Modular and heterogeneous logical theories in DOL

Till Mossakowski
joint work with Răzvan Diaconescu and Andrzej Tarlecki

GI-Fachgruppentreffen ALGO/KP/LogInf, Nov 24, 2017

Computer science uses logic in many ways

- programming languages
- formal specification and verification
- databases, WWW, artificial intelligence
- ontologies
- algorithms & complexity
- (semi-)automated theorem proving
- metatheory
- ...

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$, $\top$, $\bot$
modal logics	$\dots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\dots$, i, $@i.\varphi$
temporal logics	$\dots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\dots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\dots$, $t = u$, $P(t_1, \dots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\ldots$, $t = u$, $P(t_1, \ldots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$
description logics	$C \sqsubseteq D \qquad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\ldots$, $t = u$, $P(t_1, \ldots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$
description logics	$C \sqsubseteq D \quad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$
higher-order logics	$\ldots$, $P\ t$, $\exists P.\varphi$, $\forall P.\varphi$, $\iota x.\varphi \quad t ::= f\ t \mid \lambda x.t \mid \ldots$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\ldots$, $t = u$, $P(t_1, \ldots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$
description logics	$C \sqsubseteq D \qquad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$
higher-order logics	$\ldots$, $P\ t$, $\exists P.\varphi$, $\forall P.\varphi$, $\iota x.\varphi \qquad t ::= f\ t \mid \lambda x.t \mid \ldots$
type theories	$\ldots$, $\forall \alpha.\forall f : \alpha \rightarrow \alpha.\varphi$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\Diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\ldots$, $t = u$, $P(t_1, \ldots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$
description logics	$C \sqsubseteq D \qquad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$
higher-order logics	$\ldots$, $P\ t$, $\exists P.\varphi$, $\forall P.\varphi$, $\iota x.\varphi \qquad t ::= f\ t \mid \lambda x.t \mid \ldots$
type theories	$\ldots$, $\forall \alpha. \forall f : \alpha \rightarrow \alpha.\varphi$
Hoare logics	$\{\varphi\}P\{\psi\} \quad P ::= x := t \mid \text{while } \varphi \text{ do } P \mid \ldots$

Plenty of logics are used in computer science

propositional logics	$p,\ \neg\varphi,\ \varphi \wedge \psi,\ \varphi \vee \psi,\ \varphi \to \psi,\ \top,\ \bot$
modal logics	$\ldots,\ \Box\varphi,\ \Diamond\varphi$
hybrid logics	$\ldots,\ i,\ @i.\varphi$
temporal logics	$\ldots,\ F\varphi,\ G\varphi,\ \varphi U \psi$
QBF	$\ldots,\ \exists p.\varphi,\ \forall p.\varphi$
first-order logics	$\ldots,\ t = u,\ P(t_1, \ldots, t_n),\ \exists x.\varphi,\ \forall x.\varphi$
description logics	$C \sqsubseteq D \qquad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$
higher-order logics	$\ldots,\ P\ t,\ \exists P.\varphi,\ \forall P.\varphi,\ \iota x.\varphi \qquad t ::= f\ t \mid \lambda x.t \mid ..$
type theories	$\ldots,\ \forall \alpha.\forall f : \alpha \to \alpha.\varphi$
Hoare logics	$\{\varphi\}P\{\psi\} \quad P ::= x := t \mid \text{while } \varphi \text{ do } P \mid \ldots$
dynamic logics	$[P]\varphi,\ \langle P \rangle \varphi \qquad P ::= a \mid P? \mid PQ \mid P \cup Q \mid P^*$

Plenty of logics are used in computer science

propositional logics	p, $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\top$, $\bot$
modal logics	$\ldots$, $\Box\varphi$, $\diamond\varphi$
hybrid logics	$\ldots$, i, $@i.\varphi$
temporal logics	$\ldots$, $F\varphi$, $G\varphi$, $\varphi U \psi$
QBF	$\ldots$, $\exists p.\varphi$, $\forall p.\varphi$
first-order logics	$\ldots$, $t = u$, $P(t_1, \ldots, t_n)$, $\exists x.\varphi$, $\forall x.\varphi$
description logics	$C \sqsubseteq D$ $\quad C ::= A \mid C \sqcup D \mid \neg C \mid \exists R.C \mid \ldots$
higher-order logics	$\ldots$, $P\ t$, $\exists P.\varphi$, $\forall P.\varphi$, $\iota x.\varphi$ $\quad t ::= f\ t \mid \lambda x.t \mid \ldots$
type theories	$\ldots$, $\forall\alpha.\forall f : \alpha \to \alpha.\varphi$
Hoare logics	$\{\varphi\}P\{\psi\}$ $\quad P ::= x := t \mid \text{while } \varphi \text{ do } P \mid \ldots$
dynamic logics	$[P]\varphi$, $\langle P\rangle\varphi$ $\quad P ::= a \mid P? \mid PQ \mid P \cup Q \mid P^*$
linear logic	$!\varphi$, $?\varphi$, $\varphi \otimes \psi$, $\varphi \oplus \psi$, $\varphi \& \psi$, $\varphi ⅋ \psi$
$\ldots$	

What do these logics have in common?

- formulas / sentences
- entailment, logical consequence
- models
- soundness, completeness
- conservative extensions
- ...

Are there definitions and theorems that we can

- introduce once and for all
- and then apply them to many logics?

What is a logic, after all?

Definition (Gentzen, Tarski, Scott)

An entailment relation (ER) $(S,\vdash)$ is a binary relation $\vdash\,\subseteq \mathscr{P}(S) \times S$ on a set S of sentences.

$(S,\vdash)$ is Tarskian, if

1. reflexivity: for any $\varphi \in S$, $\{\varphi\} \vdash \varphi$,
2. monotonicity: if $\Gamma \vdash \varphi$ and $\Gamma' \supseteq \Gamma$ then $\Gamma' \vdash \varphi$,
3. transitivity: if $\Gamma \vdash \varphi_i$, for $i \in I$, and $\Gamma \cup \{\varphi_i \mid i \in I\} \vdash \psi$, then $\Gamma \vdash \psi$.

What is a logic, after all?

Definition (Gentzen, Tarski, Scott)

An entailment relation (ER) $(S,\vdash)$ is a binary relation $\vdash\ \subseteq \mathscr{P}(S) \times S$ on a set S of sentences.

