Example: in $\mathbf{AlgSig}$

sort $String$
ops $a, \ldots, z : String$;
$\_ \,\hat{}\, \_ : String \times String \to String$

sorts $String, Nat, Array[String]$
ops $a, \ldots, z : String$;
$\_ \,\hat{}\, \_ : String \times String \to String$;
$empty : Array[String]$;
$put : Nat \times String \times Array[String] \to Array[String]$;
$get : Nat \times Array[String] \to String$

PO

sort $Elem$

sorts $Elem, Nat, Array[Elem]$
ops $empty : Array[Elem]$;
$put : Nat \times Elem \times Array[Elem] \to Array[Elem]$;
$get : Nat \times Array[Elem] \to Elem$

Tuning up the logical system

- various sets of formulae (Horn-clauses, first-order, higher-order, modal formulae, ...)
- various notions of algebra (partial algebras, relational structures, error algebras, Kripke structures, ...)
- various notions of signature (order-sorted, error, higher-order signatures, sets of propositional variables, ...)
- (various notions of signature morphisms)

No best logic for everything

Solution:

Work with an arbitrary logical system

Institution

- a category $\mathbf{Sign}$ of *signatures*
- a functor $\mathbf{Sen}\colon \mathbf{Sign} \to \mathbf{Set}$
  - $\mathbf{Sen}(\Sigma)$ is the set of Σ-*sentences*, for $\Sigma \in |\mathbf{Sign}|$
- a functor $\mathbf{Mod}\colon \mathbf{Sign}^{op} \to \mathbf{Cat}$
  - $\mathbf{Mod}(\Sigma)$ is the category of Σ-*models*, for $\Sigma \in |\mathbf{Sign}|$
- for each $\Sigma \in |\mathbf{Sign}|$, Σ-*satisfaction relation* $\models_\Sigma \subseteq |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$

subject to the *satisfaction condition*:

$$\boxed{M'|_\sigma \models_\Sigma \varphi \iff M' \models_{\Sigma'} \sigma(\varphi)}$$

where $\sigma\colon \Sigma \to \Sigma'$ in $\mathbf{Sign}$, $M' \in |\mathbf{Mod}(\Sigma')|$, $\varphi \in \mathbf{Sen}(\Sigma)$, $M'|_\sigma$ stands for $\mathbf{Mod}(\sigma)(M')$, and $\sigma(\varphi)$ for $\mathbf{Sen}(\sigma)(\varphi)$

Typical institutions

- **EQ** — equational logic
- **FOEQ** — first-order logic (with predicates and equality)
- **PEQ**, **PFOEQ** — as above, but with partial operations
- **HOL** — higher-order logic (in the HOL version)
- logics of constraints (fitted via signature morphisms)
- **CASL** — the logic of CASL: partial first-order logic with equality, predicates, generation constraints, and subsorting

And somewhat less typical examples:

- modal logics
- three-valued logics
- programming language semantics

As well as perhaps unexpected:

- no sentences
- no models
- no signatures
- trivial satisfaction relations
- sets of sentences as sentences
- sets of sentences as signatures
- classes of models as sentences
- sets of sentences as models
- ...

Let's fix an institution $\mathbf{I} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, \langle \models_\Sigma \rangle_{\Sigma \in |\mathbf{Sign}|})$ for a while.

Semantic entailment

$$\boxed{\Phi \models_\Sigma \varphi}$$

Σ-sentence φ is a semantic consequence of a set of Σ-sentences Φ if φ holds in every Σ-algebra that satisfies Φ.

BTW:

- *Models* of a set of sentences: $Mod(\Phi) = \{M \in |\mathbf{Mod}(\Sigma)| \mid M \models \Phi\}$
- *Theory* of a class of models: $Th(\mathcal{C}) = \{\varphi \mid \mathcal{C} \models \varphi\}$
- $\Phi \models \varphi \iff \varphi \in Th(Mod(\Phi))$
- Mod and Th form a *Galois connection*

Preservation of entailment

Fact:

$$\boxed{\Phi \models_\Sigma \varphi \implies \sigma(\Phi) \models_{\Sigma'} \sigma(\varphi)}$$

for $\sigma\colon \Sigma \to \Sigma'$, $\Phi \subseteq \mathbf{Sen}(\Sigma)$, $\varphi \in \mathbf{Sen}(\Sigma)$.

