LogiKEy — Logical Pluralism via Logic Embeddings in HOL

Christoph Benzmüller

U Bamberg | FU Berlin

Uni Luxembourg, 31.3.2026

Logical/Mathematical Imperialism

Choose some logical foundation and don't question them ever again.

For example, commit to a priori assumptions like:

$\exists x\,.\,x = x$	existence of objects, non-empty types
$\exists x\,.\,x = 1/0$	terms always denote, undefinedness exists
$\forall \varphi\,.\,\varphi \vee \neg\varphi$	law of excluded middle, classical logic
$\forall \varphi\psi\,.\,(\varphi \equiv \psi) \supset (\varphi = \psi)$	boolean extensionality
$\forall \varphi\psi\,.\,(\forall x\,.\,\varphi x = \psi x) \supset (\varphi = \psi)$	functional extensionality
If $H \vdash \varphi$ then $H, \psi \vdash \varphi$	monotonicity
$\ldots$	

Such choices are typically made in maths formalisation projects (like MathLib etc.)

They might be completely innaceptible for formalisations in philosophy, law, …

Logical/Mathematical Pluralism

Make logical foundations flexible and negotiable.

Support reflection on them within proof assistants.

$\exists x \,.\, x = x$

$\exists x \,.\, x = 1/0$

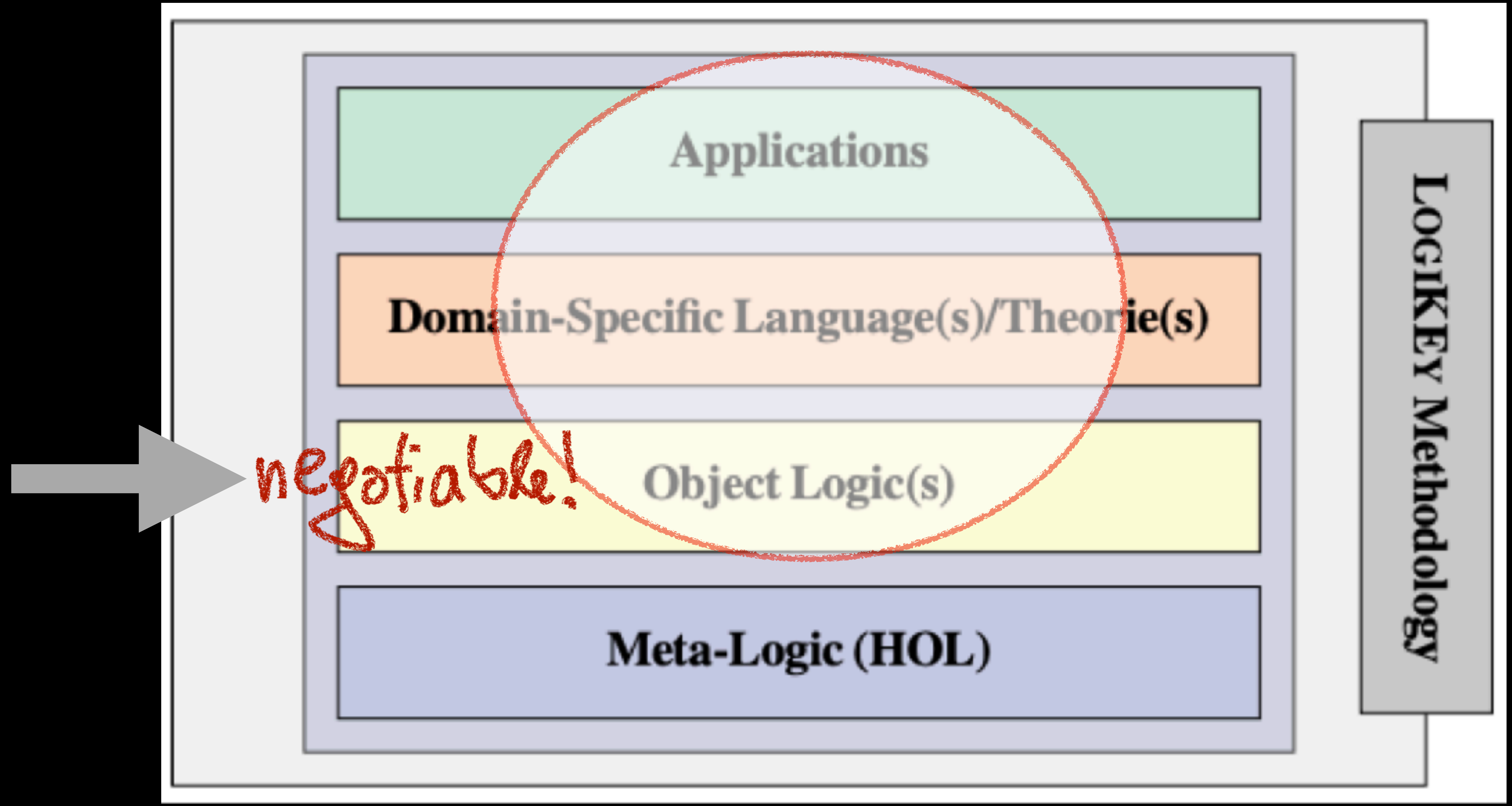
$\forall \varphi \,.\, \varphi \vee \neg \varphi$

$\forall \varphi \psi \,.\, (\varphi \equiv \psi) \supset (\varphi = \psi)$

$\forall \varphi \psi \,.\, (\forall x \,.\, \varphi x = \psi x) \supset (\varphi = \psi)$

If $H \vdash \varphi$ then $H, \psi \vdash \varphi$

$\dots$



This way proof assistants may support formalisations also in philosophy, law, …

Logical/Mathematical Pluralism

Make logical foundations flexible and negotiable.

Support reflection on them within proof assistants.

$\cancel{\exists x . x = x}$

$\cancel{\exists x . x = 1/0}$

$\cancel{\forall \varphi . \varphi \vee \neg \varphi}$

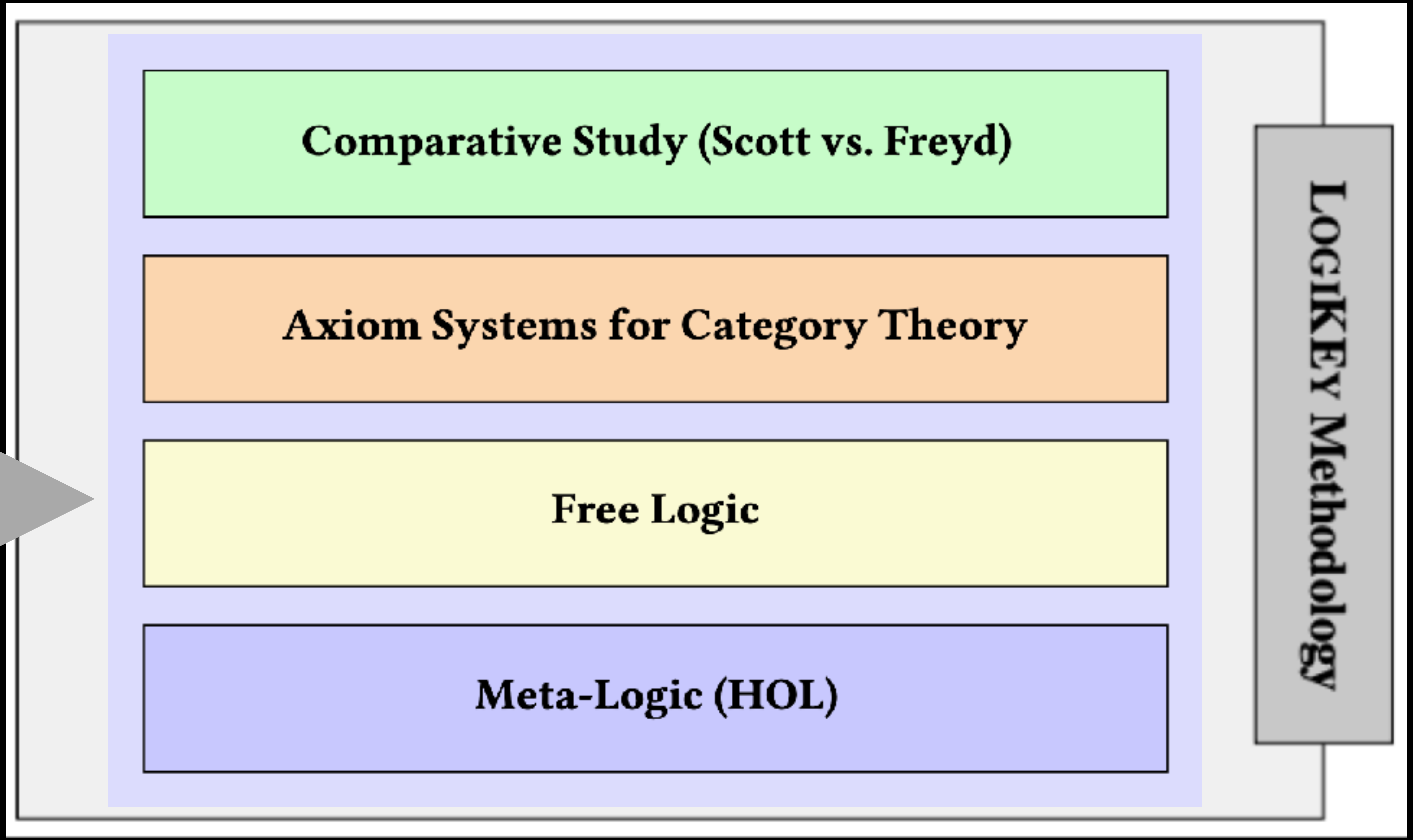
$\cancel{\forall \varphi \psi . (\varphi \equiv \psi) \supset (\varphi = \psi)}$

$\cancel{\forall \varphi \psi . (\forall x . \varphi x = \psi x) \supset (\varphi = \psi)}$

If $H \vdash \varphi$ then $H, \psi \vdash \varphi$

$\ldots$

Free Logic



Logical/Mathematical Pluralism

Make logical foundations flexible and negotiable.

Support reflection on them within proof assistants.

$\cancel{\exists x . x = x}$

$\cancel{\exists x . x = 1/0}$

$\cancel{\forall \varphi . \varphi \vee \neg \varphi}$

$\cancel{\forall \varphi \psi . (\varphi \equiv \psi) \supset (\varphi = \psi)}$

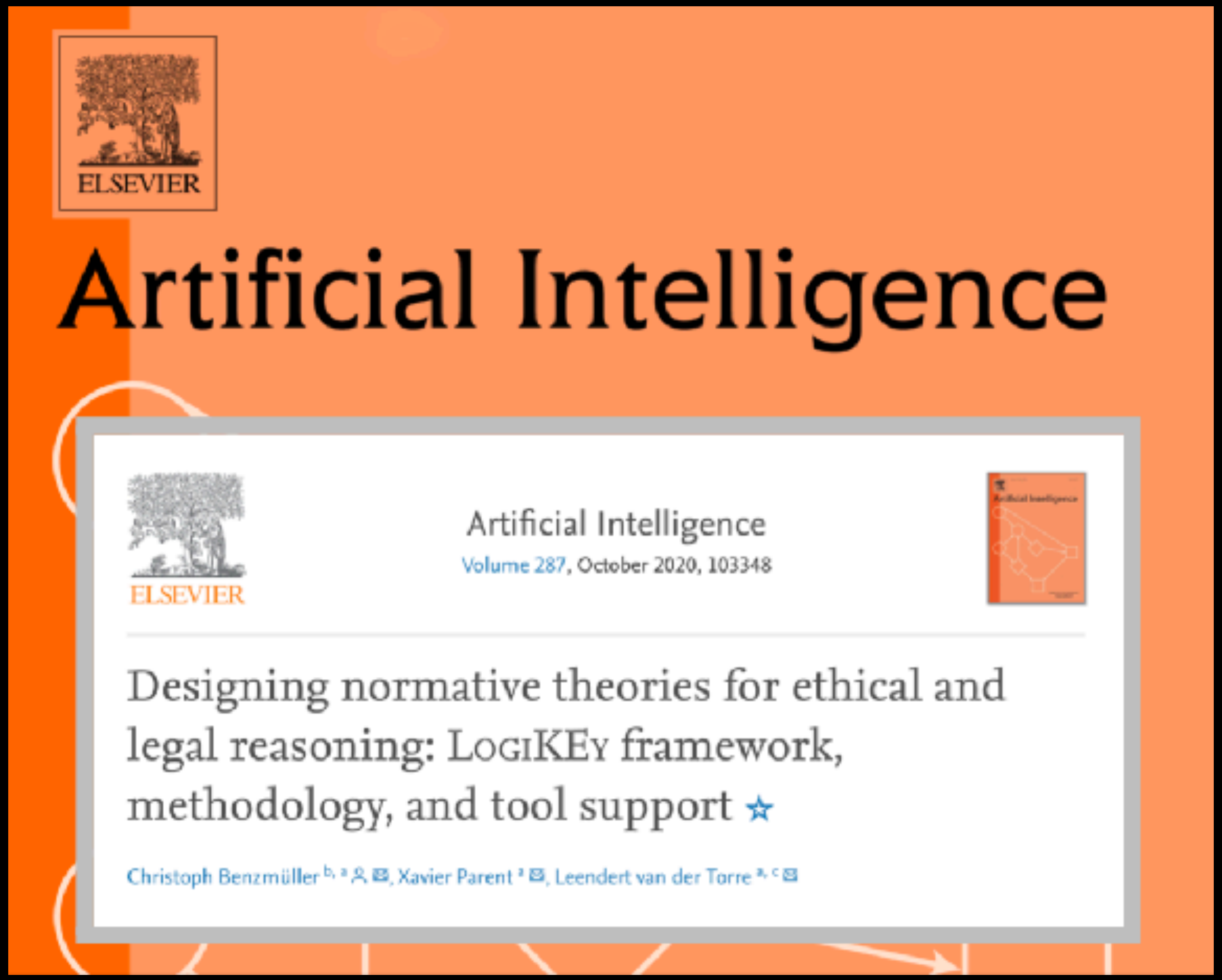
$\cancel{\forall \varphi \psi . (\forall x . \varphi x = \psi x) \supset (\varphi = \psi)}$

If $H \vdash \varphi$ then $H, \psi \vdash \varphi$

$\ldots$

Free Logic

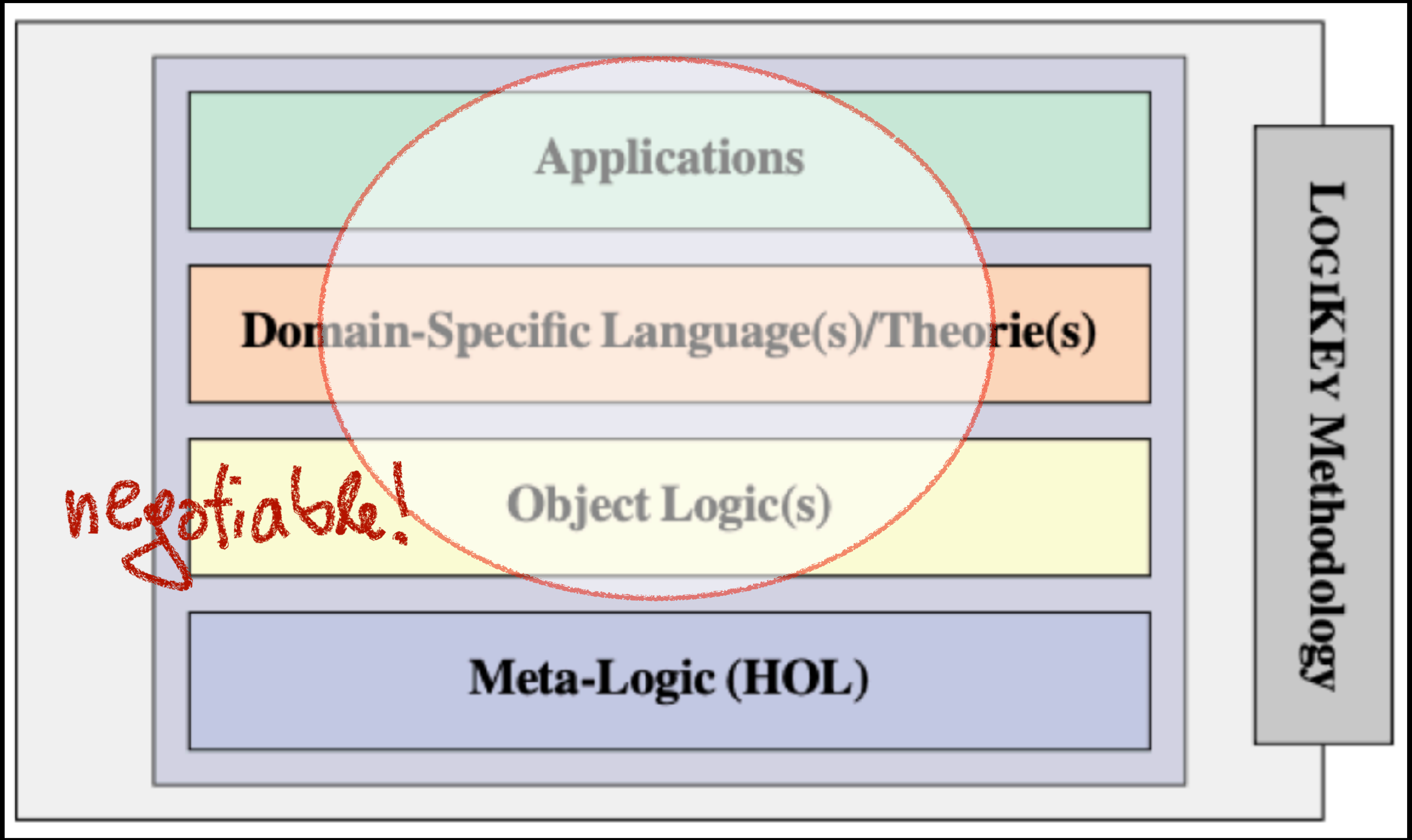
LogiKEy: Logic-pluralistic KR&R Methodology



2020

The term "LogiKEy" was introduced here.

