Comorphisms of Structured Institutions

Ionuţ Ţuţu[a,b]

[a]Department of Computer Science, Royal Holloway University of London
[b]Institute of Mathematics of the Romanian Academy, Research group of the project ID-3-0439

Abstract

In this paper we formalise the intuition of encoding an institution of structured specifications into another one by extending the concept of institution comorphism to the abstract framework of structured institutions. This allows us to define a category of comorphisms of structured institutions, and supports in this way the development of heterogeneous specification languages in which the actual structuring mechanisms may vary, in addition to the base logical systems. We consider a number of properties with practical relevance for the comorphisms between base institutions and discuss their implications in the structured setting.

Keywords: Formal methods, Specification languages, Structured specifications, Institution theory

1. Introduction

Over the last few decades areas of theoretical computing science such as formal specification have led to a rising number of logics that provide the underlying foundations of various specification languages. The fact that different aspects of the specified systems are most accurately captured through a combination of different logics has encouraged the development of heterogeneous specification languages built on top of fixed graphs of logics and translations between logics. Two of the most prominent specification languages of this type are CafeOBJ [9] and HetCasl [20]. Each of them considers a particular structuring mechanism that is defined over an arbitrary logic formalised as an institution [12]. While the underlying logics of specifications can change through logic translations (formalised as maps between institutions: institution morphisms in the case of CafeOBJ,[1] and institution comorphisms in the case of HetCasl), the structuring mechanism is fixed. For this reason, both CafeOBJ and HetCasl can be regarded as specification languages that rely on a lower-level institution-independent theory of structuring specifications.

Following recent developments on the axiomatisation of structured specifications [7], we advance in this paper a concept of translation between logics of structured specifications presented formally as a comorphism of structured institutions. This could set the foundations for heterogeneous specification languages with an additional upper level of institution independence in which one can change not only the underlying logic of specifications, but also the considered structuring mechanism.

The paper is organised as follows. In the first sections we recall the notions of institution and structured institution together with a couple of examples. Section 4 introduces the main concept that supports the additional level of institution independence for heterogeneous structured specifications, and details the connection between the comorphisms of base institutions (which are given as components of comorphisms of structured institutions) and the induced comorphisms of institutions of structured specifications. Lastly, we investigate a series of properties of institution comorphisms that can be lifted from the lower level of the underlying logics to the upper level of structured specifications.

2. Preliminaries

The theory of institutions, introduced by Goguen and Burstall [12], promotes a universal model-theoretic approach to the study of logics by abstracting the notion of truth, which is supposed to be invariant with respect to the change of notation. Since institutions are formalised using category theory, we assume that the reader is familiar with basic notions of category theory [16], as adopted in recent

Email address: ittutu@gmail.com (Ionuţ Ţuţu)

[1]Based on ideas from [19, 5], in more recent papers such as [6] the institution-theoretic semantics of CafeOBJ has been equivalently described in terms of institution comorphisms.

works on institutions [5] and algebraic specifications [24]. In what follows we recall the concept of institution, which formalises the intuitive notion of logical system, including syntax, semantics and the satisfaction between them.

Definition 2.1. An *institution* $\mathcal{I}$ consists of

- a category $\mathbb{S}\mathrm{ig}^{\mathcal{I}}$ whose objects are called *signatures*,
- a *sentence functor* $\mathsf{Sen}^{\mathcal{I}}\colon \mathbb{S}\mathrm{ig}^{\mathcal{I}} \to \mathbb{S}\mathrm{et}$ defining for every signature Σ the set $\mathsf{Sen}^{\mathcal{I}}(\Sigma)$ of Σ-*sentences* and for every signature morphism φ the *sentence translation map* $\mathsf{Sen}^{\mathcal{I}}(\varphi)$,
- a *model functor* $\mathsf{Mod}^{\mathcal{I}}\colon (\mathbb{S}\mathrm{ig}^{\mathcal{I}})^{op} \to \mathbb{C}\mathrm{at}$ defining for each signature Σ the category $\mathsf{Mod}^{\mathcal{I}}(\Sigma)$ of Σ-*models* and Σ-*model homomorphisms*, and for each signature morphism φ the *reduct functor* $\mathsf{Mod}^{\mathcal{I}}(\varphi)$,
- for every signature Σ, a binary Σ-*satisfaction relation* $\models^{\mathcal{I}}_{\Sigma} \subseteq |\mathsf{Mod}^{\mathcal{I}}(\Sigma)| \times \mathsf{Sen}^{\mathcal{I}}(\Sigma)$,

such that the *satisfaction condition*

$$M' \models^{\mathcal{I}}_{\Sigma'} \mathsf{Sen}^{\mathcal{I}}(\varphi)(\rho) \quad \text{iff} \quad \mathsf{Mod}^{\mathcal{I}}(\varphi)(M') \models^{\mathcal{I}}_{\Sigma} \rho$$

holds for any signature morphism $\varphi\colon \Sigma \to \Sigma'$, any Σ'-model M' and any Σ-sentence ρ.

We may omit the superscripts or subscripts from the notations of the components of institutions when there is no risk of ambiguity. For example, if the considered institution and signature are clear, we may denote $\models^{\mathcal{I}}_{\Sigma}$ by $\models$. We may also denote the sentence translation $\mathsf{Sen}^{\mathcal{I}}(\varphi)$ by $\varphi(\_)$ and the reduct functor $\mathsf{Mod}^{\mathcal{I}}(\varphi)$ by $\_\!\upharpoonright_{\varphi}$. For $M = M'\!\upharpoonright_{\varphi}$, we say that M is the φ-*reduct* of M' and that M' is a φ-*expansion* of M.

There are numerous examples of logics formalised as institutions, used as foundation for both specification and programming languages. One of the most widely used is (many-sorted) first-order logic, which was first presented as an institution in [12]. We recall here from [5] the equational fragment of first-order logic obtained by discarding the relation symbols.

Example 2.2 (First-Order Equational Logic – FOEQL)**.** A *first-order equational signature* (S, F) consists of a set S of *sorts* and a family $F = \{F_{w\to s} \mid w \in S^*, s \in S\}$ of sets of *operation symbols* indexed by *arities* and *sorts*.