If the reduct $\_|_\sigma \colon |\mathbf{Mod}(\Sigma')| \to |\mathbf{Mod}(\Sigma)|$ *is surjective, then*

$$\boxed{\Phi \models_\Sigma \varphi \iff \sigma(\Phi) \models_{\Sigma'} \sigma(\varphi)}$$

Adding provability

Add to institution:

- *proof-theoretic entailment*:

$$\vdash_\Sigma \subseteq \wp(\mathbf{Sen}(\Sigma)) \times \mathbf{Sen}(\Sigma)$$

for each signature $\Sigma \in |\mathbf{Sign}|$, closed under
  - weakening, reflexivity, transitivity (cut)
  - translation along signature morphisms

Require:

- *soundness*: $\Phi \vdash_\Sigma \varphi \implies \Phi \models_\Sigma \varphi$

(?) *completeness*: $\Phi \models_\Sigma \varphi \implies \Phi \vdash_\Sigma \varphi$

Amalgamation

$$\begin{tikzcd}[column sep=small]
 & & M' \arrow[dll, blue] \arrow[drr, blue] & & \\
 & & \Sigma' & & \\
M_1 = M'|_{\sigma_2'} \arrow[ddrr, blue] & \Sigma_1 \arrow[ur, "\sigma_2'"] & \text{PO} & \Sigma_2 \arrow[ul, "\sigma_1'"'] & M'|_{\sigma_1'} = M_2 \arrow[ddll, blue] \\
 & & \Sigma \arrow[ul, "\sigma_1"] \arrow[ur, "\sigma_2"'] & & \\
 & & M_1|_{\sigma_1} = M_2|_{\sigma_2} & &
\end{tikzcd}$$

Given $M_1 \in |\mathbf{Mod}(\Sigma_1)|$ and $M_2 \in |\mathbf{Mod}(\Sigma_2)|$ with $M_1|_{\sigma_1} = M_2|_{\sigma_2}$, there is a unique $M' \in |\mathbf{Mod}(\Sigma')|$ with $M'|_{\sigma_1'} = M_2$ and $M'|_{\sigma_2'} = M_1$

Adding amalgamation

Assume:

- the model functor $\mathbf{Mod}\colon \mathbf{Sign}^{op} \to \mathbf{Cat}$ is *continuous* (maps colimits of signatures to limits of model categories)

Fact: $\mathbf{Alg}\colon \mathbf{AlgSig}^{op} \to \mathbf{Cat}$ *is continuous.*

Amalgamation property: *Amalgamation property follows for a pushout in* $\mathbf{Sign}$ *if* $\mathbf{Mod}$ *maps it to a pullback in* $\mathbf{Cat}$*:*

$$\begin{tikzcd}
\Sigma_1 \arrow[r, "\sigma_2'"] & \Sigma' & & \mathbf{Mod}(\Sigma_1) \arrow[d, "-|_{\sigma_1}"'] & \mathbf{Mod}(\Sigma') \arrow[l, "-|_{\sigma_2'}"'] \arrow[d, "-|_{\sigma_1'}"] \\
\Sigma \arrow[u, "\sigma_1"] \arrow[r, "\sigma_2"'] \arrow[ur, phantom, "\text{PO}"] & \Sigma_2 \arrow[u, "\sigma_1'"'] \arrow[r, mapsto, "\mathbf{Mod}"] & {} & \mathbf{Mod}(\Sigma) \arrow[ur, phantom, "\text{PB}"] & \mathbf{Mod}(\Sigma_2) \arrow[l, "-|_{\sigma_2}"]
\end{tikzcd}$$

Adding interpolation

$\mathbf{I}$ has the *interpolation property* for a pushout in $\mathbf{Sign}$

$$\begin{tikzcd}
 & \Sigma' & \\
\Sigma_1 \arrow[ur, "\sigma_2'"] & & \Sigma_2 \arrow[ul, "\sigma_1'"'] \\
 & \Sigma \arrow[ul, "\sigma_1"] \arrow[ur, "\sigma_2"'] &
\end{tikzcd}$$

if for all $\varphi_1 \in \mathbf{Sen}(\Sigma_1)$ and $\varphi_2 \in \mathbf{Sen}(\Sigma_2)$ such that $\sigma_2'(\varphi_1) \models_{\Sigma'} \sigma_1'(\varphi_2)$ there is $\theta \in \mathbf{Sen}(\Sigma)$ such that $\varphi_1 \models_{\Sigma_1} \sigma_1(\theta)$ and $\sigma_2(\theta) \models_{\Sigma_2} \varphi_2$.

Fact: $\mathbf{FOEQ}$ *has the interpolation property for all pushouts of morphisms injective on sorts.*

WORK IN AN ARBITRARY INSTITUTION

... adding extra structure and assumptions only if really needed ...

Revised rough analogy

module interface	$\rightsquigarrow$	signature
module	$\rightsquigarrow$	model
module specification	$\rightsquigarrow$	class of models