Origins: 2007-2008



LogiKEy: focused so far on **shallow** embeddings

This talk: integration of **deep and shallow** embeddings

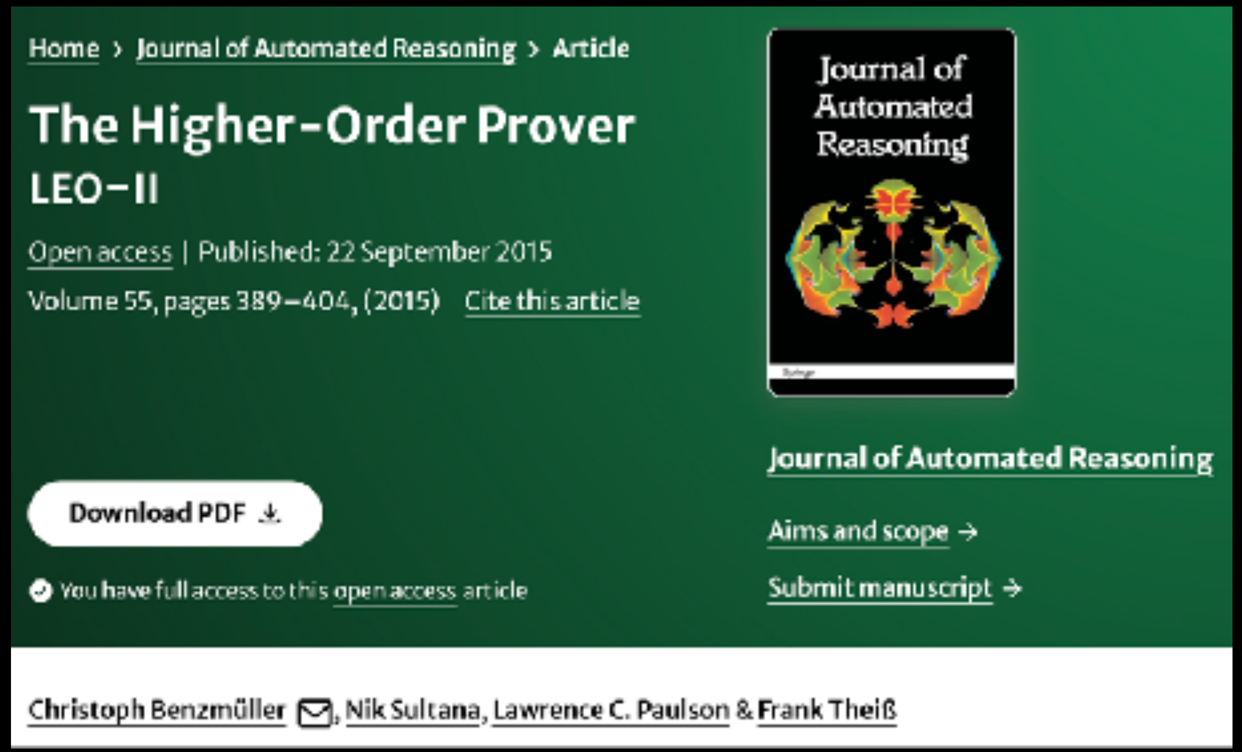
LogiKEy: Logic-pluralistic KR&R Methodology

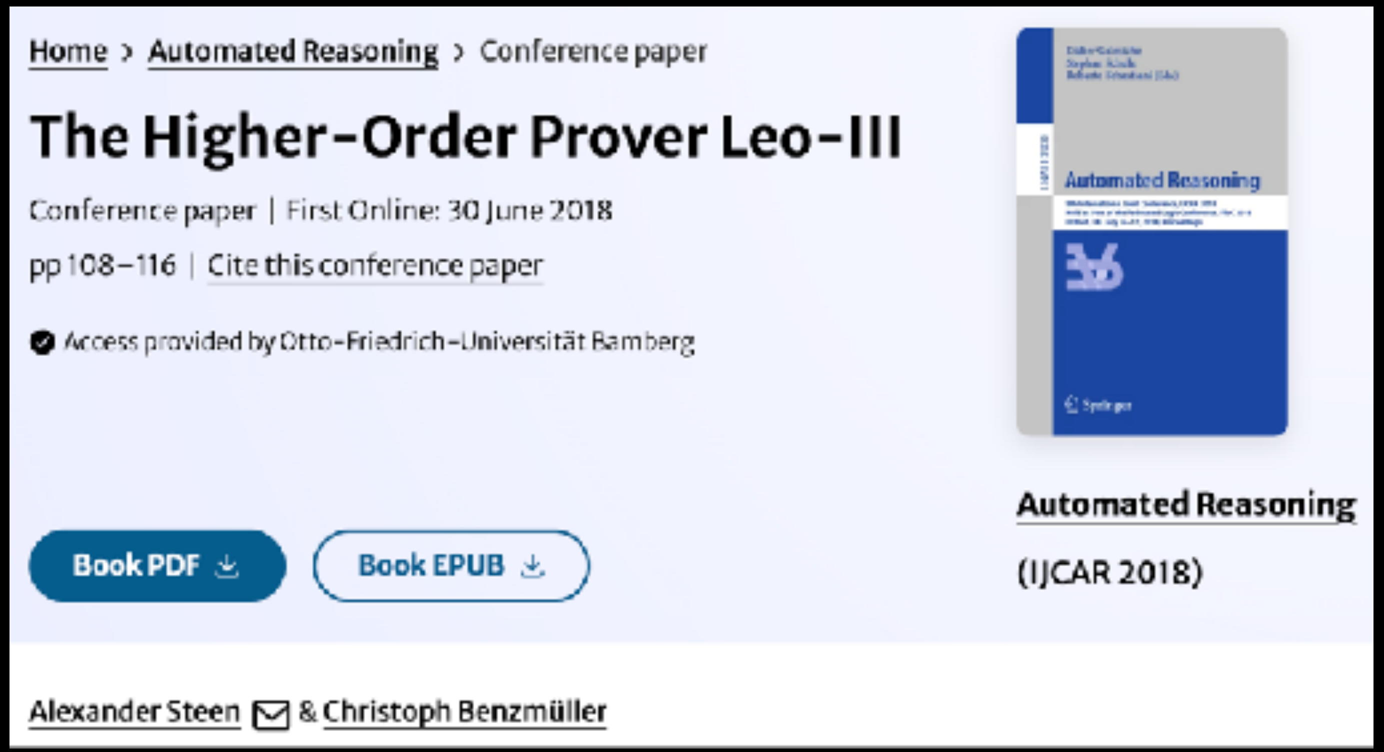
Selection of logic embeddings:

- Multimodal Logics [Andrews Festschrift, 2008]
- Access Control Logic [SEP, 2009]
- Intuitionistic Logic [LJ IGPL, 2010]
- Quantified Multimodal Logic [Log Univers, 2012]
- Quantified Conditional Logic [J Phil Log, 2016]
- Multivalued Logic [LLP, 2016]
- Free Logic [JAR, 2020]
- Dyadic Deontic Logic [AIJ, 2020] [JANCL, 2024]
- (Hyper-)Intensional Higher-Order Modal Logic [RSL, 2020]
- Public Announcement Logic [JLC, 2022]
- Logic of Right to Know [ECAI, 2025]
- …

Quantified Multimodal Logics at work: Computational Metaphysics

HOL ITP & ATP





Other provers could be used instead:

- ITPs: Lean, HOL4, Coq, Agda, …
- ATPs: Vampire, E, Zipperpin, CVC5, …

Isabelle/HOL: Still my favorite tool for **logic education** (and choice for this work)

- Proof automation with **Sledgehammer** (FO & HO ATPs, SMT solvers)
- Model- and countermodel-finding with **Nitpick**
- Powerful **User Interface**

LogiKEy in education started a decade ago

Forschen lernen im digitalen Zeitalter

Projekt von Informatiker Christoph Benzmüller mit dem Lehrpreis der Freien Universität Berlin ausgezeichnet

02.05.2016

Die Entscheidung sei in diesem Jahr besonders schwergefallen, sagte Klaus Hoffmann-Holland, Vizepräsident für Forschung und Lehre der Freien Universität Berlin und Mitglied der Jury für den Lehrpreis, anlässlich der Preisverleihung. Deshalb wurde in diesem Jahr nicht nur ein herausragendes Lehrprojekt ausgezeichnet, es wurden auch zwei weitere gewürdigt. Wie groß die Vielfalt der Angebote zur forschungsorientierten Lehre an der Freien Universität ist, zeigte eine Podiumsdiskussion im Anschluss an die Preisverleihung.

Ausgezeichnet mit dem Lehrpreis: Informatiker Christoph Benzmüller.
Bildquelle: Manuel Krane

- C. Benzmüller, M. Wisniewski, and A. Steen. "Computational Metaphysics". 2015 central teaching award of FU Berlin. 2015. url: https://dx.doi.org/10.13140/RG.2.1.3535.2568

- M. Wisniewski, A. Steen, and C. Benzmüller. "Einsatz von Theorembeweisern in der Lehre". In: Hochschuldidaktik der Informatik: 7. Fachtagung des GI-Fachbereichs Informatik und Ausbildung/ Didaktik der Informatik. Ed. by A. Schwill and U. Lucke. Commentarii informaticae didacticae (CID). Potsdam, Germany: Universitätsverlag Potsdam, 2016, pp. 81– 92. url: https://publishup.uni-potsdam.de/opus4-ubp/frontdoor/index/index/docId/9485

https://page.mi.fu-berlin.de/cbenzmueller/papers/LogiKEy_in_Education.pdf

Many related papers on deep and shallow embeddings in HOL since the early 1990s — see references in CADE 30 paper (and references therein)

Rest of the talk:

- Interlinked logic enbeddings in LogiKEy: deep & shallow
- Reasoning at meta- and object level
- Faithfulness proofs — can they be automated?
- Evaluation — which type of embedding performs best in practice?

PML

2 Propositional modal logic (PML)

Syntax

The *syntax* of PML is defined by the grammar $\boxed{\varphi, \psi := a \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi,}$ where $a \in \mathcal{S}$ are propositional symbols declared in a (nonempty) signature $\mathcal{S}$. Further logical connectives can be defined as usual, e.g.: $\varphi \vee \psi := \neg\varphi \supset \psi$, $\varphi \wedge \psi := \neg(\varphi \supset \neg\psi)$, $\Diamond\varphi := \neg\Box\neg\varphi$, $\top := p \supset p$ (for some $p \in \mathcal{S}$), and $\bot := \neg\top$.

A *semantics* for PML, called relational semantics or possible-world semantics, determines the truth of a formula φ relative to a so-called possible world w and a relational model. A relational model is a tuple $\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle$, where $\mathcal{W}$ is a set of possible worlds (called the universe), $\mathcal{R}$ is a binary relation on W (called the accessibility relation), $\mathcal{V}$ is a valuation function that assigns a truth value to each symbol $a \in \mathcal{S}$ in each world $w \in \mathcal{W}$, where $\mathcal{S}$ is the signature of propositional symbols. $\langle\mathcal{W}, \mathcal{R}\rangle$ is called a *frame*. Formally, the truth of a formula φ with respect to a given model $\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle$ and a possible world $w \in \mathcal{W}$ is defined as follows:

Semantics

$$\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models a \text{ iff } \mathcal{V}(a)(w) \text{ is true}$$
$$\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \neg\varphi \text{ iff not } \langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \varphi$$
$$\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \varphi \supset \psi \text{ iff } \langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \varphi \text{ implies } \langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \psi$$
$$\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \Box\varphi \text{ iff for all } v \text{ with } \mathcal{R}wv \text{ we have } \langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, v \models \varphi$$

Note that in this recursive definition of truth the only parameter that may be recursively modified is the possible world w in the $\Box$-clause, which states that a formula φ is "necessary" in a world w iff φ is true at all reachable worlds v from w. Based on the above definitions a notion of validity can be defined: A formula φ is valid (in so called base modal logic **K**) iff $\langle\mathcal{W}, \mathcal{R}, \mathcal{V}\rangle, w \models \varphi$ for all $\mathcal{W}$, $\mathcal{R}$, $\mathcal{V}$, and $w \in \mathcal{W}$.