$(S,\vdash)$ is Tarskian, if

1. reflexivity: for any $\varphi \in S$, $\{\varphi\} \vdash \varphi$,
2. monotonicity: if $\Gamma \vdash \varphi$ and $\Gamma' \supseteq \Gamma$ then $\Gamma' \vdash \varphi$,
3. transitivity: if $\Gamma \vdash \varphi_i$, for $i \in I$, and $\Gamma \cup \{\varphi_i \mid i \in I\} \vdash \psi$, then $\Gamma \vdash \psi$.

Definition

A theory $\Gamma \subseteq S$ is consistent if $\Gamma \nvdash \varphi$ for some φ.

ER for propositional logic

Example (Propositional logic)

Propositional logic (**PL**) has sentences given by the following grammar

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \to \varphi_2 \mid \top \mid \bot$$

where p denotes propositional variables.

$\vdash$ is the minimal Tarskian entailment relation satisfying:

$$\frac{\Gamma \vdash \varphi, \quad \Gamma \vdash \psi}{\Gamma \vdash \varphi \wedge \psi} \qquad \frac{\Gamma, \varphi \vdash \chi, \quad \Gamma, \psi \vdash \chi}{\Gamma, \varphi \vee \psi \vdash \chi} \qquad \frac{\Gamma \vdash \varphi \to \psi}{\Gamma, \varphi \vdash \psi}$$

$$\frac{}{\Gamma \vdash \top} \qquad \frac{}{\bot \vdash \varphi} \qquad \frac{\Gamma \vdash \neg\varphi}{\Gamma, \varphi \vdash \bot} \qquad \frac{\Gamma \vdash \neg\neg\varphi}{\Gamma \vdash \varphi}$$

ER for modal logic

Example (Modal logic)

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \rightarrow \varphi_2 \mid \top \mid \bot \mid \Box\varphi \mid \Diamond\varphi$$

$\vdash$ is the minimal Tarskian entailment relation satisfying the rules for propositional logic plus:

$$\frac{}{\Gamma \vdash \Box(\varphi \rightarrow \psi) \rightarrow (\Box\varphi \rightarrow \Box\psi)} \qquad \frac{\vdash \varphi}{\vdash \Box\varphi} \qquad \frac{\Gamma \vdash \Diamond\varphi}{\Gamma \vdash \neg\Box\neg\varphi}$$

ER for first-order logic (without function symbols)

Example (First-order logic)

$$
\begin{aligned}
t ::= \quad & x \mid c \\
\varphi ::= \quad & P(t_1, \ldots, t_n) \mid \exists x.\varphi \mid \forall x.\varphi \mid \\
& \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \to \varphi_2 \mid \top \mid \bot
\end{aligned}
$$

$\vdash$ is the minimal Tarskian entailment relation satisfying the rules for propositional logic plus:

$$\frac{\Gamma \vdash \varphi(t)}{\Gamma \vdash \exists x.\varphi(x)} \qquad \frac{\Gamma, \varphi(c) \vdash \psi}{\Gamma, \exists x.\varphi(x) \vdash \psi} \ (c \text{ does not occur in } \Gamma,\ \varphi,\ \psi)$$

$$\frac{\Gamma \vdash \forall x.\varphi(x)}{\Gamma \vdash \varphi(t)} \qquad \frac{\Gamma \vdash \varphi(c)}{\Gamma \vdash \forall x.\varphi(x)} \ (c \text{ does not occur in } \Gamma,\ \varphi)$$

Morphisms of entailment relations

Definition

An entailment relation morphism $\alpha\colon (S_1, \vdash^1) \longrightarrow (S_2, \vdash^2)$ is a function $\alpha\colon S_1 \longrightarrow S_2$ such that

$$\Gamma \vdash^1 \varphi \quad \text{implies} \quad \alpha(\Gamma) \vdash^2 \alpha(\varphi)$$

If the converse holde, α is conservative.
ERs and ER morphisms form a category $\mathbb{ER}$.

Observation:

> *If we have a conservative ER morphism and a theorem prover for $\vdash^2$, we can borrow it for $\vdash^1$.*

For propositional and first-order logic, there are many automated theorem provers, but not for modal logic.

Translating modal logic into first-order logic

Example

A conservative ER morphism Modal $\to$ FOL is defined by

$$\begin{aligned}
&\alpha_x(p) = p(x)\\
&\alpha_x(\Box\varphi) = \forall y.R(x,y) \to \alpha_y(\varphi)\\
&\alpha_x(\Diamond\varphi) = \exists y.R(x,y) \wedge \alpha_y(\varphi)\\
&\alpha_x(\neg\varphi) = \neg\alpha_x(\varphi)\\
&\ldots\\
&\alpha(\varphi) = \forall x.\alpha_x(\varphi)
\end{aligned}$$

Proof of ER property: induction over proofs.
Proof of conservativity property is more complicated
$\Rightarrow$ use model theory.

Adding model theory

Definition (Goguen, Burstall)

A satisfaction system $(S, \mathscr{M}, \models)$ consists of

- a set of S of sentences,
- a category $\mathscr{M}$ of models and model homomorphisms, and
- a binary relation $\models \subseteq |\mathscr{M}| \times S$, the satisfaction relation.

Definition (Logical consequence)

$\Gamma \models \varphi$ iff for all $M \in \mathscr{M}$, $M \models \Gamma$ implies $M \models \varphi$.

Logics

Definition (Logical consequence)

A logic $(S, \vdash, \mathscr{M}, \models)$ consists of

- an entailment relation $(S, \vdash)$, and
- a satisfaction system $(S, \mathscr{M}, \models)$,

such that soundness holds:

$$\Gamma \vdash \varphi \text{ implies } \Gamma \models \varphi$$

A logic is complete, if

$$\Gamma \models \varphi \text{ implies } \Gamma \vdash \varphi$$

Satisfaction system for propositional logic

Example (Propositional logic)

sentences as above

models maps from propositional variables to $\{true, false\}$

model homomorphisms $M_1 \to M_2$ iff
$(M_1(p) = true$ implies $M_2(p) = true)$

satisfaction $M \models \varphi$ iff
$M(\varphi) = true$ according to standard truth tables

Proposition

Propositional logic is sound and complete.

