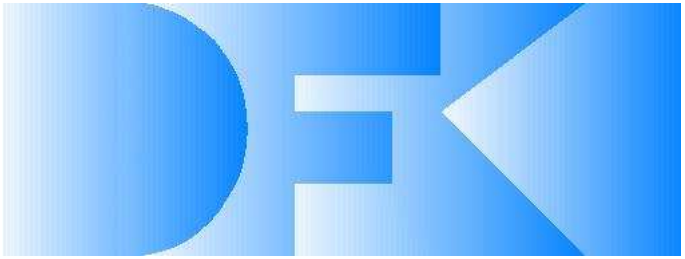
Heterogeneous Proofs, with an Example About Relation Algebras

Till Mossakowski

IFIP WG 1.3 meeting, 30 June 2006

DFKI Lab Bremen

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH



- Branch of the German Research Center for Artificial Intelligence, Saarbrücken and Kaiserslautern
- Initial phase: 3 years
- Project SAMS: Safe autonomous mobile systems (3 open positions)
- Project ForRBAC: Formal specification and verification of role base access control policies (1 open position)

Overview

- Motivation
- Institutions with proofs
- Structured specification and their proofs
- Heterogeneous specifications and their proofs
- The heterogeneous tool set
- A sample heterogeneous proof

Motivation

"There is a population explosion among the logical systems used in computer science." (Joseph Goguen)

"It is a fact of life that no single perspective, no single formalization of level of abstraction suffices to represent a system and reason about its behaviour." (José Meseguer)

"As can be seen, a plethora of formalisms for the verification of programs, and, in particular, for the verification of concurrent programs has been proposed. Their relationship is almost clear and for many different formalisms we already know if translations between them exist." (Klaus Schneider)

Motivation (cont'd)

- multiple viewpoints are used when specifying complex software intensive systems
- changes in the formalisms may be needed in the course of software development
- even for one and the same mathematical formalism, there are many slightly varying input languages
- the occasional use of a more complex formalism should not destroy the benefits of mainly using a simpler formalism

$\Rightarrow$ How can we integrate formalisms and tools?

Institutions with proofs

Institutions

- category $\mathbf{Sign}$ of *signatures*,
- sentence functor $\mathbf{Sen}\colon \mathbf{Sign} \longrightarrow \mathbf{Set}$
- a functor $\mathbf{Mod}\colon \mathbf{Sign}^{op} \longrightarrow \mathcal{CAT}$
- satisfaction relation $\models_\Sigma \subseteq |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$,

such that

$$M' \models_{\Sigma'} \mathbf{Sen}(\sigma)(\varphi) \Leftrightarrow \mathbf{Mod}(\sigma)(M)' \models_\Sigma \varphi$$

or shortly

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

Examples of Logics Formalized as Institutions

- propositional, first-order, higher-order logic, polymorphic logics
- logics of partial functions
- modal logic, epistemic logic, deontic logic, description logics, logics of knowledge and belief, agent logics
- μ-calculus, dynamic logic
- spatial logics, temporal logics, process logics, object logics
- intuitionistic logic
- linear logic, non-monotone logics, fuzzy logics
- paraconsistent logic, database query languages

Institutions With Proofs

There are several approaches in the literature:

- entailments $\Gamma \vdash_\Sigma \varphi$ ($\Gamma =$ set of sentences)
  π-institutions [Fiadeiro & Sernadas 88], entailment systems [Meseguer 89]
- proof categories: proofs $\varphi \xrightarrow{p} \psi$ [Goguen & Burstall 85]
- multicategories: proofs $\Gamma \xrightarrow{p} \varphi$ for finite Γ [Meseguer 87]
- power-ordered proof categories: proofs $\Gamma \xrightarrow{p} \varphi$ for (in)finite Γ
  [Mossakowski, Goguen, Diaconescu & Tarlecki 05]
- preorder-enriched categories: proofs reductions $\varphi \xrightarrow{p_2} \psi \geq \varphi \xrightarrow{p_1} \psi$
  [Mossakowski, Rabe, Schröder, Goguen, de Pavia - unpublished]

Here, we just use entailment systems.

Entailment systems

- category $\mathbf{Sign}$ of *signatures*,
- sentence functor $\mathbf{Sen}\colon \mathbf{Sign} \longrightarrow \mathbf{Set}$
- entailment relation $\vdash_\Sigma \subseteq \mathcal{P}(\mathbf{Sen}(\Sigma)) \times \mathbf{Sen}(\Sigma)$, such that the following conditions are satisfied:

  (a) reflexivity: for any $\varphi \in \mathbf{Sen}(\Sigma)$, $\{\varphi\} \vdash_\Sigma \varphi$,

  (b) monotonicity: if $\Gamma \vdash_\Sigma \varphi$ and $\Gamma' \supseteq \Gamma$ then $\Gamma' \vdash_\Sigma \varphi$,

  (c) transitivity: if $\Gamma \vdash_\Sigma \varphi_i$, for $i \in I$, and $\Gamma \cup \{\varphi_i \mid i \in I\} \vdash_\Sigma \psi$, then $\Gamma \vdash_\Sigma \psi$,

  (d) translation: if $\Gamma \vdash_\Sigma \varphi$, then $\sigma(\Gamma) \vdash_{\Sigma'} \sigma(\varphi)$ (for $\sigma\colon \Sigma \longrightarrow \Sigma'$)

Structured specification and their proofs

Structured specifications over an arbitrary institution

$$SP ::= \langle \Sigma, \Gamma \rangle \mid SP \cup SP \mid \sigma(SP) \mid \sigma^{-1}(SP)$$

. . . and their semantics

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

$$Sig(SP_1 \cup SP_2) = Sig(SP_1) = Sig(SP_2)$$
$$\mathbf{Mod}(SP_1 \cup SP_2) = \mathbf{Mod}(SP_1) \cap \mathbf{Mod}(SP_2)$$

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

$$Sig((\sigma : \Sigma_1 \longrightarrow \Sigma_2)^{-1}(SP)) = \Sigma_1$$
$$\mathbf{Mod}((\sigma : \Sigma_1 \longrightarrow \Sigma_2)^{-1}(SP)) = \{M|_\sigma \mid M \in \mathbf{Mod}(SP)\}$$

Proof calculus for entailment (Borzyszkowski)

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

$$(sum1)\ \frac{SP_1 \vdash \varphi}{SP_1 \cup SP_2 \vdash \varphi} \qquad (sum2)\ \frac{SP_1 \vdash \varphi}{SP_1 \cup SP_2 \vdash \varphi}$$

$$(trans)\ \frac{SP \vdash \varphi}{\sigma(SP) \vdash \sigma(\varphi)} \qquad (derive)\ \frac{SP \vdash \sigma(\varphi)}{\sigma^{-1}(SP) \vdash \varphi}$$

Proof calculus for refinement (Borzyszkowski)

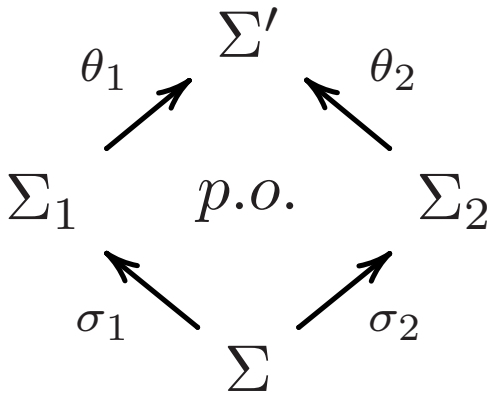
$$(Basic)\ \frac{SP \vdash \Gamma}{\langle \Sigma, \Gamma \rangle \rightsquigarrow SP} \qquad (Sum)\ \frac{SP_1 \rightsquigarrow SP \quad SP_2 \rightsquigarrow SP}{SP_1 \cup SP_2 \rightsquigarrow SP}$$

$$(Trans_1)\ \frac{SP \rightsquigarrow \theta(SP') \quad \theta = \sigma^{-1}}{\sigma(SP) \rightsquigarrow SP'} \qquad (Trans_2)\ \frac{SP \rightsquigarrow \sigma^{-1}(SP')}{\sigma(SP) \rightsquigarrow SP'}$$

