HAL
open science

Displayed Categories

Benedikt Ahrens, Peter Lefanu Lumsdaine

To cite this version:

Benedikt Ahrens, Peter Lefanu Lumsdaine. Displayed Categories. 2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017), Sep 2017, Oxford, United Kingdom. 10.4230/LIPIcs.FSCD.2017.5 . hal-01584770

HAL Id: hal-01584770

https://inria.hal.science/hal-01584770v1

Submitted on 9 Sep 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Displayed Categories*

Benedikt Ahrens[1] and Peter LeFanu Lumsdaine[2]

1 **Inria, Nantes, France**
`benedikt.ahrens@inria.fr`
2 **Department of Mathematics, Stockholm University, Stockholm, Sweden**
`p.l.lumsdaine@math.su.se`

Abstract

We introduce and develop the notion of *displayed categories*.

A displayed category over a category $\mathcal{C}$ is equivalent to 'a category $\mathcal{D}$ and functor $F : \mathcal{D} \to \mathcal{C}$', but instead of having a single collection of 'objects of $\mathcal{D}$' with a map to the objects of $\mathcal{C}$, the objects are given as a family indexed by objects of $\mathcal{C}$, and similarly for the morphisms. This encapsulates a common way of building categories in practice, by starting with an existing category and adding extra data/properties to the objects and morphisms.

The interest of this seemingly trivial reformulation is that various properties of functors are more naturally defined as properties of the corresponding displayed categories. Grothendieck fibrations, for example, when defined as certain functors, use equality on objects in their definition. When defined instead as certain displayed categories, no reference to equality on objects is required. Moreover, almost all examples of fibrations in nature are, in fact, categories whose standard construction can be seen as going via displayed categories.

We therefore propose displayed categories as a basis for the development of fibrations in the type-theoretic setting, and similarly for various other notions whose classical definitions involve equality on objects.

Besides giving a conceptual clarification of such issues, displayed categories also provide a powerful tool in computer formalisation, unifying and abstracting common constructions and proof techniques of category theory, and enabling modular reasoning about categories of multi-component structures. As such, most of the material of this article has been formalised in Coq over the `UniMath` library, with the aim of providing a practical library for use in further developments.

1998 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases Category theory, Dependent type theory, Computer proof assistants, Coq, Univalent mathematics

Digital Object Identifier 10.4230/LIPIcs.FSCD.2017.5

1 Introduction

It is often said that reference to equality of objects of categories is in general both undesirable and unnecessary.

* This material is based upon work supported by the National Science Foundation under agreement No. DMS-1128155 and CMU 1150129-338510. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work has partly been funded by the CoqHoTT ERC Grant 637339, and by the Swedish Research Council (VR) Grant 2015-03835 *Constructive and category-theoretic foundations of mathematics*.

© Benedikt Ahrens and Peter LeFanu Lumsdaine;
licensed under Creative Commons License CC-BY
2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017).
Editor: Dale Miller; Article No. 5; pp. 5:1–5:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

There are some topics, however, whose development does appear to require it. One example often given is the definition of *(Grothendieck) fibrations* (and their relatives): functors $p : \mathcal{D} \to \mathcal{C}$ equipped with a lifting property providing (among other things) an object d of $\mathcal{D}$ such that pd is equal to a previously given object c of $\mathcal{C}$. A similar example is the property of *creating limits*; see [8, Remark 5.3.7] for an explicit discussion of this example.[1]

In examples of fibrations (or creation of limits), however, one virtually never has cause to speak explicitly of equality of objects; and equally in their basic general theory.

How is this avoidance achieved? In the general development, equality occurs only within the notion of 'objects of $\mathcal{D}$ over c', for objects c of $\mathcal{C}$. And in examples, there is almost always an obvious alternative notion of 'object $\mathcal{D}$ over c', trivially equivalent to 'objects of $\mathcal{D}$ whose projection is equal to c', but expressible without mentioning equality of objects.

Specifically, objects of $\mathcal{D}$ typically consist of objects of $\mathcal{C}$ equipped with extra structure or data; 'an object of $\mathcal{D}$ over c' is then understood to mean 'a choice of the extra data for c'. For instance, in showing that the forgetful functor $\mathsf{grp} \to \mathsf{set}$ creates limits, one doesn't construct a group and then note that its underlying set is the desired one; one simply constructs a group structure on the set.

The notion of *displayed categories* makes this explicit. A displayed category over $\mathcal{C}$ consists of a family of types $\mathcal{D}_c$ (of 'objects over c'), indexed by objects c of $\mathcal{C}$, and similarly sets of morphisms indexed by morphisms of $\mathcal{C}$, along with suitable composition and identity operations to ensure that the total collections of objects and morphisms form a category (with a projection functor to $\mathcal{C}$). This is entirely equivalent to the data of a category with a functor to $\mathcal{C}$, just as 'a family of sets indexed by X' is equivalent to 'a set with a function to X'.

If fibrationhood (or creating limits, etc.) is now defined not as a property of a functor but instead as a property of a displayed category, no mention of equality of objects is required. Equality of objects is used only for turning an arbitrary functor into a displayed category; but this is rarely needed in practice, since most natural examples of fibrations already arise from displayed categories. For instance, the standard definition of the category of groups can be read as the total category of the displayed category over set whose objects are group structures on sets.