Deep vs. Shallow (deep)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

> "[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

$$\varphi, \psi := \mathsf{at}(a) \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$$\varphi, \psi := a \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$\langle W, R, V\rangle, w \vDash \mathsf{at}(a)$ iff $V(a)(w)$ is true

$\langle W, R, V\rangle, w \vDash \neg\varphi$ iff not $\langle W, R, V\rangle, w \vDash \varphi$

$\langle W, R, V\rangle, w \vDash \varphi \supset \psi$ iff $\langle W, R, V\rangle, w \vDash \varphi$ implies $\langle W, R, V\rangle, w \vDash \psi$

$\langle W, R, V\rangle, w \vDash \Box\varphi$ iff for all $v \in W$ with $Rwv : \langle W, R, V\rangle, v \vDash \varphi$

$\vDash \varphi$ **iff** $\langle W, R, V\rangle, v \vDash \varphi$ **for all** W, R, V, v

$$\vDash \quad \neg\Box a \supset \neg(\Box\neg(a \supset b))$$

Deep vs. Shallow (deep)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

> "[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

```
—‹Deep embedding (of propositional modal logic in HOL)›
datatype PML = AtmD 𝒮 ("_ᵈ") | NotD PML ("¬ᵈ") | ImpD PML PML (infixr "⊃ᵈ" 93) | BoxD PML ("□ᵈ")
```

```
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
primrec RelativeTruthD :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒PML⇒bool" ("⟨_,_,_⟩,_ ⊨ᵈ _") where
    "⟨W,R,V⟩, w ⊨ᵈ aᵈ        = (V a w)"
  | "⟨W,R,V⟩, w ⊨ᵈ ¬ᵈφ       = (¬ ⟨W,R,V⟩, w ⊨ᵈ φ)"
  | "⟨W,R,V⟩, w ⊨ᵈ φ ⊃ᵈ ψ    = (⟨W,R,V⟩, w ⊨ᵈ φ  ⟶  ⟨W,R,V⟩, w ⊨ᵈ ψ)"
  | "⟨W,R,V⟩, w ⊨ᵈ □ᵈφ       = (∀v:W. R w v ⟶ ⟨W,R,V⟩, v ⊨ᵈ φ)"
—‹Definition of validity›
definition ValD ("⊨ᵈ _") where "(⊨ᵈ φ) ≡ (∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨ᵈ φ)"
```

Superscript
d
for "deep"

```
⊨ᵈ ¬ᵈ (□ᵈ aᵈ) ⊃ᵈ ¬ᵈ (□ᵈ (¬ᵈ (aᵈ ⊃ᵈ bᵈ)))
```

Deep vs. Shallow (shallow — maximal)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

"[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

A shallow embedding eschews the algebraic datatype, and hence the explicit representation of the abstract syntax of the language; instead, the language is defined directly in terms of its semantics."

$$\varphi, \psi := \mathsf{at}(a) \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$$\varphi, \psi := a \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$\langle W, R, V\rangle, w \vDash \mathsf{at}(a)$ iff $V(a)(w)$ is true

$\langle W, R, V\rangle, w \vDash \neg\varphi$ iff not $\langle W, R, V\rangle, w \vDash \varphi$

$\langle W, R, V\rangle, w \vDash \varphi \supset \psi$ iff $\langle W, R, V\rangle, w \vDash \varphi$ implies $\langle W, R, V\rangle, w \vDash \psi$

$\langle W, R, V\rangle, w \vDash \Box\varphi$ iff for all $v \in W$ with Rwv : $\langle W, R, V\rangle, v \vDash \varphi$

$\vDash \varphi$ **iff** $\langle W, R, V\rangle, v \vDash \varphi$ **for all** W, R, V, v

at	$\lambda a\,.\,\lambda W R V w\,.\,V a w$
$\neg$	$\lambda\varphi\,.\,\lambda W R V w\,.\,\neg(\varphi\, W R V w)$
$\supset$	$\lambda\varphi\,.\,\lambda\psi\,.\,\lambda W R V w\,.\,(\varphi\, W R V w) \to (\psi\, W R V w)$
$\Box$	$\lambda\varphi\,.\,\lambda W R V w\,.\,\forall v{:}W\,.\,R w v \to (\varphi\, W R V v)$
$\vDash$	$\lambda\varphi\,.\,\forall W R V w\,.\,(\varphi\, W R V w)$

$$\vDash \quad \neg\Box a \supset \neg(\Box\neg(a \supset b))$$

Deep vs. Shallow (shallow — maximal)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

“[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

A shallow embedding eschews the algebraic datatype, and hence the explicit representation of the abstract syntax of the language; instead, the language is defined directly in terms of its semantics.”

Superscript *s* for “shallow”

```
—‹Shallow embedding (of propositional modal logic in HOL)›
type_synonym σ = "𝒲⇒ℛ⇒𝒱⇒w⇒bool"
definition AtmS::"𝒮⇒σ"     ("_^s")              where  "a^s ≡ λW R V w. V a w"
definition NegS::"σ⇒σ"     ("¬^s")              where  "¬^s φ ≡ λW R V w.  ¬(φ W R V w)"
definition ImpS::"σ⇒σ⇒σ" (infixr "⊃^s" 93)  where  "φ ⊃^s ψ ≡ λW R V w. (φ W R V w) ⟶ (ψ W R V w)"
definition BoxS::"σ⇒σ"     ("□^s")              where  "□^s φ ≡ λW R V w. ∀v:W. R w v ⟶ (φ W R V v)"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
definition RelativeTruthS :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒σ⇒bool" ("⟨_,_,_⟩,_ ⊨^s _") where "⟨W,R,V⟩,w ⊨^s φ ≡ φ W R V w"
—‹Definition of validity›
definition ValS ("⊨^s _") where "⊨^s φ  ≡  ∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨^s φ"
```

$\models^s \neg^s (\Box^s a^s) \supset^s \neg^s (\Box^s (\neg^s (a^s \supset^s b^s)))$

Deep vs. Shallow (shallow — maximal)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

"[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

A shallow embedding eschews the algebraic datatype, and hence the explicit representation of the abstract syntax of the language; instead, the language is defined directly in terms of its semantics."

$$\varphi, \psi := \mathrm{at}(a) \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$
$$\varphi, \psi := a \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$\langle W, R, V\rangle, w \vDash \mathrm{at}(a)$ iff $V(a)(w)$ is true

$\langle W, R, V\rangle, w \vDash \neg\varphi$ iff not $\langle W, R, V\rangle, w \vDash \varphi$

$\langle W, R, V\rangle, w \vDash \varphi \supset \psi$ iff $\langle W, R, V\rangle, w \vDash \varphi$ implies $\langle W, R, V\rangle, w \vDash \psi$

$\langle W, R, V\rangle, w \vDash \Box\varphi$ iff for all $v \in W$ with Rwv : $\langle W, R, V\rangle, v \vDash \varphi$

$\vDash \varphi$ **iff** $\langle W, R, V\rangle, v \vDash \varphi$ **for all** W, R, V, v

at	$\lambda a\,.\,\lambda W R V w\,.\,V a w$
$\neg$	$\lambda\varphi\,.\,\lambda W R V w\,.\,\neg(\varphi W R V w)$
$\supset$	$\lambda\varphi\,.\,\lambda\psi\,.\,\lambda W R V w\,.\,(\varphi W R V w) \to (\psi W R V w)$
$\Box$	$\lambda\varphi\,.\,\lambda W R V w\,.\,\forall v{:}W\,.\,R w v \to (\varphi W R V v)$
$\vDash$	$\lambda\varphi\,.\,\forall W R V w\,.\,(\varphi W R V w)$

$$\vDash \quad \neg\Box a \supset \neg(\Box\neg(a \supset b))$$

Deep vs. Shallow (shallow — minimal)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

"[...] a deep embedding consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

A shallow embedding eschews the algebraic datatype, and hence the explicit representation of the abstract syntax of the language; instead, the language is defined directly in terms of its semantics."

$$\varphi, \psi := \mathsf{at}(a) \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$$\varphi, \psi := a \mid \neg\varphi \mid \varphi \supset \psi \mid \Box\varphi \qquad (a \in S)$$

$\langle W, R, V\rangle, w \vDash \mathsf{at}(a)$ iff $V(a)(w)$ is true

$\langle W, R, V\rangle, w \vDash \neg\varphi$ iff not $\langle W, R, V\rangle, w \vDash \varphi$

$\langle W, R, V\rangle, w \vDash \varphi \supset \psi$ iff $\langle W, R, V\rangle, w \vDash \varphi$ implies $\langle W, R, V\rangle, w \vDash \psi$

$\langle W, R, V\rangle, w \vDash \Box\varphi$ iff for all $v \in W$ with $Rwv : \langle W, R, V\rangle, v \vDash \varphi$

$\vDash \varphi$ **iff** $\langle W, R, V\rangle, v \vDash \varphi$ **for all** W, R, V, v

consts V, R

at	$\lambda a \,.\, \lambda w \,.\, V\, a\, w$
$\neg$	$\lambda\varphi \,.\, \lambda w \,.\, \neg(\varphi\, w)$
$\supset$	$\lambda\varphi \,.\, \lambda\psi \,.\, \lambda w \,.\, (\varphi\, w) \to (\psi\, w)$
$\Box$	$\lambda\varphi \,.\, \lambda w \,.\, \forall v \,.\, R\, w\, v \to (\varphi\, v)$
$\vDash$	$\lambda\varphi \,.\, \forall w \,.\, \varphi\, w$

$$\vDash \quad \neg\Box a \supset \neg(\Box\neg(a \supset b))$$

Deep vs. Shallow (shallow — minimal)

Gibbons & Wu (2014) Folding domain-specific languages: deep and shallow embeddings (functional pearl). Doi: 10.1145/2628136.2628138

"[...] a **deep embedding** consists of a representation of the abstract syntax as an algebraic datatype, together with some functions that assign semantics to that syntax by traversing the algebraic datatype.

A **shallow embedding** eschews the algebraic datatype, and hence the explicit representation of the abstract syntax of the language; instead, the language is defined directly in terms of its semantics."

Superscript ^{m} for "min. shallow"

```
—‹The acessibility relation R and the valuation function V are introduced as constants at the meta-level HOL›
consts R::ℛ V::𝒱
—‹Shallow embedding (of propositional modal logic in HOL)›
type_synonym σ = "w⇒bool"
definition AtmM::"𝒮⇒σ"     ("_^m")          where  "a^m ≡ λw. V a w"
definition NegM::"σ⇒σ"     ("¬^m")          where  "¬^m φ ≡ λw. ¬φ w"
definition ImpM::"σ⇒σ⇒σ" (infixr "⊃^m" 93)  where  "φ ⊃^m ψ ≡ λw. φ w ⟶ ψ w"
definition BoxM::"σ⇒σ"     ("□^m")          where  "□^m φ ≡ λw. ∀v. R w v ⟶ φ v"

—‹Definition of truth of a formula relative to a model ‹(W,R,V)› and a possible world w›
definition RelativeTruthM :: "w⇒σ⇒bool" ("_ ⊨^m _") where "w ⊨^m φ ≡ φ w"
—‹Definition of validity›
definition ValM ("⊨^m _") where "⊨^m φ ≡ ∀w::w. w ⊨^m φ"
```

$$\models^m \neg^m (\Box^m a^m) \supset^m \neg^m (\Box^m (\neg^m (a^m \supset^m b^m)))$$

Example (minimal shallow):

$\models \quad \neg \Box a \supset \neg(\Box \neg (a \supset b))$

$\models^m \neg^m (\Box^m a^m) \supset^m \neg^m (\Box^m (\neg^m (a^m \supset^m b^m)))$

$\models \quad \neg \Box a \supset \neg(\Box \neg (a \supset b))$

$\neg \downarrow \lambda \varphi . \lambda w . \neg(\varphi\, w)$

$\models \quad (\lambda \varphi . \lambda w . \neg(\varphi\, w)) \Box a \supset (\lambda \varphi . \lambda w . \neg(\varphi\, w))(\Box (\lambda \varphi . \lambda w . \neg(\varphi\, w))(a \supset b))$

$\Box \downarrow \lambda \varphi . \lambda w . \forall v . R\, w\, v \rightarrow \varphi\, v$

$\models \quad (\lambda \varphi . \lambda w . \neg(\varphi\, w))((\lambda \varphi . \lambda w . \forall v . R\, w\, v \rightarrow \varphi\, v)\, a) \supset (\lambda w . \neg(((\lambda \varphi . \lambda w . \forall v . R\, w\, v \rightarrow \varphi\, v)(\lambda \varphi . \lambda w . \neg(\varphi\, w))(a \supset b))\, w))$

$\supset \downarrow \lambda \varphi . \lambda \psi . \lambda w . (\varphi\, w) \rightarrow (\psi\, w)$

$\models \quad \ldots \quad \ldots$

$\text{at} \downarrow \lambda a . \lambda w . (V\, a\, w)$

$\models \quad \ldots \quad \ldots$

$\models \downarrow \lambda \varphi . \forall w . (\varphi\, w)$

$\ldots \quad \ldots \quad \ldots$

$\downarrow \beta$-reduction

$\forall w . \neg(\forall v . R\, w\, v \rightarrow V\, a\, v) \rightarrow \neg(\forall v . R\, w\, v \rightarrow \neg(V\, a\, v \rightarrow V\, b\, v))$

$\downarrow$ simplification

$\forall w . (\exists v . R\, w\, v \wedge \neg V\, a\, v) \rightarrow (\exists v . R\, w\, v \wedge (V\, a\, v \rightarrow V\, b\, v))$

Standard translation of PML to FOL via unfolding and β-reduction in HOL

Example (all embeddings): $\models \quad \neg\Box a \supset \neg(\Box\neg(a \supset b))$

shallow minimal ($.^m$)

$\models^m \neg^m (\Box^m a^m) \supset^m \neg^m (\Box^m (\neg^m (a^m \supset^m b^m)))$

$\forall w.\ (\exists v.\ R\ w\ v \wedge \neg V\ a\ v) \longrightarrow (\exists v.\ R\ w\ v \wedge (V\ a\ v \longrightarrow V\ b\ v))$

shallow maximal ($.^s$)

$\models^s \neg^s (\Box^s a^s) \supset^s \neg^s (\Box^s (\neg^s (a^s \supset^s b^s)))$

$\forall W\ R\ V\ x.$
$\quad W\ x \longrightarrow$
$\quad (\exists xa.\ W\ xa \wedge R\ x\ xa \wedge \neg V\ a\ xa) \longrightarrow (\exists xa.\ W\ xa \wedge R\ x\ xa \wedge (V\ a\ xa \longrightarrow V\ b\ xa))$

deep ($.^d$)

$\models^d \neg^d (\Box^d a^d) \supset^d \neg^d (\Box^d (\neg^d (a^d \supset^d b^d)))$

$\forall W\ R\ V\ x.$
$\quad W\ x \longrightarrow$
$\quad (\exists xa.\ W\ xa \wedge R\ x\ xa \wedge \neg V\ a\ xa) \longrightarrow (\exists xa.\ W\ xa \wedge R\ x\ xa \wedge (V\ a\ xa \longrightarrow V\ b\ xa))$

Example (all embeddings):

$\cancel{\models \neg\Box a \supset \neg(\Box\neg(a \supset b))}$

$\models \neg\Box\varphi \supset \neg(\Box\neg(\varphi \supset \psi))$

shallow minimal ($.^m$)

$\models^m \neg^m (\Box^m \varphi) \supset^m \neg^m (\Box^m (\neg^m (\varphi \supset^m \psi)))$

$\forall$w. ($\exists$v. R w v $\wedge$ $\neg$ φ v) $\longrightarrow$ ($\exists$v. R w v $\wedge$ (φ v $\longrightarrow$ ψ v))

shallow maximal ($.^s$)

$\models^s \neg^s (\Box^s \varphi) \supset^s \neg^s (\Box^s (\neg^s (\varphi \supset^s \psi)))$

```
∀W R V x.
   W x ⟶
   (∃xa. W xa ∧ R x xa ∧ ¬ φ W R V xa) ⟶
   (∃xa. W xa ∧ R x xa ∧ (φ W R V xa ⟶ ψ W R V xa))
```

deep ($.^d$)

$\models^d \neg^d (\Box^d \varphi) \supset^d \neg^d (\Box^d (\neg^d (\varphi \supset^d \psi)))$

```
∀W R V x.
   W x ⟶
   (∃xa.
      W xa ∧
      R x xa ∧
      ¬ rec_PML (λa W R V. V a) (λφ φ' W R V w. ¬ φ' W R V w)
         (λφ ψ φ' ψ' W R V w. φ' W R V w ⟶ ψ' W R V w)
         (λφ φ' W R V w. ∀x. W x ⟶ R w x ⟶ φ' W R V x) φ W R V xa) ⟶
   (∃xa.
      W xa ∧
      R x xa ∧
      (rec_PML (λa W R V. V a) (λφ φ' W R V w. ¬ φ' W R V w)
        (λφ ψ φ' ψ' W R V w. φ' W R V w ⟶ ψ' W R V w)
        (λφ φ' W R V w. ∀x. W x ⟶ R w x ⟶ φ' W R V x) φ W R V xa ⟶
       rec_PML (λa W R V. V a) (λφ φ' W R V w. ¬ φ' W R V w)
        (λφ ψ φ' ψ' W R V w. φ' W R V w ⟶ ψ' W R V w)
        (λφ φ' W R V w. ∀x. W x ⟶ R w x ⟶ φ' W R V x) ψ W R V xa))
```

