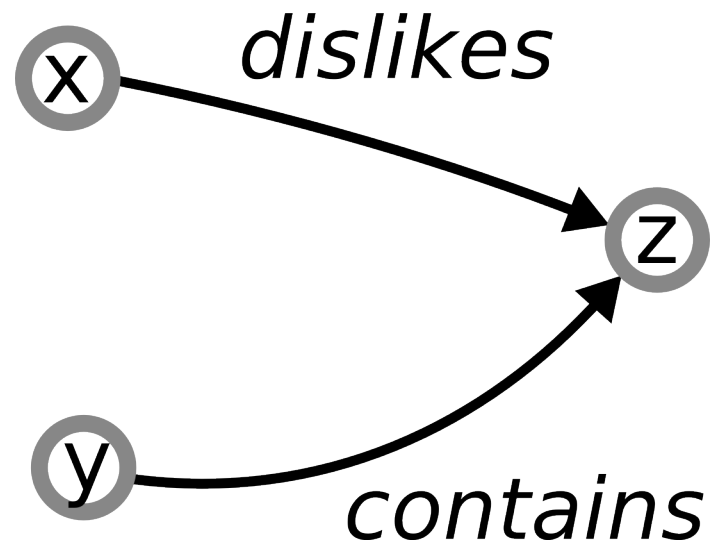
DL Rules

Restrict to rules that could
(indirectly) be encoded with DL
anyway.*

*) rules with “tree-shaped” bodies

DL Rules*

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow$
 $dislikes(x,y)$



*) rules with “tree-shaped” bodies

DL Rules

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$ ✓

$\top \sqsubseteq \forall orderedDish.Dish$ ✓

$sebastian: \exists orderedDish.ThaiCurry$ ✓

$NutAllergic(x) \wedge NutProduct(y) \rightarrow dislikes(x,y)$ ✓

$orderedDish(x,y) \wedge dislikes(x,y) \rightarrow Unhappy(x)$ ✗

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow dislikes(x,y)$ ✓

$\rightarrow NutAllergic(sebastian)$ ✓

$\rightarrow NutProduct(peanutOil)$ ✓

DL-safe rules + DL Rules

DL-safe rules + DL Rules

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$ ✓

$\top \sqsubseteq \forall orderedDish.Dish$ ✓

$sebastian: \exists orderedDish.ThaiCurry$ ✓

$NutAllergic(x) \wedge NutProduct(y) \rightarrow dislikes(x,y)$ ✓

$orderedDish(x,y) \wedge dislikes(x,y) \rightarrow Unhappy(x)$ 🚫

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow dislikes(x,y)$ ✓

$\rightarrow NutAllergic(sebastian)$ ✓

$\rightarrow NutProduct(peanutOil)$ ✓



DL-safe rules + DL Rules



Desired conclusion does not follow

It is still computationally expensive

DL-safe rules: ExpTime

DL Rules: like DL, i.e. NExpTime for OWL DL

Tractable Profiles in OWL 2

Tractable Profiles in OWL 2

OWL RL: *Horn logic* fragment,
similar to DLP, no existentials

OWL EL: includes existentials, based
on DL *EL++*

Regaining Tractability: OWL 2 EL

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$ ✓

$\top \sqsubseteq \forall orderedDish.Dish$ ✓

$sebastian: \exists orderedDish.ThaiCurry$ ✓

$NutAllergic(x) \wedge NutProduct(y) \rightarrow dislikes(x,y)$ ✗

$orderedDish(x,y) \wedge dislikes(x,y) \rightarrow Unhappy(x)$ ✗

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow dislikes(x,y)$ ✗

$\rightarrow NutAllergic(sebastian)$ ✓

$\rightarrow NutProduct(peanutOil)$ ✓

Regaining Tractability: OWL 2 RL

ThaiCurry $\sqsubseteq \exists \text{contains}.\{\text{peanutOil}\}$ ✓

\top $\sqsubseteq \forall \text{orderedDish}.\text{Dish}$ ✓

sebastian: $\exists \text{orderedDish}.\text{ThaiCurry}$ ✗

NutAllergic(*x*) \wedge *NutProduct*(*y*) \rightarrow *dislikes*(*x*,*y*) ✗

orderedDish(*x*,*y*) \wedge *dislikes*(*x*,*y*) \rightarrow *Unhappy*(*x*) ✗

dislikes(*x*,*z*) \wedge *Dish*(*y*) \wedge *contains*(*y*,*z*) \rightarrow *dislikes*(*x*,*y*) ✗

\rightarrow *NutAllergic*(*sebastian*) ✓

\rightarrow *NutProduct*(*peanutOil*) ✓

OWL EL: PTime complete



OWL RL: PTime complete

OWL EL: PTime complete



OWL RL: PTime complete



OWL EL+RL:

N2ExpTime complete

Bringing it all together: ELP

Bringing it all together: ELP

DL Rules for OWL EL
+
Conjunctions of Roles*
+
DL-safe *variables*

*) those roles must be “simple”



Theorem

**Inferencing in ELP is PTime
complete.**

Bringing it all together: ELP

$ThaiCurry \sqsubseteq \exists contains.\{peanutOil\}$ ✓

$\top \sqsubseteq \forall orderedDish.Dish$ ✓

$sebastian: \exists orderedDish.ThaiCurry$ ✓

$NutAllergic(x) \wedge NutProduct(y) \rightarrow dislikes(x,y)$ ✓

$orderedDish(x,y) \wedge dislikes(x,y) \rightarrow Unhappy(x)$ ✓

$dislikes(x,z) \wedge Dish(y) \wedge contains(y,z) \rightarrow dislikes(x,y)$ ✓

$\rightarrow NutAllergic(sebastian)$ ✓

$\rightarrow NutProduct(peanutOil)$ ✓

Bringing it all together: ELP

→ *Unhappy(sebastian)*



Note

**ELP supports inferencing in
OWL EL and OWL RL.**

Understanding DL-safety

$\text{ThaiCurry} \sqsubseteq \exists \text{contains.} \mathbf{FishProduct}$

$\top \sqsubseteq \forall \text{orderedDish.Dish}$

markus: $\exists \text{orderedDish.ThaiCurry}$

$\mathbf{Vegetarian}(x) \wedge \mathbf{FishProduct}(y) \rightarrow \text{dislikes}(x,y)$

$\text{orderedDish}(x,y) \wedge \text{dislikes}(x,y) \rightarrow \text{Unhappy}(x)$

$\text{dislikes}(x,z) \wedge \text{Dish}(y) \wedge \text{contains}(y,z) \rightarrow \text{dislikes}(x,y)$

$\rightarrow \mathbf{Vegetarian}(\text{markus})$

Understanding DL-safety

Unhappy(markus)

cannot be concluded

Towards Implementation



Theorem

Inferencing in ELP can be reduced in linear time to inferencing in 3-variable datalog.

Reasoning through Datalog

- Transformation to datalog is completely syntactic.
- Each axiom/rule can be transformed individually.
- Datalog engines can be used as blackbox.
- Instance and subsumption checking directly in datalog.



Summary



ELP: DL-based tractable rule language

- Almost completely expressible in OWL 2
- Support for OWL EL and OWL RL
- Linear-time conversion to 3-var datalog
→ simple implementation strategy

Happy(markus)

Happy(sebastian)

Happy(pascal)



[Full paper available at <http://korrekt.org/page/ELP>]