We therefore propose that displayed categories should be taken as a basis for the development of fibrations (and creation of limits, and various other notions), in particular in the type-theoretic setting, where dealing with equality on objects is more practically problematic than in classical foundations.

We do not believe we are introducing something mathematically novel here; we are simply making explicit an aspect of how mathematicians already deal with certain kinds of examples in practice. The payoffs, however, are twofold.

Firstly, since this concept has been previously un-articulated, it has not been consistently appreciated that it resolves the 'problematic' issue of fibrations (and various other notions) apparently requiring use of equality on objects. Besides providing conceptual clarification, this should help in future work with disentangling which constructions genuinely *do* require use of equality on objects, and may require extra work or assumptions to develop in type-theoretic settings.

Secondly, by making this common informal technique precise, we make it available for use in computer formalisation, where a difference between the formal definitions given and the

[1] Both of these definitions have analogues in which the equality is weakened to isomorphism; but the strict versions have nonetheless remained in more general currency.

approach used in practice cannot be so blithely elided as it can for human mathematicians. To that end, most constructions and results of the present paper have been formalised in the proof assistant Coq, over the `UniMath` library, with the goal of providing a practical library for re-use in further developments.

While that development is in univalent type theory, for the present article we work in an 'agnostic' logical setting: all results may be understood either in type theory with univalence, or in a classical set-theoretic foundation.

1.1 Outline

We begin, in § 2, by laying out precisely the agnostic type-theoretic foundation in which we work, and recalling the basic background of category theory in this setting.

In § 3, we then set up the core definitions and constructions of displayed categories, along with various examples which will be used as running illustrations through the following sections.

Following this, in § 4, we consider *creation of limits*, a first simple example of a classical property of functors which can be stated and developed more cleanly as a property of displayed categories.

In § 5, we move to the central such example: fibrations, along with their cousins isofibrations, discrete fibrations, and so on. We set out the displayed-category definitions of these, and set out some of the basic results and constructions over this definition.

This provides a basis for the theory and application of fibrations in the type-theoretic setting. In § 6, we use this to define *comprehension categories* – a categorical axiomatisation of type dependency – bringing together several of the tools set up in earlier sections.

Finally, in § 7, we consider *univalence* of displayed categories. The main result there is that the total category of a univalent displayed category (suitably defined) over a univalent base category is univalent. This generalises the *structure identity principle* of [9, §9.8].

Throughout the article, many proofs would be almost word-for-word the same as standard proofs of the corresponding results about classically-defined fibrations (resp. creation of limits, etc), since displayed categories are exactly a formal abstraction of the language already used in such proofs. We therefore omit these, to avoid repeating well-known material – but we invite the reader to recall the standard proofs, and see how directly they transfer.

Most other proofs are also either omitted or just briefly sketched, if they are either routine, available in detail in the formalisation, or both.

We follow Voevodsky in writing 'Problem', rather than 'Theorem', 'Proposition', etc., to denote proof-relevant results.

1.2 Formalisation

Most results of the present article have been formalised in Coq, over the `UniMath` library of Voevodsky et al. [10].

The primary goal of the formalisation is to provide a library for use in further work. We have therefore focused in it on the results and constructions we expect to be useful in such work. In particular, we have not formalised the comparisons with classical definitions: these are not needed for the development of fibrations etc. based on displayed categories, but rather form a justification that this approach is 'correct' from a classical point of view.

The full code of the formalisation is available on Github, at `https://github.com/UniMath/TypeTheory`, in the subdirectory `TypeTheory/Displayed_Cats`. Instructions for

installation and compilation can be found in the repository's `README.md` file. A browsable version is available at `https://unimath.github.io/TypeTheory/`.

Definitions, constructions, and results included in the formalisation are labelled below with their corresponding identifiers, as e.g. `disp_cat`. The main branch of the formalisation remains under development, so these identifiers may change in future. However, the version current at time of writing will remain permanently available under the tag `2017-displayed-cats-fscd`.

The material of the present paper constitutes about 5,000 lines of code.

2 Background

2.1 Logical setting

All the material of the present paper may be understood either in the univalent setting, or in classical set-theoretic foundations.

Precisely, our background setting throughout is Martin-Löf's intensional type theory, with: Σ-types, with the strong η rule; identity types; Π-types, also with η, and functional extensionality; $\mathsf{0}$, $\mathsf{1}$, $\mathsf{2}$, and N; propositional truncation; and two universes closed under all these constructions.

This setting is *agnostic* about equality on types: it assumes neither univalence, nor UIP. It is therefore expected to be compatible both with the addition of univalence, and with the interpretation of types as classical sets.

Some type-theoretic issues trivialise under the classical reading – for instance, the consideration of transport along equalities, which is unnecessary classically. Some topics also become less interesting there, as they admit only degenerate examples: in particular, the material on univalent categories. The reader interested only in the classical setting may therefore ignore these aspects.

2.2 Type-theoretic background

We mostly follow the terminology standardised in the HoTT book [9]. A brief, but sufficient, overview is given in [2], among other places.

We depart from it (and type-theoretic tradition in general) in writing just *existence* for what is called *mere existence* in [9], since this is what corresponds (under the interpretation of types as sets) to the standard mathematical usage of *existence*.