Signature morphisms $\varphi\colon (S, F) \to (S', F')$ reflect the structure of signatures. They are defined by functions $\varphi^{st}\colon S \to S'$ between the sets of sorts, and families of functions $\{\varphi^{op}_{w\to s}\colon F_{w\to s} \to F'_{\varphi^{st}(w)\to\varphi^{st}(s)} \mid w \in S^*, s \in S\}$ between the sets of operation symbols.

For every signature (S, F), a *model* M interprets each sort $s \in S$ as a set M_s, and each operation symbol $\sigma \in F_{w\to s}$ as a function $M_\sigma\colon M_w \to M_s$, where $M_w = M_{s_1} \times \cdots \times M_{s_n}$ for $w = s_1 \cdots s_n$. *Homomorphisms* $h\colon M \to N$ are families of functions $\{h_s\colon M_s \to N_s \mid s \in S\}$ such that $h_s(M_\sigma(m)) = N_\sigma(h_w(m))$ for all $\sigma \in F_{w\to s}$ and $m \in M_w$, where $h_w\colon M_w \to N_w$ denotes the canonical extension of h to w-tuples.

With respect to *model reducts*, for a morphism of signatures $\varphi\colon (S, F) \to (S', F')$ and an (S', F')-model M', $M'\!\upharpoonright_{\varphi}$ is defined as the (S, F)-model M given by $M_x = M'_{\varphi(x)}$ for every sort or operation symbol x of (S, F).

The *sentences* are usual first-order sentences built over equational atoms as follows. For every signature (S, F) and sort $s \in S$, the set $T_{F,s}$ of F-*terms* of sort s is the least set such that $\sigma(t) \in T_{F,s}$ for all $\sigma \in F_{w\to s}$ and all tuples $t \in T_{F,w}$, where $T_{F,w} = T_{F,s_1} \times \cdots \times T_{F,s_n}$ for $w = s_1 \cdots s_n$. Then the set of (S, F)-sentences is the smallest set containing the *equational atoms* $t =_s t'$ (for $s \in S$ and $t, t' \in T_{F,s}$) that is closed under Boolean connectives and quantification over sets of first-order variables, which are triples $(x, s, F_{\varepsilon\to s})$,[2] often denoted $x\colon s$, where x is the name of the variable (distinct from the names of other variables) and $s \in S$ is its sort.

The *translation of sentences* along a signature morphism $\varphi\colon (S, F) \to (S', F')$ is defined inductively on the structure of (S, F)-sentences and naturally renames the sorts and operation symbols of (S, F) according to φ. Note that on variables φ maps $(x, s, F_{\varepsilon\to s})$ to $\left(x, \varphi^{st}(s), F'_{\varepsilon\to\varphi^{st}(s)}\right)$.

The *satisfaction* between models and sentences is the usual Tarskian satisfaction defined inductively on the structure of sentences and based on the valuation of terms in models. For instance, given an (S, F)-model M and a universally quantified (S, F)-sentence $(\forall X)\rho$, it holds that $M \models_{(S,F)} (\forall X)\rho$ if and only if $M' \models_{(S,F\cup X)} \rho$ for all expansions M' of M along the inclusion $(S, F) \subseteq (S, F \cup X)$, where $(S, F \cup X)$ denotes the extension of (S, F) with the elements of X as new symbols of constants.

Building on the work of Russell on mathematical logic and type theory [22], higher-order logic with Henkin semantics has been developed in [2, 14], and later integrated into the framework of algebraic specifications in [18]. As in [18] and in more recent institution-theoretic works such as [24, 3] we consider here a simplified version of higher-order logic that only takes into account λ-free terms; this does not limit the expressive power since for any term $\lambda(x\colon s).\,t$ we can consider a new constant σ and a uni-

[2] ε denotes the empty sequence.

versal sentence of the form $(\forall x\colon s)\,\sigma\,x = t$.[3]

Example 2.3 (Higher-Order Logic with Henkin Semantics – HNK)**.** A *higher-order signature* (S, F) consists of a set S of *basic types* or *sorts* and a family $F = \{F_s \mid s \in \overrightarrow{S}\}$ of sets of *constant (operation) symbols*, indexed by *S-types* $s \in \overrightarrow{S}$, where $\overrightarrow{S}$ is the least set such that $S \subseteq \overrightarrow{S}$ and $s_1 \to s_2 \in \overrightarrow{S}$ whenever $s_1, s_2 \in \overrightarrow{S}$.

A *signature morphism* $\varphi\colon (S, F) \to (S', F')$ comprises functions $\varphi^{st}\colon S \to \overrightarrow{S'}$ and $\{\varphi^{op}_s\colon F_s \to F'_{\varphi^{type}(s)} \mid s \in \overrightarrow{S}\}$, where $\varphi^{type}\colon \overrightarrow{S} \to \overrightarrow{S'}$ is the canonical extension of φ^{st} given by $\varphi^{type}(s_1 \to s_2) = \varphi^{type}(s_1) \to \varphi^{type}(s_2)$.

The *models* M of a higher-order signature (S, F) interpret the types $s \in \overrightarrow{S}$ as sets M_s, the constant symbols $\sigma \in F_s$ as elements $M_\sigma \in M_s$, and define injective maps $[\![\_]\!]^M_{s_1 \to s_2}\colon M_{s_1 \to s_2} \to [M_{s_1} \to M_{s_2}]$, where $[M_{s_1} \to M_{s_2}]$ denotes the set of functions from M_{s_1} to M_{s_2}, for types $s_1, s_2 \in \overrightarrow{S}$. *Model homomorphisms* $h\colon M \to N$ are families of functions $\{h_s\colon M_s \to N_s \mid s \in \overrightarrow{S}\}$ such that $h_s(M_\sigma) = N_\sigma$ for any $s \in \overrightarrow{S}$ and $\sigma \in F_s$, and the following diagram commutes, for all $s_1, s_2 \in \overrightarrow{S}$ and $f \in M_{s_1 \to s_2}$.

$$\begin{array}{ccc} M_{s_1} & \xrightarrow{\;[\![f]\!]^M_{s_1 \to s_2}\;} & M_{s_2} \\ {\scriptstyle h_{s_1}}\big\downarrow & & \big\downarrow{\scriptstyle h_{s_2}} \\ N_{s_1} & \xrightarrow[\;[\![h_{s_1 \to s_2}(f)]\!]^N_{s_1 \to s_2}\;]{} & N_{s_2} \end{array}$$

The *model reducts* are defined likewise to FOEQL. For any signature morphism $\varphi\colon (S, F) \to (S', F')$ and any (S', F')-model M', the reduct $M'{\upharpoonright}_\varphi$ is the (S, F)-model M given by $M_x = M'_{\varphi(x)}$ for any type or constant operation symbol x of (S, F), and $[\![\_]\!]^M_{s_1 \to s_2} = [\![\_]\!]^{M'}_{\varphi^{type}(s_1) \to \varphi^{type}(s_2)}$ for any types s_1 and s_2 of (S, F).