Schematic formulas φ lead to more complex proof problems

Interlinked Embeddings

shared preliminaries

```
—‹Type declarations common for both the deep and shallow embedding›
  typedecl w —‹Type for possible worlds›
  typedecl 𝒮 —‹Type for propositional constant symbols›
  consts p::𝒮 q::𝒮 —‹Some propositional constant symbols›
  type_synonym 𝒲 = "w⇒bool"  —‹Type for sets of possible worlds›
  type_synonym ℛ = "w⇒w⇒bool"  —‹Type for accessibility relations›
  type_synonym 𝒱 = "𝒮⇒w⇒bool"  —‹Type for valuation functions›
```

shallow maximal ($.^{s}$)

```
theory PMLinHOL_shallow imports PMLinHOL_preliminaries          (* Christoph Benzmüller, 2025 *)
begin
—‹Shallow embedding (of propositional modal logic in HOL)›
  type_synonym σ = "𝒲⇒ℛ⇒𝒱⇒w⇒bool"
  definition AtmS::"𝒮⇒σ"     ("_ˢ")             where   "aˢ ≡ λW R V w. V a w"
  definition NegS::"σ⇒σ"     ("¬ˢ")             where   "¬ˢ φ ≡ λW R V w. ¬(φ W R V w)"
  definition ImpS::"σ⇒σ⇒σ" (infixr "⊃ˢ" 93)  where   "φ ⊃ˢ ψ ≡ λW R V w. (φ W R V w) ⟶ (ψ W R V w)"
  definition BoxS::"σ⇒σ"     ("□ˢ")             where   "□ˢ φ ≡ λW R V w. ∀v:W. R w v ⟶ (φ W R V v)"
—‹Further logical connectives as definitions›
  definition OrS   (infixr "∨ˢ" 92)  where  "φ ∨ˢ ψ ≡ ¬ˢφ ⊃ˢ ψ"
  definition AndS  (infixr "∧ˢ" 95)  where  "φ ∧ˢ ψ ≡ ¬ˢ(φ ⊃ˢ ¬ˢψ)"
  definition DiaS  ("◇ˢ")            where  "◇ˢφ ≡ ¬ˢ (□ˢ (¬ˢφ)) "
  definition TopS  ("⊤ˢ")            where  "⊤ˢ ≡ pˢ ⊃ˢ pˢ"
  definition BotS  ("⊥ˢ")            where  "⊥ˢ ≡ ¬ˢ ⊤ˢ"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
  definition RelativeTruthS :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒σ⇒bool" ("⟨_,_,_⟩,_ ⊨ˢ _") where "⟨W,R,V⟩,w ⊨ˢ φ ≡ φ W R V w"
—‹Definition of validity›
  definition ValS ("⊨ˢ _") where "⊨ˢ φ ≡ ∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨ˢ φ"
—‹Collection of definitions in a bag called DefS›
  named_theorems DefS declare AtmS_def[DefS,simp] NegS_def[DefS,simp] ImpS_def[DefS,simp]
   BoxS_def[DefS,simp] OrS_def[DefS,simp] AndS_def[DefS,simp] DiaS_def[DefS,simp] TopS_def[DefS,simp]
   BotS_def[DefS,simp] RelativeTruthS_def[DefS,simp] ValS_def[DefS,simp]
end
```

deep ($.^{d}$)

```
theory PMLinHOL_deep imports PMLinHOL_preliminaries          (* Christoph Benzmüller, 2025 *)
begin
—‹Deep embedding (of propositional modal logic in HOL)›
  datatype PML = AtmD 𝒮 ("_ᵈ") | NotD PML ("¬ᵈ") | ImpD PML PML (infixr "⊃ᵈ" 93) | BoxD PML ("□ᵈ")
—‹Further logical connectives as definitions›
  definition OrD    (infixr "∨ᵈ" 92)  where   "φ∨ᵈψ ≡ ¬ᵈφ ⊃ᵈ ψ"
  definition AndD   (infixr "∧ᵈ" 95)  where   "φ∧ᵈψ ≡ ¬ᵈ(φ ⊃ᵈ ¬ᵈψ)"
  definition DiaD   ("◇ᵈ_")           where   "◇ᵈφ ≡ ¬ᵈ(□ᵈ(¬ᵈφ)) "
  definition TopD   ("⊤ᵈ")            where   "⊤ᵈ ≡ pᵈ ⊃ᵈ pᵈ"
  definition BotD   ("⊥ᵈ")            where   "⊥ᵈ ≡ ¬ᵈ ⊤ᵈ"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
  primrec RelativeTruthD :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒PML⇒bool" ("⟨_,_,_⟩,_ ⊨ᵈ _") where
     "⟨W,R,V⟩, w ⊨ᵈ aᵈ        = (V a w)"
   | "⟨W,R,V⟩, w ⊨ᵈ ¬ᵈφ       = (¬ ⟨W,R,V⟩, w ⊨ᵈ φ)"
   | "⟨W,R,V⟩, w ⊨ᵈ φ ⊃ᵈ ψ   = (⟨W,R,V⟩, w ⊨ᵈ φ ⟶ ⟨W,R,V⟩, w ⊨ᵈ ψ)"
   | "⟨W,R,V⟩, w ⊨ᵈ □ᵈφ       = (∀v:W. R w v ⟶ ⟨W,R,V⟩, v ⊨ᵈ φ)"
—‹Definition of validity›
  definition ValD ("⊨ᵈ _") where "(⊨ᵈ φ) ≡ (∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨ᵈ φ)"
—‹Collection of definitions in a bag called DefD›
  named_theorems DefD declare OrD_def[DefD,simp] AndD_def[DefD,simp] DiaD_def[DefD,simp]
   TopD_def[DefD,simp] BotD_def[DefD,simp] RelativeTruthD_def[DefD,simp] ValD_def[DefD,simp]
end
```

shallow minimal ($.^{m}$)

```
theory PMLinHOL_shallow_minimal  imports PMLinHOL_preliminaries          (* Christoph Benzmüller, 2025 *)
begin
—‹The acessibility relation R and the valuation function V are introduced as constants at the meta-level HOL›
  consts R::ℛ V::𝒱
—‹Shallow embedding (of propositional modal logic in HOL)›
  type_synonym σ = "w⇒bool"
  definition AtmM::"𝒮⇒σ"     ("_ᵐ")             where    "aᵐ ≡ λw. V a w"
  definition NegM::"σ⇒σ"     ("¬ᵐ")             where    "¬ᵐ φ ≡ λw. ¬φ w"
  definition ImpM::"σ⇒σ⇒σ" (infixr "⊃ᵐ" 93)  where   "φ ⊃ᵐ ψ ≡ λw. φ w ⟶ ψ w"
  definition BoxM::"σ⇒σ"     ("□ᵐ")             where    "□ᵐ φ ≡ λw. ∀v. R w v ⟶ φ v"
—‹Further logical connectives as definitions›
  definition OrM    (infixr "∨ᵐ" 92)  where  "φ ∨ᵐ ψ ≡ ¬ᵐφ ⊃ᵐ ψ"
  definition AndM   (infixr "∧ᵐ" 95)  where  "φ ∧ᵐ ψ ≡ ¬ᵐ(φ ⊃ᵐ ¬ᵐψ)"
  definition DiaM   ("◇ᵐ_")           where  "◇ᵐφ ≡ ¬ᵐ (□ᵐ (¬ᵐφ)) "
  definition TopM   ("⊤ᵐ")            where  "⊤ᵐ ≡ pᵐ ⊃ᵐ pᵐ"
  definition BotM   ("⊥ᵐ")            where  "⊥ᵐ ≡ ¬ᵐ ⊤ᵐ"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
  definition RelativeTruthM :: "w⇒σ⇒bool" ("_ ⊨ᵐ _") where "w ⊨ᵐ φ ≡ φ w"
—‹Definition of validity›
  definition ValM ("⊨ᵐ _") where "⊨ᵐ φ  ≡ ∀w::w. w ⊨ᵐ φ"
—‹Collection of definitions in a bag called DefM›
  named_theorems DefM declare AtmM_def[DefM,simp] NegM_def[DefM,simp] ImpM_def[DefM,simp]
   BoxM_def[DefM,simp] OrM_def[DefM,simp] AndM_def[DefM,simp] DiaM_def[DefM,simp] TopM_def[DefM,simp]
   BotM_def[DefM,simp] RelativeTruthM_def[DefM,simp] ValM_def[DefM,simp]
end
```

Faithfulness automated

deep ($.^d$)

```
theory PMLinHOL_deep imports PMLinHOL_preliminaries                    (* Christoph Benzmüller, 2025 *)
begin
—‹Deep embedding (of propositional modal logic in HOL)›
 datatype PML = AtmD 𝒮 ("_^d") | NotD PML ("¬^d") | ImpD PML PML (infixr "⊃^d" 93) | BoxD PML ("□^d")
—‹Further logical connectives as definitions›
 definition OrD   (infixr "∨^d" 92)  where  "φ∨^dψ ≡ ¬^dφ ⊃^d ψ"
 definition AndD  (infixr "∧^d" 95)  where  "φ∧^dψ ≡ ¬^d(φ ⊃^d ¬^dψ)"
 definition DiaD  ("◇^d_")           where  "◇^dφ ≡ ¬^d(□^d(¬^dφ)) "
 definition TopD  ("⊤^d")            where  "⊤^d ≡ p^d ⊃^d p^d"
 definition BotD  ("⊥^d")            where  "⊥^d ≡ ¬^d ⊤^d"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
 primrec RelativeTruthD :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒PML⇒bool" ("⟨_,_,_⟩,_ ⊨^d _") where
    "⟨W,R,V⟩, w ⊨^d a^d         = (V a w)"
  | "⟨W,R,V⟩, w ⊨^d ¬^dφ        = (¬ ⟨W,R,V⟩, w ⊨^d φ)"
  | "⟨W,R,V⟩, w ⊨^d φ ⊃^d ψ     = (⟨W,R,V⟩, w ⊨^d φ ⟶ ⟨W,R,V⟩, w ⊨^d ψ)"
  | "⟨W,R,V⟩, w ⊨^d □^dφ        = (∀v:W. R w v ⟶ ⟨W,R,V⟩, v ⊨^d φ)"
—‹Definition of validity›
 definition ValD ("⊨^d _") where "(⊨^d φ) ≡ (∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨^d φ)"
—‹Collection of definitions in a bag called DefD›
 named_theorems DefD declare OrD_def[DefD,simp] AndD_def[DefD,simp] DiaD_def[DefD,simp]
  TopD_def[DefD,simp] BotD_def[DefD,simp] RelativeTruthD_def[DefD,simp] ValD_def[DefD,simp]
end
```

shallow maximal ($.^s$)

```
theory PMLinHOL_shallow imports PMLinHOL_preliminaries                 (* Christoph Benzmüller, 2025 *)
begin
—‹Shallow embedding (of propositional modal logic in HOL)›
 type_synonym σ = "𝒲⇒ℛ⇒𝒱⇒w⇒bool"
 definition AtmS::"𝒮⇒σ"      ("_^s")             where  "a^s ≡ λW R V w. V a w"
 definition NegS::"σ⇒σ"      ("¬^s")             where  "¬^s φ ≡ λW R V w. ¬(φ W R V w)"
 definition ImpS::"σ⇒σ⇒σ" (infixr "⊃^s" 93)     where  "φ ⊃^s ψ ≡ λW R V w. (φ W R V w) ⟶ (ψ W R V w)"
 definition BoxS::"σ⇒σ"      ("□^s")             where  "□^s φ ≡ λW R V w. ∀v:W. R w v ⟶ (φ W R V v)"
—‹Further logical connectives as definitions›
 definition OrS   (infixr "∨^s" 92)  where  "φ ∨^s ψ ≡ ¬^sφ ⊃^s ψ"
 definition AndS  (infixr "∧^s" 95)  where  "φ ∧^s ψ ≡ ¬^s(φ ⊃^s ¬^sψ)"
 definition DiaS  ("◇^s")            where  "◇^sφ ≡ ¬^s (□^s (¬^sφ)) "
 definition TopS  ("⊤^s")            where  "⊤^s ≡ p^s ⊃^s p^s"
 definition BotS  ("⊥^s")            where  "⊥^s ≡ ¬^s ⊤^s"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
 definition RelativeTruthS :: "𝒲⇒ℛ⇒𝒱 ⇒w⇒σ⇒bool" ("⟨_,_,_⟩,_ ⊨^s _") where "⟨W,R,V⟩,w ⊨^s φ ≡ φ W R V w"
—‹Definition of validity›
 definition  ValS ("⊨^s _") where "⊨^s φ  ≡ ∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨^s φ"
—‹Collection of definitions in a bag called DefS›
 named_theorems DefS declare AtmS_def[DefS,simp] NegS_def[DefS,simp] ImpS_def[DefS,simp]
  BoxS_def[DefS,simp] OrS_def[DefS,simp] AndS_def[DefS,simp] DiaS_def[DefS,simp] TopS_def[DefS,simp]
  BotS_def[DefS,simp] RelativeTruthS_def[DefS,simp] ValS_def[DefS,simp]
end
```

shallow minimal ($.^m$)

```
theory PMLinHOL_shallow_minimal  imports PMLinHOL_preliminaries        (* Christoph Benzmüller, 2025 *)
begin
—‹The acessibility relation R and the valuation function V are introduced as constants at the meta-level HOL›
 consts R::ℛ V::𝒱
—‹Shallow embedding (of propositional modal logic in HOL)›
 type_synonym σ = "w⇒bool"
 definition AtmM::"𝒮⇒σ"      ("_^m")             where  "a^m ≡ λw. V a w"
 definition NegM::"σ⇒σ"      ("¬^m")             where  "¬^m φ ≡ λw.  ¬φ w"
 definition ImpM::"σ⇒σ⇒σ" (infixr "⊃^m" 93)     where  "φ ⊃^m ψ ≡ λw. φ w ⟶ ψ w"
 definition BoxM::"σ⇒σ"      ("□^m")             where  "□^m φ ≡ λw. ∀v. R w v ⟶ φ v"
—‹Further logical connectives as definitions›
 definition OrM   (infixr "∨^m" 92)  where  "φ ∨^m ψ ≡ ¬^mφ ⊃^m ψ"
 definition AndM  (infixr "∧^m" 95)  where  "φ ∧^m ψ ≡ ¬^m(φ ⊃^m ¬^mψ)"
 definition DiaM  ("◇^m_")           where  "◇^mφ ≡ ¬^m (□^m (¬^mφ)) "
 definition TopM  ("⊤^m")            where  "⊤^m ≡ p^m ⊃^m p^m"
 definition BotM  ("⊥^m")            where  "⊥^m ≡ ¬^m ⊤^m"
—‹Definition of truth of a formula relative to a model ‹⟨W,R,V⟩› and a possible world w›
 definition RelativeTruthM :: "w⇒σ⇒bool" ("_ ⊨^m _") where "w ⊨^m φ ≡ φ w"
—‹Definition of validity›
 definition  ValM ("⊨^m _") where "⊨^m φ  ≡ ∀w::w. w ⊨^m φ"
—‹Collection of definitions in a bag called DefM›
 named_theorems DefM declare AtmM_def[DefM,simp] NegM_def[DefM,simp] ImpM_def[DefM,simp]
  BoxM_def[DefM,simp] OrM_def[DefM,simp] AndM_def[DefM,simp] DiaM_def[DefM,simp] TopM_def[DefM,simp]
  BotM_def[DefM,simp] RelativeTruthM_def[DefM,simp] ValM_def[DefM,simp]
end
```

```
theory PMLinHOL_faithfulness  imports PMLinHOL_deep PMLinHOL_shallow PMLinHOL_shallow_minimal  (* C.B., 2025 *)
begin
—‹Mappings: deep to maximal shallow and deep to minimal shallow›
 primrec DpToShMax ("⦇_⦈") where  "⦇a^d⦈  = a^s" | "⦇¬^d φ⦈ = ¬^s ⦇φ⦈" | "⦇φ ⊃^d ψ⦈ = ⦇φ⦈ ⊃^s ⦇ψ⦈" | "⦇□^d φ⦈ = □^s ⦇φ⦈"
 primrec DpToShMin  ("⟦_⟧") where "⟦a^d⟧  = a^m" | "⟦¬^d φ⟧ = ¬^m ⟦φ⟧" | "⟦φ ⊃^d ψ⟧ = ⟦φ⟧ ⊃^m ⟦ψ⟧" | "⟦□^d φ⟧ = □^m ⟦φ⟧"
—‹Proving faithfulness between deep and maximal shallow›
 theorem Faithful1a:   "∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨^d φ ⟷ ⟨W,R,V⟩,w ⊨^s ⦇φ⦈"     apply induct by auto
 theorem Faithful1b:   "⊨^d φ ⟷ ⊨^s ⦇φ⦈"                                      using Faithful1a by auto
—‹Proving faithfulness between deep and minimal shallow›
 theorem Faithful2:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨^d φ ⟷ w ⊨^m ⟦φ⟧"           apply induct by auto
—‹Proving faithfulness maximal shallow and minimal shallow›
 theorem Faithful3:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨^s ⦇φ⦈ ⟷ w ⊨^m ⟦φ⟧"        apply induct by auto
—‹Additional check for soundness for the minimal shallow embedding›
 lemma Sound1: "⊨^m ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨^d φ)" by (smt Faithful2 DefM DefD RelativeTruthD.simps ext[of ψ "⟦x⊃^dx⟧"])
 lemma Sound2: "⊨^m ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨^m ⟦φ⟧)" using Sound1 by blast
end
```