We will make frequent use of *dependent paths/equalities* [9, §6.2] Specifically, in a type family B_x indexed by $x : A$, we will write dependent equalities as e.g. $p : y_0 =_e y_1$, where $e : x_0 =_A x_1$ and $y_i : B_{x_i}$. We omit explicit mention of the type family B, since it will always be clear from context. The base A will often moreover be a set, in which case $y_0 =_e y_1$ does not depend on the base path e, so we suppress this and write just $y_0 =_* y_1$.

We will mostly ignore size issues; we would really like to think of everything as being universe-polymorphic. For concreteness, however, type may be understood always as the smaller of our two assumed universes, with types in this universe referred to as *small*, and similarly set as meaning the type or category of small sets, and so on.

2.3 Categories

We mostly follow the approach to category theory in the type-theoretic setting established in [2]. We depart however from their terminology, writing *categories* for what [2] calls precat-

egories (since it is these that becomes the standard definition under the set interpretation), and writing *univalent categories* for what [2] calls categories.

Specifically, in a *category* $\mathcal{C}$, the hom-sets $\mathcal{C}(a,b)$ are required to be sets, but the type $\mathcal{C}_0$ of objects is allowed to be an arbitrary type. A category $\mathcal{C}$ is *univalent* if for all $a,b:\mathcal{C}$, the canonical map $\mathsf{idtoiso}_{a,b} : (a=b) \to \mathsf{Iso}_{\mathcal{C}}(a,b)$ is an equivalence: informally, if 'equality of objects is isomorphism in $\mathcal{C}$'.

Following the `UniMath` library, 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 Displayed categories

In this section, we set out the basic definitions of displayed categories, displayed functors, and displayed natural transformations, along with key constructions on them, and examples which will act as running illustrations throughout the paper.

3.1 Definition and examples

▶ **Definition 1** (`disp_cat`). Given a category $\mathcal{C}$, a *displayed category* $\mathcal{D}$ *over* $\mathcal{C}$ consists of

- for each object $c:\mathcal{C}$, a type $\mathcal{D}_c$ of 'objects over c';
- for each morphism $f : a \to b$ of $\mathcal{C}$, $x : D_a$ and $y : D_b$, a set of 'morphisms from x to y over f', denoted $\mathsf{hom}_f(x,y)$ or $x \to_f y$;
- for each $c:\mathcal{C}$ and $x:\mathcal{D}_c$, a morphism $1_x : x \to_{1_c} x$;
- for all morphisms $f : a \to b$ and $g : b \to c$ in $\mathcal{C}$ and objects $x : \mathcal{D}_a$ and $y : \mathcal{D}_b$ and $z : \mathcal{D}_c$, a function

$$\mathsf{hom}_f(x,y) \times \mathsf{hom}_g(y,z) \to \mathsf{hom}_{f\cdot g}(x,z)\,,$$

denoted like ordinary composition by $(\bar{f},\bar{g}) \mapsto \bar{f}\cdot\bar{g} : x \to_{f\cdot g} z$, where $\bar{f} : x \to_f y$ and $\bar{g} : y \to_g z$,

such that, for all suitable inputs, we have:

- $\bar{f}\cdot 1_y =_* \bar{f}$,
- $1_x \cdot \bar{f} =_* \bar{f}$,
- $\bar{f}\cdot(\bar{g}\cdot\bar{h}) =_* (\bar{f}\cdot\bar{g})\cdot\bar{h}$.

Note that the axioms are all *dependent* equalities, over equalities of morphisms in $\mathcal{C}$: for instance, if $\bar{f} : x \to_f y$, then $\bar{f}\cdot 1_y : x \to_{f\cdot 1_b} y$, so the displayed right unit axiom $\bar{f}\cdot 1_y =_* \bar{f}$ is over the ordinary right unit axiom $f \cdot 1_b = f$ of $\mathcal{C}$. This will be typical in what follows: equations in displayed categories will be modulo analogous equations in $\mathcal{C}$, which we will usually suppress without further comment.

As promised, any displayed category over $\mathcal{C}$ induces an ordinary category over $\mathcal{C}$:

▶ **Definition 2** (`total_category`, `pr1_category`). Let $\mathcal{D}$ be a displayed category $\mathcal{D}$ over $\mathcal{C}$. The *total category* of $\mathcal{D}$, written $\int\mathcal{D}$ (or $\int_{\mathcal{C}}\mathcal{D}$, or $\int_{c:\mathcal{C}}\mathcal{D}_c$) is defined as follows:

- objects are pairs (a,x) where $a:\mathcal{C}$ and $x:\mathcal{D}_a$; in other words, the type of objects is

$$(\textstyle\int\mathcal{D})_0 := \sum_{a:\mathcal{C}} \mathcal{D}_a\,,$$

- morphisms $(a,x) \to (b,y)$ are pairs $(f,\bar{f})$ where $f : a \to b$ and $\bar{f} : x \to_f y$; in other words,

$$(\textstyle\int\mathcal{D})\big((a,x),(b,y)\big) := \sum_{f:\mathcal{C}(a,b)} \mathsf{hom}_f(x,y)\,,$$

- composition and identities in $\int\mathcal{D}$ are induced straightforwardly from those of $\mathcal{C}$ and $\mathcal{D}$, and similarly for the axioms.

The evident *forgetful functor* $\pi_1^{\mathcal{D}} : \int\mathcal{D} \to \mathcal{C}$ simply takes the first projection, on both objects and morphisms.