Satisfaction system for modal logic

Example (Modal logic)

- sentences as above
- a model M consists of
  - a non-empty set W of worlds,
  - a binary accessibility relation $R \subseteq W \times W$,
  - a map maps from propositional variables and worlds to $\{true, false\}$
- satisfaction
  - $M, w \models p$ iff $M(p, w) = true$
  - $M, w \models \Box\varphi$ iff for all $v \in W$ with $R(w, v)$, $M, v \models \varphi$
  - $M, w \models \diamond\varphi$ iff for somd $v \in W$ with $R(w, v)$, $M, v \models \varphi$
  - $M, w \models \neg\varphi$ iff $M, w \not\models \varphi$ etc.
  - $M \models \varphi$ iff for all $w \in W$, $\quad M, w \models \varphi$

Satisfaction system for modal logic (cont'd)

Proposition

Modal logic is sound and complete.

Satisfaction system for first-order logic

Example (First-order logic)

- sentences as above
- models: a first-order M model consist of
  - a non-empty set $|M|$ called universe,
  - an element $M_c \in |M|$ for each constant c,
  - an n-ary relation M_P on $|M|$ for each n-ary predicate symbol P
- satisfaction
  - $M, v \models P(t_1, \ldots, t_n)$ iff $(v^{\#}(t_1), \ldots, v^{\#}(t_n)) \in M_P$
  - $M, v \models \forall x.\varphi$ iff for all ξ differing from v at most for x, $\quad M, \xi \models \varphi$
  - $M, v \models \exists x.\varphi$ iff forsome ξ differing from v at most for x, $M, \xi \models \varphi$
  - $M, v \models \neg\varphi$ iff $M, v \not\models \varphi$ etc.
  - $M \models \varphi$ iff for all v, $\quad M, v \models \varphi$

Satisfaction system for first-order logic (cont'd)

Proposition

First-order logic is sound and complete.

Morphisms of satisfaction systems

Definition (Goguen, Burstall)

A satisfaction system morphism $(\alpha, \beta): (S_1, \mathscr{M}_1, \models_1) \longrightarrow (S_2, \mathscr{M}_2, \models_2)$ consists of

- a sentence translation function $\alpha: S_1 \longrightarrow S_2$, and
- a model reduction functor $\beta: \mathscr{M}_2 \longrightarrow \mathscr{M}_1$, such that

$$M_2 \models_2 \alpha(\varphi_1) \text{ iff } \beta(M_2) \models_1 \varphi_1$$

(satisfaction condition).

This gives us a category $\mathbb{S}at$ of satisfaction systems and satisfaction system morphisms.

Translating modal logic into first-order logic

Example

A satisfaction system morphism Modal $\rightarrow$ FOL is defined by

- sentence translation as above
- a first-order model is reduced to a modal model by
  - taking the universe as set of worlds
  - taking the interpretation of the binary predicate R as accessibility relation
  - taking the interpretation of the unary predicate p as interpretation of the propositional variable p

Proposition

The satisfaction condition holds.
Proof: induction over formulas.

Semantic proof of conservative ER morphism property

Theorem (Cerioli, Meseguer)

Let $(S_1, \vdash_1, \mathscr{M}_1, \models_1)$ and $(S_2, \vdash_2, \mathscr{M}_2, \models_2)$ be two sound and complete logics and a satisfaction system morphism

$$(\alpha, \beta) \colon (S_1, \mathscr{M}_1, \models_1) \longrightarrow (S_2, \mathscr{M}_2, \models_2)$$

be given.
If β is surjectivive, then α is a conservative ER morphism

$$\alpha \colon (S_1, \vdash_1) \longrightarrow (S_2, \vdash_2).$$

Indexing over signatures

We have been imprecise at various places.

Indexing over signatures

We have been imprecise at various places.
Strictly speaking, we need to index over signatures. Signatures are vocabularies of non-logical (=user-defined) symbols.

Indexing over signatures

We have been imprecise at various places.
Strictly speaking, we need to index over signatures. Signatures are vocabularies of non-logical (=user-defined) symbols.
Entailment:

- set Sig of signatures, and
- family of ERs $(\mathrm{Sen}(\Sigma), \vdash_\Sigma)_{\Sigma \in \mathrm{Sig}}$

Indexing over signatures

We have been imprecise at various places.
Strictly speaking, we need to index over signatures. Signatures are vocabularies of non-logical (=user-defined) symbols.

Entailment:

- set Sig of signatures, and
- family of ERs $(\mathrm{Sen}(\Sigma), \vdash_\Sigma)_{\Sigma \in \mathrm{Sig}}$

Satisfaction:

- set Sig of signatures, and
- family of satisfaction systems $(\mathrm{Sen}(\Sigma), \mathrm{Mod}(\Sigma), \models_\Sigma)_{\Sigma \in \mathrm{Sig}}$

However, this is not the whole story!

Within this framework, we can study

- abstract logical connectives
- logic translations
- logic combination
- consistency strength, expressiveness
- ...

However, this is not the whole story!

Within this framework, we can study

- abstract logical connectives
- logic translations
- logic combination
- consistency strength, expressiveness
- ...

However, we cannot study

- conservative extensions
- modular logical theories
- abstract quantifiers
- Craig interpolation, Robinson consistency, Beth definability
- elementary diagrams
- ...

Indexing over signature morphisms

Definition (Fiadeiro, Meseguer)

An entailment system is a functor $I\colon \mathrm{Sig} \longrightarrow \mathbb{ER}$, where Sig is the category of signatures.

This gives us

- a graph Sig of signatures and signature morphisms,
- for each signature Σ, an identity morphism $id_\Sigma : \Sigma \longrightarrow \Sigma$,
- a composition operation $\circ$ on signature morphisms,
- for each $\Sigma \in \mathrm{Sig}$, an ER $(\mathrm{Sen}(\Sigma), \vdash_\Sigma)$,
- for each signature morphism $\sigma_1 : \Sigma_1 \longrightarrow \Sigma_2 \in \mathrm{Sig}$, an ER morphism $I(\sigma)\colon (\mathrm{Sen}(\Sigma_1), \vdash_{\Sigma_1}) \longrightarrow (\mathrm{Sen}(\Sigma_2), \vdash_{\Sigma_2})$, by abuse of notation also denoted by σ.

Sample entailment systems

Example (Entailment system for propositional logic)

signatures sets of propositional variables

ERs $(\mathrm{Sen}(\Sigma), \vdash_\Sigma)$ as before, but built over Σ

ER morphisms $\sigma(\varphi)$ replaces symbols in φ along σ.
We have

$$\Gamma \vdash_{\Sigma_1} \varphi \text{ implies } \sigma(\Gamma) \vdash_{\Sigma_2} \sigma(\varphi)$$

Further examples:
modal logic, first-order logic, and many more.