Faithfulness automated

⦇.⦈ - Mapping of deep formulas to maximal shallow formulas

⟦.⟧ - Mapping of deep formulas to minimal shallow formulas

```
theory PMLinHOL_faithfulness  imports PMLinHOL_deep PMLinHOL_shallow PMLinHOL_shallow_minimal  (* C.B., 2025 *)
begin
—‹Mappings: deep to maximal shallow and deep to minimal shallow›
  primrec DpToShMax ("⦇_⦈") where  "⦇aᵈ⦈  = aˢ" | "⦇¬ᵈ φ⦈ = ¬ˢ ⦇φ⦈" | "⦇φ ⊃ᵈ ψ⦈ = ⦇φ⦈ ⊃ˢ ⦇ψ⦈" | "⦇□ᵈ φ⦈ = □ˢ ⦇φ⦈"
  primrec DpToShMin  ("⟦_⟧") where "⟦aᵈ⟧  = aᵐ" | "⟦¬ᵈ φ⟧ = ¬ᵐ ⟦φ⟧" | "⟦φ ⊃ᵈ ψ⟧ = ⟦φ⟧ ⊃ᵐ ⟦ψ⟧" | "⟦□ᵈ φ⟧ = □ᵐ ⟦φ⟧"
—‹Proving faithfulness between deep and maximal shallow›
  theorem Faithful1a:    "∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨ᵈ φ ⟷ ⟨W,R,V⟩,w ⊨ˢ ⦇φ⦈"     apply induct by auto
  theorem Faithful1b:    "⊨ᵈ φ ⟷ ⊨ˢ ⦇φ⦈"                                        using Faithful1a by auto
—‹Proving faithfulness between deep and minimal shallow›
  theorem Faithful2:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨ᵈ φ ⟷ w ⊨ᵐ ⟦φ⟧"             apply induct by auto
—‹Proving faithfulness maximal shallow and minimal shallow›
  theorem Faithful3:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨ˢ ⦇φ⦈ ⟷ w ⊨ᵐ ⟦φ⟧"          apply induct by auto
—‹Additional check for soundness for the minimal shallow embedding›
  lemma Sound1: "⊨ᵐ ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨ᵈ φ)" by (smt Faithful2 DefM DefD RelativeTruthD.simps ext[of ψ "⟦x⊃ᵈx⟧"])
  lemma Sound2: "⊨ᵐ ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨ᵐ ⟦φ⟧)" using Sound1 by blast
end
```

Faithfulness automated

A deep formula is valid if and only if its maximal shallow translation is valid.

```
theory PMLinHOL_faithfulness  imports PMLinHOL_deep PMLinHOL_shallow PMLinHOL_shallow_minimal  (* C.B., 2025 *)
begin
—‹Mappings: deep to maximal shallow and deep to minimal shallow›
  primrec DpToShMax ("⦇_⦈") where  "⦇a^d⦈  = a^s" | "⦇¬^d φ⦈ = ¬^s ⦇φ⦈" | "⦇φ ⊃^d ψ⦈ = ⦇φ⦈ ⊃^s ⦇ψ⦈" | "⦇□^d φ⦈ = □^s ⦇φ⦈"
  primrec DpToShMin  ("⟦_⟧") where "⟦a^d⟧  = a^m" | "⟦¬^d φ⟧ = ¬^m ⟦φ⟧" | "⟦φ ⊃^d ψ⟧ = ⟦φ⟧ ⊃^m ⟦ψ⟧" | "⟦□^d φ⟧ = □^m ⟦φ⟧"
—‹Proving faithfulness between deep and maximal shallow›
  theorem Faithful1a:    "∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨^d φ ⟷ ⟨W,R,V⟩,w ⊨^s ⦇φ⦈"     apply induct by auto
  theorem Faithful1b:    "⊨^d φ ⟷ ⊨^s ⦇φ⦈"                                     using Faithful1a by auto
—‹Proving faithfulness between deep and minimal shallow›
  theorem Faithful2:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨^d φ ⟷ w ⊨^m ⟦φ⟧"            apply induct by auto
—‹Proving faithfulness maximal shallow and minimal shallow›
  theorem Faithful3:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨^s ⦇φ⦈ ⟷ w ⊨^m ⟦φ⟧"           apply induct by auto
—‹Additional check for soundness for the minimal shallow embedding›
  lemma Sound1: "⊨^m ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨^d φ)" by (smt Faithful2 DefM DefD RelativeTruthD.simps ext[of ψ "⟦x⊃^dx⟧"])
  lemma Sound2: "⊨^m ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨^m ⟦φ⟧)" using Sound1 by blast
end
```

Faithfulness automated

Similar relative faithfulness theorems for the
- deep and minimal shallow embedding, and for the
- maximal shallow and minimal shallow embedding

```
theory PMLinHOL_faithfulness  imports PMLinHOL_deep PMLinHOL_shallow PMLinHOL_shallow_minimal  (* C.B., 2025 *)
begin
—‹Mappings: deep to maximal shallow and deep to minimal shallow›
  primrec DpToShMax ("⦇_⦈") where  "⦇aᵈ⦈  = aˢ" | "⦇¬ᵈ φ⦈ = ¬ˢ ⦇φ⦈" | "⦇φ ⊃ᵈ ψ⦈ = ⦇φ⦈ ⊃ˢ ⦇ψ⦈" | "⦇□ᵈ φ⦈ = □ˢ ⦇φ⦈"
  primrec DpToShMin  ("⟦_⟧") where "⟦aᵈ⟧  = aᵐ" | "⟦¬ᵈ φ⟧ = ¬ᵐ ⟦φ⟧" | "⟦φ ⊃ᵈ ψ⟧ = ⟦φ⟧ ⊃ᵐ ⟦ψ⟧" | "⟦□ᵈ φ⟧ = □ᵐ ⟦φ⟧"
—‹Proving faithfulness between deep and maximal shallow›
  theorem Faithful1a:    "∀W R V. ∀w:W. ⟨W,R,V⟩,w ⊨ᵈ φ ⟷ ⟨W,R,V⟩,w ⊨ˢ ⦇φ⦈"     apply induct by auto
  theorem Faithful1b:    "⊨ᵈ φ ⟷ ⊨ˢ ⦇φ⦈"                                      using Faithful1a by auto
—‹Proving faithfulness between deep and minimal shallow›
  theorem Faithful2:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨ᵈ φ ⟷ w ⊨ᵐ ⟦φ⟧"            apply induct by auto
—‹Proving faithfulness maximal shallow and minimal shallow›
  theorem Faithful3:     "∀w. ⟨(λx::w. True),R,V⟩,w ⊨ˢ ⦇φ⦈ ⟷ w ⊨ᵐ ⟦φ⟧"           apply induct by auto
—‹Additional check for soundness for the minimal shallow embedding›
  lemma Sound1: "⊨ᵐ ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨ᵈ φ)" by (smt Faithful2 DefM DefD RelativeTruthD.simps ext[of ψ "⟦x⊃ᵈx⟧"])
  lemma Sound2: "⊨ᵐ ψ ⟷ (∃φ. ψ=⟦φ⟧ ∧ ⊨ᵐ ⟦φ⟧)" using Sound1 by blast
end
```

Experiments

Conducted with all embeddings — but presented here for the minimal shallow embedding

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal          (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨^m φ ⊃^m (ψ ⊃^m φ)" by auto
 lemma H2: "⊨^m (φ ⊃^m (ψ ⊃^m γ)) ⊃^m ((φ ⊃^m ψ) ⊃^m (φ ⊃^m γ)) " by auto
 lemma H3: "⊨^m (¬^mφ  ⊃^m ¬^mψ) ⊃^m (ψ ⊃^m φ)" by auto
 lemma MP: "⊨^m φ ⟹ ⊨^m (φ  ⊃^m ψ) ⟹ ⊨^m ψ" by auto
```

Hilbert axioms and modus ponens rule are implied

$$\text{H1:}\quad \vDash \varphi \supset (\psi \supset \varphi)$$

$$\text{H2:}\quad \vDash (\varphi \supset (\psi \supset \gamma)) \supset ((\varphi \supset \psi) \supset (\varphi \supset \gamma))$$

$$\text{H3:}\quad \vDash (\neg\varphi \supset \neg\psi) \supset (\psi \supset \varphi)$$

$$\text{MP:}\quad \vDash \varphi \quad \Longrightarrow \quad \vDash \varphi \supset \psi \quad \Longrightarrow \quad \vDash \psi$$

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal          (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨ᵐ φ ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
 lemma H2: "⊨ᵐ (φ ⊃ᵐ (ψ ⊃ᵐ γ)) ⊃ᵐ ((φ ⊃ᵐ ψ) ⊃ᵐ (φ ⊃ᵐ γ)) " by auto
 lemma H3: "⊨ᵐ (¬ᵐφ  ⊃ᵐ ¬ᵐψ) ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
 lemma MP: "⊨ᵐ φ ⟹ ⊨ᵐ (φ  ⊃ᵐ ψ) ⟹ ⊨ᵐ ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
 lemma HCderived1: "⊨ᵐ (φ ⊃ᵐ φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
```

Simple automated proof of $\varphi \supset \varphi$ at level of this Hilbert calculus

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal          (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨ᵐ φ ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
 lemma H2: "⊨ᵐ (φ ⊃ᵐ (ψ ⊃ᵐ γ)) ⊃ᵐ ((φ ⊃ᵐ ψ) ⊃ᵐ (φ ⊃ᵐ γ)) " by auto
 lemma H3: "⊨ᵐ (¬ᵐφ  ⊃ᵐ ¬ᵐψ) ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
 lemma MP: "⊨ᵐ φ ⟹ ⊨ᵐ (φ  ⊃ᵐ ψ) ⟹ ⊨ᵐ ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
 lemma HCderived1: "⊨ᵐ (φ ⊃ᵐ φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
  proof -  have 1: "⊨ᵐ φ ⊃ᵐ ((ψ ⊃ᵐ φ) ⊃ᵐ φ)" using H1 by auto
           have 2: "⊨ᵐ (φ ⊃ᵐ ((ψ ⊃ᵐ φ) ⊃ᵐ φ)) ⊃ᵐ ((φ ⊃ᵐ (ψ ⊃ᵐ φ)) ⊃ᵐ (φ ⊃ᵐ φ))" using H2 by auto
           have 3: "⊨ᵐ (φ ⊃ᵐ (ψ ⊃ᵐ φ)) ⊃ᵐ (φ ⊃ᵐ φ)" using 1 2 MP by meson
           have 4: "⊨ᵐ φ ⊃ᵐ (ψ  ⊃ᵐ φ)" using H1 by auto
           thus ?thesis using 3 4 MP by meson qed
```

Interactive proof of $\varphi \supset \varphi$ in this Hilbert calculus

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal        (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨m φ ⊃m (ψ ⊃m φ)" by auto
 lemma H2: "⊨m (φ ⊃m (ψ ⊃m γ)) ⊃m ((φ ⊃m ψ) ⊃m (φ ⊃m γ)) " by auto
 lemma H3: "⊨m (¬mφ  ⊃m ¬mψ) ⊃m (ψ ⊃m φ)" by auto
 lemma MP: "⊨m φ ⟹ ⊨m (φ  ⊃m ψ) ⟹ ⊨m ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
 lemma HCderived1: "⊨m (φ ⊃m φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
  proof -  have 1: "⊨m φ ⊃m ((ψ ⊃m φ) ⊃m φ)" using H1 by auto
           have 2: "⊨m (φ ⊃m ((ψ ⊃m φ) ⊃m φ)) ⊃m ((φ ⊃m (ψ ⊃m φ)) ⊃m (φ ⊃m φ))" using H2 by auto
           have 3: "⊨m (φ ⊃m (ψ ⊃m φ)) ⊃m (φ ⊃m φ)" using 1 2 MP by meson
           have 4: "⊨m φ ⊃m (ψ  ⊃m φ)" using H1 by auto
           thus ?thesis using 3 4 MP by meson qed
 lemma HCderived2: "⊨m φ ⊃m (¬mφ  ⊃m ψ) " by (metis H1 H2 H3 MP)
 lemma HCderived3: "⊨m (¬mφ  ⊃m φ) ⊃m φ" by (metis H1 H2 H3 MP)
 lemma HCderived4: "⊨m (φ ⊃m ψ) ⊃m (¬mψ ⊃m ¬mφ) " by auto
```

Further automated proofs of simple example problems

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal         (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨^m φ ⊃^m (ψ ⊃^m φ)" by auto
 lemma H2: "⊨^m (φ ⊃^m (ψ ⊃^m γ)) ⊃^m ((φ ⊃^m ψ) ⊃^m (φ ⊃^m γ)) " by auto
 lemma H3: "⊨^m (¬^mφ  ⊃^m ¬^mψ) ⊃^m (ψ ⊃^m φ)" by auto
 lemma MP: "⊨^m φ ⟹ ⊨^m (φ  ⊃^m ψ) ⟹ ⊨^m ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
 lemma HCderived1: "⊨^m (φ ⊃^m φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
  proof -  have 1: "⊨^m φ ⊃^m ((ψ ⊃^m φ) ⊃^m φ)" using H1 by auto
           have 2: "⊨^m (φ ⊃^m ((ψ ⊃^m φ) ⊃^m φ)) ⊃^m ((φ ⊃^m (ψ ⊃^m φ)) ⊃^m (φ ⊃^m φ))" using H2 by auto
           have 3: "⊨^m (φ ⊃^m (ψ ⊃^m φ)) ⊃^m (φ ⊃^m φ)" using 1 2 MP by meson
           have 4: "⊨^m φ ⊃^m (ψ  ⊃^m φ)" using H1 by auto
           thus ?thesis using 3 4 MP by meson qed
 lemma HCderived2: "⊨^m φ ⊃^m (¬^mφ  ⊃^m ψ) " by (metis H1 H2 H3 MP)
 lemma HCderived3: "⊨^m (¬^mφ  ⊃^m φ) ⊃^m φ" by (metis H1 H2 H3 MP)
 lemma HCderived4: "⊨^m (φ ⊃^m ψ) ⊃^m (¬^mψ ⊃^m ¬^mφ) " by auto
—‹Modal logic: the schematic necessitation rule and distribution axiom are implied›
 lemma Nec: "⊨^m φ ⟹ ⊨^m □^mφ" by (smt DefM)
 lemma Dist:  "⊨^m □^m(φ ⊃^m ψ) ⊃^m (□^mφ ⊃^m □^mψ) " by auto
```

Necessitation rule and distribution axiom are implied

Nec: $\models \varphi \quad \Longrightarrow \quad \models \Box\varphi$

Dist: $\models \Box(\varphi \supset \psi) \supset (\Box\varphi \supset \Box\psi)$

Experiments