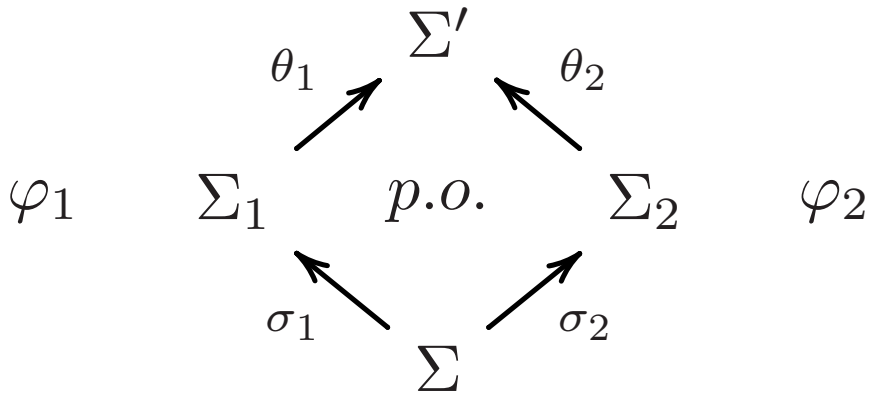
$$(Derive)\ \frac{SP \rightsquigarrow SP''}{\sigma^{-1}(SP) \rightsquigarrow SP'} \qquad \text{if } \sigma\colon SP' \longrightarrow SP'' \text{ is a conservative extension}$$

$$(Trans\text{-}equiv)\ \frac{\theta(\sigma(SP)) \rightsquigarrow SP'}{\theta \circ \sigma(SP) \rightsquigarrow SP'}$$

Craig interpolation

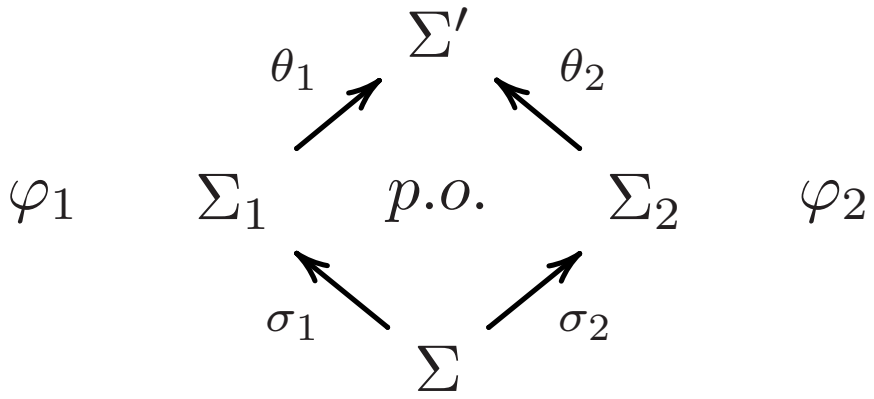


Craig interpolation



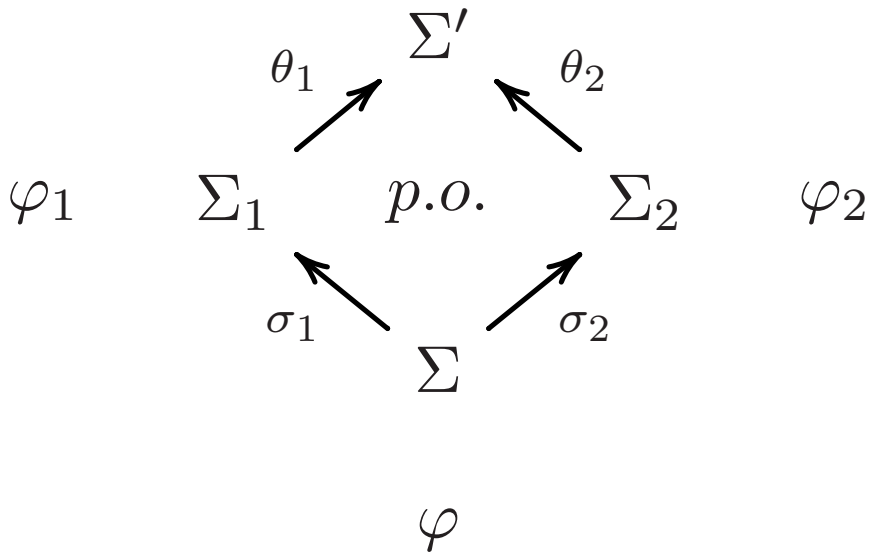
Craig interpolation

$$\theta_1(\varphi_1) \models \theta_2(\varphi_2)$$



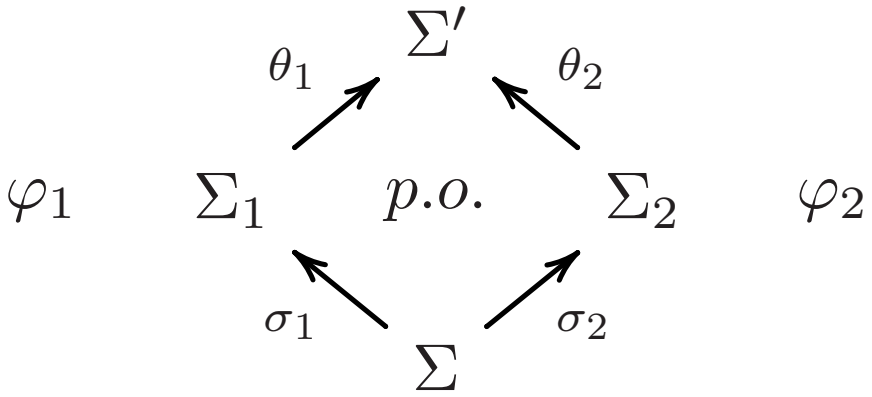
Craig interpolation

$$\theta_1(\varphi_1) \models \theta_2(\varphi_2)$$



Craig interpolation

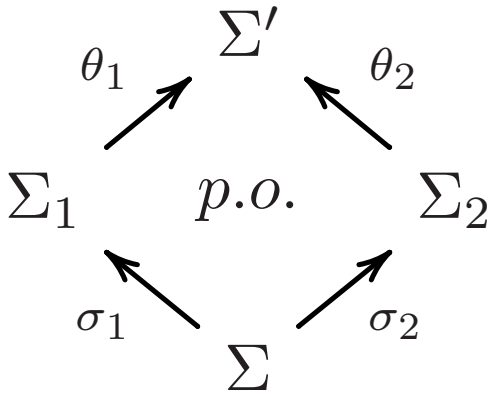
$$\theta_1(\varphi_1) \models \theta_2(\varphi_2)$$



