Categorical Structures for Type Theory in Univalent Foundations[*]

Benedikt Ahrens[1], Peter LeFanu Lumsdaine[2], and Vladimir Voevodsky[3]

1   **Inria, Nantes, France**
     `benedikt.ahrens@inria.fr`
2   **Department of Mathematics, Stockholm University, Stockholm, Sweden**
     `p.l.lumsdaine@math.su.se`
3   **Institute for Advanced Study, Princeton, NJ, USA**
     `vladimir@ias.edu`

─── **Abstract** ───

In this paper, we analyze and compare three of the many algebraic structures that have been used for modeling dependent type theories: *categories with families*, *split type-categories*, and *representable maps of presheaves*. We study these in the setting of univalent foundations, where the relationships between them can be stated more transparently. Specifically, we construct maps between the different structures and show that these maps are equivalences under suitable assumptions.

We then analyze how these structures transfer along (weak and strong) equivalences of categories, and, in particular, show how they descend from a category (not assumed univalent/saturated) to its Rezk completion. To this end, we introduce *relative universes*, generalizing the preceding notions, and study the transfer of such relative universes along suitable structure.

We work throughout in (intensional) dependent type theory; some results, but not all, assume the univalence axiom. All the material of this paper has been formalized in Coq, over the UniMath library.

1   Introduction

Various kinds of categorical structures have been introduced in which to interpret type theories: for instance, categories with families, C-systems, categories with attributes, and

so on. The aim of these structures is to encompass and abstract away the structural rules of type theories – weakening, substitution, and so on – that are independent of the specific logical type and term formers of the type theory under consideration.

For a given kind of categorical structures and a given type theory one expects furthermore to be able to build, from the syntax, an initial object of a category or 2-category whose objects are categorical structures of this kind equipped with suitable extra operations 'modelling' the type and term formers of the type theory under consideration. This universal property then provides a means to construct interpretations of the syntax, by assembling the objects of the desired interpretation into another instance of such a structure.

It is natural to ask if, and in what sense, these various categorical structures are equivalent, or otherwise related. The equivalences, differences, and comparisons between them are often said to be well-known, but few precise statements exist in the literature.

The goals of the present work are twofold. Firstly, to give some such comparisons precisely and carefully. And secondly, to illustrate how in univalent foundations, such comparisons can be approached in a different and arguably more straightforward fashion than in classical settings.

More specifically, the paper falls into two main parts:

- comparison between categories with families (CwF's) and split type-categories;
- interaction between CwF structures and Rezk completion of categories, and comparison between CwF structures and representable maps of presheaves, all via comparison with *relative universes* (introduced in the present work).

Our constructions and equivalences may be summed up in the following diagram:

$$\mathsf{splty}(\mathcal{C}) \xleftrightarrow{\;\simeq\;} \mathsf{cwf}(\mathcal{C}) \xleftrightarrow{\;\simeq\;} \mathsf{relu}(\mathsf{y}_\mathcal{C}) \longrightarrow \mathsf{relu}(\mathsf{y}_{\mathsf{RC}(\mathcal{C})}) \xleftrightarrow{\;\simeq\;} \mathsf{cwf}(\mathsf{RC}(\mathcal{C}))$$

$$\mathsf{rep}(\mathcal{C}) \xleftrightarrow{\;\simeq\;} \mathsf{relwku}(\mathsf{y}_\mathcal{C}) \xleftrightarrow{\;\simeq\;} \mathsf{relwku}(\mathsf{y}_{\mathsf{RC}(\mathcal{C})}) \xleftrightarrow{\;\simeq\;} \mathsf{rep}(\mathsf{RC}(\mathcal{C}))$$

All proofs and constructions of the article have been formalized in Coq, over the `UniMath` library. We take this as licence to err on the side of indulgence in focusing on the key ideas of constructions and suppressing less-enlightening technical details, for which we refer readers to the formalization. An overview the formalization is given in Section 6. Throughout, we annotate results with their corresponding identifier in the formalization, in `teletype_font`.

2 Background

2.1 Categorical structures for type theory

In this short section we briefly review some of the various categorical structures for dependent type theory introduced in the literature. We do not aim to give a comprehensive survey of the field, but just to recall what pertains to the present paper.

The first introduced were Cartmell's *contextual categories* [4, 5], since studied by Voevodsky under the name of *C-systems* [15, 13].

Pitts [10, Def. 6.3.3] introduced *split type-categories*, originally just as *type-categories*; Van den Berg and Garner [12, Def. 2.2.1] (whom we follow here) later reorganized Pitts' definition to isolate the splitness conditions, and hence allow what they call *(non-split) type-categories*. These have also been studied as *categories with attributes* by Hofmann [8] and others.[1]

---

[1] Cartmell originally used the name *categories with attributes* for a slightly different notion [4, §3.2].

Categories with families were defined by Dybjer [6] to make explicit the data of *terms*, not taken as primary in the previous approaches. A functional variant of Dybjer's definition, which we follow in the present paper, was suggested by Fiore [7, Appendix], and studied under the name *natural models* by Awodey [3] (see also footnote to Def. 32 below).

The notions of *universes* and *universe categories*, which we generalize in the present work to *universes relative to a functor*, were introduced by Voevodsky in [13, Def. 2.1].

2.2 The agnostic, univalent, and classical settings

The background setting of the present work is intensional type theory, assuming throughout: Σ-types, with the strong η rule; identity types; Π-types, also with η, and functional extensionality; 0, 1, 2, and N; propositional truncation (axiomatised as in [11, §3.7]); and two universes closed under all these constructions.

All the above is *agnostic* about equality on types – it is not assumed either to be univalent, or to be always a proposition – and hence is expected to be compatible with the interpretation of types as classical sets, as well as homotopical interpretations such as the simplicial set model. In particular, our main definitions of categorical structures use only this background theory, and under the classical interpretation they become the established definitions from the literature. Similarly, most of the comparison maps we construct rely only on this, so can be understood in the classical setting.

Other results, however – essentially, all non-trivial equivalences of types we prove – assume additionally the univalence axiom; these therefore hold only in the univalent setting, and are not compatible with the classical interpretation.

We mostly follow the type-theoretic vocabulary standardized in [11]. A brief, but sufficient, overview is given in [1], among other places. By Set we denote the category of h-sets of a fixed, but unspecified universe.

We depart from it (and other type-theoretic traditions) in using *existence* for what is called *mere existence* in [11] and *weak existence* by Howard [9], since this is what corresponds to the standard mathematical usage of *existence*.

2.3 Comparing structures in the univalent setting

Suppose one wishes to show that two kinds of mathematical object are equivalent – say, one-object groupoids and groups. What precise statement should one aim for?

In the classical setting, the most obvious candidate – a bijection of the sets (or classes) of these objects – is not at all satisfactory. On the one hand, the natural back-and-forth constructions may well fail to form a bijection. On the other hand, the axiom of choice may imply that bijections exist even when the objects involved are quite unrelated.