```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal        (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
 lemma H1: "⊨m φ ⊃m (ψ ⊃m φ)" by auto
 lemma H2: "⊨m (φ ⊃m (ψ ⊃m γ)) ⊃m ((φ ⊃m ψ) ⊃m (φ ⊃m γ)) " by auto
 lemma H3: "⊨m (¬mφ  ⊃m ¬mψ) ⊃m (ψ ⊃m φ)" by auto
 lemma MP: "⊨m φ ⟹ ⊨m (φ  ⊃m ψ) ⟹ ⊨m ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
 lemma HCderived1: "⊨m (φ ⊃m φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
  proof -  have 1: "⊨m φ ⊃m ((ψ ⊃m φ) ⊃m φ)" using H1 by auto
           have 2: "⊨m (φ ⊃m ((ψ ⊃m φ) ⊃m φ)) ⊃m ((φ ⊃m (ψ ⊃m φ)) ⊃m (φ ⊃m φ))" using H2 by auto
           have 3: "⊨m (φ ⊃m (ψ ⊃m φ)) ⊃m (φ ⊃m φ)" using 1 2 MP by meson
           have 4: "⊨m φ ⊃m (ψ  ⊃m φ)" using H1 by auto
           thus ?thesis using 3 4 MP by meson qed
 lemma HCderived2: "⊨m φ ⊃m (¬mφ  ⊃m ψ) " by (metis H1 H2 H3 MP)
 lemma HCderived3: "⊨m (¬mφ  ⊃m φ) ⊃m φ" by (metis H1 H2 H3 MP)
 lemma HCderived4: "⊨m (φ ⊃m ψ) ⊃m (¬mψ ⊃m ¬mφ) " by auto
—‹Modal logic: the schematic necessitation rule and distribution axiom are implied›
 lemma Nec: "⊨m φ ⟹ ⊨m □mφ" by (smt DefM)
 lemma Dist:  "⊨m □m(φ ⊃m ψ) ⊃m (□mφ ⊃m □mψ) " by auto
—‹Correspondence theory: correct statements›
 lemma cM:  "reflexive R ⟷ (∀φ. ⊨m □mφ ⊃m φ)" by auto
 lemma cBa: "symmetric R ⟶ (∀φ. ⊨m φ ⊃m □m◇mφ)" by auto
 lemma cBb: "symmetric R ⟵ (∀φ. ⊨m φ ⊃m □m◇mφ)" by (metis DefM)
 lemma c4a: "transitive R ⟶ (∀φ. ⊨m □mφ ⊃m □m(□mφ))" by (smt DefM)
 lemma c4b: "transitive R ⟵ (∀φ. ⊨m □mφ ⊃m □m(□mφ))" by auto
```

Sahlqvist correspondences automated

Experiments

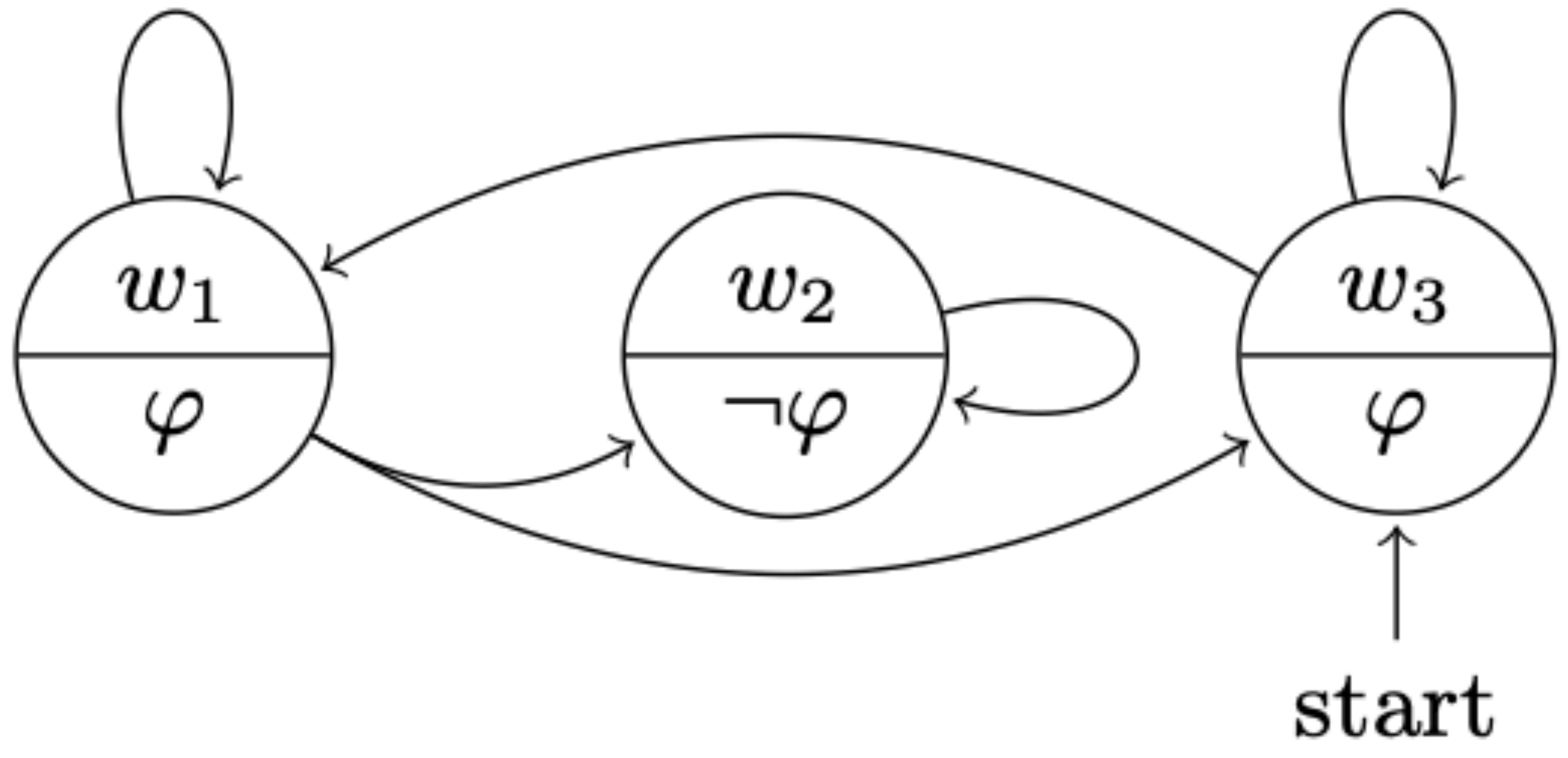
```
theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal        (* Christoph Benzmüller, 2025 *)
begin
—‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
  lemma H1: "⊨^m φ ⊃^m (ψ ⊃^m φ)" by auto
  lemma H2: "⊨^m (φ ⊃^m (ψ ⊃^m γ)) ⊃^m ((φ ⊃^m ψ) ⊃^m (φ ⊃^m γ)) " by auto
  lemma H3: "⊨^m (¬^mφ  ⊃^m ¬^mψ) ⊃^m (ψ ⊃^m φ)" by auto
  lemma MP: "⊨^m φ ⟹ ⊨^m (φ  ⊃^m ψ) ⟹ ⊨^m ψ" by auto
—‹Reasoning with the Hilbert calculus: interactive and fully automated›
  lemma HCderived1: "⊨^m (φ ⊃^m φ)" —‹sledgehammer(H1 H2 H3 MP) returns: by (metis H1 H2 MP)›
   proof -  have 1: "⊨^m φ ⊃^m ((ψ ⊃^m φ) ⊃^m φ)" using H1 by auto
            have 2: "⊨^m (φ ⊃^m ((ψ ⊃^m φ) ⊃^m φ)) ⊃^m ((φ ⊃^m (ψ ⊃^m φ)) ⊃^m (φ ⊃^m φ))" using H2 by auto
            have 3: "⊨^m (φ ⊃^m (ψ ⊃^m φ)) ⊃^m (φ ⊃^m φ)" using 1 2 MP by meson
            have 4: "⊨^m φ ⊃^m (ψ  ⊃^m φ)" using H1 by auto
            thus ?thesis using 3 4 MP by meson qed
  lemma HCderived2: "⊨^m φ ⊃^m (¬^mφ  ⊃^m ψ) " by (metis H1 H2 H3 MP)
  lemma HCderived3: "⊨^m (¬^mφ  ⊃^m φ) ⊃^m φ" by (metis H1 H2 H3 MP)
  lemma HCderived4: "⊨^m (φ ⊃^m ψ) ⊃^m (¬^mψ ⊃^m ¬^mφ) " by auto
—‹Modal logic: the schematic necessitation rule and distribution axiom are implied›
  lemma Nec: "⊨^m φ ⟹ ⊨^m □^mφ" by (smt DefM)
  lemma Dist:  "⊨^m □^m(φ ⊃^m ψ) ⊃^m (□^mφ ⊃^m □^mψ) " by auto
—‹Correspondence theory: correct statements›
  lemma cM:  "reflexive R ⟷ (∀φ. ⊨^m □^mφ ⊃^m φ)" by auto
  lemma cBa: "symmetric R ⟶ (∀φ. ⊨^m φ ⊃^m □^m◇^mφ)" by auto
  lemma cBb: "symmetric R ⟵ (∀φ. ⊨^m φ ⊃^m □^m◇^mφ)" by (metis DefM)
  lemma c4a: "transitive R ⟶ (∀φ. ⊨^m □^mφ ⊃^m □^m(□^mφ))" by (smt DefM)
  lemma c4b: "transitive R ⟵ (∀φ. ⊨^m □^mφ ⊃^m □^m(□^mφ))" by auto
—‹Correspondence theory: incorrect statements›
  lemma "reflexive R ⟶ (∀φ. ⊨^m □^mφ ⊃^m □^m(□^mφ))" nitpick[card w=3,show_all] oops —‹nitpick: Counterexample›
```

Incorrect correspondence claim

$$\text{reflexive}\, R \longrightarrow (\forall \varphi\, .\, \vDash \Box \varphi \supset \Box\Box \varphi)$$

Experiments

```
1 theory PMLinHOL_shallow_minimal_tests imports PMLinHOL_shallow_minimal        (* Christoph Benzmüller, 2025 *)
2 begin
3 —‹Hilbert calculus: proving that the schematic axioms and rules are implied by the embedding›
4   lemma H1: "⊨ᵐ φ ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
5   lemma H2: "⊨ᵐ (φ ⊃ᵐ (ψ ⊃ᵐ γ)) ⊃ᵐ ((φ ⊃ᵐ ψ) ⊃ᵐ (φ ⊃ᵐ γ)) " by auto
6   lemma H3: "⊨ᵐ (¬ᵐφ  ⊃ᵐ ¬ᵐψ) ⊃ᵐ (ψ ⊃ᵐ φ)" by auto
7   lemma MP: "⊨ᵐ φ ⟹ ⊨ᵐ (φ  ⊃ᵐ ψ) ⟹ ⊨ᵐ ψ" by auto
```




```
Nitpick found a counterexample for card w = 3:
  Skolem constants:
    ??.BoxM.v = w₁
    ??.BoxM.v = w₂
    ??.ValM.w = w₃
    φ = (λx. _)(w₁ := True, w₂ := False, w₃ := True)
  Constant:
    R =
      (λx. _)
      (w₁ := (λx. _)(w₁ := True, w₂ := True, w₃ := True),
       w₂ := (λx. _)(w₁ := False, w₂ := True, w₃ := False),
       w₃ := (λx. _)(w₁ := True, w₂ := False, w₃ := True))
```

Fig. 5. Counterexample reported by **nitpick** in line 28 of Fig. 4

[See paper at LPLR 2025]

```
27 —‹Correspondence theory: incorrect statements›
28   lemma "reflexive R ⟶ (∀φ. ⊨ᵐ □ᵐφ ⊃ᵐ □ᵐ(□ᵐφ))" nitpick[card w=3,show_all] oops —‹nitpick: Counterexample›
```

Incorrect correspondence claim — counterexample by Nitpick

$$\text{reflexive}\, R \longrightarrow (\forall\varphi\,.\, \vDash \Box\varphi \supset \Box\Box\varphi)$$

Experiments

modal reasoning

```
theory PMLinHOL_shallow_minimal_further_tests imports PMLinHOL_shallow_minimal_tests          (* C.B., 2025 *)
begin
—‹Implied modal principle›
 lemma K_Dia: "⊨ᵐ (□ᵐ(φ ⊃ᵐ ψ)) ⊃ᵐ ((◇ᵐφ) ⊃ᵐ ◇ᵐψ)"    by auto
—‹Example 6.10 of Sider (2009) Logic for Philosophy›
 lemma T1a: "⊨ᵐ □ᵐpᵐ ⊃ᵐ ((◇ᵐqᵐ) ⊃ᵐ ◇ᵐ (pᵐ ∧ᵐ qᵐ))" by auto —‹fast automation in meta-logic HOL›
 lemma T1b: "⊨ᵐ □ᵐpᵐ ⊃ᵐ ((◇ᵐqᵐ) ⊃ᵐ ◇ᵐ (pᵐ ∧ᵐ qᵐ))"
  proof -                                   —‹alternative interactive proof in modal object logic K›
   have 1: "⊨ᵐ pᵐ ⊃ᵐ (qᵐ ⊃ᵐ (pᵐ ∧ᵐ qᵐ))"          unfolding AndM_def using H1 H2 H3 MP by metis
   have 2: "⊨ᵐ □ᵐ(pᵐ ⊃ᵐ (qᵐ ⊃ᵐ (pᵐ ∧ᵐ qᵐ)))"                         using 1 Nec      by metis
   have 3: "⊨ᵐ □ᵐpᵐ ⊃ᵐ □ᵐ(qᵐ ⊃ᵐ (pᵐ ∧ᵐ qᵐ))"                         using 2 Dist MP  by metis
   have 4: "⊨ᵐ (□ᵐ(qᵐ ⊃ᵐ (pᵐ ∧ᵐ qᵐ))) ⊃ᵐ ((◇ᵐqᵐ) ⊃ᵐ ◇ᵐ(pᵐ ∧ᵐ qᵐ))"  using K_Dia      by metis
   have 5: "⊨ᵐ □ᵐpᵐ ⊃ᵐ ((◇ᵐqᵐ) ⊃ᵐ ◇ᵐ (pᵐ ∧ᵐ qᵐ))"                    using 3 4 H1 H2 MP by metis
   thus ?thesis . qed
end
```

inspecting, and reasoning with, the Löb formula