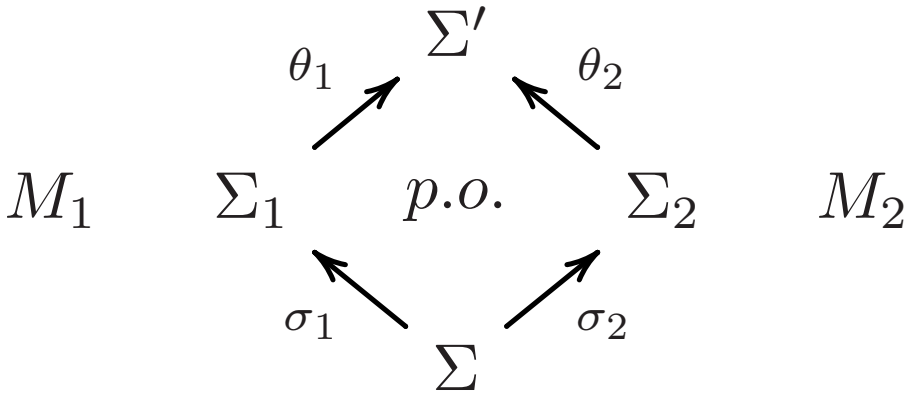
$$\varphi$$

$$\varphi_1 \models \sigma_1(\varphi) \text{ and } \sigma_2(\varphi) \models \varphi_2.$$