Indexing over signature morphisms (cont'd)

Definition (Goguen, Burstall)

An institution is a functor $I \colon \mathrm{Sig} \longrightarrow \mathbb{S}at$.

This gives us

- a graph Sig of signatures and signature morphisms, $(\dots)$
- for each $\Sigma \in \mathrm{Sig}$, a satisfaction system $(\mathrm{Sen}(\Sigma), \mathrm{Mod}(\Sigma), \models_\Sigma)$,
- for each signature morphism $\sigma_1 : \Sigma_1 \longrightarrow \Sigma_2 \in \mathrm{Sig}$, a satisfaction system morphism $I(\sigma) \colon (\mathrm{Sen}(\Sigma_1), \mathrm{Mod}(\Sigma_1), \models_{\Sigma_1}) \longrightarrow (\mathrm{Sen}(\Sigma_2), \mathrm{Mod}(\Sigma_2), \models_{\Sigma_2})$, by abuse of notation also denoted by $(\sigma, \_|_\sigma)$.

Sample institutions

Example (Institutions for propositional logic)

signatures sets of propositional variables

Sat. systems $(\mathrm{Sen}(\Sigma), \mathrm{Mod}(\Sigma), \models_\Sigma)$ as before, but built over Σ

Sat. syst. morphisms
- $\sigma(\varphi)$ replaces symbols in φ along σ
- $M|_\sigma$ interprets p as $M|_\sigma(p) := M_{\sigma(p)}$
  We have
  $$M_2|_\sigma \models_{\Sigma_1} \varphi_1 \text{ iff } M_2 \models_{\Sigma_2} \sigma(\varphi_1)$$

Further examples:
modal logic, first-order logic, and many more.

Indexing over signature morphisms (cont'd)

Definition (Meseguer)

A logic is an institution equipped with an entailment system, agreeing on signatures and sentences.

Abstraction via institutions

Institution independent notions and theorems,
languages, calculi, and software tools

Semantics, calculi and proof tools of particular institutions

Structured ontologies, models, specifications (OMS) over an arbitrary institution

$$
\begin{array}{llllr}
 & & & & \textit{DOLnotation} \\
O & ::= & \langle \Sigma, \Gamma \rangle & \text{basic specification} & \\
 & | & O_1 \cup O_2 & \text{union} & O_1 \textbf{ and } O_2 \\
 & | & \sigma(O) & \text{translation} & O \textbf{ with } \sigma \\
 & | & O|_\sigma & \text{hiding} & O \textbf{ hide } \sigma
\end{array}
$$

. . . and their semantics

Definition (Signature and model class of an OMS)

$Sig(\langle\Sigma,\Gamma\rangle) = \Sigma$
$\mathrm{Mod}(\langle\Sigma,\Gamma\rangle) = \{M \in \mathrm{Mod}(\Sigma) \mid M \models \Gamma\}$

$Sig(O_1 \cup O_2) = Sig(O_1) = Sig(O_2)$
$\mathrm{Mod}(O_1 \cup O_2) = \mathrm{Mod}(O_1) \cap \mathrm{Mod}(O_2)$

$Sig(\sigma : \Sigma_1 \longrightarrow \Sigma_2(O)) = \Sigma_2$
$\mathrm{Mod}(\sigma(O)) = \{M \in \mathrm{Mod}(\Sigma_2) \mid M|_\sigma \in \mathrm{Mod}(O)\}$

$Sig(O|_{\sigma : \Sigma_1 \longrightarrow \Sigma_2}) = \Sigma_1$
$\mathrm{Mod}(O|_{\sigma : \Sigma_1 \longrightarrow \Sigma_2}) = \{M|_\sigma \mid M \in \mathrm{Mod}(O)\}$

Distributed Ontology, Model and Specification Language

DOL

- DOL has been adopted as an OMG-Standard (under my leadership)
- combines modularity, interoperability and language heterogeneity
- continuous formal semantics, based on institutions
  - Ontologies, models and specifications are logical theories
- cooperation of different communities:
  - Ontologies, UML, specification

T. Mossakowski, C. Lange, O. Kutz (2012). Three Semantics for the Core of the Distributed Ontology Language, FOIS 2012. Best paper award

Theory Morphisms

Definition

A theory morphism $\sigma : (\Sigma_1, \Gamma_1) \to (\Sigma_2, \Gamma_2)$ is a signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ such that

$$\text{for } M \in \mathrm{Mod}(\Sigma_2, \Gamma_2), \text{ we have } M|_\sigma \in \mathrm{Mod}(\Sigma_1, \Gamma_1)$$

Extensions are theory morphisms:

$$(\Sigma, \Gamma) \textbf{ then } (\Delta_\Sigma, \Delta_\Gamma)$$

leads to the theory morphism

$$(\Sigma, \Gamma) \xrightarrow{\iota} (\Sigma \cup \Delta_\Sigma, \iota(\Gamma) \cup \Delta_\Gamma)$$

Proof: $M \models \iota(\Gamma) \cup \Delta_\Gamma$ implies $M|_\iota \models \Gamma$ by the satisfaction condition.

Interpretations (views, refinements)

- **interpretation** *name* : O_1 **to** $O_2 = \sigma$
- σ is a signature morphism (if omitted, assumed to be identity)
- expresses that σ is a theory morphism $O_1 \rightarrow O_2$

```
logic CASL.FOL=
spec RichBooleanAlgebra =
  BooleanAlgebra
then %def
  pred __ <= __ : Elem * Elem;
  forall x,y:Elem
  . x <= y <=> x cap y = x %(leq_def)%
end
interpretation order_in_BA :
  PartialOrder to RichBooleanAlgebra
end
```

Family Ontology

```
logic OWL
ontology Family2 =
  Class: Person
  Class: Woman SubClassOf: Person
  ObjectProperty: hasChild
  Class: Mother
        EquivalentTo: Woman and hasChild some Person
  Individual: mary Types: Woman Facts: hasChild  john
  Individual: john Types: Person
  Individual: mary
       Types: Annotations: Implied "true"^^xsd:boolean
              Mother
end
```

Interpretation in OWL

```
logic OWL
ontology Family_alt =
  Class: Human
  Class: Female
  Class: Woman EquivalentTo: Human and Female
  ObjectProperty: hasChild
  Class: Mother
         EquivalentTo: Female and hasChild some Human
end

interpretation i : Family_alt to Family2 =
  Human |-> Person, Female |-> Woman
end
```

Criterion for Theory Morphisms

Theorem

A signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ *is a theory morphism* $\sigma : (\Sigma_1, \Gamma_1) \to (\Sigma_2, \Gamma_2)$ *iff*

$$\Gamma_2 \models_{\Sigma_2} \sigma(\Gamma_1)$$

Proof.