To give a more meaningful statement, one may define suitable morphisms, corral the objects into two categories, and construct an equivalence of categories between these. Sometimes, one needs to go further, and construct an equivalence of higher categories, or spaces.

In the univalent setting, however, life is simpler. The most straightforward candidate, an equivalence of *types* between the types of the two kinds of objects, is already quite satisfactory and meaningful, corresponding roughly to an equivalence between the *groupoids* of such objects in the classical setting (or higher groupoids, etc.).

This is not to dismiss the value given by constructing (say) an equivalence of categories However, defining the morphisms and so on is no longer *required* in order to give a meaningful equivalence between the two kinds of objects.

Another advantage of a comparison in terms of equivalence of types is its *uniformity*. Indeed, the one notion of equivalence of types can serve to compare objects that naturally form the elements of sets, or the objects of categories, or bicategories, etc.

In the present paper, therefore, we take advantage of this: two of our main results are such equivalences, between the types of categories with families and of split type-categories, and between the types of representable maps of presheaves on a category $\mathcal{C}$, and of CwF structures on its Rezk completion.

(We have focused here on equivalences, but the principle applies equally for other comparisons: for instance, an injection of types carries more information than an injection of sets/classes, corresponding roughly to a full and faithful map of (possibly higher) groupoids.)

2.4   Categories in the univalent setting

The fundamentals of category theory were transferred to the univalent setting in [1]. Two primary notions of category are defined, there called *precategories* and *categories*. We change terminology, calling their precategories *categories* (since it is this that becomes the traditional definition under the set interpretation), and their categories *univalent categories*.

Specifically, a *category* $\mathcal{C}$ (in our terminology) consists of:

- a type $\mathcal{C}_0$, its *objects*;
- for each $a, b : \mathcal{C}_0$, a set $\mathcal{C}(a, b)$, the *morphisms* or *maps* from a to b;
- together with identity and composition operations satisfying the usual axioms.

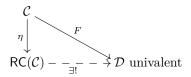We emphasize that the hom-sets $\mathcal{C}(a, b)$ are required to be sets, but $\mathcal{C}_0$ is allowed to be an arbitrary type.

In any category $\mathcal{C}$, there is a canonical map from equalities of objects to isomorphisms, $\mathsf{idtoiso}_{a,b} : (a =_{\mathcal{C}_0} b) \to \mathsf{Iso}_{\mathcal{C}}(a, b)$. We say that $\mathcal{C}$ is *univalent* if for all $a, b : \mathcal{C}_0$, this map $\mathsf{idtoiso}_{a,b}$ is an equivalence: informally, if 'equality of objects is isomorphism'.

A central example is the category Set of sets (in some universe). Univalence of this category follows directly from the univalence axiom for the corresponding universe. It follows in turn that $\mathsf{PreShv}(\mathcal{C})$, the category of presheaves on a category $\mathcal{C}$, is also always univalent. We write $\mathsf{y}_{\mathcal{C}} : \mathcal{C} \to \mathsf{PreShv}(\mathcal{C})$ for the Yoneda embedding.

In properties of functors, we distinguish carefully between existence properties and chosen data. We say a functor $F : \mathcal{C} \to \mathcal{D}$ is *essentially surjective* if for each $d : \mathcal{D}_0$, there exist some $c : \mathcal{C}_0$ and isomorphism $i : Fc \cong d$, and is *split essentially surjective* if it is equipped with an operation giving, for each $d : \mathcal{D}_0$, some such c and i. A *weak equivalence* is a functor that is full, faithful, and essentially surjective.

An important construction from [1] is the *Rezk completion* $\mathsf{RC}(\mathcal{C})$ of a category $\mathcal{C}$, the 'free' univalent category on $\mathcal{C}$. Precisely, $\mathsf{RC}(\mathcal{C})$ is univalent, and there is a weak equivalence $\eta_{\mathcal{C}} : \mathcal{C} \to \mathsf{RC}(\mathcal{C})$, enjoying the expected universal property: any functor from $\mathcal{C}$ to a univalent category factorizes uniquely through $\eta_{\mathcal{C}}$.

Note that the main definitions make sense in the agnostic background setting, and under the classical interpretation become the standard definitions; but the Rezk completion construction and the univalence of Set and $\mathsf{PreShv}(\mathcal{C})$ rely additionally on the univalence axiom.

Indeed, [1] additionally assumes *resizing axioms*, in order to show that $\mathsf{RC}(\mathcal{C})$ is no larger than $\mathcal{C}$ That is, a priori, the type $\mathsf{RC}(\mathcal{C})_0$ lives in a higher universe than the type $\mathcal{C}_0$. However,

for the present paper, this size issue is not a problem; so we do not need to assume resizing rules.

As usual, we will write $f : a \to b$ for $f : \mathcal{C}(a, b)$ in arbitrary categories, and will write $c : \mathcal{C}$ rather than $c : \mathcal{C}_0$. We write composition in the 'diagrammatic' order; that is, the composite of $f : a \to b$ and $g : b \to c$ is denoted $f \cdot g : a \to c$.

3 Equivalence between categories with families and split type-categories

3.1 CwF's and type-categories

For the most part, we take care to follow established definitions closely. We depart from most literature however in one way: we do not take CwF's or type-categories (or other similar structures) to include a terminal object. This does not interact in any way with the rest of the structure, so does not affect the equivalences we construct below. We do this since both versions (with and without terminal object) seem useful for the development of the theory; and it is easier to equip objects with extra structure later than to remove it.

▶ **Definition 1** ([7, Appendix]; `cwf_structure`, `cwf`). A *category with families (à la Fiore)* consists of:

0. a category $\mathcal{C}$, together with

1. presheaves $\mathsf{Ty}, \mathsf{Tm} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$;

2. a natural transformation $\mathsf{p} : \mathsf{Tm} \to \mathsf{Ty}$; and

3. for each object $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, a representation of the fiber of p over A, i.e.
   a. an object $\Gamma.A : \mathcal{C}$ and map $\pi_A : \Gamma.A \to \Gamma$,
   b. an element $\mathsf{te}_A : \mathsf{Tm}(\Gamma.A)$, such that $\mathsf{p}_{\Gamma.A}(\mathsf{te}_A) = \pi_A^* A : \mathsf{Ty}(\Gamma.A)$,
   c. and such that the induced commutative square

$$
\begin{array}{ccc}
\mathsf{y}(\Gamma.A) & \xrightarrow{\widehat{\mathsf{te}_A}} & \mathsf{Tm} \\
{\scriptstyle \mathsf{y}(\pi_A)} \downarrow \quad {\scriptstyle \lrcorner} & & \downarrow {\scriptstyle \mathsf{p}} \\
\mathsf{y}(\Gamma) & \xrightarrow{\widehat{A}} & \mathsf{Ty}
\end{array}
$$

is a pullback. (Here e.g. $\widehat{A}$ denotes the Yoneda transpose of an element of a presheaf.) By a *CwF structure* on a category $\mathcal{C}$, we mean the data of items 1–3 above.