Given a signature (S, F), the family $\{T_{F,s} \mid s \in \overrightarrow{S}\}$ of *F-terms* is the least family of sets such that $F_s \subseteq T_{F,s}$ for all $s \in \overrightarrow{S}$, and $(t\,t_1) \in T_{F,s_2}$ for all terms $t \in T_{F,s_1 \to s_2}$ and $t_1 \in T_{F,s_1}$. The *sentences* over (S, F) are built from *equational atoms* $t =_s t'$ (where $s \in \overrightarrow{S}$ and $t, t' \in T_{F,s}$), by repeated applications of Boolean connectives and quantification over sets of higher-order variables. Their *translation* along signature morphisms is also defined in an inductive manner by replacing every type and every constant symbol by their images under the considered morphism.

[3]Note that the presence of λ-terms imposes the existence of certain functions in Henkin models, which can generally be expressed in the λ-free setting only through infinitely many universal sentences of the form $(\forall x\colon s)\,\sigma\,x = t$.

Similarly to FOEQL, the definition of the *satisfaction relation* relies on the interpretation of terms in models, which extends the interpretation of constant symbols as follows: $M_{(t\,t_1)} = [\![M_t]\!]^M_{s_1 \to s_2}(M_{t_1})$, for any (S, F)-model M and any two terms $t \in T_{F,s_1 \to s_2}$ and $t_1 \in T_{F,s_1}$.

3. Structured institutions

The concept of structured institution was introduced in [7] in the context of a uniform theory of structured specifications that is independent of any particular choice of structuring operators, one that unifies the main institution-independent approaches to structuring specifications: the property-oriented approach of Goguen and Burstall [12, 10], and the model-oriented approach of Sannella and Tarlecki [23, 24]. This novel axiomatic framework of structured specifications considers an additional level of institution independence that corresponds to the model theory of structured specifications and is related to the underlying logic through a particular type of institution morphism [12].

Definition 3.1 (Structured institution)**.** We say that an institution $\mathcal{I}' = (\mathbb{S}\mathrm{ig}', \mathsf{Sen}', \mathsf{Mod}', \models')$ is *structured* over a base institution $\mathcal{I} = (\mathbb{S}\mathrm{ig}, \mathsf{Sen}, \mathsf{Mod}, \models)$ through a *structuring functor* $\Phi\colon \mathbb{S}\mathrm{ig}' \to \mathbb{S}\mathrm{ig}$ if

– $\mathsf{Sen}'(\Sigma') = \mathsf{Sen}(\Phi(\Sigma'))$, for every $\mathcal{I}'$-signature Σ', and $\mathsf{Sen}'(\varphi') = \mathsf{Sen}(\Phi(\varphi'))$, for every $\mathcal{I}'$-signature morphism φ',

– $\mathsf{Mod}'(\Sigma') \subseteq \mathsf{Mod}(\Phi(\Sigma'))$ is a full subcategory inclusion, for every $\mathcal{I}'$-signature Σ', and the diagram bellow commutes, for every $\mathcal{I}'$-signature morphism $\varphi'\colon \Sigma'_1 \to \Sigma'_2$, and

$$\begin{array}{ccc} \mathsf{Mod}'(\Sigma'_2) & \xrightarrow{\;\subseteq\;} & \mathsf{Mod}(\Phi(\Sigma'_2)) \\ {\scriptstyle \_{\upharpoonright}_{\varphi'}}\big\downarrow & & \big\downarrow{\scriptstyle \_{\upharpoonright}_{\Phi(\varphi')}} \\ \mathsf{Mod}'(\Sigma'_1) & \xrightarrow[\;\subseteq\;]{} & \mathsf{Mod}(\Phi(\Sigma'_1)) \end{array}$$

– for every $\mathcal{I}'$-signature Σ', every Σ'-model M' and every Σ'-sentence ρ',

$$M' \models'_{\Sigma'} \rho' \quad \text{iff} \quad M' \models_{\Phi(\Sigma')} \rho'.$$

We usually denote structured institutions $\mathcal{I}'$ over $\mathcal{I}$ through Φ as triples $(\mathcal{I}', \mathcal{I}, \Phi)$. When the base institution $\mathcal{I}$ and the structuring functor Φ can be easily inferred we may choose to denote the structured institution simply by $\mathcal{I}'$.

Closure under isomorphisms. We assume that all structured institutions $\mathcal{I}'$ over $\mathcal{I}$ through Φ are *closed under isomorphisms*, i.e. for any $\mathcal{I}'$-signature Σ' and any two isomorphic $\Phi(\Sigma')$-models M and N, M is a Σ'-model if and only if N is a Σ'-model.

Example 3.2 (Presentations)**.** The *presentations* [12] of an institution $\mathcal{I} = (\mathbb{S}\mathrm{ig}, \mathsf{Sen}, \mathsf{Mod}, \models)$ are pairs (Σ, E) consisting of signatures Σ and sets of Σ-sentences E. They form a category $\mathbb{P}\mathrm{res}$ whose arrows $\varphi\colon (\Sigma, E) \to (\Sigma', E')$ are signature morphisms $\varphi\colon \Sigma \to \Sigma'$ such that $E' \models \varphi(E)$.

The institution $\mathcal{I}^{pres} = (\mathbb{P}\mathrm{res}, \mathsf{Sen}^{pres}, \mathsf{Mod}^{pres}, \models^{pres})$ of presentations over $\mathcal{I}$ is obtained by extending the sentence functor, the model functor and the satisfaction relation from base signatures to presentations. $\mathcal{I}^{pres}$ is structured over $\mathcal{I}$ through the functor $\mathsf{Sig}\colon \mathbb{P}\mathrm{res} \to \mathbb{S}\mathrm{ig}$ that maps every presentation (Σ, E) to Σ.