By the satisfaction condition. $\square$

Conservative Extensions

Definition

A theory morphism $\sigma : T_1 \to T_2$ is consequence-theoretically conservative (ccons), if for each $\phi_1 \in \mathrm{Sen}(\Sigma_1)$

$$T_2 \models \sigma(\phi_1) \text{ implies } T_1 \models \phi_1 .$$

(no "new" facts over the "old" signature)

Definition

A theory morphism $\sigma : T_1 \to T_2$ is model-theoretically conservative (mcons), if for each $M_1 \in \mathrm{Mod}(T_1)$, there is a σ-expansion

$$M_2 \in \mathrm{Mod}(T_2) \text{ with } (M_2)|_\sigma = M_1$$

A General Theorem

Theorem

If $\sigma : T_1 \to T_2$ is mcons, then it is also ccons.

Proof.

Assume that $\sigma : T_1 \to T_2$ is mcons.
Let ϕ_1 be a formula, such that $T_2 \models_{\Sigma_2} \sigma(\phi_1)$.
Let M_1 be a model $M_1 \in \mathrm{Mod}(T_1)$. By assumption there is a model $M_2 \in \mathrm{Mod}(T_2)$ with $M_2|_\sigma = M_1$. Since $T_2 \models_{\Sigma_2} \sigma(\phi_1)$, we have $M_2 \models \sigma(\phi_1)$. By the satisfaction condition $M_2|_\sigma \models_{\Sigma_1} \phi_1$. Hence $M_1 \models \phi_1$. Altogether $T_1 \models_{\Sigma_1} \phi_1$. □

Logical notions for OMS

Definition (Logical consequence for OMS)

$$O \models \varphi \qquad \text{iff} \qquad M \models \varphi \text{ for all } M \in \mathrm{Mod}(O)$$

Definition (OMS refinement)

$$O \rightsquigarrow O' \qquad \text{iff} \qquad \mathrm{Mod}(O') \subseteq \mathrm{Mod}(O)$$

Proof calculus for entailment (Borzyszkowski)

$$(CR)\ \frac{\{O \vdash \varphi_i\}_{i \in I} \quad \{\varphi_i\}_{i \in I} \vdash \varphi}{O \vdash \varphi} \qquad (basic)\ \frac{\varphi \in \Gamma}{\langle \Sigma, \Gamma \rangle \vdash \varphi}$$

$$(sum1)\ \frac{O_1 \vdash \varphi}{O_1 \cup O_2 \vdash \varphi} \qquad (sum2)\ \frac{O_1 \vdash \varphi}{O_1 \cup O_2 \vdash \varphi}$$

$$(trans)\ \frac{O \vdash \varphi}{\sigma(O) \vdash \sigma(\varphi)} \qquad (derive)\ \frac{O \vdash \sigma(\varphi)}{O|_\sigma \vdash \varphi}$$

Soundness means: $O \vdash \varphi$ implies $O \models \varphi$
Completeness means: $O \models \varphi$ implies $O \vdash \varphi$

Proof calculus for refinement (Borzyszkowski)

$$(\mathit{Basic})\ \frac{O \vdash \Gamma}{\langle \Sigma, \Gamma \rangle \rightsquigarrow O} \qquad (\mathit{Sum})\ \frac{O_1 \rightsquigarrow O \quad O_2 \rightsquigarrow O}{O_1 \cup O_2 \rightsquigarrow O}$$

$$(\mathit{Trans})\ \frac{O \rightsquigarrow O'|_\sigma}{\sigma(O) \rightsquigarrow O'}$$

$$(\mathit{Derive})\ \frac{O \rightsquigarrow O''}{O|_\sigma \rightsquigarrow O'}$$

if $\sigma : O' \longrightarrow O''$
is a conservative extension

Soundness means: $O_1 \rightsquigarrow O_2$ implies $O_1 \leadsto O_2$
Completeness means: $O_1 \leadsto O_2$ implies $O_1 \rightsquigarrow O_2$

Craig-Robinson interpolation

Definition

$$\begin{array}{ccc} \Psi & \begin{tikzcd} \Sigma \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & \Sigma_1 \arrow[d, "\theta_1"] \\ \Sigma_2 \arrow[r, "\theta_2"'] & \Sigma' \end{tikzcd} & (1)\ \Psi_1 \models \varphi_1(\Psi) \\ (2)\ \varphi_2(\Psi) \cup \Gamma_2 \models \Psi_2 & & (0)\ \theta_1(\Psi_1) \cup \theta_2(\Gamma_2) \models \theta_2(\Psi_2) \end{array}$$

A commutative square admits Craig-Robinson interpolation,
if for all finite $\Psi_1 \subseteq \mathrm{Sen}(\Sigma_1)$, $\Psi_2, \Gamma_2 \subseteq \mathrm{Sen}(\Sigma_2)$,
if (0), then there exists a finite $\Psi \subseteq \mathrm{Sen}(\Sigma)$ with (1) and (2).

$\mathscr{I}$ has Craig-Robinson interpolation if all signature pushouts admit Craig-Robinson interpolation.

Soundness and Completeness

Theorem (Borzyszkowski, Tarlecki, Diaconescu)

Under the assumptions that

- *the institution admits Craig-Robinson interpolation,*
- *the institution is weakly semi-exact, and*
- *the entailment system is complete,*

the calculus for structured entailment and refinement is sound and complete.

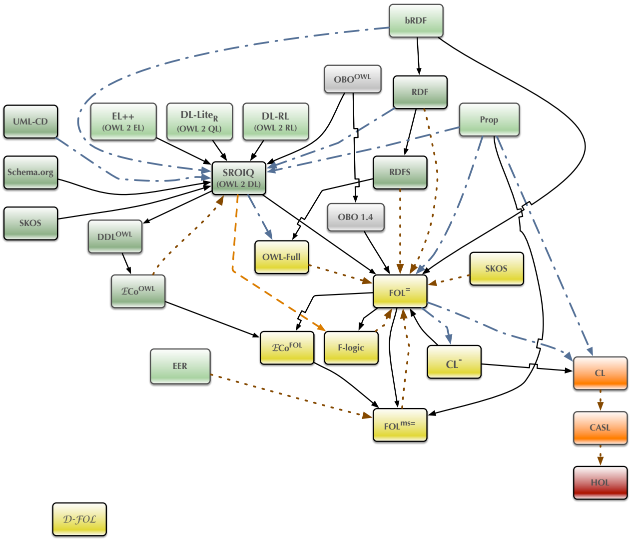
For refinement, we need an oracle for conservative extensions.
Weak semi-exactness = Mod maps pushouts to weak pullbacks

Heterogeneous specification

Definition

A heterogeneous logical environment ($\mathscr{HLE}$) (or indexed coinstitution) is diagram of institutions and comorphisms.