▶ **Example 3** (`disp_grp`). The *category of groups* can be defined as the total category of a displayed category grp, over set:

- grp_X is the set of group structures on the set X;
- given a function $f : X \to Y$ and group structures (μ, e) on X and (μ', e') on Y, $\mathsf{hom}_f((\mu, e), (\mu', e'))$ is (the type representing) the proposition 'f is a homorphism with respect to (μ, e), (μ', e')';
- the displayed composition 'operation' is the fact that the composite of homomorphisms is a homomorphism; similarly for the identity;
- the axioms are trivial, since the displayed hom-sets are propositions.

The total category of this is exactly the usual category of groups.

▶ **Example 4** (`disp_top`). The category of topological spaces can be defined as the total category of the displayed category top over set:

- top_X is the set of topologies on the set X;
- given a function $f : X \to Y$ and topologies T on X and T' on Y, $\mathsf{hom}_f(T, T')$ is the proposition 'f is continuous with respect to X and Y'.

▶ **Example 5** (`disp_over_unit`). Any category can be viewed as a displayed category over the terminal category.

▶ **Example 6** (`disp_full_sub`). Let $P : \mathcal{C} \to \mathsf{type}$ be a (type-valued) predicate on the objects of $\mathcal{C}$. Then there is an associated displayed category, with object family exactly P, and with $\mathsf{hom}_f(y, y') := 1$ for all $f : x \to x'$, $y : Px$, and $y' : Px'$. The operations and axioms are trivial.

Its total category is the full subcategory of $\mathcal{C}$ on objects satisfying the predicate P.

The preceding few examples are instances of a common situation:

▶ **Proposition 7** (`full_pr1_category`, `faithful_pr1_category`, `fully_faithful_pr1_category`). *Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. If every displayed hom-set $\mathsf{hom}_f(y, y')$ of $\mathcal{D}$ is a proposition (resp. inhabited, contractible) then $\pi_1 : \int\mathcal{D} \to \mathcal{C}$ is faithful (full, fully faithful).* ◀

Besides the total category, a displayed category also possesses *fibre* categories:

▶ **Definition 8** (`fiber_category`). Given a displayed category $\mathcal{D}$ over $\mathcal{C}$, and an object $c : \mathcal{C}$, define the *fibre category* $\mathcal{D}_c$ of $\mathcal{D}$ over c as the category with objects $\mathcal{D}_c$ and with morphisms $\mathsf{hom}(x, y) := \mathsf{hom}_{1_c}(x, y)$. Composition and identity are induced by that of $\mathcal{D}$.

In general these may not be so well behaved as the total category; they will typically be interesting and well-behaved just when $\mathcal{D}$ is an isofibration (Def. 30).

▶ Remark. In choosing notation and terminology for examples of displayed categories, a question arises: should one name displayed categories according to their total category, or according to their fibres?

This problem arises already with fibrations in the classical setting; so we follow for the most part the usual compromises used there. Specifically, when a given total category has a particularly canonical displaying—for example, groups displayed over sets—we will use the

same name for the displayed category and its total category, so for example $G : \mathsf{grp}$ denotes a group, while $(\mu, e) : \mathsf{grp}_X$ is a group structure on X. On the other hand, when different displayed categories have equivalent total categories—for instance, the product $\mathcal{C} \times \mathcal{C}'$ may be displayed over either $\mathcal{C}$ or $\mathcal{C}'$—then we will adopt different notation to distinguish these, usually based on the resulting fibre categories.

Other examples we will meet below include:

- any product $\mathcal{C} \times \mathcal{C}'$, displayed over its first factor as $\mathsf{const}_{\mathcal{C}}\, \mathcal{C}'$ (Example 14);
- the arrow category $\mathcal{C}^{\to}$, in several ways: displayed over $\mathcal{C}^2$, with fibres hom-sets; and displayed over $\mathcal{C}$, with fibres either the slices or the coslices of $\mathcal{C}$ (Example 16);
- categories of algebras for endofunctors and monads (Examples 17, 18).

We postpone their full definitions until we have a few more tools set up.

▶ Remark. Definitions very closely equivalent to that of displayed categories—that is, "fibred" presentations of a functor into a fixed base category—can also be recovered from more sophisticated categorical structures in several ways: as a lax 2-functor or double functor from the base category into the bicategory or double category of spans, or as a normal lax 2-functor/double functor into the bicategory/double category of distributors (as observed by Bénabou in [4, §7]), or as a double profunctor from the base category to the terminal double category.[2]

3.2 Displayed functors and natural transformations

Another occurrence of equality of objects is in various definitions where diagrams of functors are assumed to commute on the nose. For instance, comprehension categories involve a fibration $p : \mathcal{T} \to \mathcal{C}$, and a functor $\chi : \mathcal{T} \to \mathcal{C}^{\to}$, such that $\chi \cdot \mathsf{cod} = p$ [7, Theorem 9.3.4]; similar conditions occur in the definition of functorial factorisations, in the theory of weak factorisation systems (among many other places). It is typically clear that the definitions could also be phrased without equality of objects, at some cost in concision and clarity.

However, they are almost always of the form $G \cdot \pi_1^{\mathcal{D}} = F$, where G is a functor into the total category of some displayed category, and F is a previously-given functor into the base. Indeed, they are often of the more specialised form $G \cdot \pi_1^{\mathcal{D}} = F \cdot \pi_1^{\mathcal{D}'}$. By axiomatising this situation, as *displayed functors* over functors into the base, such definitions can be stated without equality of objects, with no loss of concision.