Example 3.3 (Structured specifications)**.** Given an institution $\mathcal{I} = (\mathbb{S}\mathrm{ig}, \mathsf{Sen}, \mathsf{Mod}, \models)$ and a class of signature morphisms $\mathcal{T}$, the $\mathcal{T}$-structured specifications [1] of $\mathcal{I}$ are obtained through the specification building operators listed below. The semantics of every structured specification *SP* is described by its signature $\mathrm{Sig}[SP]$ and its class of models $\mathrm{Mod}[SP]$.

PRES. Any *finite presentation* (Σ, E) is a structured specification such that

$$\mathrm{Sig}[(\Sigma, E)] = \Sigma,$$
$$\mathrm{Mod}[(\Sigma, E)] = \{M \in |\mathsf{Mod}(\Sigma)| \mid M \models E\}.$$

UNION. For any two specifications SP_1 and SP_2 with the same signature Σ, their *union* $SP_1 \cup SP_2$ is also a structured specification, with

$$\mathrm{Sig}[SP_1 \cup SP_2] = \Sigma,$$
$$\mathrm{Mod}[SP_1 \cup SP_2] = \mathrm{Mod}[SP_1] \cap \mathrm{Mod}[SP_2].$$

TRANS. For any specification *SP* and any signature morphism $\varphi\colon \mathrm{Sig}[SP] \to \Sigma$ in $\mathcal{T}$, the *φ-translation* of *SP*, denoted $SP \star \varphi$, is a structured specification having

$$\mathrm{Sig}[SP \star \varphi] = \Sigma,$$
$$\mathrm{Mod}[SP \star \varphi] = \{M \in |\mathsf{Mod}(\Sigma)| \mid M\restriction_{\varphi} \in \mathrm{Mod}[SP]\}.$$

The $\mathcal{T}$-structured specifications over $\mathcal{I}$ form a category $\mathbb{S}\mathrm{pec}$ [8, 24] whose arrows $\varphi\colon SP \to SP'$ are signature morphisms $\varphi\colon \mathrm{Sig}[SP] \to \mathrm{Sig}[SP']$ such that for every model $M' \in \mathrm{Mod}[SP']$ it holds that $M'\restriction_{\varphi} \in \mathrm{Mod}[SP]$.

Similarly to the case of presentations, the sentence functor, the model functor and the satisfaction relation can be naturally extended from signatures to structured specifications. We obtain in this way the institution $\mathcal{I}^{spec} = (\mathbb{S}\mathrm{pec}, \mathsf{Sen}^{spec}, \mathsf{Mod}^{spec}, \models^{spec})$ of $\mathcal{T}$-structured specifications over $\mathcal{I}$, which is structured over $\mathcal{I}$ through the functor $\mathsf{Sig}\colon \mathbb{S}\mathrm{pec} \to \mathbb{S}\mathrm{ig}$ that maps every structured specification *SP* to $\mathrm{Sig}[SP]$.

4. Moving between structured institutions

Institution comorphisms capture the intuitive notion of embedding simpler logical systems into more complex ones. They were originally discussed in [17] under the name of plain maps, and in [25] under the name of institution representations.

Definition 4.1 (Comorphism of institutions)**.** Given two institutions $\mathcal{I}_i = (\mathbb{S}\mathrm{ig}_i, \mathsf{Sen}_i, \mathsf{Mod}_i, \models_i)$, with $i \in \{1, 2\}$, an *institution comorphism* $(\Psi, \alpha, \beta)\colon \mathcal{I}_1 \to \mathcal{I}_2$ consists of

- a *signature functor* $\Psi\colon \mathbb{S}\mathrm{ig}_1 \to \mathbb{S}\mathrm{ig}_2$,
- a natural transformation $\alpha\colon \mathsf{Sen}_1 \Rightarrow \Psi\,;\mathsf{Sen}_2$, and
- a natural transformation $\beta\colon \Psi^{op}\,;\mathsf{Mod}_2 \Rightarrow \mathsf{Mod}_1$

such that the following *satisfaction condition* holds for any $\mathcal{I}_1$-signature Σ, $\Psi(\Sigma)$-model M and Σ-sentence ρ.

$$M\ (\models_2)_{\Psi(\Sigma)}\ \alpha_\Sigma(\rho) \quad \text{iff} \quad \beta_\Sigma(M)\ (\models_1)_\Sigma\ \rho$$

It is often the case that the codomain of an institution comorphism is the institution of presentations of a logical system that is simpler than the one given by the domain of the comorphism. Such encodings $\mathcal{I} \to \mathcal{I}'^{pres}$ are called simple theoroidal comorphisms in [13].

The following encoding of higher-order logic into first-order logic has been first outlined in [18]. More detailed presentations can be found in [5, 3].

Example 4.2. There exists a comorphism of institutions $(\Psi, \alpha, \beta)\colon \underline{\mathrm{HNK}} \to \underline{\mathrm{FOEQL}}^{pres}$ that maps

- every HNK-signature (S, F) to the FOEQL-presentation $\Psi(S, F) = ((\overrightarrow{S}, \overrightarrow{F}), E_{(S,F)})$, where $\overrightarrow{S}$ is the (infinite) set of S-types, $\overrightarrow{F}$ is the family of sets of operation symbols given by

$$\overrightarrow{F}_{w \to s} = \begin{cases} F_s & \text{if } w \text{ is empty,} \\ \{\mathrm{apply}_w\} & \text{if } w = (s_1 \to s)s_1 \\ & \quad \text{for some } s_1 \in \overrightarrow{S}, \\ \emptyset & \text{otherwise,} \end{cases}$$

and $E_{(S,F)}$ is the set of $(\overrightarrow{S}, \overrightarrow{F})$-sentences of the form

$$(\forall f, g\colon s_1 \to s_2)$$
$$((\forall x\colon s_1)\ \mathrm{apply}(f, x) = \mathrm{apply}(g, x)) \to f = g,$$

– every $((\overrightarrow{S}, \overrightarrow{F}), E_{(S,F)})$-model $\overrightarrow{M}$ to the (S, F)-model $\beta_{(S,F)}(\overrightarrow{M}) = M$ defined by $M_x = \overrightarrow{M}_x$ for every type or constant symbol x of (S, F), and

$$[\![f]\!]^{M}_{s_1 \to s_2}(m) = \overrightarrow{M}_{\mathrm{apply}}(f, m)$$

for every $s_1, s_2 \in \overrightarrow{S}$, $f \in M_{s_1 \to s_2}$ and $m \in M_{s_1}$,