Some $\mathscr{HLE}$ for ontologies



- subinstitution
- theoroidal subinstitution
- simultaneously exact and model-expansive comorphisms
- model-expansive comorphisms

grey: no fixed expressivity

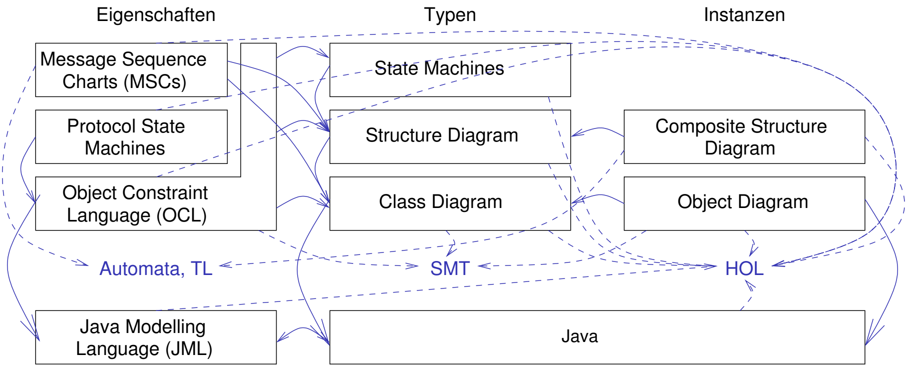
green: decidable ontology languages

yellow: semi-decidable

orange: some second-order constructs

red: full second-order logic

Some $\mathscr{HLE}$ for UML and Java



Institution comorphisms

Definition

An institution comorphism $\rho : \mathscr{I} \to \mathscr{I}'$ consists of:

- a functor $\Phi : \mathbf{Sign} \to \mathbf{Sign}'$;
- a natural transformation $\alpha : \mathbf{Sen} \to \mathbf{Sen}' \circ \Phi$; and
- a natural transformation $\beta : \mathbf{Mod}' \circ (\Phi)^{op} \to \mathbf{Mod}$,

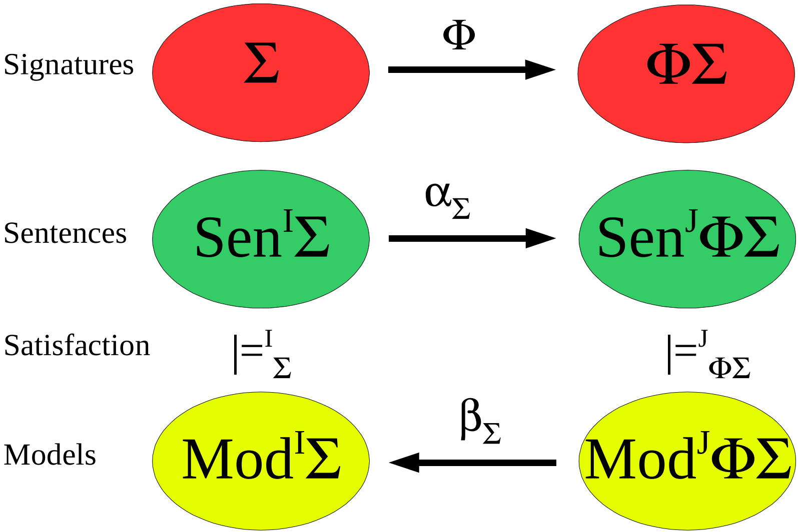
such that

$$M' \models'_{\Phi(\Sigma)} \alpha_\Sigma(\varphi) \iff \beta_\Sigma(M') \models_\Sigma \varphi$$

[*Satisfaction condition*]

Institution comorphisms

Institution comorphisms



Heterogeneous structuring operations

heterogeneous translation: For any $\mathscr{I}$-specification O, $\rho(O)$ is a specification with:

$$Sig[\rho(O)] := \Phi(Sig[O])$$

$$Mod[\rho(O)] := \beta^{-1}_{Sig[O]}(Mod[O])$$

heterogeneous hiding: For any $\mathscr{I}'$-specification O' and signature Σ with $Sig[O'] = \Phi(\Sigma)$, $O'|^{\Sigma}_{\rho}$ is a specification with:

$$Sig[O'|^{\Sigma}_{\rho}] := \Sigma$$

$$Mod[O'|^{\Sigma}_{\rho}] := \beta_{\Sigma}(Mod[O'])$$

A heterogeneous proof calculus

$$(\textit{het-trans})\ \frac{O \vdash \varphi}{\rho(O) \vdash \alpha(\varphi)} \qquad (\textit{het-derive})\ \frac{O \vdash \alpha(\varphi)}{O|_{\rho}^{\Sigma} \vdash \varphi}$$

$$(\textit{borrowing})\ \frac{\rho(O) \vdash \alpha(\varphi)}{O \vdash \varphi} \qquad \text{if } \rho \text{ is model-expansive}$$

$$(\textit{Het-snf})\ \frac{O' \vdash \sigma(\alpha(\varphi))}{O \vdash \varphi} \qquad \text{if } \textit{hsnf}(O) = (O'|_{\sigma})|_{\rho}^{\Sigma}$$

A heterogeneous proof calculus for refinement

$$(\textit{Het-Trans})\ \frac{O \rightsquigarrow O'|^{\Sigma}_{\rho}}{\rho(O) \rightsquigarrow O'}$$

$$(\textit{Het-Derive})\ \frac{O \rightsquigarrow O''}{O|^{\Sigma}_{\rho} \rightsquigarrow O'} \quad \text{if } \rho\colon O' \longrightarrow O'' \text{ is a conservative extension}$$

Conservativity of $\rho = (\Phi, \alpha, \beta)\colon O' \longrightarrow O''$ means that for each model $M' \in \mathrm{Mod}(SP')$, there is a model $M'' \in \mathrm{Mod}(SP'')$ with $\beta(M'') = M'$.