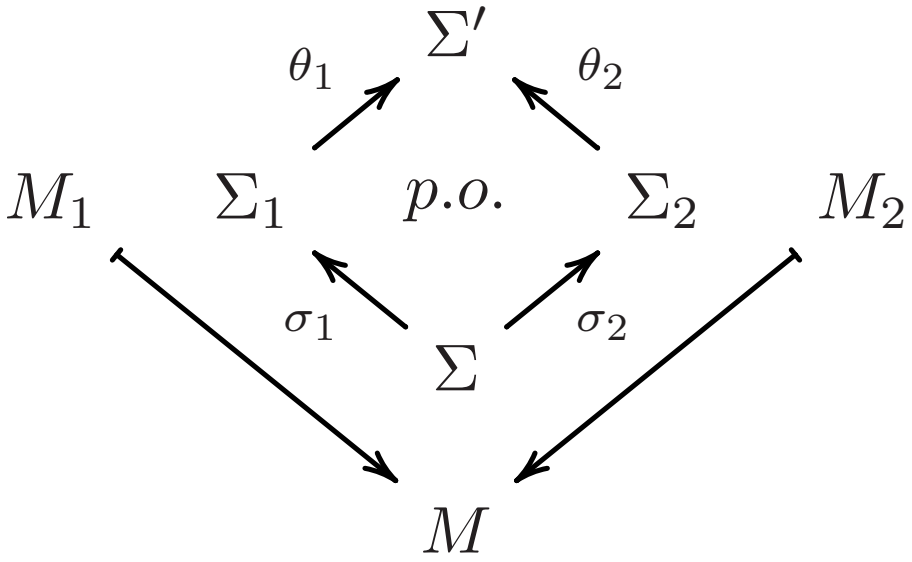
(Weak) amalgamation



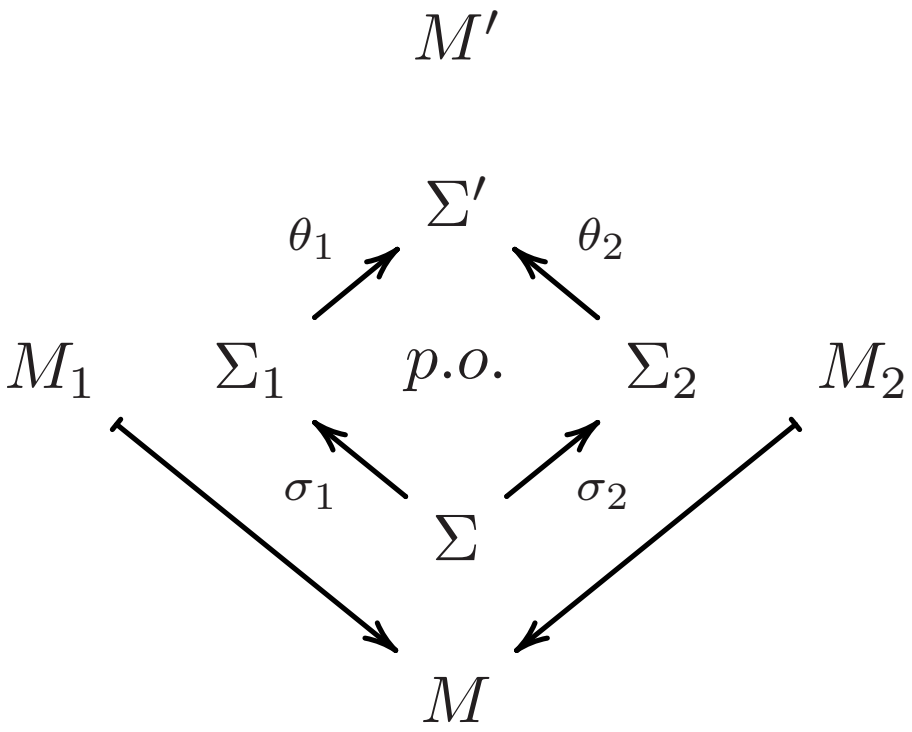
(Weak) Amalgamation



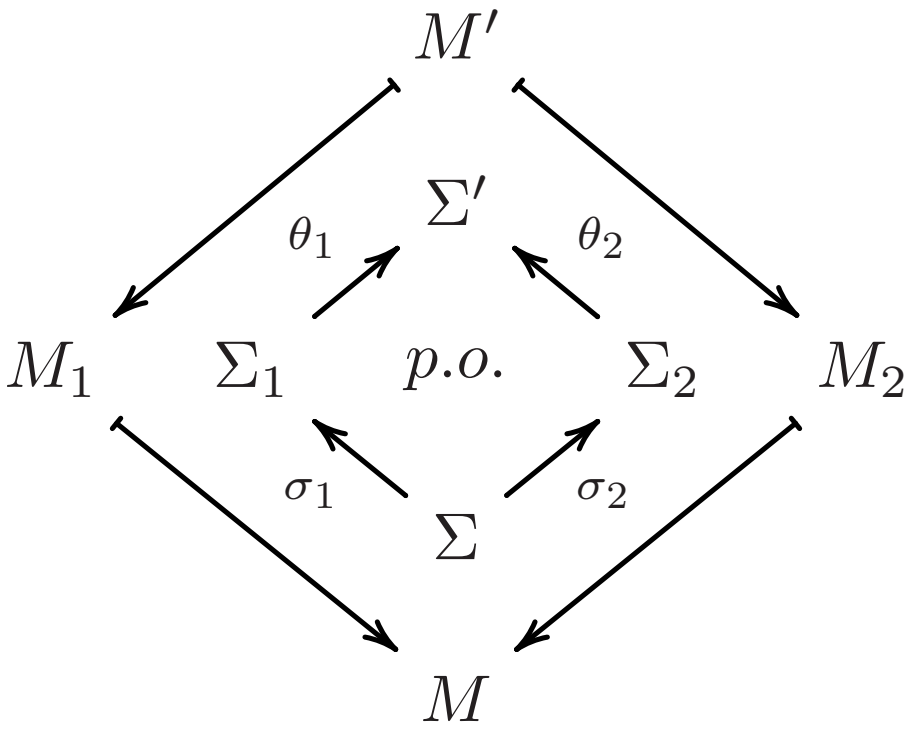
(Weak) Amalgamation



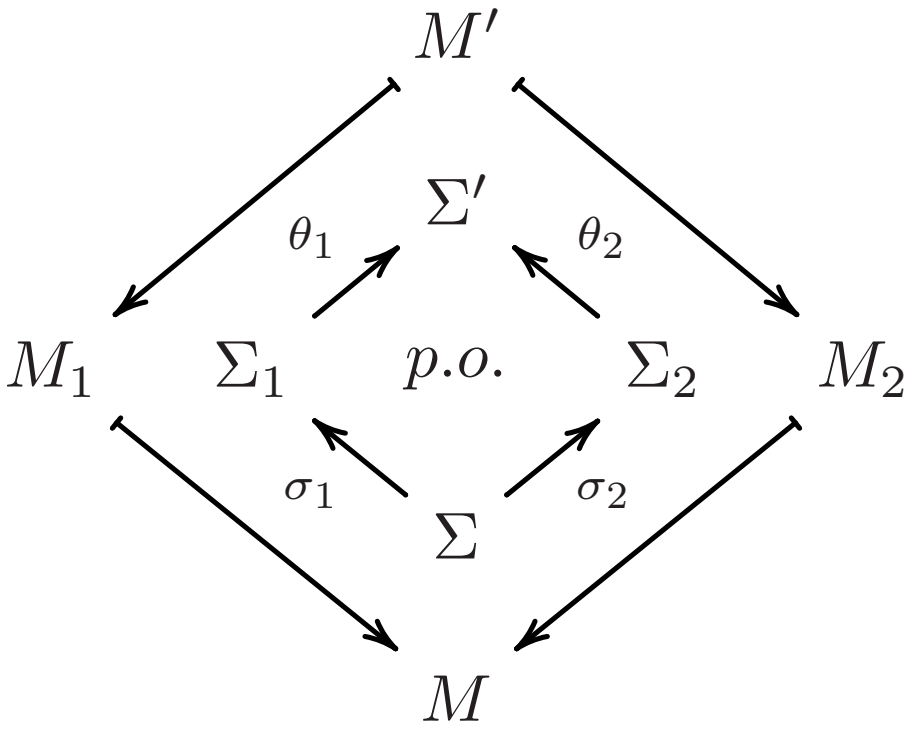
(Weak) Amalgamation



(Weak) Amalgamation



(Weak) Amalgamation



In case of amalgamation, M' must be unique.

I is called (weakly) semi-exact if it admits (weak) amalgamation for pushouts.

Soundness and Completeness

Under the assumptions that

- the institution has the Craig interpolation property,
- the institution admits weak amalgamation, and
- the institution has conjunction and implication and
- the institution is equipped with a sound and complete entailment system,

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

Note that for refinement, an
oracle for conservative extensions is needed.

Problem: Implication and Craig interpolation often fail!

Development graphs $\mathcal{S} = \langle \mathcal{N}, \mathcal{L} \rangle$

Nodes in $\mathcal{N}$: (Σ^N, Γ^N) with

- Σ^N signature,
- $\Gamma^N \subseteq \mathbf{Sen}(\Sigma^N)$ set of local axioms.

Links in $\mathcal{L}$:

- global $M \xrightarrow{\sigma} N$, where $\sigma : \Sigma^M \to \Sigma^N$,
- local $M \xrightarrow{\sigma} N$ where $\sigma : \Sigma^M \to \Sigma^N$, or
- hiding $M \xrightarrow[h]{\sigma} N$ where $\sigma : \Sigma^N \to \Sigma^M$
  going against the direction of the link.

Semantics of development graphs

$\mathbf{Mod}_S(N)$ consists of those Σ^N-models n for which

1. n satisfies the local axioms Γ^N,
2. for each $K \xrightarrow{\sigma} N \in \mathcal{S}$, $n|_\sigma$ is a K-model,
3. for each $K \xrightarrow{\sigma} N \in \mathcal{S}$,
   $n|_\sigma$ satisfies the local axioms Γ^K,
4. for each $K \xrightarrow[h]{\sigma} N \in \mathcal{S}$,
   n has a σ-expansion k (i.e. $k|_\sigma = n$) that is a K-model.

Theorem links

Theorem links come in two versions:

- global theorem links $M \xrightarrow{\sigma} N$, where $\sigma\colon \Sigma^M \longrightarrow \Sigma^N$,
  - $\mathcal{S} \models M \xrightarrow{\sigma} N$ iff for all $n \in \mathbf{Mod}_S(N)$, $n|_\sigma \in \mathbf{Mod}_S(M)$.
- local theorem links $M \overset{\sigma}{\dashrightarrow} N$, where $\sigma\colon \Sigma^M \longrightarrow \Sigma^N$,
  - $\mathcal{S} \models M \overset{\sigma}{\dashrightarrow} N$ iff for all $n \in \mathbf{Mod}_S(N)$, $n|_\sigma \models \Gamma^M$.
- the calculus reduces these to local proof obligations.

Completeness of the calculus

Theorem If the institution has weak amalgamation and is equipped with a sound and complete entailment system, then the calculus for development graphs is sound and complete.

Again, an oracle for conservative extensions is needed.

Heterogeneous specifications and their proofs

Heterogeneous Specification

Recall the three modes of heterogeneous specification (Andrzej's talk):

- "universal" logics/logical frameworks
- Focused heterogeneous specification
- Distributed heterogeneous specification

Here, we concentrate on the distributed case, which is most general.

Basic data for heterogeneous specification

- a diagram of institutions and (theoroidal) (semi) (co)morphisms
- some of the institutions equipped with an entailment system, preferably supported by proof tool(s)

Institution Morphisms Versus Comorphisms

$$\begin{array}{ccc} \mathbf{Sign}^I & \xrightarrow{\Phi} & \mathbf{Sign}^J \\ \mathbf{Sen}^I & \xleftarrow{\alpha} & \mathbf{Sen}^J \circ \Phi \\ \mathbf{Mod}^I & \xrightarrow{\beta} & \mathbf{Mod}^J \circ \Phi^{op} \end{array}$$

morphisms $\mu = (\Phi, \alpha, \beta)$

$$\begin{array}{ccc} \mathbf{Sign}^I & \xrightarrow{\Phi} & \mathbf{Sign}^J \\ \mathbf{Sen}^I & \xrightarrow{\alpha} & \mathbf{Sen}^J \circ \Phi \\ \mathbf{Mod}^I & \xleftarrow{\beta} & \mathbf{Mod}^J \circ \Phi^{op} \end{array}$$

comorphisms $\rho = (\Phi, \alpha, \beta)$

Heterogeneous (Grothendieck) Signature Morphisms

$$(I_1, \Sigma_1) \xrightarrow{(\mu,\sigma)} (I_2, \Sigma_2)$$

morphisms

where $\mu\colon I_2 \longrightarrow I_1$, $\sigma\colon \Sigma_1 \longrightarrow \Phi(\Sigma_2)$

$$(I_1, \Sigma_1) \xrightarrow{(\rho,\sigma)} (I_2, \Sigma_2)$$

comorphisms

where $\rho\colon I_1 \longrightarrow I_2$, $\sigma\colon \Phi(\Sigma_1) \longrightarrow \Sigma_2$

. . . and their decomposition:

$$(I_1, \Sigma_1) \xrightarrow{(id,\sigma)} (I_1, \Phi(\Sigma_2)) \xrightarrow{(\mu,id)} (I_2, \Sigma_2)$$

morphisms

$$(I_1, \Sigma_1) \xrightarrow{(\rho,id)} (I_2, \Phi(\Sigma_1)) \xrightarrow{(id,\sigma)} (I_2, \Sigma_2)$$

comorphisms

Heterogeneous reducts

The decomposition of heterogeneous signature morphisms:

$$(I_1, \Sigma_1) \xrightarrow{(id,\sigma)} (I_1, \Phi(\Sigma_2)) \xrightarrow{(\mu,id)} (I_2, \Sigma_2) \quad \text{morphisms}$$

$$(I_1, \Sigma_1) \xrightarrow{(\rho,id)} (I_2, \Phi(\Sigma_1)) \xrightarrow{(id,\sigma)} (I_2, \Sigma_2) \quad \text{comorphisms}$$

leads to model reducts:

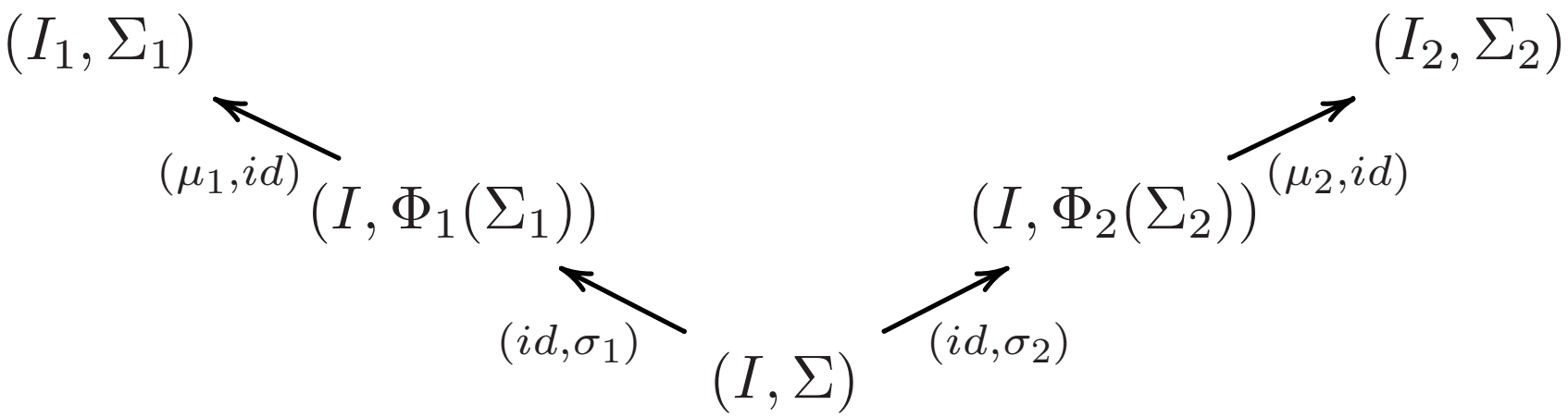
$$\mathbf{Mod}(I_1, \Sigma_1) \xleftarrow{-|\sigma} \mathbf{Mod}(I_1, \Phi(\Sigma_2)) \xleftarrow{\beta_{\Sigma_2}} \mathbf{Mod}(I_2, \Sigma_2)$$

morphisms

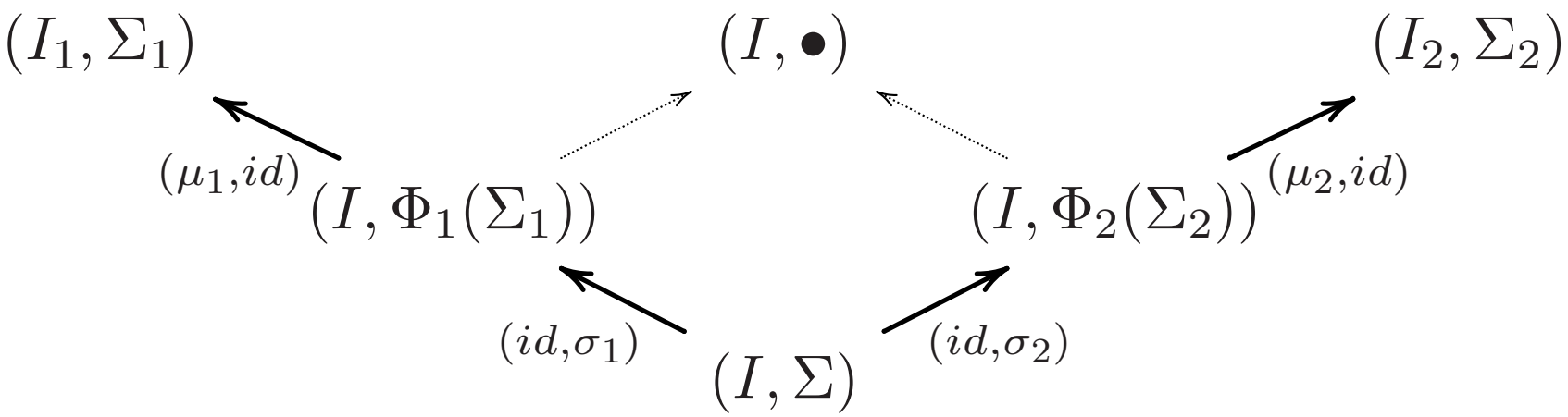
$$\mathbf{Mod}(I_1, \Sigma_1) \xleftarrow{\beta_{\Sigma_1}} \mathbf{Mod}(I_2, \Phi(\Sigma_1)) \xleftarrow{-|\sigma} \mathbf{Mod}(I_2, \Sigma_2)$$

comorphisms

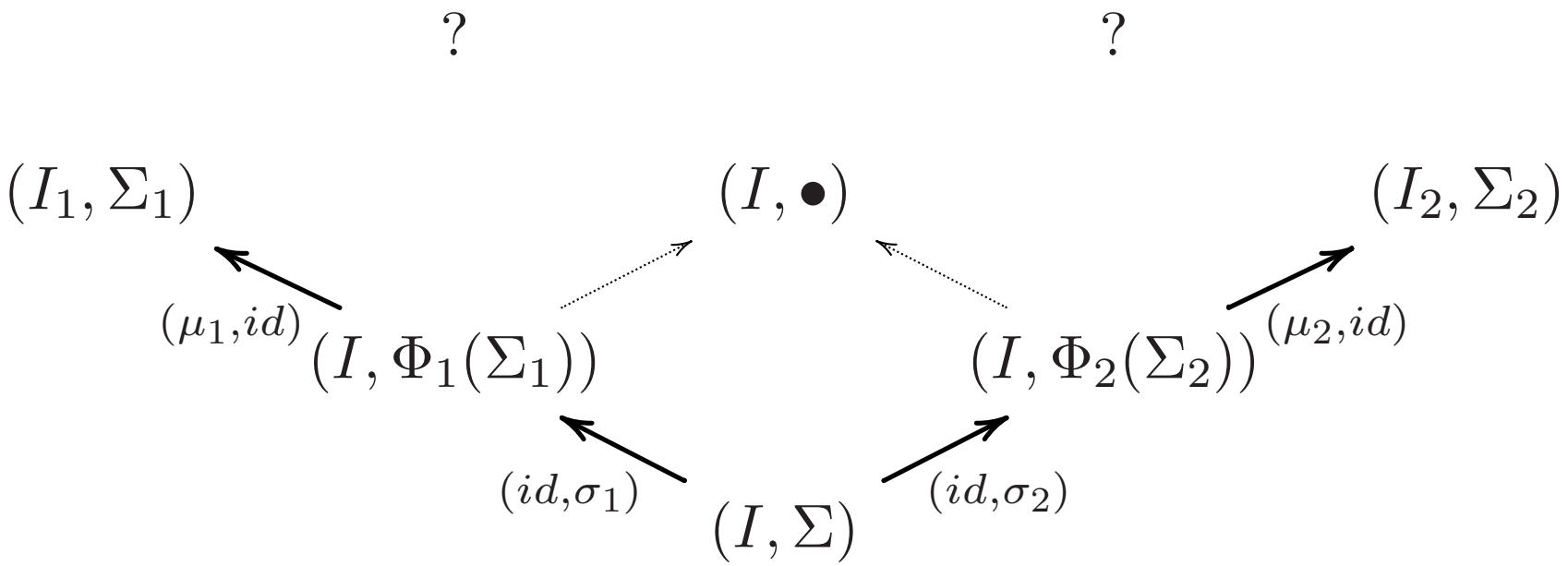
Heterogeneous (Weak) Amalgamation with Morphisms