▶ Remark. This is a reformulation, due to Fiore, of Dybjer's original definition of CwF's [6, Def. 1], replacing the single functor $\mathcal{C} \to \mathsf{Fam}$ by the map of presheaves $\mathsf{p} : \mathsf{Tm} \to \mathsf{Ty}$.

▶ **Definition 2.** Let $\mathcal{C}$ be a CwF, $\Gamma : \mathcal{C}$ an object, and $e : A = B$ an equality of elements of $\mathsf{Ty}(\Gamma)$. We write $\Delta_e : \Gamma.A \cong \Gamma.B$ for the induced isomorphism $\mathsf{idtoiso}(\mathsf{ap}_{x \mapsto \Gamma.x}(e))$.

Since $\mathsf{Ty}(\Gamma)$ is a set, we will sometimes suppress e and write just $\Delta_{A,B}$. We will also use this notation in other situations with a family Ty and operation $\Gamma, A \mapsto \Gamma.A$ as in a CwF.

▶ **Definition 3** ([10]; `typecat_structure`, `is_split_typecat`, `split_typecat_structure`). A *type-category* consists of:

0. a category $\mathcal{C}$, together with

1. for each object $\Gamma : \mathcal{C}$, a type $\mathsf{Ty}(\Gamma)$,

2. for each $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, an object $\Gamma.A : \mathcal{C}$

3. for each such Γ, A, a morphism $\pi_A : \Gamma.A \to \Gamma$,

4. for each map $f : \Gamma' \to \Gamma$, a 'reindexing' function $\mathsf{Ty}(\Gamma) \to \mathsf{Ty}(\Gamma')$, denoted $A \mapsto f^* A$,

5. for each Γ, $A : \mathsf{Ty}(\Gamma)$, and $f : \Gamma' \to \Gamma$, a morphism $\mathsf{q}(f, A) : \Gamma'.f^* A \to \Gamma.A$,

6. such that for each such Γ, A, Γ', f, the following square commutes and is a pullback:

$$
\begin{array}{ccc}
\Gamma'.f^* A & \xrightarrow{\;\;\mathsf{q}(f,A)\;\;} & \Gamma.A \\
{\scriptstyle \pi_{f^* A}} \downarrow & \lrcorner & \downarrow {\scriptstyle \pi_A} \\
\Gamma' & \xrightarrow[\;\;\;\;f\;\;\;\;]{} & \Gamma
\end{array}
$$

A type-category is *split* if:

7. for each Γ, the type $\mathsf{Ty}(\Gamma)$ is a set;

8. for each Γ and $A : \mathsf{Ty}(\Gamma)$, we have equalities
   a. $e : 1_\Gamma^* A = A$, and
   b. $\mathsf{q}(1_\Gamma, A) = \Delta_e \; : \; \Gamma.1_\Gamma^* A \to \Gamma.A$; and

9. for $f' : \Gamma'' \to \Gamma')$, $f : \Gamma' \to \Gamma$, and $A : \mathsf{Ty}(\Gamma)$, we have equalities
   a. $e' : (f' \cdot f)^* A = f'^* f^* A$, and
   b. $\mathsf{q}(f' \cdot f, A) = \Delta_{e'} \cdot \mathsf{q}(f', f^* A) \cdot \mathsf{q}(f, A) \; : \; \Gamma''.(f' \cdot f)^* A \to \Gamma.A$.

In the present work, we will only consider split type-categories. Non-split type-categories are however also of great importance, especially in the agnostic/univalent settings, since classical methods for constructing split ones may no longer work.

3.2 Equivalence between CwF's and split type-categories

The main goal of this section is to construct an equivalence of types between the type of CwF's and the type of split type-categories. In outline, we proceed as follows:

Firstly, we specialize to giving an equivalence between CwF structures and split type-category structures over a fixed base category $\mathcal{C}$.

Secondly, we further abstract out the shared part of these, decomposing them into

- *object extension structures*, the shared structure common to CwF's and split type-categories, and
- structures comprising the remaining data of CwF structures and split type-category structures, which we call *term-structures* and q-*morphism structures* respectively.

Finally, we give an equivalence between term-structures and q-morphism structures over a given category and object extension structure. We do this by defining a *compatibility* relation between them, and showing that for each term-structure there exists a unique compatible q-morphism structure, and vice versa.

$$
\begin{array}{ccccc}
 & & \displaystyle\sum_{X,Y,Z} \mathsf{compat}_X(Y,Z) & & \\
 & {\scriptstyle\simeq}\nearrow & & \nwarrow{\scriptstyle\simeq} & \\
\mathsf{splty}(\mathcal{C}) \xleftrightarrow{\;\simeq\;} \displaystyle\sum_{X:\mathsf{objext}(\mathcal{C})} \mathsf{qmor}(X) & & & & \displaystyle\sum_{X:\mathsf{objext}(\mathcal{C})} \mathsf{tmstr}(X) \xleftrightarrow{\;\simeq\;} \mathsf{cwf}(\mathcal{C}) \\
 & \searrow & & \swarrow & \\
 & & \mathsf{objext}(\mathcal{C}) & &
\end{array}
$$

▶ **Definition 4** (`obj_ext_structure`). A *(split) object extension structure* on a category $\mathcal{C}$ consists of:

1. a functor $\mathsf{Ty} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$;
2. for each $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$,
   a. an object $\Gamma.A$, and
   b. a *projection* morphism $\pi_A : \mathcal{C}(\Gamma.A, \Gamma)$.

In general, one might want to loosen the setness and functoriality conditions on Ty, and hence distinguish the present definition as the split version of a more general notion. In this paper, however, we do not consider the non-split case, so take *object extension structures* to mean the split ones throughout.

▶ **Definition 5** (`term_fun_structure`)**.** Let $\mathcal{C}$ be a category equipped with an object extension structure X. A *(functional) term-structure* over X consists of:
1. a presheaf $\mathsf{Tm} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$, and natural transformation $\mathsf{p} : \mathsf{Tm} \to \mathsf{Ty}$;
2. for each object $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, an element $\mathsf{te}_A : \mathsf{Tm}(\Gamma.A)$, such that $(\Gamma.A, \pi_A, \mathsf{te}_A)$ form a representation of the fiber of p over A as in item 3 of Def. 1.

One might say *functional* to distinguish these from *familial* term-structures, which would correspond analogously to CwF's in the sense of Dybjer, with $\mathsf{Tm}(\Gamma)$ a family indexed by $\mathsf{Ty}(\Gamma)$. For the present paper, however, we work only with the functional ones, so call these simply *term-structures*.