– every sentence ρ of (S, F) to the sentence $\alpha_{(S,F)}(\rho)$ of $(\overrightarrow{S}, \overrightarrow{F})$ through the canonical extension of the translations of terms $\{\alpha^{tm}_s\colon T_{F,s} \to T_{\overrightarrow{F},s} \mid s \in \overrightarrow{S}\}$ given by $\alpha^{tm}_s(\sigma) = \sigma$ for $s \in \overrightarrow{S}$ and $\sigma \in F_s$, and $\alpha^{tm}_{s_2}(t\,t_1) = \mathrm{apply}(\alpha^{tm}_{s_1 \to s_2}(t), \alpha^{tm}_{s_1}(t_1))$ for $s_1, s_2 \in \overrightarrow{S}$, $t \in T_{F,s_1 \to s_2}$ and $t_1 \in T_{F,s_1}$.

Definition 4.3 (Comorphism of structured institutions)**.** Let $\mathcal{I}'_i = (\mathbb{S}\mathrm{ig}'_i, \mathsf{Sen}'_i, \mathsf{Mod}'_i, \models'_i)$ be a structured institution over $\mathcal{I}_i = (\mathbb{S}\mathrm{ig}_i, \mathsf{Sen}_i, \mathsf{Mod}_i, \models_i)$ through Φ_i, for $i \in \{1, 2\}$. A *comorphism of structured institutions*, or *structured comorphism*, $(\mathcal{I}'_1, \mathcal{I}_1, \Phi_1) \to (\mathcal{I}'_2, \mathcal{I}_2, \Phi_2)$ consists of

– a functor $\Psi'\colon \mathbb{S}\mathrm{ig}'_1 \to \mathbb{S}\mathrm{ig}'_2$, and

– a comorphism $(\Psi, \alpha, \beta)\colon \mathcal{I}_1 \to \mathcal{I}_2$ between the base institutions $\mathcal{I}_1$ and $\mathcal{I}_2$

such that the diagram below is commutative,

$$\begin{array}{ccc} \mathbb{S}\mathrm{ig}'_1 & \xrightarrow{\Psi'} & \mathbb{S}\mathrm{ig}'_2 \\ {\scriptstyle\Phi_1}\downarrow & & \downarrow{\scriptstyle\Phi_2} \\ \mathbb{S}\mathrm{ig}_1 & \xrightarrow[\Psi]{} & \mathbb{S}\mathrm{ig}_2 \end{array}$$

and the following condition holds for every $\mathcal{I}'_1$-signature Σ' and every $\Phi_2(\Psi'(\Sigma'))$-model M.

$$M \in |\mathsf{Mod}'_2(\Psi'(\Sigma'))| \quad \text{iff} \quad \beta_{\Phi_1(\Sigma')}(M) \in |\mathsf{Mod}'_1(\Sigma')|$$

Example 4.4 (Representing structured specifications over HNK as presentations over FOEQL^{pres})**.** Let HNK^{spec} be the institution of $\mathcal{T}$-structured specifications over HNK (as described in Example 3.3), and FOEQL$^{2\text{-}pres}$ the institution of presentations over FOEQL^{pres} (as described in Example 3.2). The comorphism (Ψ, α, β) considered in Example 4.2 can be extended to a comorphism of structured institutions $(\Psi', \Psi, \alpha, \beta)\colon$ HNK^{spec} $\to$ FOEQL$^{2\text{-}pres}$ by defining, for every HNK-specification SP,

$$\Psi'(SP) = (\Psi(\mathrm{Sig}[SP]), \alpha_{\mathrm{Sig}[SP]}(\mathrm{Ax}[SP]))$$

where the set $\mathrm{Ax}[SP]$ of axioms of SP is given by

– $\mathrm{Ax}[(\Sigma, E)] = E$ for every HNK-presentation (Σ, E),

– $\mathrm{Ax}[SP_1 \cup SP_2] = \mathrm{Ax}[SP_1] \cup \mathrm{Ax}[SP_2]$ for every pair of HNK-specifications SP_1, SP_2 with the same signature, and

– $\mathrm{Ax}[SP \star \varphi] = \varphi(\mathrm{Ax}[SP])$ for every HNK-specification SP and every morphism $\varphi\colon \mathrm{Sig}[SP] \to \Sigma$.

This extension relies on the essential property that every specification SP over a given institution $\mathcal{I}$ is semantically equivalent with the $\mathcal{I}$-presentation $(\mathrm{Sig}[SP], \mathrm{Ax}[SP])$ [8]. In fact, the above comorphism of structured institutions can be factored as the composition of two fundamentally institution-independent constructions:

1. the flattening structured comorphism that maps every structured specification over HNK to a semantically equivalent presentation, and
2. the canonical extension of (Ψ, α, β) to a structured comorphism between the institutions of presentations built over HNK and FOEQL^{pres}.

Proposition 4.5. *The comorphisms of structured institutions can be composed in a natural way in terms of their components. The composition is associative and has left and right identities given by identity comorphisms for the base institutions and identity functors for the categories of structured specifications. Therefore, the comorphisms of structured institutions form a category* co$\mathbb{S}$truc$\mathbb{I}$ns.

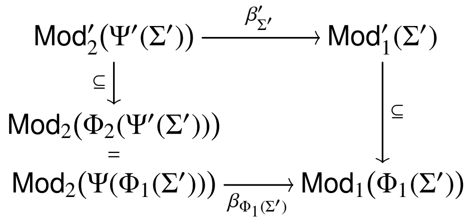
Proposition 4.6. *Under the notations and hypotheses of Definition 4.3, any comorphism of structured institutions* $(\Psi', \Psi, \alpha, \beta)\colon (\mathcal{I}'_1, \mathcal{I}_1, \Phi_1) \to (\mathcal{I}'_2, \mathcal{I}_2, \Phi_2)$ *determines a comorphism of institutions* $(\Psi', \alpha', \beta')\colon \mathcal{I}'_1 \to \mathcal{I}'_2$ *with*