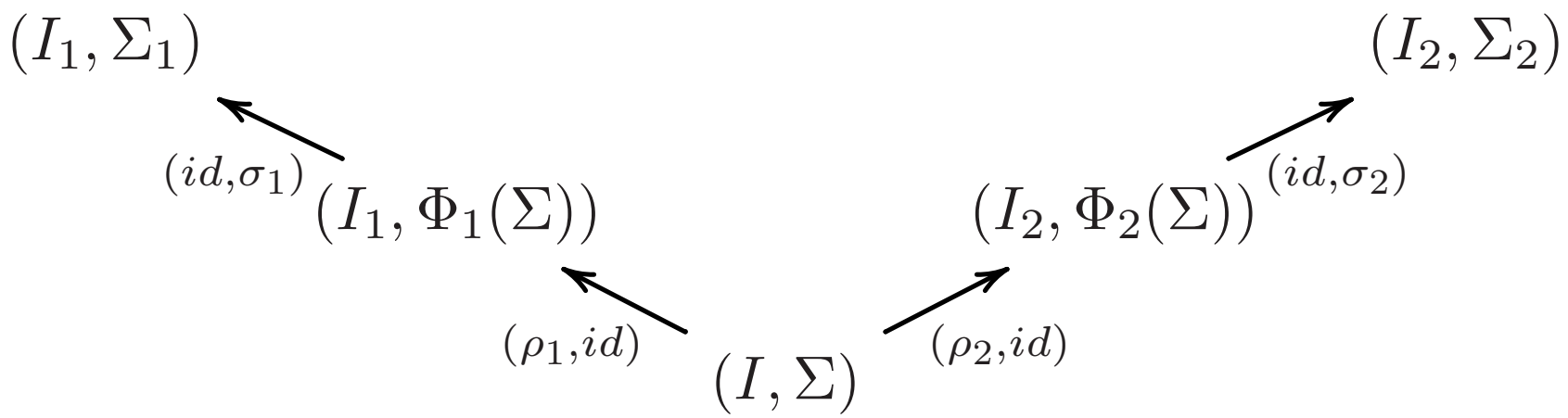
Heterogeneous (Weak) Amalgamation with Morphisms



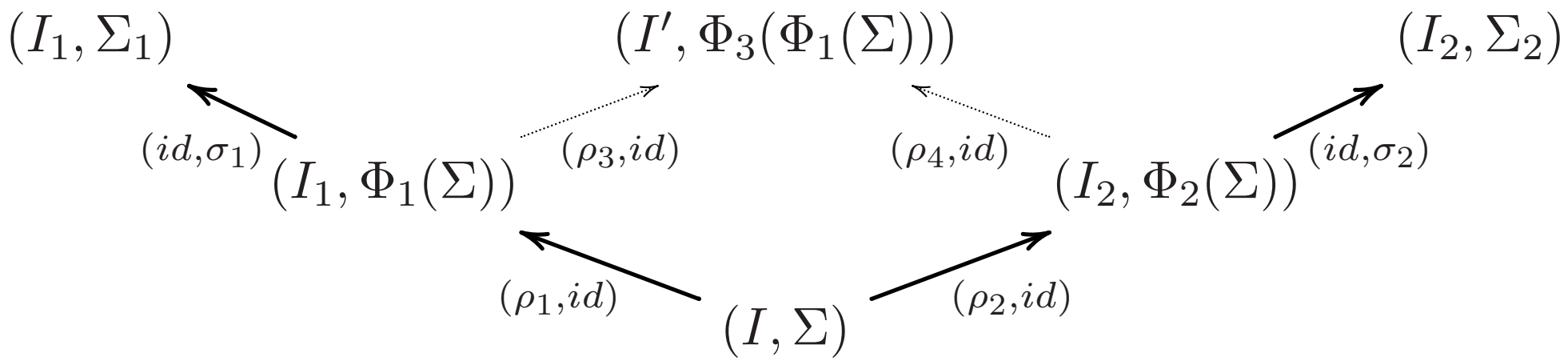
Heterogeneous (Weak) Amalgamation with Morphisms



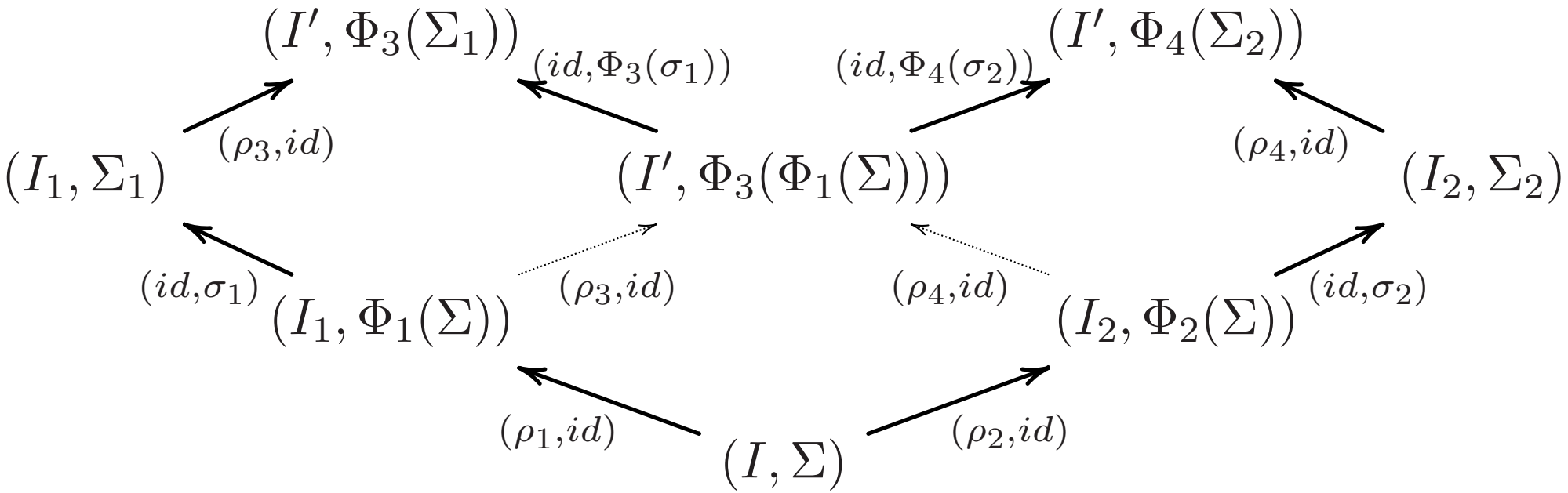
Heterogeneous (Weak) Amalgamation with Comorphisms