▶ **Problem 6.** *Given a category $\mathcal{C}$, to construct an equivalence between CwF structures on $\mathcal{C}$ and pairs (X, Y) of an object extension structure X on $\mathcal{C}$ and a term-structure Y over X.*

▶ **Construction 7** (for Problem 6; `weq_cwf_cwf'_structure`)**.** Mathematically, this is essentially trivial, as is visibly evident from the definitions: just a matter of reordering and reassociating Σ-types, and distributing Π-types over Σ-types. It is perhaps worth noting however that this distributivity requires functional extensionality. ◀

▶ **Definition 8** (`qq_morphism_structure`)**.** Let $\mathcal{C}$ be a category equipped with an object extension structure X. A *(split)* q-*morphism structure* over X consists of:
1. for each $f : \Gamma' \to \Gamma$ and $A : \mathsf{Ty}(\Gamma)$, a map $\mathsf{q}(f, A) : \Gamma.A \to \Gamma$, such that following square commutes and is a pullback

$$
\begin{array}{ccc}
\Gamma'.f^*A & \xrightarrow{\ \mathsf{q}(f,A)\ } & \Gamma.A \\
{\scriptstyle \pi_{f^*A}}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle \pi_A} \\
\Gamma' & \xrightarrow[\ \ f\ \ ]{} & \Gamma
\end{array}
$$

   and such that
2. for each Γ and $A : \mathsf{Ty}(\Gamma)$, $\mathsf{q}(1_\Gamma, A) = \Delta_{1_\Gamma^* A, A} : \Gamma.1_\Gamma^* A \to \Gamma.A$; and
3. for each $f' : \Gamma'' \to \Gamma'$, $f : \Gamma' \to \Gamma$, and $A : \mathsf{Ty}(\Gamma)$,

$$
\mathsf{q}(f' \cdot f, A) = \Delta_{(f' \cdot f)^* A, f'^* f^* A} \cdot \mathsf{q}(f', f^*A) \cdot \mathsf{q}(f, A) \ : \ \Gamma''.(f' \cdot f)^* A \to \Gamma.A.
$$

   Here the suppressed equalities on the Δ terms come from the functoriality axioms of Ty.

▶ **Problem 9.** *Given a category $\mathcal{C}$, to construct an equivalence between split type-category structures on $\mathcal{C}$ and pairs (X, Z) of an object extension structure X on $\mathcal{C}$ and a q-morphism structure Z over X.*

▶ **Construction 10** (for Problem 9; `weq_standalone_to_regrouped`)**.** Much like Construction 7, simply a matter of wrangling Π- and Σ-types. ◀

For most of the remainder of this section, we fix a category $\mathcal{C}$ and object extension structure X on $\mathcal{C}$. We can now explicitly define constructions going back and forth between term-structures and q-morphism structures over X, preparatory to showing that they form an equivalence. Before we do so, we define the *compatibility* relation between them.

▶ **Definition 11.** Let Y be a term-structure and Z a q-morphism structure over X. Say that Y and Z are *compatible* if for all $f : \Gamma \to \Gamma'$ and $A : \mathsf{Ty}(\Gamma)$, $\mathsf{te}_{f^*A} = \mathsf{q}(f, A)^* \mathsf{te}_A$.

▶ **Problem 12.** *Given a term-structure Y over X, to construct a q-morphism structure over X compatible with Y.*

▶ **Construction 13** (for Problem 12; `compatible_qq_from_term`). Given Y and $f : \Gamma \to \Gamma'$ and $A : \mathsf{Ty}(\Gamma)$, the term-structure axioms give a pullback square

$$
\begin{array}{ccc}
\mathcal{C}(\Gamma'.f^*A, \Gamma.A) & \xrightarrow{g \mapsto g^*\mathsf{te}_A} & \mathsf{Tm}(\Gamma'.f^*A) \\
{\scriptstyle -\cdot\pi_A} \downarrow & \lrcorner & \downarrow {\scriptstyle \mathsf{p}_{\Gamma'.f^*A}} \\
\mathcal{C}(\Gamma'.f^*A, \Gamma) & \xrightarrow{g \mapsto g^*A} & \mathsf{Ty}(\Gamma'.f^*A)
\end{array}
$$

So we may take $\mathsf{q}(f, A)$ to be the unique map $\Gamma'.f^*A \to \Gamma.A$ such that $\mathsf{q}(f, A) \cdot \pi_A = \pi_{f^*A} \cdot f$ and $\mathsf{q}(f, A)^* \mathsf{te}_A = \mathsf{te}_{f^*A}$. Verification that this forms a compatible q-morphism structure is essentially routine calculation. ◀

▶ **Problem 14.** *Given a q-morphism structure Y over X, to construct a term-structure over X compatible with Y.*

▶ **Construction 15** (for Problem 14; `compatible_term_from_qq`). This construction is rather more involved; we only sketch it here, and refer to the formalization for full details.

Briefly, $\mathsf{Tm}(\Gamma)$ is the set of pairs (A, s), where $A : \mathsf{Ty}(\Gamma)$, and $s : \Gamma \to \Gamma.A$ is a section of π_A. Its functorial action f^* involves pulling back sections along the pullback squares given by the q-morphism structure. Finally, the universal element $\mathsf{te}_A : \mathsf{Tm}(\Gamma)$ is the pair $(\pi_A^*A, \delta_{\pi_A})$, where $\delta_{\pi_A} : \Gamma.A \to \Gamma.A.\pi_A^*A$ is the diagonal map of the pullback square for A and π_A given by the q-morphism structure. ◀

▶ **Problem 16.** *(Assuming Univalence.) To give an equivalence between term-structures and q-morphism structures over X, whose underlying functions are as given in the two preceding constructions.*

From this equivalence, to derive an equivalence between the type of pairs (X, Y) of an object extension structure and a term-structure, and of pairs (X, Z) an object extension structure and a q-morphism structure.

▶ **Construction 17** (for Problem 16; `weq_cwf'_sty'`). As intimated above, we proceed by showing that for each term-structure, the compatible q-morphism structure constructed above is in fact the *unique* compatible such structure, and vice versa.

This equivalence immediately induces an equivalence of pair types, which is the identity on the first component carrying the object extension structure.

Note in particular that for this result – unlike in the constructions above – we rely essentially on the univalence axiom. ◀

▶ **Problem 18.** *To construct an equivalence between cwf structures and split type-category structures on a category $\mathcal{C}$.*

▶ **Construction 19** (for Problem 18; `weq_sty_cwf`). By composing the equivalences of Constructions 7, 10, and 17:

$$\mathsf{splty}(\mathcal{C}) \simeq \sum_{X:\mathsf{object}(\mathcal{C})} \mathsf{qmor}(X) \simeq \sum_{X:\mathsf{objext}(\mathcal{C})} \mathsf{tmstr}(X) \simeq \mathsf{cwf}(\mathcal{C}) \ . \qquad\qquad\blacktriangleleft$$

The back-and-forth constructions above (though not compatibility) are also sketched in e.g. [8, Sections 3.1, 3.2] (note that Hofmann's *categories with attributes* are what we call *split type-categories*). Hofmann also mentions that – in our terminology – going from q-morphism structures to term-structures and back yields the original q-morphism structure. However, working in set-theoretic foundations, the same is not true for the other direction – the original term presheaf is not recovered up to equality. This is exactly where the univalence axiom comes to our rescue: it allows us to conclude that, by showing that the obtained term presheaf is *isomorphic* to the original one, the two are identical. Hence we obtain an equivalence of types. Note however that our proof that the maps are indeed an equivalence is slightly different, to avoid the difficult direct construction of an identity between two term-structures.