```
theory PMLinHOL_shallow_minimal_Loeb_tests  imports PMLinHOL_shallow_minimal
begin
 —‹Löb axiom: with the minimal shallow embedding automated reasoning tools are still partly responsive›
 lemma Loeb1M:  "∀φ. ⊨ᵐ □ᵐ( □ᵐφ ⊃ᵐφ) ⊃ᵐ □ᵐφ" nitpick[card w=1,card 𝒮=1] oops —‹nitpick: Counterexample.›
 lemma Loeb2M:  "(conversewf R ∧ transitive R) ⟶ (∀φ. ⊨ᵐ □ᵐ(□ᵐφ ⊃ᵐφ) ⊃ᵐ □ᵐφ)"
  by (smt (verit, del_insts) BoxM_def ImpM_def RelativeTruthM_def ValM_def)
 lemma Loeb3aM: "conversewf R ⟵ (∀φ. ⊨ᵐ □ᵐ(□ᵐφ ⊃ᵐφ) ⊃ᵐ □ᵐφ)" unfolding DefM by blast
 lemma Loeb3bM: "transitive R ⟵ (∀φ. ⊨ᵐ □ᵐ(□ᵐφ ⊃ᵐφ) ⊃ᵐ □ᵐφ)"  —‹sledgehammer: no proof›
  proof assume 1: "∀φ. ⊨ᵐ □ᵐ (□ᵐ φ ⊃ᵐ φ) ⊃ᵐ □ᵐ φ"
      have 2: "∀φ. ⊨ᵐ □ᵐφ ⊃ᵐ □ᵐ((□ᵐ(□ᵐφ ∧ᵐ φ)) ⊃ᵐ ((□ᵐφ) ∧ᵐ φ))" by auto
      show "transitive R"  by (smt (z3) 1 2 DefM(10,11,2,3,4,6)) qed
 lemma Loeb3M: "(conversewf R ∧ transitive R) ⟵ (∀φ. ⊨ᵐ □ᵐ(□ᵐφ ⊃ᵐφ) ⊃ᵐ □ᵐφ)"
  using Loeb3aM Loeb3bM by force
end
```

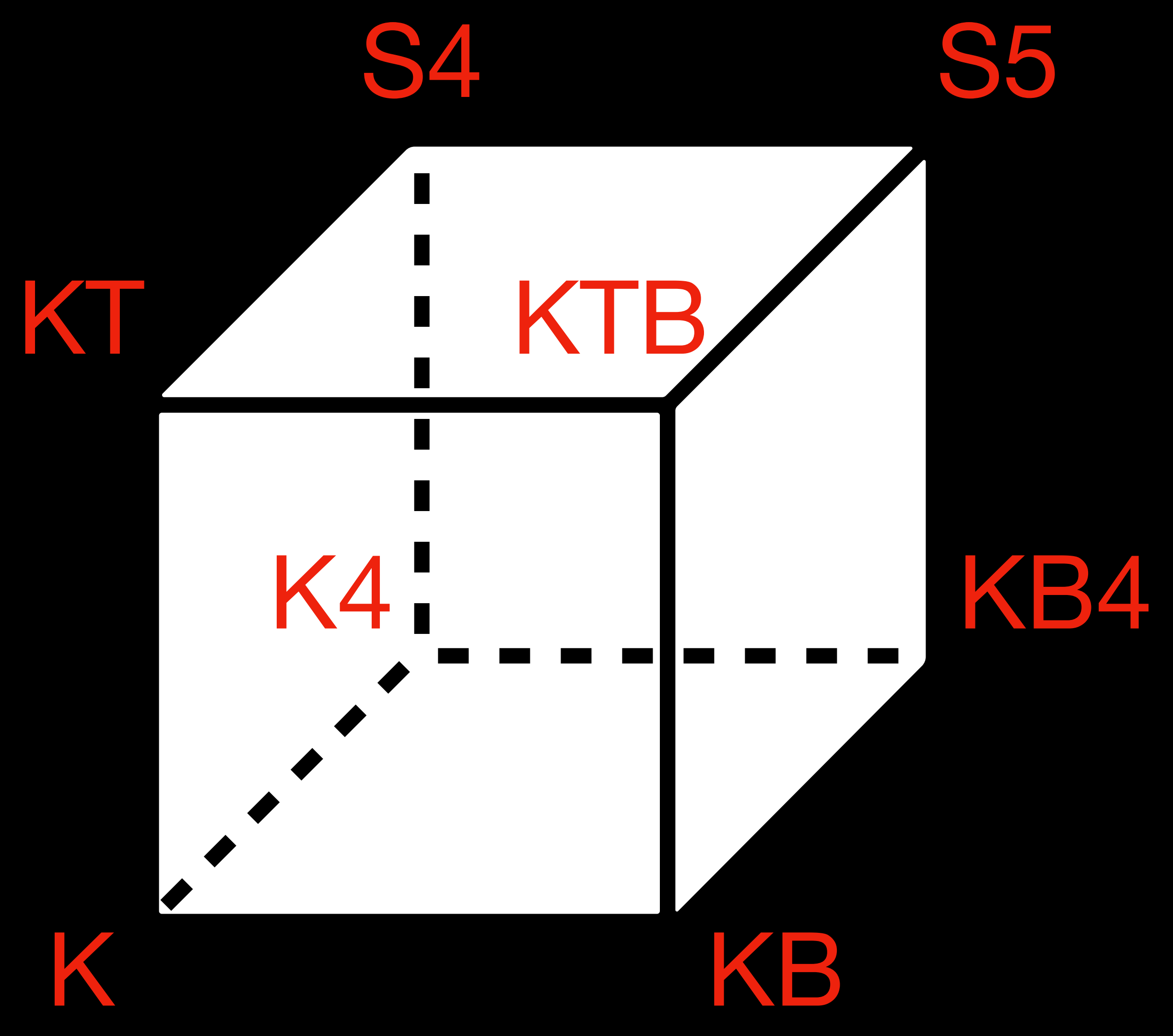
PML Evaluation

(new — extending results from CADE 2025)

Evaluation

F_1: $\Diamond\Diamond\varphi \supset \Diamond\varphi$
F_2: $\Diamond\Box\varphi \supset \Box\Diamond\varphi$
F_3: $\Diamond\Box\varphi \supset \Box\varphi$
F_4: $\Box\Diamond\Box\Diamond\varphi \supset \Box\Diamond\varphi$
F_5: $\Diamond(\varphi \wedge \Diamond\psi)) \supset (\Diamond\varphi \wedge \Diamond\psi)$
F_6: $((\Box(\varphi \supset \psi)) \wedge \Diamond\Box\neg\psi) \supset \neg\Diamond\psi$
F_7: $\Diamond\varphi \supset \Box(\varphi \vee \Diamond\varphi)$
F_8: $\Diamond(\Box\varphi) \supset (\varphi \vee \Diamond\varphi)$
F_9: $(\Box\Diamond\varphi \wedge \Box\Diamond\neg\varphi) \supset \Diamond\Diamond\varphi$
F_{10}: $(\Box(\varphi \supset \Box\psi) \wedge \Box\Diamond\neg\psi) \supset \neg\Box\psi$

(schematic)
problem formulas



AxT:	$\forall\varphi\,.\vDash \Box\varphi \supset \varphi$
AxB:	$\forall\varphi\,.\vDash \varphi \supset \Box\Diamond\varphi$
$Ax4$:	$\forall\varphi\,.\vDash \Box\varphi \supset \Box\Box\varphi$

$SemT$:	reflexive(R)
$SemB$:	symmetric(R)
$Sem4$:	transitive(R)

S5:	$AxT \wedge AxB \wedge Ax4$	$\vDash F_n$
KTB:	$AxT \wedge AxB$	$\vDash F_n$
S4:	$AxT \wedge$ $Ax4$	$\vDash F_n$
KB4:	$AxB \wedge Ax4$	$\vDash F_n$
KT:	AxT	$\vDash F_n$
KB:	AxB	$\vDash F_n$
K4:	$Ax4$	$\vDash F_n$
K:		$\vDash F_n$

80 problems
with/without “apply simp”
160 calls to sledgehammer and nitpick

$M \vDash F$

S5:	$SemT \wedge SemB \wedge Sem4$	$\vDash F_n$
KTB:	$SemT \wedge SemB$	$\vDash F_n$
S4:	$SemT \wedge$ $Sem4$	$\vDash F_n$
KB4:	$SemB \wedge Sem4$	$\vDash F_n$
KT:	$SemT$	$\vDash F_n$
KB:	$SemB$	$\vDash F_n$
K4:	$Sem4$	$\vDash F_n$
K:		$\vDash F_n$

80 problems
with/without “apply simp”
160 calls to sledgehammer and nitpick

$N \vDash F$

	nitpick			sledgehammer	
	A	B	C	D	E
Problems	counterex.	no counterex.	unknown	proof found	no proof
$M \models^m F$, $N \models^m F$		minimal shallow		minimal shallow	
$M \models^s F$, $N \models^s F$		maximal shallow		maximal shallow	
$M \models^d F$, $N \models^d F$		deep		deep	

Sledgehammer and Nitpick were run in default settings in Isabelle 2025

Evaluation

A MacBook Pro 16,1 (6-Core Intel Core i7 processor, 2,6 GHz, 6 Cores, 256 KB L2 Cache per Core, 12 MB L3 Cache, 16 GB Memory) and Isabelle 2025-1-RC1 in standard configuration was used.

		sledgehammer[14] (38 THM)									nitpick (42 CSA)				
		1s	2s	5s	10s	20s	40s	80s	$\sum$	%	2s	10s	40s	$\sum$	%
$M \models^m F$		9	9	12	19	24	34	37	144	45%	42	42	42	126	100%
$M \models^m F$	simp	18	36	37	**38**	**38**	**38**	**38**	243	91%	42	42	42	126	100%
$N \models^m F$		38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$N \models^m F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^s F$		0	0	0	23	23	24	29	99	37%	16	16	16	48	38%
$M \models^s F$	simp	7	26	26	28	28	28	28	181	68%	16	16	16	48	38%
$N \models^s F$		7	7	8	37	37	37	37	170	64%	42	42	42	126	100%
$N \models^s F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^d F$		7	12	21	24	26	30	30	150	56%	0	6	6	12	10%
$M \models^d F$	simp	0	7	13	26	28	28	28	130	49%	2	8	8	18	14%
$N \models^d F$		25	38	38	38	38	38	38	253	95%	6	24	24	54	43%
$N \models^d F$	simp	30	32	38	38	38	38	38	252	95%	2	8	8	18	14%

Fig. 10. Performance results of the automated PML formula classification task conducted with sledgehammer and nitpick; for the problem set $M \models^m F$ a perfect classification is achieved (with or without the use of simp) already for a very short timeout of 2s (and also 1s), meaning that for each formula F in this mode either a proof is reported, if F is a theorem (THM), or a counterexample is presented, if F is a countersatisfiable (CSA). There were no conflicting results reported by sledgehammer and nitpick.

minimal shallow with modal axioms

minimal shallow with semantic axioms

maximal shallow with modal axioms

maximal shallow with semantic axioms

deep with modal axioms

deep with semantic axioms

Results

- Winner(s) — perfect classification — “no counterex.”/“proof” for all valid statement and “counterex.” else

Evaluation

A MacBook Pro 16,1 (6-Core Intel Core i7 processor, 2,6 GHz, 6 Cores, 256 KB L2 Cache per Core, 12 MB L3 Cache, 16 GB Memory) and Isabelle 2025-1-RC1 in standard configuration was used.

		sledgehammer[14] (38 THM)									nitpick (42 CSA)				
		1s	2s	5s	10s	20s	40s	80s	$\sum$	%	2s	10s	40s	$\sum$	%
$M \models^m F$		9	9	12	19	24	34	37	144	45%	42	42	42	126	100%
$M \models^m F$	simp	18	36	37	**38**	**38**	**38**	**38**	243	91%	42	42	42	126	100%
$N \models^m F$		38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$N \models^m F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^s F$		0	0	0	23	23	24	29	99	37%	16	16	16	48	38%
$M \models^s F$	simp	7	26	26	28	28	28	28	181	68%	16	16	16	48	38%
$N \models^s F$		7	7	8	37	37	37	37	170	64%	42	42	42	126	100%
$N \models^s F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^d F$		7	12	21	24	26	30	30	150	56%	0	6	6	12	10%
$M \models^d F$	simp	0	7	13	26	28	28	28	130	49%	2	8	8	18	14%
$N \models^d F$		25	38	38	38	38	38	38	253	95%	6	24	24	54	43%
$N \models^d F$	simp	30	32	38	38	38	38	38	252	95%	2	8	8	18	14%

Fig. 10. Performance results of the automated PML formula classification task conducted with sledgehammer and nitpick; for the problem set $M \models^m F$ a perfect classification is achieved (with or without the use of simp) already for a very short timeout of 2s (and also 1s), meaning that for each formula F in this mode either a proof is reported, if F is a theorem (THM), or a counterexample is presented, if F is a countersatisfiable (CSA). There were no conflicting results reported by sledgehammer and nitpick.

minimal shallow with modal axioms

minimal shallow with semantic axioms

maximal shallow with modal axioms

maximal shallow with semantic axioms

deep with modal axioms

deep with semantic axioms

Results

- Winner — perfect classification — “no counterex.”/“proof” for all valid statement and “counterex.” else
- Second best

Evaluation

A MacBook Pro 16,1 (6-Core Intel Core i7 processor, 2,6 GHz, 6 Cores, 256 KB L2 Cache per Core, 12 MB L3 Cache, 16 GB Memory) and Isabelle 2025-1-RC1 in standard configuration was used.

		sledgehammer[14] (38 THM)									nitpick (42 CSA)				
		1s	2s	5s	10s	20s	40s	80s	$\sum$	%	2s	10s	40s	$\sum$	%
$M \models^m F$		9	9	12	19	24	34	37	144	45%	42	42	42	126	100%
$M \models^m F$	simp	18	36	37	**38**	**38**	**38**	**38**	243	91%	42	42	42	126	100%
$N \models^m F$		38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$N \models^m F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^s F$		0	0	0	23	23	24	29	99	37%	16	16	16	48	38%
$M \models^s F$	simp	7	26	26	28	28	28	28	181	68%	16	16	16	48	38%
$N \models^s F$		7	7	8	37	37	37	37	170	64%	42	42	42	126	100%
$N \models^s F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^d F$		7	12	21	24	26	30	30	150	56%	0	6	6	12	10%
$M \models^d F$	simp	0	7	13	26	28	28	28	130	49%	2	8	8	18	14%
$N \models^d F$		25	38	38	38	38	38	38	253	95%	6	24	24	54	43%
$N \models^d F$	simp	30	32	38	38	38	38	38	252	95%	2	8	8	18	14%

Fig. 10. Performance results of the automated PML formula classification task conducted with sledgehammer and nitpick; for the problem set $M \models^m F$ a perfect classification is achieved (with or without the use of simp) already for a very short timeout of 2s (and also 1s), meaning that for each formula F in this mode either a proof is reported, if F is a theorem (THM), or a counterexample is presented, if F is a countersatisfiable (CSA). There were no conflicting results reported by sledgehammer and nitpick.

minimal shallow with modal axioms

minimal shallow with semantic axioms

maximal shallow with modal axioms

maximal shallow with semantic axioms

deep with modal axioms

deep with semantic axioms

Results

- Winner — perfect classification — “no counterex.”/“proof” for all valid statement and “counterex.” else
- Second best
- Third best

Evaluation

A MacBook Pro 16,1 (6-Core Intel Core i7 processor, 2,6 GHz, 6 Cores, 256 KB L2 Cache per Core, 12 MB L3 Cache, 16 GB Memory) and Isabelle 2025-1-RC1 in standard configuration was used.

		sledgehammer[14] (38 THM)									nitpick (42 CSA)				
		1s	2s	5s	10s	20s	40s	80s	Σ	%	2s	10s	40s	Σ	%
$M \models^m F$		9	9	12	19	24	34	37	144	45%	42	42	42	126	100%
$M \models^m F$	simp	18	36	37	**38**	**38**	**38**	**38**	243	91%	42	42	42	126	100%
$N \models^m F$		38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$N \models^m F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^s F$		0	0	0	23	23	24	29	99	37%	16	16	16	48	38%
$M \models^s F$	simp	7	26	26	28	28	28	28	181	68%	16	16	16	48	38%
$N \models^s F$		7	7	8	37	37	37	37	170	64%	42	42	42	126	100%
$N \models^s F$	simp	38	38	38	38	38	38	38	266	100%	42	42	42	126	100%
$M \models^d F$		7	12	21	24	26	30	30	150	56%	0	6	6	12	10%
$M \models^d F$	simp	0	7	13	26	28	28	28	130	49%	2	8	8	18	14%
$N \models^d F$		25	**38**	**38**	**38**	**38**	**38**	**38**	253	95%	6	24	24	54	43%
$N \models^d F$	simp	30	32	**38**	**38**	**38**	**38**	**38**	252	95%	2	8	8	18	14%

Fig. 10. Performance results of the automated PML formula classification task conducted with `sledgehammer` and `nitpick`; for the problem set $M \models^m F$ a perfect classification is achieved (with or without the use of `simp`) already for a very short timeout of 2s (and also 1s), meaning that for each formula F in this mode either a proof is reported, if F is a theorem (THM), or a counterexample is presented, if F is a countersatisfiable (CSA). There were no conflicting results reported by `sledgehammer` and `nitpick`.

minimal shallow with modal axioms