▶ **Definition 9** (`disp_functor`). Let $F : \mathcal{C} \to \mathcal{C}'$ be a functor, and $\mathcal{D}$, $\mathcal{D}'$ displayed categories over $\mathcal{C}$ and $\mathcal{C}'$ respectively. A *(displayed) functor G from $\mathcal{D}$ to $\mathcal{D}'$ over F* consists of:

- maps $G_c : \mathcal{D}_c \to \mathcal{D}'_{Fc}$, for each $c : \mathcal{C}$ (which we usually write just as G, omitting c); and
- maps $\mathsf{hom}_f(x, y) \to \mathsf{hom}_{Ff}(Gx, Gy)$, for each $f : c \to c'$ in $\mathcal{C}$;
- satisfying the evident dependent analogues of the usual functor laws.

A displayed functor G over F straightforwardly induces a *total functor* between total categories, written $\int G : \int \mathcal{D} \to \int \mathcal{D}'$, such that $\int G \cdot \pi_1^{\mathcal{D}'} = \pi_1^{\mathcal{D}} \cdot F$. Indeed, displayed functors are precisely equivalent to such functors between total categories. We often therefore call the total functor just G.

Similarly, a functor G over F induces *fibre functors* $G_c : \mathcal{D}_c \to \mathcal{D}'_{Fc}$, for each $c : \mathcal{C}$.

A useful special case is when F is the identity functor of $\mathcal{C}$, in which case we call G just a functor over $\mathcal{C}$; this is precisely equivalent to a functor between the total categories strictly over C in the usual sense.

[2] Our thanks to Mike Shulman and an anonymous referee for pointing out some of these reformulations.

▶ **Definition 10** (`disp_nat_trans`). Let $F, F' : \mathcal{C} \to \mathcal{C}'$ be functors, $\alpha : F \to F'$ a natural transformation, and G and G' displayed functors from $\mathcal{D}$ to $\mathcal{D}'$ over F and F' respectively. A *displayed natural transformation β from G to G' over α* consists of

- for each $c : \mathcal{C}$ and $x : \mathcal{D}_c$, a morphism $\beta_c(x) : \mathsf{hom}_{\alpha(c)}(G(x), G'(x))$
- such that for any $f : \mathcal{C}(c, c')$ and $\bar{f} : \mathsf{hom}_f(x, y)$, $G\bar{f} \cdot \alpha(c') =_* \alpha(c) \cdot G'\bar{f}$.

Just as ordinary functors and natural transormations form a functor category, their displayed versions form a displayed category over the functor category between the bases:

▶ **Definition 11** (`disp_functor_cat`). Given categories $\mathcal{C}$ and $\mathcal{C}'$, and displayed categories $\mathcal{D}$ and $\mathcal{D}'$ over $\mathcal{C}$ and $\mathcal{C}'$ respectively, there is a displayed category $[\mathcal{D}, \mathcal{D}']$ over $[\mathcal{C}, \mathcal{C}']$, defined as follows:

- objects over $F : \mathcal{C} \to \mathcal{C}'$ are displayed functors from $\mathcal{D}$ to $\mathcal{D}'$ over F;
- morphisms over $\alpha : F \to G$ from F' to G' are displayed natural transformations from F' to G' over α;
- composition and identity are given by pointwise composition and identity.

Displayed analogues of usual lemmas on the functor category hold; for instance:

▶ **Lemma 12** (`is_disp_functor_precat_iso_iff_pointwise_iso`). *A displayed natural transformation is an isomorphism in the displayed functor category if and only if it is an isomorphism pointwise.*

We could now go on and define *displayed adjunctions* over adjunctions between the bases, *displayed equivalences* over equivalences of the base, and so on. From these, one gets adjunctions and equivalences, respectively, of total categories. A very useful special case is that of displayed adjunctions and equivalences over the identity in the base, yielding adjunctions and equivalences of total categories leaving the first components of objects untouched.

These definitions are provided in the formalisation; indeed, the original motivation of the present work and formalisation was to have these available, in order to construct an equivalence of univalent categories between CwF-structures and split type-category structures on a fixed base category (cf. the equivalence of types of [3, Construction 19]). However, an account of this is beyond the scope of the present paper.

One may also naturally ask what structure the total collections of displayed categories and natural transformations form. We expect that they should form a bicategory when the base category is held fixed, and more generally a displayed bicategory over the bicategory of categories; but this again is beyond the scope of the present work.

3.3 Constructions on displayed categories

To efficiently construct our remaining key examples, we set up some basic general constructions on displayed categories.

▶ **Definition 13** (`reindex_disp_cat`). Let $\mathcal{D}$ be a displayed cat over $\mathcal{C}$, and $F : \mathcal{C}' \to \mathcal{C}$ a functor. Then $F^*\mathcal{D}$, the *pullback of $\mathcal{D}$ along F*, is the displayed category over $\mathcal{C}'$ defined by

- $(F^*\mathcal{D})_c := \mathcal{D}_{Fc}$
- $\mathsf{hom}_f^{F^*\mathcal{D}}(d, d') := \mathsf{hom}_{Ff}^{\mathcal{D}}(d, d')$

with the evident composition and identities. There is an evident displayed functor $F^*\mathcal{D} \to \mathcal{D}$ over F.