– $\alpha' = \Phi_1\alpha$*, and*

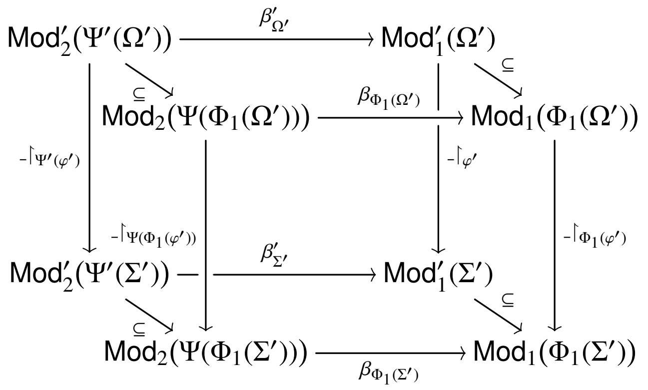
– $\beta'_{\Sigma'}(X') = \beta_{\Phi_1(\Sigma')}(X')$*, for every* $\mathcal{I}'_1$*-signature* Σ' *and every* Σ'*-model (or model homomorphism)* X'*.*

Proof. We first show that α' and β' are natural transformations from Sen'_1 to $\Psi' ; \mathsf{Sen}'_2$ and from $\Psi'^{op} ; \mathsf{Mod}'_2$ to Mod'_1, respectively. Since the naturality of α' follows immediately from its definition, we focus on β'.

Let us notice that by the definition of comorphisms of structured institutions, for every $\mathcal{I}'_1$-signature Σ' and every $\Psi'(\Sigma')$-model M', $\beta'_{\Sigma'}(M') = \beta_{\Phi_1(\Sigma')}(M')$ is a Σ'-model. Moreover, for any $\Psi'(\Sigma')$-model homomorphism $h' \colon M' \to N'$, $\beta'_{\Sigma'}(h') = \beta_{\Phi_1(\Sigma')}(h')$ is a $\Phi_1(\Sigma')$-model homomorphism $\beta'_{\Sigma'}(M') \to \beta'_{\Sigma'}(N')$. Because $\mathsf{Mod}'_1(\Sigma')$ is a full subcategory of $\mathsf{Mod}_1(\Phi_1(\Sigma'))$ (by the definition of structured institutions), it follows that $\beta'_{\Sigma'}(h')$ is a Σ'-model homomorphism. Therefore, the map $\beta'_{\Sigma'}$ is a functor $\mathsf{Mod}'_2(\Psi'(\Sigma')) \to \mathsf{Mod}'_1(\Sigma')$ such that the following diagram is commutative.



This ensures that the top and bottom squares in the cube diagram depicted below are commutative. In addition, by the definition of structured institutions, we know that the left and right squares are also commutative, for every $\mathcal{I}'_1$-signature morphism $\varphi' \colon \Sigma' \to \Omega'$. The naturality of β (in combination with the fact that the inclusion functor $\mathsf{Mod}'_1(\Sigma') \subseteq \mathsf{Mod}_1(\Phi_1(\Sigma'))$ is mono) completes the commutativity of the entire diagram, and thus also the proof of the naturality of β'.



In order to establish the satisfaction condition for the comorphism (Ψ', α', β'), let Σ' be an $\mathcal{I}'_1$-signature, M' a $\Psi'(\Sigma')$-model and ρ' a Σ'-sentence. We deduce that

$M' \ (\models'_2)_{\Psi'(\Sigma')} \ \alpha'_{\Sigma'}(\rho')$
iff $M' \ (\models'_2)_{\Psi'(\Sigma')} \ \alpha_{\Phi_1(\Sigma')}(\rho')$
(by the definition of α')
iff $M' \ (\models_2)_{\Phi_2(\Psi'(\Sigma'))} \ \alpha_{\Phi_1(\Sigma')}(\rho')$
(by the structuring of $\mathcal{I}'_2$)
iff $M' \ (\models_2)_{\Psi(\Phi_1(\Sigma'))} \ \alpha_{\Phi_1(\Sigma')}(\rho')$
(because $\Psi' \mathbin{;} \Phi_2 = \Phi_1 \mathbin{;} \Psi$)
iff $\beta_{\Phi_1(\Sigma')}(M') \ (\models_1)_{\Phi_1(\Sigma')} \ \rho'$
(by the satisfaction condition of (Ψ, α, β))
iff $\beta_{\Phi_1(\Sigma')}(M') \ (\models'_1)_{\Sigma'} \ \rho'$
(by the structuring of $\mathcal{I}'_1$)
iff $\beta'_{\Sigma'}(M') \ (\models'_1)_{\Sigma'} \ \rho'$
(by the definition of β').

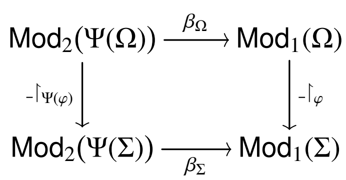
$\square$

The following properties of comorphisms allow the transfer of numerous features of institutions; for instance, conservativeness implies the reflection of semantic consequence, in addition to its preservation, which holds for any comorphism. They have been studied in a series of works on structured specifications [1], heterogeneity [4, 19], expressiveness of logical systems [15], interpolation [3] and definability [11].

Definition 4.7. A comorphism (Ψ, α, β) between institutions $\mathcal{I}_1$ and $\mathcal{I}_2$

is *conservative* or *model-expansive* if and only if β_Σ is surjective on models, for all $\mathcal{I}_1$-signatures Σ;

has *model amalgamation* if and only if for every $\mathcal{I}_1$-signature morphism $\varphi \colon \Sigma \to \Omega$, every Ω-model N_1 and every $\Psi(\Sigma)$-model M_2 such that $N_1{\upharpoonright_\varphi} = \beta_\Sigma(M_2)$, there exists a unique $\Psi(\Omega)$-model N_2, called the *amalgamation* of N_1 and M_2, such that $\beta_\Omega(N_2) = N_1$ and $N_2{\upharpoonright_{\Psi(\varphi)}} = M_2$; if the amalgamation N_2 is no longer required to be unique then (Ψ, α, β) is said to have *weak model amalgamation*;



is *persistently liberal* if and only if for every $\mathcal{I}_1$-signature Σ, the functor β_Σ admits a left adjoint such that the unit of the adjunction consists solely of isomorphisms.