In the absence of the univalence axiom, the constructed maps back and forth can still be used to compare term-structures and q-morphism structures: by defining suitable notions of morphisms of those structures, these maps underlie an equivalence of categories. We have also formalized this equivalence of categories, and will report on it in a forthcoming article.

4 Relative universes and a transfer construction

In this section we introduce the notion of *universes relative to a functor $J : \mathcal{C} \to \mathcal{D}$*. This notion generalizes the universes studied in [13], and is inspired by the generalization of monads to relative monads [2].

4.1 Relative universes and weak universes

▶ **Definition 20** (`fpullback`). Let $J : \mathcal{C} \to \mathcal{D}$ be a functor, and $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ a morphism of $\mathcal{D}$.

Given an object X of $\mathcal{C}$ and morphism $f : J(X) \to \mathsf{U}$ in $\mathcal{D}$, a *J-pullback of p along f* consists of an object X' of $\mathcal{C}$ and morphisms $p' : X' \to X$ and $Q : J(X') \to \tilde{\mathsf{U}}$, such that the following square commutes and is a pullback:

$$
\begin{array}{ccc}
J(X') & \xrightarrow{\ Q\ } & \tilde{\mathsf{U}} \\
{\scriptstyle J(p')}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle \mathsf{p}} \\
J(X) & \xrightarrow[\ f\ ]{} & \mathsf{U}
\end{array}
$$

▶ **Definition 21** (`relative_universe`, `weak_relative_universe`). Let $J : \mathcal{C} \to \mathcal{D}$ be a functor, as above.

A *J-universe structure* on a map $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ of $\mathcal{D}$ is a function giving, for each object X in $\mathcal{C}$ and map $f : J(X) \to \mathsf{U}$, a J-pullback $(X.f, \mathsf{p}_f, \mathsf{Q}(X, f))$ of p along f. A *universe relative to J*, or briefly a *J-relative universe*, is a map p equipped with a J-universe structure.

A *weak universe relative to J* is a map $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ such that for all suitable X, f, there exists some J-pullback of p along f.

A *universe* in $\mathcal{C}$, as defined in [13], is exactly a universe relative to the identity functor $\mathsf{Id}_\mathcal{C}$. We will see in Section 5 that universes relative to the Yoneda embedding $\mathsf{y}_\mathcal{C} : \mathcal{C} \to \mathsf{PreShv}(\mathcal{C})$ correspond precisely to CwF structures on $\mathcal{C}$.

▶ **Lemma 22** (`isaprop_rel_universe_structure`). *Suppose $J : \mathcal{C} \to \mathcal{D}$ is full and faithful, and $\mathcal{C}$ is univalent. Then for any morphism* $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ *in $\mathcal{D}$, and object $X : \mathcal{C}$ and $f : J(X) \to \mathsf{U}$, the type of J-pullbacks of p along f is a proposition. Similarly, for any such* p, *the type of J-universe structures on* p *is a proposition.*

Proof. The first statement is similar to the argument that pullbacks in univalent categories are unique. The second follows directly from the first. ◀

▶ **Corollary 23** (`weq_relative_universe_weak_relative_universe`). *When $J : \mathcal{C} \to \mathcal{D}$ is fully faithful and $\mathcal{C}$ is univalent, the forgetful function from universes to weak universes relative to J is an equivalence.*

4.2   Transfer constructions

We give three constructions for transferring (weak) relative universes from one functor to another. The first, with all data assumed to be given explicitly, is the most straightforward. However, it does not suffice for transfers to the Rezk completion, since the embedding $\eta_\mathcal{C} : \mathcal{C} \to \mathsf{RC}(\mathcal{C})$ is only a weak equivalence, not in general *split* essentially surjective. The second and third constructions are therefore adaptations of the first to require only essential surjectivity.

▶ **Problem 24.** *Given a square of functors commuting up to natural isomorphism, as in*

$$
\begin{array}{ccc}
\mathcal{C} & \xrightarrow{\ J\ } & \mathcal{D} \\
{\scriptstyle R}\downarrow & \overset{\nwarrow}{}_{\alpha} & \downarrow{\scriptstyle S} \\
\mathcal{C}' & \xrightarrow[\ J'\ ]{} & \mathcal{D}',
\end{array}
$$

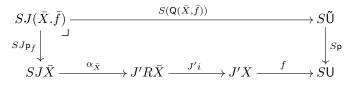*such that*
- *S preserves pullbacks,*
- *S is split full, and*
- *R is split essentially surjective,*

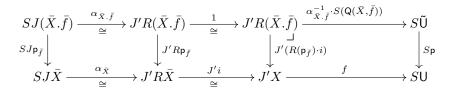*and given a J-relative universe structure on a map $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ in $\mathcal{D}$, to construct a J'-universe structure on $S(\mathsf{p})$ in $\mathcal{D}'$.*

▶ **Construction 25** (for Problem 24; `rel_universe_structure_induced_with_ess_split`). Given X in $\mathcal{C}'$ and $f : J'(X) \to S\mathsf{U}$, we need to construct a J'-pullback of $S(\mathsf{p})$ along f.

Split essential surjectivity of R gives some $\bar{X} : \mathcal{C}$ and isomorphism $i : R(\bar{X}) \cong X$. We therefore have $\alpha_{\bar{X}} \cdot J'i \cdot f : SJ\bar{X} \to S\mathsf{U}$; so split fullness of S gives us some $\bar{f} : J\bar{X} \to \mathsf{U}$ with $S\bar{f} = \alpha_{\bar{X}} \cdot J'i \cdot f$. Taking the given J-pullback of p along $\bar{f}$ and mapping it forward under S, we get a pullback square in $\mathcal{D}'$:

$$
\begin{array}{ccc}
SJ(\bar{X}.\bar{f}) & \xrightarrow{\ \ S(\mathsf{Q}(\bar{X},\bar{f}))\ \ } & S\tilde{\mathsf{U}} \\
{\scriptstyle SJ\mathsf{p}_{\bar{f}}}\downarrow \quad \lrcorner & & \downarrow{\scriptstyle S\mathsf{p}} \\
SJ\bar{X} \xrightarrow{\ \alpha_{\bar{X}}\ } J'R\bar{X} & \xrightarrow{\ J'i\ } J'X \xrightarrow{\ f\ } & S\mathsf{U}
\end{array}
$$