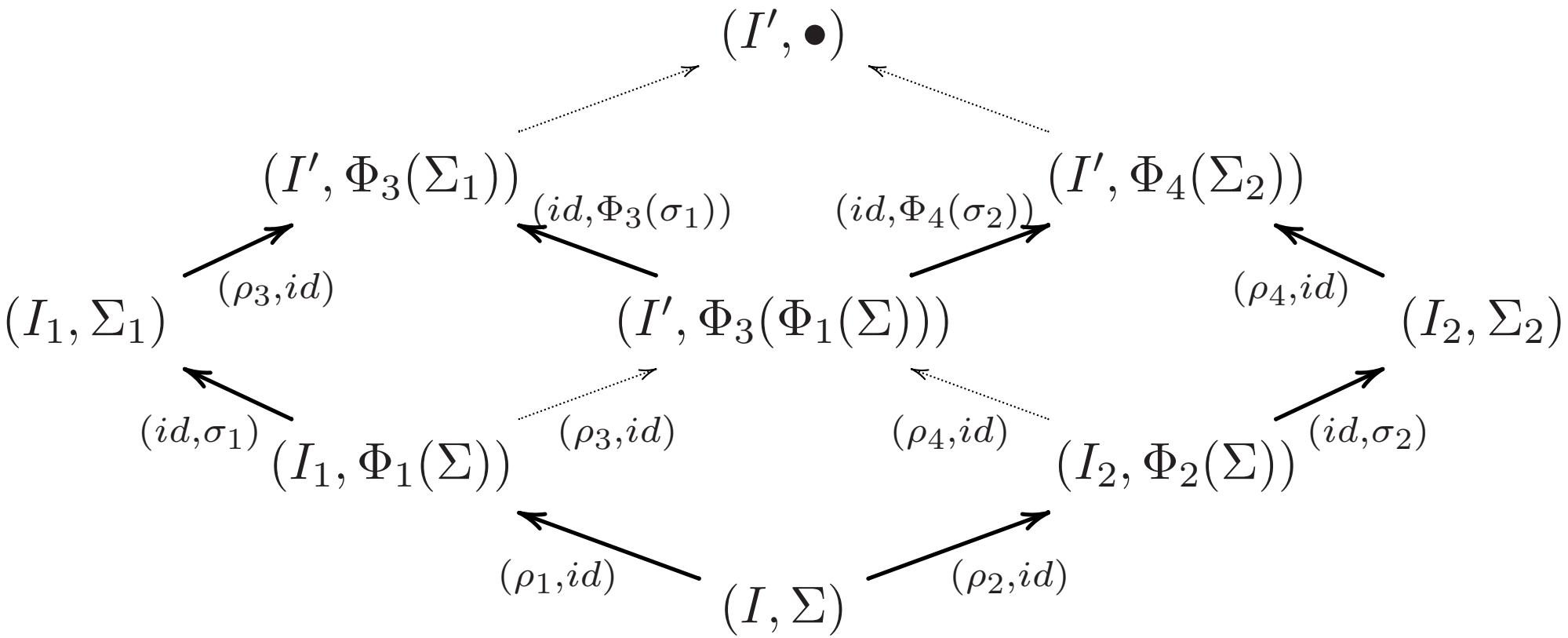
Heterogeneous (Weak) Amalgamation with Comorphisms



Heterogeneous (Weak) Amalgamation with Comorphisms



Heterogeneous (Weak) Amalgamation with Comorphisms



Heterogeneous (Weak) Amalgamation

The comorphism-based Grothendieck institution admits (weak) amalgamation under (weak) amalgamation assumptions about the indexing, the individual institutions and the comorphisms.

For the morphism-based Grothendieck institution, additionally right-adjointness of the Φ is needed. But then one can pass over to the adjoint comorphisms [Arrais Fiadeiro 96].

Completeness for Heterogeneous Development Graphs

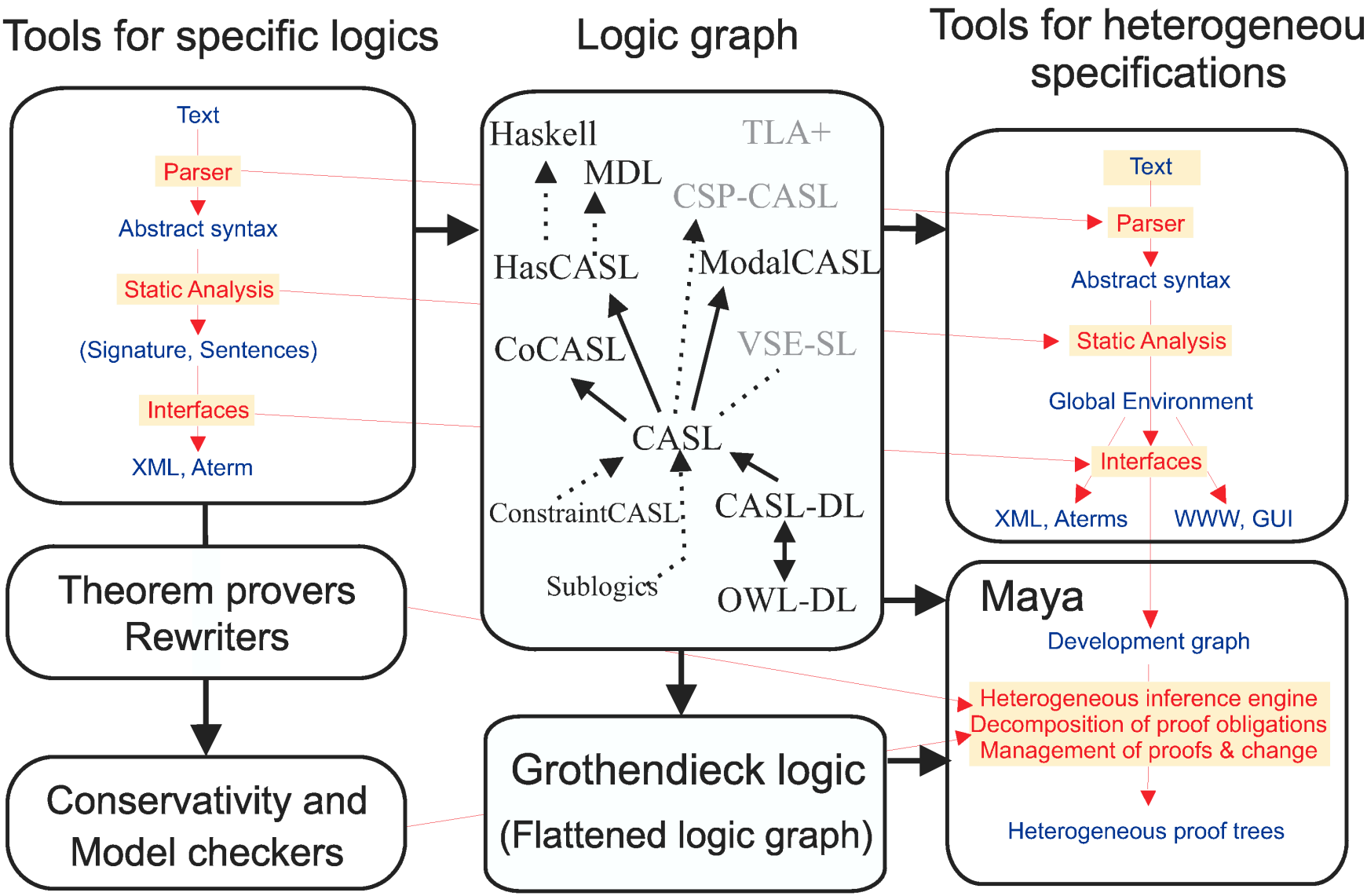
Theorem For an indexed coinstitution $\mathcal{I}\colon Ind^* \longrightarrow \mathbf{CoIns}$, the proof calculus for heterogeneous development graphs is sound. If, moreover,

- $\mathcal{I}$ is quasi-exact,
- all institution comorphisms in $\mathcal{I}$ are weakly exact,
- there is a set $\mathcal{L}$ of institutions in $\mathcal{I}$ that are equipped with sound and complete entailment systems, and the rule system is extended with a (sound and complete) oracle for conservative extension for each institution in $\mathcal{L}$,
- all institutions in $\mathcal{L}$ are quasi-semi-exact,
- from each institution in $\mathcal{I}$, there is some model-expansive comorphism in $\mathcal{I}$ going into some institution in $\mathcal{L}$,
- there is some set $\mathcal{D}$ of index morphisms in Ind such that such that index morphisms complementing $d \in \mathcal{D}$ is weak amalgamability squares are mapped to model-expansive comorphisms,

then the proof calculus complete for those heterogeneous development graphs that use hiding links are only with signature morphisms whose comorphism component is in $\mathcal{D}$.