▶ **Example 14** (`disp_cartesian`). Given any categories $\mathcal{C}, \mathcal{C}'$, the *constant displayed category over $\mathcal{C}$ with fibre $\mathcal{C}'$*, denoted $\mathsf{const}_{\mathcal{C}}\,\mathcal{C}'$ (or just $\mathsf{const}\,\mathcal{C}'$, when $\mathcal{C}$ is implicit), is the pullback along the unique functor $\mathcal{C} \to 1$ of $\mathcal{C}'$, seen as a displayed category over 1.

There is an evident equivalence from the total category $\int_{\mathcal{C}} \mathsf{const}\,\mathcal{C}'$ to the product $\mathcal{C} \times \mathcal{C}'$, strictly over $\mathcal{C}$.

▶ **Definition 15** (`sigma_disp_cat`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$, and $\mathcal{E}$ a displayed category over $\int_{\mathcal{C}} \mathcal{D}$. The Σ-*category* of $\mathcal{E}$ over $\mathcal{D}$, denoted $\sum_{\mathcal{D}} \mathcal{E}$, is the displayed category over $\mathcal{C}$ defined as follows:

- $\left(\sum_{\mathcal{D}} \mathcal{E}\right)_x := \sum_{y:\mathcal{D}_x} \mathcal{E}_{(x,y)}$
- $\mathsf{hom}_f((y,z),(y',z')) := \sum_{\bar{f}:y \to_f y'} \mathsf{hom}_{(f,\bar{f})}(z,z')$
- operations defined componentwise from those of $\mathcal{D}$ and $\mathcal{E}$.

There is an evident equivalence of total categories $\int_{\left(\int_{\mathcal{C}} \mathcal{D}\right)} \mathcal{E} \to \int_{\mathcal{C}} \left(\sum_{\mathcal{D}} \mathcal{E}\right)$ over $\mathcal{C}$.

▶ **Example 16** (`disp_arrow`, `disp_domain`, `disp_codomain`). The arrow category has three different displayed incarnations:

1. By $\mathcal{C}^{\to}$, we mean the displayed category over $\mathcal{C} \times \mathcal{C}$ with
   - ${\mathcal{C}^{\to}}_{x,y} := \mathsf{hom}^{\mathcal{C}}(x,y)$
   - $\mathsf{hom}^{\mathcal{C}^{\to}}_{h,k}(f,g) := (f \cdot k = h \cdot g)$, i.e. the proposition that the resulting square commutes.

   As our notation suggests, the total category of this is the usual arrow category of $\mathcal{C}$.
2. Pulling this back along the canonical equivalence $\int_{\mathcal{C}} \mathsf{const}\,\mathcal{C} \to \mathcal{C} \times \mathcal{C}$, and taking the Σ-category of the result, we obtain a displayed category over $\mathcal{C}$ which we denote $-\backslash\mathcal{C}$, since its fibre categories are just the co-slices of $\mathcal{C}$. Its total category is equivalent over $\mathcal{C}$ to $\mathsf{dom} : \mathcal{C}^{\to} \to \mathcal{C}$.
3. If in the previous example, we instead pull back along the equivalence $\int_{\mathcal{C}} \mathsf{const}\,\mathcal{C} \to \mathcal{C} \times \mathcal{C}$ that swaps the two components, we get instead the displayed category of *slices* of $\mathcal{C}$, with total category equivalent over $\mathcal{C}$ to $\mathsf{cod} : \mathcal{C}^{\to} \to \mathcal{C}$.

▶ **Example 17** (`disp_cat_functor_alg`). Suppose $F : \mathcal{C} \to \mathcal{C}$ is an endofunctor. Then F-*algebras* naturally form a displayed category F-Alg over $\mathcal{C}$, with

- $F\text{-}\mathsf{Alg}_c := \mathsf{hom}^{\mathcal{C}}(Fc, c)$
- $\mathsf{hom}_f(\alpha,\beta) := (\alpha \cdot f = Ff \cdot \beta)$, i.e. the proposition that $f : a \to b$ is an algebra homomorphism $(a,\alpha) \to (b,\beta)$.

The total category is the usual category F-Alg. We will sometimes write F-EndAlg to distinguish this from categories of *monad* algebras.

▶ **Example 18** (`disp_cat_monad_alg`). Suppose (T,μ,η) is a monad on $\mathcal{C}$. The full subcategory of F-EndAlg consisting of the *monad* algebras for (T,μ,η) can be seen as a displayed category over F-EndAlg, as in Example 6. Taking the Σ-category of this yields the monad-algebras (T,μ,η)-MonAlg as a displayed category over $\mathcal{C}$.

As usual, we write just T-Alg when there is no risk of confusion.

4 Creation of limits

Creation of limits is our first example of a concept which can be profitably reformulated in terms of displayed categories.

As a property of functors, it is a standard and fruitful tool in category theory. It has however often been viewed with some mistrust for involving equalities of objects: see, for example, [8, Remark 5.3.7].

If formulated instead as a property of displayed categories, it involves no equalities of objects:

▶ **Definition 19** (`creates_limit`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$, J a graph, and F a diagram of shape J in $\int\mathcal{D}$. Given a limiting cone λ for the diagram $F \cdot \pi_1 : J \to \mathcal{C}$ in $\mathcal{C}$, with vertex $x : \mathcal{C}$, we say that $\mathcal{D}$ *creates a limit for* F *over* λ if

- there is a unique cone on F over λ; that is, a unique object $d : \mathcal{D}_x$ and family of arrows $\mu_j : \mathsf{hom}_{\lambda_j}(d, \pi_2 F(j))$ such that the pairs (λ_j, μ_j) form a cone on F in $\int\mathcal{D}$;
- and, furthermore, this unique cone $(\lambda_j, \mu_j)_{j:J}$ is limiting.