The maps $R(\mathsf{p}_{\bar{f}}) \cdot i : R(\bar{X}.\bar{f}) \to X$ and $\alpha_{\bar{X}.\bar{f}}^{-1} \cdot S(\mathsf{Q}(\bar{X}, \bar{f})) : J'R(\bar{X}.\bar{f}) \to S\tilde{\mathsf{U}}$ now give the desired J'-pullback of $S\mathsf{p}$ along f, since the square they give – the right-hand square below – is isomorphic to the pullback obtained above.

$$SJ(\bar{X}.\bar{f}) \xrightarrow[\cong]{\alpha_{\bar{X}.\bar{f}}} J'R(\bar{X}.\bar{f}) \xrightarrow[\cong]{1} J'R(\bar{X}.\bar{f}) \xrightarrow{\alpha_{\bar{X}.\bar{f}}^{-1} \cdot S(\mathsf{Q}(\bar{X},\bar{f}))} S\tilde{\mathsf{U}}$$

with vertical maps $SJ\mathsf{p}_{\bar{f}}$, $J'R\mathsf{p}_{\bar{f}}$, $J'(R(\mathsf{p}_{\bar{f}})\cdot i)$, $S\mathsf{p}$

$$SJ\bar{X} \xrightarrow[\cong]{\alpha_{\bar{X}}} J'R\bar{X} \xrightarrow[\cong]{J'i} J'X \xrightarrow{f} S\mathsf{U}$$
◀

▶ **Problem 26.** *Given a square of functors as in Problem 24, such that*

- *S preserves pullbacks,*
- *S is full,*
- *R is essentially surjective,*
- *$\mathcal{C}'$ is univalent,*
- *J' is fully faithful,*

and given a J-relative universe structure on a map $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ in $\mathcal{D}$, to construct a J'-universe structure on $S(\mathsf{p})$ in $\mathcal{D}'$.

▶ **Construction 27** (for Problem 26; `rel_universe_structure_induced_with_ess_surj`). Mostly the same as Construction 25. The only problematic steps are finding $(\bar{X}, i)$ and $\bar{f}$ as above, since the hypotheses which provided them there have now been weakened to existence properties. However, our new hypotheses that $\mathcal{C}'$ is univalent and J' fully faithful allow us to apply Lemma 22 to see that the goal of a J'-pullback is a proposition; so the existence properties do in fact suffice. ◀

▶ **Lemma 28** (`is_universe_transfer`). *Suppose given a square of functors as in Problem 24, such that*

- *S preserves pullbacks,*
- *S is full,*
- *R is essentially surjective.*

If $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ in $\mathcal{D}$ is a weak J-relative universe in $\mathcal{C}$, then $S(\mathsf{p})$ is a weak J'-relative universe in $\mathcal{D}'$.

Proof. Again, mostly the same as Construction 25. Now our goal is just to show, for each suitable X, f, that there exists some J'-pullback of $S\mathsf{p}$ along f. This is by construction a proposition; so, again, we can obtain objects from our existence hypotheses whenever required. ◀

▶ **Lemma 29** (`isweq_is_universe_transfer`). *Given functors satisfying the hypotheses of Lemma 28, if additionally R is full and S is faithful, then a map $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$ in $\mathcal{D}$ is a weak J-relative universe if and only if $S(\mathsf{p})$ is a weak J'-relative universe.*

Proof. One implication is exactly Lemma 28. For the other, assume that $S(\mathsf{p})$ is a weak J'-relative universe; we must show that p is a weak J-relative universe.

Given $X : \mathcal{C}$ and $f : J(X) \to \mathsf{U}$, we need to show there exists a J-pullback for p along f. Since the goal is just existence, we can take witnesses for existence hypotheses as needed. In particular, we can take some $X' : \mathcal{C}'$, $p' : X' \to RX$ and $Q : J'X' \to S\tilde{\mathsf{U}}$ forming a J-pullback for $S\mathsf{p}$ along $\alpha_X^{-1} \cdot Sf : J'RX \to S\mathsf{U}$.

By essential surjectivity of R, we can find some $\bar{X}' : \mathcal{C}$ and isomorphism $i : R\bar{X}' \cong X'$; and similarly by its fullness, we can take some arrow $\bar{p}' : \bar{X}' \to X$ such that $R(\bar{p}') = i \cdot p'$. Finally, fullness of S gives a map $\bar{Q} : J\bar{X}' \to \tilde{\mathsf{U}}$ with $S\bar{Q} = \alpha_{\bar{X}'} \cdot J'i \cdot Q$.

But now $(\bar{X}', \bar{p}', \bar{Q})$ form a J-pullback for p along f, since under S, the square they give becomes isomorphic to the original J'-pullback square of (X', p', Q), and S reflects limits, as it is full and faithful.

$$\begin{array}{ccccccc}
SJ\bar{X}' & \xrightarrow[\cong]{\alpha_{\bar{X}'}} & J'R\bar{X}' & \xrightarrow[\cong]{J'i} & J'X' & \xrightarrow{Q} & S\tilde{\mathsf{U}} \\
{\scriptstyle SJ\bar{p}'}\downarrow & & {\scriptstyle J'R\bar{p}'}\downarrow & & {\scriptstyle J'p'}\downarrow\;\lrcorner & & \downarrow{\scriptstyle S\mathsf{p}} \\
SJ\bar{X} & \xrightarrow[\cong]{\alpha_X} & J'R\bar{X} & \xrightarrow[\cong]{1} & J'X & \xrightarrow{\alpha_X^{-1}\cdot Sf} & S\mathsf{U}
\end{array}$$

◀

▶ **Problem 30.** *Given a square of functors as in Problem 24, such that*
- $\mathcal{D}$ *and* $\mathcal{D}'$ *are both univalent,*
- S *is an equivalence, and*
- R *is essentially surjective and full,*

to construct an equivalance between weak J-relative universes and weak J'-relative universes.
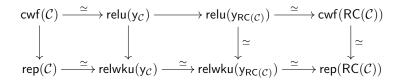
▶ **Construction 31** (for Problem 30; `weq_weak_relative_universe_transfer`). The equivalence S of univalent categories induces an equivalence between morphisms in $\mathcal{D}$ and in $\mathcal{D}'$. Lemma 29 implies that this restricts to an equivalence between weak relative universes as desired. ◀

<div style="border-left: 4px solid orange; padding-left: 8px;">**5**</div> ## 5 CwF structures, representable maps of presheaves and the Rezk completion

In this section, we show that CwF structures can be seen as relative universes, and hence apply the results of the previous section to transfer CwF structures along weak equivalences of categories: in particular, from a category to its Rezk completion.

We also consider *representable maps of presheaves*, corresponding similarly to relative *weak* universes, and use the results on relative universes to elucidate their relationship to CwF structures.