The Heterogeneous Tool Set

Architecture of the heterogeneous tool set Hets



General Design of HETS

- per institution: signature category, sentence functor, parser, static analyser, (optional) prover
- per comorphism: signature and sentence translation
- object-orientierted interface for institutions
  (via multiparameter type classes in Haskell)
- separation of heterogeneous level and logic-specific instances of the interface
- 60.000 lines Haskell (about the half is logic independent)
- heterogeneous proof management

Signature Categories in Haskell

```
class Category lid sign morphism
                | lid -> sign, lid -> morphism where
      ide :: lid -> sign -> morphism
      comp :: lid -> morphism -> morphism -> Maybe morphism
      dom, cod :: lid -> morphism -> sign
```

Instances have to provide types `lid`, `sign`, `morphism` and operations `ide`, `comp`, `dom`, `cod`

Further Type Classes

- `Sentences` Sentences, sentence translation
- `Syntax` Abstract syntax for basic specs, parser
- `StaticAnalysis` Static analysis for basic specs, infrastructure of institution with qualified symbols
- `Logic` Sublogic analysis, theorem provers, consistency checkers

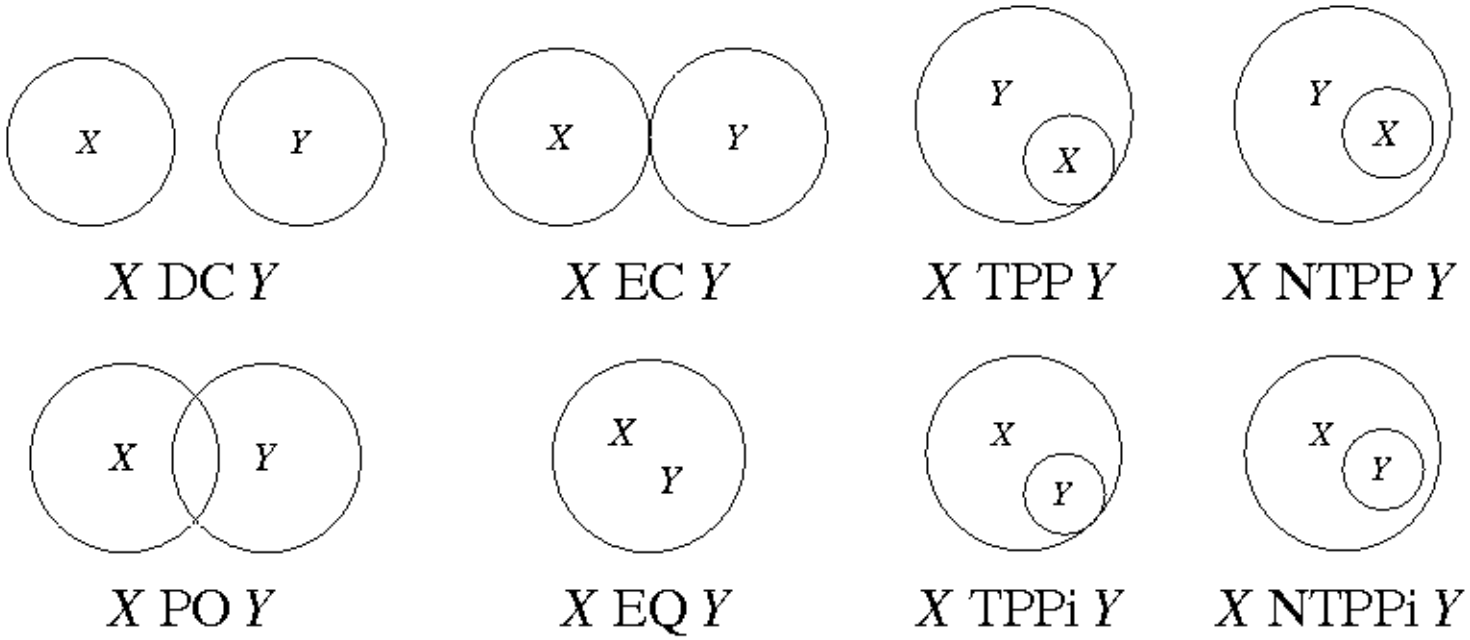
From Genericity to Heterogeneity

- SML functors achieve genericity over an arbitrary logic
- Heterogeneity needs to handle several logics at once
- Solution: Haskell existential types over type classes

```
-- Grothendieck signatures
data G_sign = forall lid sign morphism ...
              . Logic lid sign morphism ...
              => G_sign lid sign
```

A sample heterogeneous proof

A verification example: the Region Connection Calculus



The RCC8 Composition Table

Table 1: The composition table of RCC-8

∘	DC	EC	PO	TPP	NTPP	TPPi	NTPPi
DC	1	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC	DC
EC	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPP, TPPi, EQ	DC, EC, PO, TPP, NTPP	EC, PO, TPP, NTPP	PO, TPP, NTPP	, DC, EC	DC
PO	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPPi, NTPPi	1	PO, TPP, NTPP	PO, TPP, NTPP	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPPi, NTPPi
TPP	DC	DC, EC	DC, EC, PO, TPP, NTPP	TPP, NTPP	NTPP	DC, EC, PO, TPP, TPPi, EQ	DC, EC, PO, TPPi, NTPPi
NTPP	DC	DC	DC, EC, PO, TPP, NTPP	NTPP	NTPP	DC, EC, PO, TPP, NTPP	1
TPPi	DC, EC, PO, TPPi, NTPPi	EC, PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPP, TPPi, EQ	PO, TPP, NTPP	TPPi, NTPPi	NTPPi
NTPPi	DC, EC, PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPP, TPPi, NTPP, NTPPi, EQ	NTPPi	NTPPi

Verification of the RCC8 Composition Table

Verification goal: closed discs in a metric space satisfy the axioms of the region connection calculus (RCC).
This goal can be split into two subgoals:

1. Verification that closed discs in a metric satisfy Bennett's connectedness axioms (a few higher-order theorems, proved using the interactive theorem prover Isabelle).

2. Verification that Bennett's connectedness axioms imply the standard RCC axioms (many first-order theorems, proved using the automated theorem prover SPASS).

The Heterogeneous Integration Framework Initiative (HiFi)

- Aims at integration of existing tools
- Formal interoperability among different approaches
- Combination of high-level and low-level specifications
- Viewpoint specifications (like in UML, but formal)
- HETS could help with this integration
- Open, collaborative effort, based on free software
- `http://www.informatik.uni-bremen.de/hifi/`