More generally, we say that $\mathcal{D}$ *creates limits of shape* $\mathcal{J}$ (or *creates small limits*, etc.) if, for any diagram F as above over $\mathcal{J}$ (resp. over any small $\mathcal{J}$), and every limiting cone λ on $F \cdot \pi_1$ in $\mathcal{C}$, $\mathcal{D}$ creates a limit for F over λ.

It is routine to check that this does indeed correspond to the standard notion:

▶ **Proposition 20.** *A displayed category $\mathcal{D}$ over a category $\mathcal{C}$ creates limits of shape $\mathcal{J}$, in our sense, if and only the functor $\pi_1^{\mathcal{D}} : \int\mathcal{D} \to \mathcal{C}$ creates limits in the classical sense.*

It of course follows immediately from this that the displayed definition implies the various standard consequences of creation of limits. In fact, however, the proofs from the displayed definition are at least as direct as the standard proofs; for instance,

▶ **Proposition 21** (`total_limits`, `pr1_preserves_limit`). *Suppose the category $\mathcal{C}$ has limits of shape $\mathcal{J}$, and the displayed category $\mathcal{D}$ over $\mathcal{C}$ creates limits of shape $\mathcal{J}$. Then $\int\mathcal{D}$ has all such limits, and $\pi_1^{\mathcal{D}} : \int\mathcal{D} \to \mathcal{C}$ preserves them.* ◀

Moreover, all the main standard examples of functors that create limits can be seen as the forgetful functors associated to displayed categories.

▶ **Example 22** (`creates_limits_functor_alg`). For any endofunctor $F : \mathcal{C} \to \mathcal{C}$, the displayed category of F-algebras over $\mathcal{C}$ creates all limits. Likewise, for any monad T on $\mathcal{C}$, the displayed category of T-algebras over $\mathcal{C}$ creates all limits.

5 Fibrations

We consider, in this section, three important variations of fibrations of categories: Grothendieck fibrations (and their dual, opfibrations); isofibrations; and discrete fibrations.

We depart from some classical literature in defining fibrations by default to be *cloven* – that is, to include an operation providing all lifts required. (This is not novel: it has been preferred also by other authors, to avoid indiscriminate use of the axiom of choice.) We distinguish the case where liftings are merely known to exist as *weak* fibrations.

5.1 Fibrations and opfibrations

▶ **Definition 23** (`is_cartesian`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. A map $\bar{f} : \mathsf{hom}_f(d', d)$ of $\mathcal{D}$ over $f : c' \to c$ is *cartesian* if for each $g : c'' \to c'$, $d'' : \mathcal{D}_{c''}$, and $\bar{h} : \mathsf{hom}_{f \cdot g}(d'', d)$, there is a unique $\bar{g} : \mathsf{hom}_g(d'', d')$ such that $\bar{g} \cdot \bar{f} =_* \bar{h}$.

▶ **Definition 24** (`cartesian_lift`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. A *cartesian lift* of $f : \mathcal{C}(c', c)$ and $d : \mathcal{D}_c$ consists of an object $d' : \mathcal{D}_{c'}$ and cartesian map $\bar{f} : \mathsf{hom}_f(d', d)$.

▶ **Definition 25** (`cleaving`, `fibration`, `weak_fibration`). A *cleaving* for a displayed category $\mathcal{D}$ over a category $\mathcal{C}$ is a function giving, for each $f : c' \to c$ and $d : \mathcal{D}_c$, a cartesian lift of f and d. A *(cloven) fibration* over $\mathcal{C}$ is a displayed category equipped with a cleaving. A *weak fibration* is a displayed category such that for each such f, d as above, there exists some cartesian lift.

All the above have evident duals: opcartesian maps and lifts, and weak/cloven opfibrations. Again, these all correspond straightforwardly to their classical versions:

▶ **Proposition 26.** *A map in a total category $\int_{\mathcal{C}} \mathcal{D}$ is cartesian in our sense (resp. opcartesian) exactly it is cartesian (opcartesian) with respect to $\pi_1^{\mathcal{D}}$ in the classical sense. A displayed category $\mathcal{D}$ is a cloven (resp. weak) fibration in our sense exactly if $\pi_1^{\mathcal{D}}$ is one in the classical sense (i.e. [8, Def. 5.3.5], read unchanged in the univalent setting).* ◀

As with the standard definition, cartesian lifts are unique up to isomorphism. Proposition 45 below shows that when $\mathcal{D}$ is univalent, they are literally unique.