Heterogeneous completeness

Theorem

For a heterogeneous logical environment $\mathcal{HLE} : \mathcal{G} \longrightarrow co\mathcal{INS}$ (with some of the institutions having entailment systems), the proof calculi for heterogeneous specifications are sound for $\mathcal{I}^{\mathcal{HLE}}/\equiv$. If

1. *$\mathcal{HLE}$ is quasi-exact,*
2. *all institution comorphisms in $\mathcal{HLE}$ are weakly exact,*
3. *there is a set $\mathcal{L}$ of institutions in $\mathcal{HLE}$ that come with complete entailment systems,*
4. *all institutions in $\mathcal{L}$ are quasi-semi-exact,*
5. *from each institution in $\mathcal{HLE}$, there is some model-expansive comorphism in $\mathcal{HLE}$ going into some institution in $\mathcal{L}$,*

the proof calculus for entailments between heterogeneous specifications and sentences is complete over $\mathcal{I}^{\mathcal{HLE}}$.

Institutional model theory

1. Choose a proof of a meta theorem in logic.
2. Extract its essence by leaving out the irrelevant details and by identifying the conceptual structure and the causalities underlying the result.
3. Formulate the conceptual structure at the level of an abstract institution.
4. Lift the proof considered to the level of an abstract institution, shaping an abstract, generic scope of the result.
5. Determine the actual scope by analysing the abstract conditions used in the proof.

Craig interpolation (Ci) in abstract institutions

$$
\begin{array}{ccc}
\Sigma_1 \cap \Sigma_2 & \longrightarrow & \Sigma_1 \\
\downarrow & & \downarrow \\
\Sigma_2 & \longrightarrow & \Sigma_1 \cup \Sigma_2
\end{array}
$$

$\rho_1 \vdash_{\Sigma_1 \cup \Sigma_2} \rho_2$ iff
exists ρ s.th. $\rho_1 \vdash_{\Sigma_1} \rho$, $\rho \vdash_{\Sigma_2} \rho_2$.

Craig interpolation (Ci) in abstract institutions

$$\begin{tikzcd} \Sigma_1 \cap \Sigma_2 \arrow[r] \arrow[d] & \Sigma_1 \arrow[d] \\ \Sigma_2 \arrow[r] & \Sigma_1 \cup \Sigma_2 \end{tikzcd}$$

$\rho_1 \models_{\Sigma_1 \cup \Sigma_2} \rho_2$ iff
exists ρ s.th. $\rho_1 \models_{\Sigma_1} \rho$, $\rho \models_{\Sigma_2} \rho_2$.

Craig interpolation (Ci) in abstract institutions

$$\begin{tikzcd} \Sigma \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & \Sigma_1 \arrow[d, "\theta_1"] \\ \Sigma_2 \arrow[r, "\theta_2"'] & \Sigma' \end{tikzcd}$$

$\theta_1(\rho_1) \models_{\Sigma'} \theta_2(\rho_2)$ iff
exists ρ s.th. $\rho_1 \models_{\Sigma_1} \varphi_1(\rho)$, $\varphi_2(\rho) \models_{\Sigma_2} \rho_2$.

Craig interpolation (Ci) in abstract institutions

$$
\begin{array}{ccc}
\Sigma & \xrightarrow{\varphi_1} & \Sigma_1 \\
{\scriptstyle \varphi_2}\downarrow & & \downarrow{\scriptstyle \theta_1} \\
\Sigma_2 & \xrightarrow[\theta_2]{} & \Sigma'
\end{array}
$$

$\theta_1(E_1) \models_{\Sigma'} \theta_2(E_2)$ iff
exists E s.th. $E_1 \models_{\Sigma_1} \varphi_1(E)$, $\varphi_2(E) \models_{\Sigma_2} E_2$.

$(\mathscr{L},\mathscr{R})$-interpolation

Expecting Ci for all pushout squares is too much! (single sorted **FOL** yes, but many-sorted **FOL**, **EQL**, **HCL** no, etc.)

Solution: restrict (abstractly) $\varphi_1 \in \mathscr{L}$, $\varphi_2 \in \mathscr{R}$.

$$\begin{tikzcd}
\Sigma \arrow[r, "\varphi_1 \in \mathscr{L}"] \arrow[d, "\varphi_2 \in \mathscr{R}"'] & \Sigma_1 \arrow[d] \\
\Sigma_2 \arrow[r] & \Sigma'
\end{tikzcd}$$

$(\mathscr{L},\mathscr{R})$-interpolation

Expecting Ci for all pushout squares is too much! (single sorted **FOL** yes, but many-sorted **FOL**, **EQL**, **HCL** no, etc.)

Solution: restrict (abstractly) $\varphi_1 \in \mathscr{L}$, $\varphi_2 \in \mathscr{R}$.

$$\begin{tikzcd} \Sigma \arrow[r, "\varphi_1 \in \mathscr{L}"] \arrow[d, "\varphi_2 \in \mathscr{R}"'] & \Sigma_1 \arrow[d] \\ \Sigma_2 \arrow[r] & \Sigma' \end{tikzcd}$$

many-sorted **FOL**: $\mathscr{L}$ or $\mathscr{R}$ injective on the sorts.
EQL, **HCL**: $\mathscr{R}$ injective or $\mathscr{L}$ injective on sorts and ‘encapsulates’ operations.

Remembering Birkhoff

$$\mathscr{A}^{**} = HSP(\mathscr{A})$$

- $\mathscr{A}^{*} = \{(\forall X)t = t' \mid \mathscr{A} \models (\forall X)t = t'\}$;
- $\mathscr{A}^{**} = \{A \mid A \models \mathscr{A}^{*}\}$;
- P products, S subalgebras, H homomorphic images (quotients).

Birkhoff institutions

Very abstract definition of the (coarse-level) scope of Birkhoff variety thm.

Institution enhanced with

- $\mathscr{F}$ (designated) class of filters with $\{\{*\}\} \in \mathscr{F}$;
- $\mathrm{Mod}(\Sigma)$ has (categorical) $\mathscr{F}$-filtered products;
- $\mathscr{B}_\Sigma \subseteq |\mathrm{Mod}(\Sigma)| \times |\mathrm{Mod}(\Sigma)|$ closed under iso and

$$\mathscr{M}^{**} = \mathscr{B}_\Sigma^{-1}(\mathscr{F}\mathscr{M})$$

Captures hundreds of Birkhoff style axiomatizability results (e.g. Nemeti-Andreka, etc.)