Fact 4.8. Let $(\Psi, \alpha, \beta) \colon \underline{\mathrm{HNK}} \to \underline{\mathrm{FOEQL}}^{pres}$ be the comorphism defined in Example 4.2. For any $\underline{\mathrm{HNK}}$-signature (S, F) the model functor $\beta_{(S,F)}$ is an isomorphism

$$\mathsf{Mod}^{\underline{\mathrm{FOEQL}}^{pres}}(\Psi(S, F)) \cong \mathsf{Mod}^{\underline{\mathrm{HNK}}}(S, F).$$

As a result, the comorphism (Ψ, α, β) enjoys trivially all three properties considered in Definition 4.7.

Corollary 4.9. *Under the notations and hypotheses of Proposition 4.6, if the comorphism* (Ψ, α, β) *is conservative / has (weak) model amalgamation / is persistently liberal then* (Ψ', α', β') *has these properties as well.*

Proof. Conservativeness. Let Σ' be an $\mathcal{I}'_1$-signature and N' a Σ'-model. Since $\mathcal{I}'_1$ is structured over $\mathcal{I}_1$ through Φ_1 it follows that N' is also a $\Phi_1(\Sigma')$-model. By the conservativeness of (Ψ, α, β) we deduce that there exists a $\Psi(\Phi_1(\Sigma'))$-model M' such that $\beta_{\Phi_1(\Sigma')}(M') = N'$, and thus $\beta_{\Phi_1(\Sigma')}(M') \in |\mathsf{Mod}'_1(\Sigma')|$. Hence, by the definition of comorphisms of structured institutions, M' is a $\Psi'(\Sigma')$-model. Moreover, $\beta'_{\Sigma'}(M') = \beta_{\Phi_1(\Sigma')}(M') = N'$.

(Weak) Model amalgamation. We analyse the commutative cube of functors considered in the proof of Proposition 4.6. Let $\varphi' : \Sigma' \to \Omega'$ be a morphism of $\mathcal{I}'_1$-signatures, N'_1 an Ω'-model and M'_2 a $\Psi'(\Sigma')$-model such that $N'_1 \restriction_{\varphi'} = \beta'_{\Sigma'}(M'_2)$. Since $\mathcal{I}'_1$ and $\mathcal{I}'_2$ are structured over $\mathcal{I}_1$ and $\mathcal{I}_2$, respectively, we deduce that N'_1 is a $\Phi_1(\Omega')$-model and M'_2 is a $\Psi(\Phi_1(\Sigma'))$-model such that

$$N'_1 \restriction_{\Phi_1(\varphi')} = N'_1 \restriction_{\varphi'} = \beta'_{\Sigma'}(M'_2) = \beta_{\Phi_1(\Sigma')}(M'_2).$$

It follows that there exists a (unique) $\Psi(\Phi_1(\Omega'))$-model N_2 such that $\beta_{\Phi_1(\Omega')}(N_2) = N'_1$ and $N_2 \restriction_{\Psi(\Phi_1(\varphi'))} = M'_2$, because the comorphism (Ψ, α, β) has (weak) model amalgamation (by hypothesis). In addition, N_2 is a $\Psi'(\Omega')$-model because $(\Psi, \alpha, \beta, \Psi')$ is a comorphism of structured institutions and $\beta_{\Phi_1(\Omega')}(N_2) = N'_1 \in |\mathsf{Mod}'_1(\Omega')|$.

Liberality. For every $\mathcal{I}'_1$-signature Σ' we know by hypothesis that $\beta_{\Phi_1(\Sigma')}$ admits a left adjoint $\delta_{\Phi_1(\Sigma')}$ such that the unit component $\eta^{\Phi_1(\Sigma')}_M : M \to \beta_{\Phi_1(\Sigma')}(\delta_{\Phi_1(\Sigma')}(M))$ is an isomorphism, for every $\Phi_1(\Sigma')$-model M.

$$\begin{array}{ccc}
\mathsf{Mod}'_2(\Psi'(\Sigma')) & \xrightarrow{\beta'_{\Sigma'}} & \mathsf{Mod}'_1(\Sigma') \\
\subseteq \downarrow & & \downarrow \subseteq \\
\mathsf{Mod}_2(\Psi(\Phi_1(\Sigma'))) & \underset{\beta_{\Phi_1(\Sigma')}}{\overset{\delta_{\Phi_1(\Sigma')}}{\leftrightarrows}} & \mathsf{Mod}_1(\Phi_1(\Sigma'))
\end{array}$$

Let us prove first that $\delta_{\Phi_1(\Sigma')}$ can be restricted to a functor $\mathsf{Mod}'_1(\Sigma') \to \mathsf{Mod}'_2(\Psi'(\Sigma'))$, which is equivalent to the fact that $\delta_{\Phi_1(\Sigma')}(\mathsf{Mod}'_1(\Sigma'))$ is a subcategory of $\mathsf{Mod}'_2(\Psi'(\Sigma'))$. On objects, for every Σ'-model M', it holds that

$$\delta_{\Phi_1(\Sigma')}(M') \in |\mathsf{Mod}'_2(\Psi'(\Sigma'))| \quad \text{iff} \quad \beta_{\Phi_1(\Sigma')}(\delta_{\Phi_1(\Sigma')}(M')) \in |\mathsf{Mod}'_1(\Sigma')|$$

(by the definition of comorphisms of structured institutions). Since $M' \cong \beta_{\Phi_1(\Sigma')}(\delta_{\Phi_1(\Sigma')}(M'))$ (by hypothesis), and $M' \in |\mathsf{Mod}'_1(\Sigma')|$, we deduce by the closure of structured institutions under isomorphisms that the equivalent membership statements mentioned above are valid.
On arrows, for every Σ'-homomorphism $h' : M' \to N'$ we have that $\delta_{\Phi_1(\Sigma')}(h')$ is a $\Psi(\Phi_1(\Sigma'))$-homomorphism between the $\Psi'(\Sigma')$-models $\delta_{\Phi_1(\Sigma')}(M')$ and $\delta_{\Phi_1(\Sigma')}(N')$. It follows that $\delta_{\Phi_1(\Sigma')}(h')$ is also a $\Psi'(\Sigma')$-homomorphism because $\mathsf{Mod}'_2(\Psi'(\Sigma')) \subseteq \mathsf{Mod}_2(\Psi(\Phi_1(\Sigma')))$ is full.