minimal shallow with semantic axioms

maximal shallow with modal axioms

maximal shallow with semantic axioms

deep with modal axioms

deep with semantic axioms

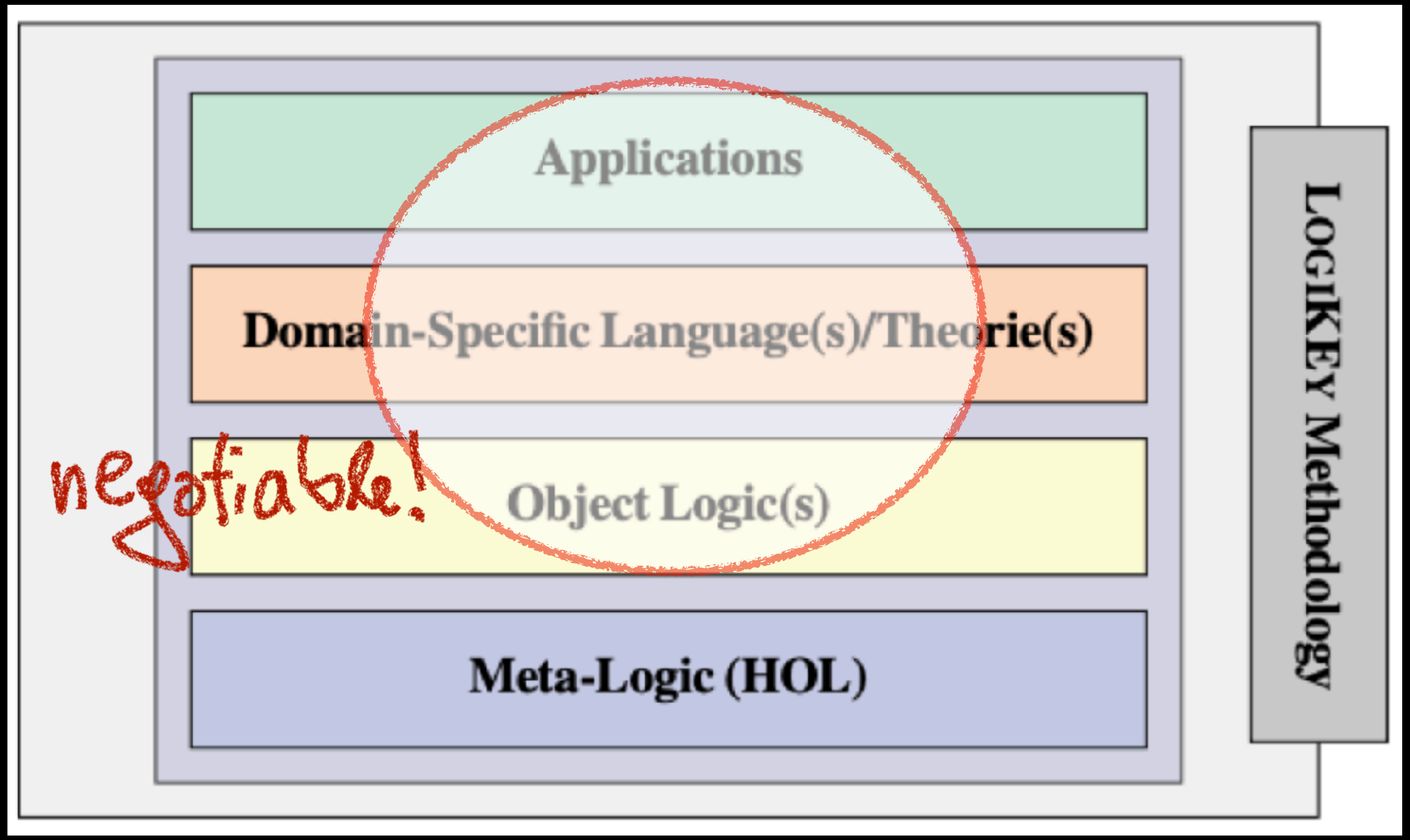
Results

- Winner — perfect classification — “no counterex.”/“proof” for all valid statement and “counterex.” else
- Second best
- Third best

Comments

- Remember that reasoning for **schematic formulas** was tested here
- Approach supports the **parallelization** of deep, minimal shallow and maximal shallow

Discussion: LogiKEy — Logical Pluralism in Proof Assistant Systems —



- supports the exploration of (expressive) **logics & logic combinations**
- supports the exploration & verification of **meta-logical properties**
- enables the **simultaneous use** of deep & shallow embeddings
- enables **integration of generic and specialist tools** (opportunities for further work)
- has **innovative interdisciplinary applications** (philosophy, law & ethics, maths, …)
- creates **uniform proofs** at the meta-level (after unfolding)
- …
- supports **uniform logic education** (since 2016 at FU Berlin & U Bamberg)

Reading

(preprints: http://christoph-benzmueller.de/publications.html)

HOL (see also the papers of Church, Henkin, Andrews and Muskens in JSL as mentioned before)

- Benzmüller, Brown, Kohlhase (2004), Higher-Order Semantics and Extensionality, J. Symb. Log., http://doi.org/10.2178/jsl/1102022211
- Benzmüller & Miller (2014), Automation of Higher-Order Logic, Handbook of the History of Logic, http://doi.org/10.1016/B978-0-444-51624-4.50005-8
- Benzmüller & Andrews (2019), Church's Type Theory, SEP, https://plato.stanford.edu/entries/type-theory-church/

Leo Provers

- Steen & Benzmüller (2021), Extensional Higher-Order Paramodulation in Leo-III, J. Autom. Reason., http://doi.org/10.1007/s10817-021-09588-x
- Benzmüller, Sultana, Paulson (2015), Thhe Higher-Order Prover Leo-II, J. Autom. Reason., http://doi.org/10.1007/s10817-015-9348-y
- Benzmüller & Kohlhase (1998), LEO – A Higher-Order Theorem Prover, CADE, http://doi.org/10.1007/BFb0054256

LogiKEy & Universal (Meta-)Logical Reasoning

- Benzmüller (2019), **Universal (meta-)logical reasoning: Recent successes.** Sci. Comp. Progr. (http://doi.org/10.1016/j.scico.2018.10.008)
- __, Parent & van der Torre (2020), **Designing normative theories for ethical and legal reasoning: LogiKEy framework, methodology, and tool support.** Artificial Intelligence. (https://doi.org/10.1016/j.artint.2020.103348)
- Benzmüller et.al. (2020), **LogiKEy Workbench: Deontic Logics, Logic Combinations and Expressive Ethical and Legal Reasoning (Isabelle/HOL Dataset)**. Data in Brief (https://doi.org/10.1016/j.dib.2020.106409)
- Bibel (2022), **Computer Kreiert Wissenschaft**. Informatik Spektrum. (https://doi.org/10.1007/s00287-022-01456-1)

Reading (cont'd)

(preprints: http://christoph-benzmueller.de/publications.html)

Legal & Normative Reasoning

- Lawniczak, L. Pasetto, C. Benzmüller, X. Li, and R. Markovich. “Reasoning with Epistemic Rights and Duties: Automating a Dynamic Logic of the Right to Know in LogiKEy”. In: ECAI 2025 - 28th European Conference on Artificial Intelligence. Ed. by I. Lynce, N. Murano, M. Vallati, S. Villata, F. Chesani, M. Milano, A. Omicini, and M. Dastani. Vol. 413. IOS Press, 2025, pp. 1623–1630. doi: 10.3233/FAIA250988
- L. Pasetto and C. Benzmüller. “Implementing the Fatio Protocol for Multi-Agent Argumentation in LogiKEy”. In: Proceedings of the 5th International Workshop on Automated Reasoning in Quantified Non-Classical Logics (ARQNL 2024), Nancy, France, July 1, 2024. Ed. by C. Benzmüller, J. Otten, and R. Ramanayake. Vol. CEUR3875. Nancy, Fance: CEUR Workshop Proceedings, CEUR-WS.org, 2024, pp. 38–47. url: https://ceurws.org/Vol-3875/ARQNL2024_paper4.pdf
- L. Lawniczak and C. Benzmüller. “Logical Modalities within the European AI Act: An Analysis”. In: Proceedings of the 20th International Conference on Artificial Intelligence and Law. Ed. by J. Maranhão. Association for Computing Machinery, New York, NY, USA, 2025. doi: 10.1145/3769126.3769202
- X. Parent and C. Benzmüller. “Conditional Normative Reasoning as a Fragment of HOL”. In: Journal of Applied Non-Classical Logics 34.4 (2024), pp. 1–32. doi: 10.1080/11663081.2024.2386917
- C. Benzmüller, A. Farjami, P. Meder, and X. Parent. “I/O Logic in HOL”. In: Journal of Applied Logics – IfCoLoG Journal of Logics and their Applications (Special Issue: Reasoning for Legal AI) 6.5 (2019). Ed. by L. Robaldo and L. van der Torre, pp. 715–732. url: https:// www.collegepublications.co.uk/ifcolog/ ?00034.
- C. Benzmüller, A. Farjami, and X. Parent. “Åqvist’s Dyadic Deontic Logic E in HOL”. In: Journal of Applied Logics – IfCoLoG Journal of Logics and their Applications (Special Issue: Reasoning for Legal AI) 6.5 (2019). Ed. by L. Robaldo and L. van der Torre, pp. 733–755. url: https://www.collegepublications.co.uk/ ifcolog/?00034
- C. Benzmüller, A. Farjami, and X. Parent. “Dyadic Deontic Logic in HOL: Faithful Embedding and MetaTheoretical Experiments”. In: New Developments in Legal Reasoning and Logic: From Ancient Law to Modern Legal Systems. Ed. by S. Rahman, M. Armgardt, N. Kvernenes, and H. Christian. Vol. 23. Logic, Argumentation & Reasoning. Springer Nature Switzerland AG, 2022. doi: 10.1007/978-3-030-70084-3_14.
- …

Reading (cont'd)

(preprints: http://christoph-benzmueller.de/publications.html)

Foundational Studies in Metaphysics

- Kirchner, Benzmüller & Zalta (2019), **Computer Science and Metaphysics: A Cross-Fertilization**. Open Philosophy (http://doi.org/10.1515/opphil-2019-0015)
- __ (2020), **Mechanizing Principia Logico-Metaphysica in Functional Type Theory**. Review of Symbolic Logic (https://doi.org/10.1017/S1755020319000297)
- Kirchner (2021). **Embedding of Abstract Object Theory in Isabelle/HOL.** Full sources. (https://github.com/ekpyron/AOT/tree/dissertation)
- __ (2022), **Computer-Verified Foundations of Metaphysics and an Ontology of Natural Numbers in Isabelle/HOL**. PhD thesis, Dep. of Maths and CS, FU Berlin.

Foundational Studies in Ethics & Law

- Fuenmayor & Benzmüller (2018), **Formalisation and Evaluation of Alan Gewirth's Proof for the Principle of Generic Consistency in Isabelle/HOL**. (https://www.isa-afp.org/entries/GewirthPGCProof.html)
- __ (2019), **Harnessing Higher-Order (Meta-)Logic to Represent and Reason with Complex Ethical Theories**. PRICAI 2019 (https://doi.org/10.1007/978-3-030-29908-8_34)
- Benzmüller & Fuenmayor (2021), **Value-oriented Legal Argumentation in Isabelle/HOL**. ITP 2021 (https://doi.org/10.4230/LIPIcs.ITP.2021.7)
- __, Fuenmayor & Lomfeld (2022), **Modelling Value-oriented Legal Reasoning in LogiKEy**. (https://arxiv.org/abs/2006.12789)

Studies on the Ontological Argument

- Benzmüller & Woltzenlogel Paleo (2014), **Automating Gödel's Ontological Proof of God's Existence with Higher-order Automated Theorem Provers.** ECAI 2014 (http://doi.org/10.3233/978-1-61499-419-0-93)
- __ (2016), **The Inconsistency in Gödel's Ontological Argument: A Success Story for AI in Metaphysics.** IJCAI 2016 (http://www.ijcai.org/Proceedings/16/Papers/137.pdf)

Reading (cont'd)

(preprints: http://christoph-benzmueller.de/publications.html)

Studies on the Ontological Argument (cont'd)

- Benzmüller & Woltzenlogel Paleo (2016), **An Object-Logic Explanation for the Inconsistency in Gödel's Ontological Theory** (Extended Abstract, Sister Conferences). KI 2016 (http://doi.org/10.1007/978-3-319-46073-4)
- __ (2017). **Computer-Assisted Analysis of the Anderson-Hájek Controversy.** Logica Universalis (http://doi.org/10.1007/s11787-017-0160-9)
- Fuenmayor & Benzmüller (2017), **Automating Emendations of the Ontological Argument in Intensional Higher-Order Modal Logic.** KI 2017 (http://doi.org/10.1007/978-3-319-67190-1_9)
- __ (2018). **A Case Study On Computational Hermeneutics: E. J. Lowe's Modal Ontological Argument.** IfCoLoG Journal of Logics and their Applications (https://www.researchgate.net/publication/333804824)
- Benzmüller & Fuenmayor (2020), **Computer-supported Analysis of Positive Properties, Ultrafilters and Modal Collapse in Variants of Gödel's Ontological Argument.** Bulletin of the Section of Logic (http://doi.org/10.18778/0138-0680.2020.08)
- Benzmüller (2020), **A (Simplified) Supreme Being Necessarily Exists, says the Computer: Computationally Explored Variants of Gödel's Ontological Argument.** KR 2020 (https://doi.org/10.24963/kr.2020/80)
- __ (2022), **A Simplified Variant of Gödel's Ontological Argument.** To appear (https://www.researchgate.net/publication/358607847)

Isabelle/HOL:

- Website: https://isabelle.in.tum.de
- Documentation: https://isabelle.in.tum.de/documentation.html

- Nipkow, Paulson & Wenzel (2002), **Isabelle/HOL: A Proof Assistant for Higher-Order Logic.** Springer (https://doi.org/10.1007/3-540-45949-9)
- Blanchette, Bulwahn & Tobias Nipkow (2011), **Automatic Proof and Disproof in Isabelle/HOL**. FroCoS 2011 (https://doi.org/10.1007/978-3-642-24364-6_2)

Reading (cont'd)

(preprints: http://christoph-benzmueller.de/publications.html)

Free Logic & Axiomatic Category Theory in Free Logic

- Scott (1967), **Existence and description in formal logic.** Reprinted in: Lambert (1991), Philosophical Application of Free Logic, OUP. (ISBN 0195061314)
- Scott (1977), **Identity and existence in intuitionistic logic.** Applications of Sheaves, Springer (https://doi.org/10.1007/BFb0061839)
- Benzmüller & Scott (2016), **Automating Free Logic in Isabelle/HOL.** ICMS 2016 (https://doi.org/10.1007/978-3-319-42432-3_6)
- __ (2018), **Axiom systems for category theory in free logic**. Archive of Formal Proofs (https://www.isa-afp.org/entries/AxiomaticCategoryTheory.html)
- __ (2020), **Automating Free Logic in HOL, with an Experimental Application in Category Theory.** JAR (http://doi.org/10.1007/s10817-018-09507-7)
- Tiemens, Scott, Benzmüller & Benda (2020), **Computer-supported Exploration of a Categorical Axiomatization of Modeloids.** RAMiCS 2020 (https://doi.org/10.1007/978-3-030-43520-2_19)
- Makarenko & Benzmüller (2020), **Positive Free Higher-Order Logic and its Automation via a Semantical Embedding**. KI 2020 (http://doi.org/10.1007/978-3-030-58285-2_9)
- Bayer (2021), **Exploring categories, formally**. BSc Thesis, Dep. of Maths and CS, FU Berlin.
- Gonus (2021), **Categorical semantics of Intuitionistic Multiplicative Linear Logic and its formalization in Isabelle/HOL**. MSc Thesis, Dep. of Maths and CS, FU Berlin.