The resulting transfers and relationships are summed up in the following diagram, whose vertical maps all simply forget chosen structure:

$$\begin{array}{ccccccc}
\mathsf{cwf}(\mathcal{C}) & \xrightarrow{\simeq} & \mathsf{relu}(\mathsf{y}_{\mathcal{C}}) & \longrightarrow & \mathsf{relu}(\mathsf{y}_{\mathsf{RC}(\mathcal{C})}) & \xrightarrow{\simeq} & \mathsf{cwf}(\mathsf{RC}(\mathcal{C})) \\
\downarrow & & \downarrow & & \downarrow{\scriptstyle\simeq} & & \downarrow{\scriptstyle\simeq} \\
\mathsf{rep}(\mathcal{C}) & \xrightarrow{\simeq} & \mathsf{relwku}(\mathsf{y}_{\mathcal{C}}) & \xrightarrow{\simeq} & \mathsf{relwku}(\mathsf{y}_{\mathsf{RC}(\mathcal{C})}) & \xrightarrow{\simeq} & \mathsf{rep}(\mathsf{RC}(\mathcal{C}))
\end{array}$$

5.1 Representable maps of presheaves

▶ **Definition 32** (`rep_map`). A map $\mathsf{p} : \mathsf{Tm} \to \mathsf{Ty}$ of presheaves on a category $\mathcal{C}$ is *representable* if for each for each $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, there exists some representation of the fiber of p over A, i.e. some $\pi_A : \Gamma.A \to \Gamma$ and $\mathsf{te}_A \in \mathsf{Tm}(\Gamma.A)$ satisfying the conditions of Definition 1, item 3.[2]

In other words, a representable map of presheaves on $\mathcal{C}$ is just like a CwF structure, except that the representations are merely assumed to exist, not included as chosen data.

---

[2] Note that in [3], Awodey takes *representable map* to mean what we call a CwF structure, i.e. to include choices of representations of the fibers; cf. [3, §1.1, Algebraic character].

Evidently, the underlying map $p : Ty \to Tm$ of any CwF structure is representable, just by forgetting its chosen representations. This can sometimes be reversed:

▶ **Lemma 33** (`isweq_from_cwf_to_rep`). *If $\mathcal{C}$ is a univalent category, then the forgetful map from CwF structures to representable maps of presheaves on $\mathcal{C}$ is an equivalence. That is, any representable map of presheaves on $\mathcal{C}$ carries a unique choice of representing data.*

Proof. If suffices to show that for a given map $p : Tm \to Ty$, and for any $\Gamma : \mathcal{C}$ and $A : Ty(\Gamma)$, representing data $(\Gamma.A, \pi_A, \mathsf{te}_A)$ for the fiber is unique if it exists.

Such data is always unique up to isomorphism, for any category $\mathcal{C}$, since pullbacks are unique up to isomorphism and $y_\mathcal{C}$ is full and faithful. But when $\mathcal{C}$ is univalent, this uniqueness up to isomorphism can be translated into literal uniqueness, as required.

(Alternatively, following Constructions 35 and 37 below, we could see this result as essentially a special case of Corollary 23 on universe structures.) ◀

5.2 Transfer of CwF structures and representable maps of presheaves

In order to apply the transfer results of the previous section, we first establish the equivalences between CwF structures on a category $\mathcal{C}$ (resp. representable maps of presheaves) and relative (weak) universes on $y_\mathcal{C}$.

▶ **Problem 34.** *Given a category $\mathcal{C}$, to construct an equivalence between $\mathsf{cwf}(\mathcal{C})$ and $\mathsf{relu}(y_\mathcal{C})$.*

▶ **Construction 35** (for Problem 34; `weq_cwf_structure_RelUnivYo`). This is a matter of reassociating components, and replacing two quantifications over elements of a presheaf – once over $Ty(\Gamma)$, once over $Tm(\Gamma.A)$ – by quantification over the respective isomorphic sets of natural transformations into Ty and Tm. ◀

▶ **Problem 36.** *Given a category $\mathcal{C}$, to construct an equivalence between $\mathsf{rep}(\mathcal{C})$ and $\mathsf{relwku}(y_\mathcal{C})$.*

▶ **Construction 37** (for Problem 36; `weq_rep_map_weakRelUnivYo`). Similar to Construction 35. ◀

Next, we make use of this to transfer CwF structures and representable maps along weak equivalences, by viewing them as relative (weak) universes and applying the transfer results for those.

▶ **Problem 38.** *Given a weak equivalence $F : \mathcal{C} \to \mathcal{D}$, where $\mathcal{D}$ is univalent, to construct a map $\mathsf{cwf}(\mathcal{C}) \to \mathsf{cwf}(\mathcal{D})$.*

▶ **Construction 39** (for Problem 38; `transfer_cwf_weak_equivalence`). We construct a map $\mathsf{relu}(y_\mathcal{C}) \to \mathsf{relu}(y_\mathcal{D})$ as an instance of Construction 27 and obtain the desired map by composition with the equivalence to CwF structures of Construction 35. Consider the diagram

Here, the functor F° given by precomposition with F^{op} is a weak equivalence between univalent categories, and hence a strong equivalence with inverse $\mathcal{S}$. The isomorphism α

is constructed as follows: Note that fully faithful functors reflect isomorphisms; we apply this for the functor F° of precomposition with F^{op}. It hence suffices to construct a natural isomorphism from $\mathsf{y}_{\mathcal{C}} \cdot \mathcal{S} \cdot F^\circ \simeq \mathsf{y}_{\mathcal{C}}$ to $F \cdot \mathsf{y}_{\mathcal{D}} \cdot F^\circ$. But this is an instance of a general isomorphism: indeed, for any functor $G : \mathcal{A} \to \mathcal{X}$, we have natural transformation from $\mathsf{y}_{\mathcal{A}}$ to $G \cdot \mathsf{y}_{\mathcal{X}} \cdot G^\circ$, and this natural transformation is an isomorphism when G is fully faithful. This ends the construction of the natural isomorphism α. The functor $\mathcal{S}$ preserves pullbacks since it is fully faithful and essentially surjective. The hypotheses of Problem 26 are easily checked, hence Construction 27 applies.   ◄

▶ **Problem 40.** *Given a weak equivalence $F : \mathcal{C} \to \mathcal{D}$, to construct an equivalence* $\mathsf{rep}(\mathcal{C}) \simeq$ $\mathsf{rep}(\mathcal{D})$.

▶ **Construction 41** (for Problem 40; `transfer_rep_map_weak_equivalence`)**.** A direct instance of Construction 31.   ◄

Putting everything together, we obtain:

▶ **Problem 42.** *For any category $\mathcal{C}$, to construct an equivalence between representable maps on $\mathcal{C}$ and CwF structures on* $\mathsf{RC}(\mathcal{C})$.