One can easily see now that the map $\delta'_{\Sigma'}$ given by $\delta'_{\Sigma'}(X') = \delta_{\Phi_1(\Sigma')}(X')$, for every Σ'-model (or homomorphism) X', defines a functor $\mathsf{Mod}'_1(\Sigma') \to \mathsf{Mod}'_2(\Psi'(\Sigma'))$ that is a left adjoint for $\beta'_{\Sigma'}$. Furthermore, the components of the unit and of the counit of this adjunction are inherited from the adjunction

$$\delta_{\Phi_1(\Sigma')} : \mathsf{Mod}_1(\Phi_1(\Sigma')) \leftrightarrows \mathsf{Mod}_2(\Psi(\Phi_1(\Sigma'))) : \beta_{\Phi_1(\Sigma')}.$$

Hence, the comorphism (Ψ', α', β') is persistently liberal. □

5. Conclusions

We have introduced comorphisms of structured institutions by extending the well-known concept of comorphism of (plain) institutions, and we have formalised in this manner the embedding of simpler structuring mechanisms into more complex ones.

The proposed framework supports the development of heterogeneous specification languages with two levels of institution independence, for the underlying institutions and also for the structuring constructs. With respect to this, we have studied properties of comorphisms that are directly related to heterogeneity such as conservativeness, model amalgamation and persistent liberality.

One of the issues to be further pursued is the enhancement of the category $\mathrm{co}\mathbb{S}\mathrm{truc}\mathbb{I}\mathrm{ns}$ of comorphisms of structured institutions with a non-trivial 2-categorical structure, which could help refine the possible translations between structured institutions. This can be achieved by upgrading the concept of institution comorphism modification [4, 21] to our structured setting.

Acknowledgements

The author would like to thank José Fiadeiro for carefully reading an early version of this work, and Răzvan Diaconescu for helpful discussions about institution morphisms and comorphisms. This research has been supported by a grant of the Romanian National Authority for Scientific Research, CNCS-UEFISCDI, project number PN-II-ID-PCE-2011-3-0439.

References

[1] Tomasz Borzyszkowski. Logical systems for structured specifications. *Theoretical Computer Science*, 286(2):197–245, 2002.

[2] Alonzo Church. A formulation of the simple theory of types. *The Journal of Symbolic Logic*, 5(2):56–68, 1940.

[3] Razvan Diaconescu. Borrowing interpolation. *Journal of Logic and Computation*, 22(3):561–586, 2012.

[4] Răzvan Diaconescu. Grothendieck institutions. *Applied Categorical Structures*, 10(4):383–402, 2002.

[5] Răzvan Diaconescu. *Institution-independent model theory*. Studies in Universal Logic. Birkhäuser, 2008.

[6] Răzvan Diaconescu. Grothendieck inclusion systems. *Applied Categorical Structures*, 19(5):783–802, 2011.

[7] Răzvan Diaconescu. An axiomatic approach to structuring specifications. *Theoretical Computer Science*, 433:20–42, 2012.

[8] Răzvan Diaconescu and Ionuţ Ţuţu. On the algebra of structured specifications. *Theoretical Computer Science*, 412(28):3145–3174, 2011.

[9] Răzvan Diaconescu and Kokichi Futatsugi. *CafeOBJ report: the language, proof techniques, and methodologies for object-oriented algebraic specification*. AMAST series in computing. World Scientific, 1998.

[10] Răzvan Diaconescu, Joseph A. Goguen, and Petros Stefaneas. Logical support for modularisation. In Gerard Huet and Gordon Plotkin, editors, *Logical Environments*, pages 83–130. Cambridge University Press, 1993.

[11] Răzvan Diaconescu and Marius Petria. Abstract beth definability in institutions. *The Journal of Symbolic Logic*, 71(3):1002–1028, 2006.

[12] Joseph A. Goguen and Rod M. Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the ACM*, 39(1):95–146, 1992.

[13] Joseph A. Goguen and Grigore Roşu. Institution morphisms. *Formal Aspects of Computing*, 13(3–5):274–307, 2002.

[14] Leon Henkin. Completeness in the theory of types. *The Journal of Symbolic Logic*, 15(2):81–91, 1950.

[15] Hans-Jörg Kreowski and Till Mossakowski. Equivalence and difference between institutions: Simulating horn clause logic with based algebras. *Mathematical Structures in Computer Science*, 5(2):189–215, 1995.

[16] Saunders Mac Lane. *Categories for the working mathematician*. Graduate texts in mathematics. Springer, 1998.

[17] José Meseguer. General logics. In Heinz-Dieter Ebbinghaus, editor, *Logic Colloquium '87*, Studies in Logic and the Foundations of Mathematics Series, pages 275–329. North-Holland, 1989.

[18] Bernhard Möller, Andrzej Tarlecki, and Martin Wirsing. Algebraic specifications of reachable higher-order algebras. In Donald Sannella and Andrzej Tarlecki, editors, *Recent Trends in Data Type Specification*, volume 332 of *Lecture Notes in Computer Science*, pages 154–169. Springer, 1987.

[19] Till Mossakowski. Comorphism-based Grothendieck logics. In Krzysztof Diks and Wojciech Rytter, editors, *Mathematical Foundations of Computer Science 2002*, volume 2420 of *Lecture Notes in Computer Science*, pages 593–604. Springer, 2002.

[20] Till Mossakowski. HetCasl – heterogeneous specification. Language summary. Technical report, CoFI: The Common Framework Initiative, 2004.

[21] Till Mossakowski. Institutional 2-cells and Grothendieck institutions. In *Algebra, Meaning, and Computation, Essays Dedicated to Joseph A. Goguen*, volume 4060 of *Lecture Notes in Computer Science*, pages 124–149. Springer, 2006.

[22] Bertrand Russell. Mathematical logic as based on the theory of types. *American Journal of Mathematics*, 30(3):222–262, 1908.

[23] Donald Sannella and Andrzej Tarlecki. Specifications in an arbitrary institution. *Information and Computation*, 76(2/3):165–210, 1988.

[24] Donald Sannella and Andrzej Tarlecki. *Foundations of Algebraic Specification and Formal Software Development*. Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2011.

[25] Andrzej Tarlecki. Moving between logical systems. In Magne Haveraaen, Olaf Owe, and Ole-Johan Dahl, editors, *Recent Trends in Data Type Specification*, volume 1130 of *Lecture Notes in Computer Science*, pages 478–502. Springer, 1995.