Generic scope of Ci by axiomatizability

Theorem

Any Birkhoff institution $(\mathrm{Sig}, \mathrm{Sen}, \mathrm{Mod}, \models, \mathscr{F}, \mathscr{B})$ *with the weak model amalgamation property, has* $(\mathscr{L}, \mathscr{R})$*-Ci when*

1. *for each* $\varphi \in \mathscr{L}$, $\mathrm{Mod}(\varphi)$ *preserves* $\mathscr{F}$*-filtered products, and*
2.
   - *for each* $\varphi \in \mathscr{R}$, $\mathrm{Mod}(\varphi)$ *lifts* $\mathscr{B}$*, or*
   - *for each* $\varphi \in \mathscr{L}$, $\mathrm{Mod}(\varphi)$ *lifts* $\mathscr{B}^{-1}$ *and model isomorphisms.*

Actual scope of Ci by axiomatizability

Clarification of detailed technical causalities in the proof of Ci by axiomatizability; their analysis determine the actual scope:

- Amalgamation and preservation of filtered products obvious in concrete situations; at abstract level their role becomes evident.
- Lifting conditions less obvious, but in applications established smoothly. Semantic in nature they represent the technical link between the closure by $\mathscr{B}$ and the syntactic restrictions on the signature morphisms.

Surprising instances

Theorem

Many sorted Horn clause logic has $(\mathscr{L},\mathscr{R})$-Ci when

1. *$\mathscr{R}$ consists of injective signature morphisms, or*
2. *$\mathscr{L}$ consists of injective on the sorts signature morphisms and 'encapsulates' the operations.*

Conclusions

- Many more notions and results from logic can be generalised:
  - ultraproducts, Los' theorem
  - saturated models
  - Gödel's completeness theorem
- OntoIOp probably will be the first international standard based on category theory
- Also software tools benefit from the category-theoretic abstraction

Main references

- J. Goguen and R. Burstall. Institutions: abstract model theory for specification and programming. J.ACM 39, 1992.
- Till Mossakowski, Joseph Goguen, Razvan Diaconescu, Andrzej Tarlecki. What is a Logic? In Jean-Yves Beziau (Ed.), Logica Universalis, pp. 111–133, 2007.
- R. Diaconescu. An institution-independent proof of Craig interpolation theorem. Studia Logica 77(1), 2004.
- M. Petria, R. Diaconescu. Abstract Beth definability in institutions. J. Symbolic Logic 71(3), 2006.
- R. Diaconescu. *Institution-independent Model Theory*. Birkhäuser Basel, 2008.

Implicit definability in abstract institutions

Signature morphism $\varphi : \ \Sigma \to \Sigma'$ defined implicitly by Σ'-theory E' when the corresponding reduct

$$\mathrm{Mod}(\varphi) : \ \mathrm{Mod}(\Sigma', E') \subseteq \mathrm{Mod}(\Sigma') \to \mathrm{Mod}(\Sigma)$$

is injective.

- generalization from extensions of signatures with individual new symbols to any signature morphism;
- simpler to do without irrelevant details such as ‘extension’, ‘individual’.

Explicit definability in abstract institutions

$\varphi : \Sigma \to \Sigma'$ defined explicitly by E' when for each pushout

$$\begin{array}{ccc} \Sigma & \xrightarrow{\varphi \in \mathscr{L}} & \Sigma' \\ {\scriptstyle \theta \in \mathscr{R}} \downarrow & & \downarrow {\scriptstyle \theta'} \\ \Sigma_1 & \xrightarrow[\varphi_1]{} & \Sigma_1' \end{array}$$

and each $\rho \in \mathrm{Sen}(\Sigma_1')$, exists finite $E_\rho \subseteq \mathrm{Sen}(\Sigma_1)$

$$E' \models_{\Sigma'} (\forall \theta')(\rho \Leftrightarrow \varphi_1(E_\rho)).$$

(not necessarily to have $\forall$ and $\Leftrightarrow$)
θ, θ' stand for signature extensions with finite blocks of first-order variables, however ‘first-order’, ‘extension’, irrelevant details.

Beth definability

φ has the definability property iff a theory defines φ explicitly whenever it defines it implicitly.

Opposite implication immediate in **FOL** but highly non-trivial for abstract institutions (skipped here).

On the role of implication (2,3)

An institution has implication when for any ρ_1, ρ_2 there exists ρ such that

$$M \models \rho \text{ iff } M \models \rho_2 \text{ whenever } M \models \rho_1.$$

Implicit reliance upon implication in traditional proof of **FOL** Beth definability.

However we may render implication unnecessary by reformulating interpolation.

Craig-Robinson interpolation (CRi) (2,3)

‘Parameterises’ each instance of interpolation by a set of ‘secondary’ premises:

$$\begin{tikzcd} \Sigma \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & \Sigma_1 \arrow[d, "\theta_1"] \\ \Sigma_2 \arrow[r, "\theta_2"'] & \Sigma' \end{tikzcd}$$

$\theta_1(E_1) \cup \theta_2(\Gamma_2) \models_{\Sigma'} \theta_2(E_2)$ iff
exists E s.th. $E_1 \models_{\Sigma_1} \varphi_1(E)$, $\varphi_2(E) \cup \Gamma_2 \models_{\Sigma_2} E_2$.

Craig-Robinson interpolation (2,3)

- CRi => Ci
- In any compact institution with implication Ci <=> CRi (e.g. **FOL**, etc.)
- However CRi may exist in the absence of implication, e.g. many sorted Horn clause logic has $(\mathscr{L},\mathscr{R})$-CRi when $\mathscr{L}$ consists of signature morphisms that are injective on the sorts and 'encapsulate' operations.

Generic scope of Beth definability by interpolation

Theorem

In any compact institution with model amalgamation and $(\mathscr{L},\mathscr{R})$-CRi such that $\mathscr{L}$, $\mathscr{R}$ that are stable under pushouts, any signature morphism in $\mathscr{L}\cap\mathscr{R}$ has the definability property.

Actual scope of Beth definability by interpolation

By analysis of the conditions of above thm.

1. Compactness: common property established by various means:
   - completeness of finitary proof calculus;
   - preservation by ultraproducts;

   (Both treated at level of abstract institutions).
2. Model amalgamation: common easy property (already discussed);
3. Stability: pure technical condition, mild in the applications;
4. the only substantial condition is CRi.

Two actual instances

Corollary

In many-sorted **FOL** *any signature morphism that is injective on sorts has the definability property.*

In a logic without implication:

Corollary

In many-sorted **HCL** *any signature morphism that is injective on sorts and 'encapsulates' the operations has the definability property.*