▶ **Construction 43** (for Problem 42; `weq_rep_map_cwf_Rezk`)**.** Construction 41, applied to $\eta_{\mathcal{C}}$, gives us an equivalence $\mathsf{rep}(\mathcal{C}) \simeq \mathsf{rep}(\mathsf{RC}(\mathcal{C}))$. On the other hand, Lemma 33 tells us that $\mathsf{cwf}(\mathsf{RC}(\mathcal{C})) \simeq \mathsf{rep}(\mathsf{RC}(\mathcal{C}))$. Composing the first of these with the inverse of the second yields the desired equivalence.   ◄

6   Guide to the accompanying formalization

All constructions and theorems of this work have been formalized in the proof assistant Coq, over the `UniMath` library of univalent mathematics [16, 14]. We rely particularly heavily on `UniMath`'s category theory library.

Our formalization can be found at `https://github.com/UniMath/TypeTheory`. The version current at time of writing will remain permanently available under the tag `2017-ALV1`.

The main library will continue development, so naming, organisation, etc. may change from what is presented here. However, the file `Articles/ALV_2017.v` will be maintained to keep the main results of this paper available and locatable over future versions of the library.

For the reader interested in exploring the formalization, we recommend starting with that file, and following backwards to find the details of definitions and constructions. A browsable version is available at `https://unimath.github.io/TypeTheory/coqdoc/master/TypeTheory.Articles.ALV_2017.html`.

The specific material of the present article amounts to about 3700 lines of code in the formalization. Additionally, this development required formalizing a further c. 1500 lines of general background material (mostly on category theory) that had not previously been given.

6.1   Type theory of the formalisation

There are some subtleties in how the type theory of the formalisation relates to what we set out in Section 2.2.

Firstly, the type theory of Coq includes many powerful features; like `UniMath` itself, we deliberately avoid most of these, staying within the fragment described in [14, §1].

Secondly, functional extensionality and univalence are provided by `UniMath` as axioms – that is, as assumptions added to the global context. We use functional extensionality freely, but do not use univalence except where explicitly required, as noted in the text.

Thirdly, in order to acquire resizing principles (not otherwise available in Coq), `UniMath` uses type-in-type, and hence is in principle inconsistent. `UniMath` itself is careful not to use any consequences of type-in-type except for the resulting resizing principles. We are even more restricted: we do not make direct or essential use of the resizing principles. We do depend on them indirectly, since propositional truncation is implemented in `UniMath` using resizing principles. However, our use of truncation is always via the interface corresponding to its standalone axiomatisation, as assumed in Section 2.2.

7 Summary and future work

The above sections complete the construction of the maps and equivalences of types promised in the introduction: in particular,

- equivalence between split type-category structures and CwF structures;
- equivalence between CwF structures and universes relative to the Yoneda embedding, and similarly between representable maps and weak such relative universes;
- transfer of CwF structures and representable maps to the Rezk completion;
- equivalence between CwF structures on a category $\mathcal{C}$ and representable maps of presheaves on its Rezk completion.

There are several natural interesting directions for further work:

- Define *categories* of all the various structures considered here; show that the comparison constructions given here are all moreover functorial, and that our equivalences of types underlie equivalences of categories. Besides its intrinsic interest, this would make more of our constructions meaningful and useful in the classical setting.
- Extend these constructions to give comparisons with other categorical structures considered for similar purposes in the literature: Dybjer's original CwF's; Cartmell's contextual categories/C-systems; comprehension categories; categories with display maps...
- Understand further the transfer of CwF structures along the Rezk completion construction: does the result enjoy an analogous universal property, making it the 'free univalent category with families' on a category $\mathcal{C}$?

Progress in some of these directions may be already found in our formalization, though not included in the present article.

References

1 Benedikt Ahrens, Krzysztof Kapulkin, and Michael Shulman. Univalent categories and the Rezk completion. *Mathematical Structures in Computer Science*, 25:1010–1039, 2015. `arXiv:1303.0584`, `doi:10.1017/S0960129514000486`.

2 Thorsten Altenkirch, James Chapman, and Tarmo Uustalu. Monads need not be endofunctors. *Logical Methods in Computer Science*, 11(1), 2015. `doi:10.2168/LMCS-11(1:3)2015`.

3 Steve Awodey. Natural models of homotopy type theory. *Mathematical Structures in Computer Science*, pages 1–46, 2016. `arXiv:1406.3219`, `doi:10.1017/S0960129516000268`.

4 John Cartmell. Generalised algebraic theories and contextual categories. *Ph.D. Thesis, Oxford University*, 1978. URL: `http://www.cs.ru.nl/~spitters/Cartmell.pdf`.

5 John Cartmell. Generalised algebraic theories and contextual categories. *Ann. Pure Appl. Logic*, 32:209–243, 1986. `doi:10.1016/0168-0072(86)90053-9`.

6 Peter Dybjer. Internal type theory. In Stefano Berardi and Mario Coppo, editors, *Types for Proofs and Programs, International Workshop TYPES'95, Torino, Italy, June 5-8, 1995, Selected Papers*, volume 1158 of *Lecture Notes in Computer Science*, pages 120–134. Springer, 1995. `doi:10.1007/3-540-61780-9_66`.

7   Marcelo Fiore. Discrete generalised polynomial functors, 2012. Slides from talk given at ICALP 2012. URL: `http://www.cl.cam.ac.uk/~mpf23/talks/ICALP2012.pdf`.

8   Martin Hofmann. Syntax and semantics of dependent types. In *Semantics and Logics of Computation*, pages 79–130. Cambridge University Press, 1997.

9   William A. Howard. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*, pages 479–490. Academic Press, 1980. Reprint of 1969 article.

10   Andrew M. Pitts. Categorical logic. In S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science, Volume 5. Algebraic and Logical Structures*, chapter 2, pages 39–128. Oxford University Press, 2000. URL: `http://www.oup.co.uk/isbn/0-19-853781-6`.

11   The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. The Univalent Foundations Program, Institute for Advanced Study, 2013. URL: `http://homotopytypetheory.org/book`.

12   Benno van den Berg and Richard Garner. Topological and simplicial models of identity types. *ACM Trans. Comput. Logic*, 13(1):3:1–3:44, January 2012. `doi:10.1145/2071368.2071371`.

13   Vladimir Voevodsky. A C-system defined by a universe category. *Theory Appl. Categ.*, 30(37):1181–1215, 2015. URL: `http://www.tac.mta.ca/tac/volumes/30/37/30-37.pdf`.

14   Vladimir Voevodsky. An experimental library of formalized mathematics based on the univalent foundations. *Math. Structures Comput. Sci.*, 25(5):1278–1294, 2015. `doi:10.1017/S0960129514000577`.

15   Vladimir Voevodsky. Subsystems and regular quotients of C-systems. In *Conference on Mathematics and its Applications, (Kuwait City, 2014)*, number 658 in Contemporary Mathematics, pages 127–137, 2016. URL: `http://arxiv.org/abs/1406.7413`.

16   Vladimir Voevodsky, Benedikt Ahrens, Daniel Grayson, et al. *UniMath*: Univalent Mathematics. URL: `https://github.com/UniMat`.