An important example in our applications of interest is the arrow category:

▶ **Proposition 27** (`cartesian_iff_isPullback`)**.** *For any category $\mathcal{C}$, consider the displayed category $\mathcal{C}/-$ of slices of $\mathcal{C}$, as in Example 16.3 above. An arrow $h : f \to_k g$ in $\mathcal{C}/-$ is cartesian exactly if its associated commuting square is a pullback. The displayed category $\mathcal{C}/-$ is a weak fibration exactly if all pullbacks exist in $\mathcal{C}$, and a (cloven) fibration just if $\mathcal{C}$ has chosen pullbacks.* ◀

Finally, we transfer the definition of *split* fibrations. It seems likely to us that – as with the hom-set condition for categories – split fibrations in the type-theoretic setting should include a setness condition in order to be as useful and well-behaved as classically:

▶ **Definition 28** (`is_split`)**.** Say a fibration $\mathcal{D}$ over $\mathcal{C}$ is *split* if:

- each $\mathcal{D}_c$ is a set; and
- the chosen lifts of idenities are identities, and the chosen lift of any composite is the composite of the individual lifts.

5.2 Isofibrations

▶ **Definition 29** (`iso_disp`)**.** Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$, and $f : c \cong c'$ an isomorphism in $\mathcal{C}$.

A map $\bar{f} : \mathsf{hom}_f(x, y)$ is a *(displayed) isomorphism* if it has a 2-sided inverse, i.e. some $\bar{g} : \mathsf{hom}_{f^{-1}}(y, x)$ such that $\bar{f} \cdot \bar{g} =_* 1_x$ and $\bar{g} \cdot \bar{f} =_* 1_y$ We write $\bar{f} : x \cong_f y$.

As with ordinary isomorphisms, the inverse of a displayed isomorphism is unique.

▶ **Definition 30** (`weak_iso_fibration`, `iso_cleaving`, `iso_fibration`)**.** Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. Say $\mathcal{D}$ is a *weak isofibration* if for each isomorphism $i : c' \cong c$ in $\mathcal{C}$ and $d : \mathcal{D}_c$, there exists some object $d' : \mathcal{D}_{c'}$ and isomorphism $\bar{i} : d' \cong_i d$. An *iso-cleaving* on $\mathcal{D}$ is a function giving, for each such i, d, some such $d', \bar{i}$. A *(cloven) isofibration* over $\mathcal{C}$ is a displayed category equipped with an iso-cleaving.

▶ **Proposition 31.** *A displayed category is a weak (resp. cloven) isofibration in our sense just if its forgetful functor is one in the classical sense.* ◀

▶ **Example 32** (`iso_cleaving_functor_alg`)**.** The displayed categories of groups, topological spaces, and similar are all naturally isofibrations over set, just as classically. More generally, so are the displayed categories of algebras for endofunctors and monads.

In fact, in the univalent setting, isofibrations often come for free:

▶ **Problem 33** (`iso_cleaving_category`)**.** *Let $\mathcal{D}$ be a displayed category over a univalent category $\mathcal{C}$. Then $\mathcal{D}$ is an isofibration.*

▶ **Construction 34** (for Problem 33)**.** Since $\mathcal{C}$ is univalent, every isomorphism $i : c \cong c'$ is uniquely of the form $\mathsf{idtoiso}(e)$. To give an iso-cleaving on $\mathcal{D}$, it therefore suffices to give, for each $e : c = c'$ and $d : D_{c'}$, some lift $\bar{i} : d' \cong_{\mathsf{idtoiso}\,e} d$. By identity elimination, the case $e := 1_c$ suffices; in this case, we take $d' := d$ and $\bar{i} := 1_d$. ◀

Assuming the univalence axiom, the examples above of grp and top over set therefore come for free. However, we note them separately (and prove them directly, in the formalisation), both to show that they do not require univalence, and to have their action explicitly.

▶ Remark. As the examples given illustrate, most fibrations and isofibrations encountered in nature are categories/functors that arise as the total category/forgetful functor of a displayed category. This, we argue, supports the idea that it is natural to take the displayed-category definitions as basic for developing fibrations and related notions, especially in the type-theoretic setting.

However, not all examples are of this form. For instance, suppose $F : \mathcal{C}' \to \mathcal{C}$ is a functor of small categories that is a complemented inclusion on objects; then the precomposition functor $F^* : \widehat{C} \to \widehat{C'}$ between their presheaf categories is an isofibration. However, in the classical setting, $\widehat{C}$ is not literally the total category of any displayed category over $\widehat{C'}$ (though it is of course isomorphic to one).

5.3 Discrete fibrations

▶ **Definition 35** (`is_discrete_fibration`)**.** Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. Say that $\mathcal{D}$ is *a discrete fibration* if

- for each $c : \mathcal{C}$, the type $\mathcal{D}_c$ is a set; and
- for any $f : \mathcal{C}(c', c)$ and $d : \mathcal{D}_c$, there is a unique $d' : \mathcal{D}_{c'}$ and $\bar{f} : \mathsf{hom}_f(d', d)$.

These lifts are automatically cartesian; so any discrete fibration is canonically a fibration (`fibration_from_discrete_fibration`), and is moreover split (`is_split_fibration_from_discrete_fibration`).

Thanks to the setness condition, discrete fibrations over a fixed base category $\mathcal{C}$ and displayed functors between them form a category; and, just as classically, we have:

▶ **Problem 36** (`forms_equivalence_disc_fib`)**.** *For any category $\mathcal{C}$, there is a (strong) equivalence of categories between $\widehat{\mathcal{C}}$ and the category of discrete fibrations over $\mathcal{C}$.* ◀

For a presheaf P on $\mathcal{C}$, the classical category of elements of P is the total category of the displayed discrete fibration given by the above equivalence.

6 Comprehension categories

We now turn briefly to *comprehension categories* and *categories with attributes*, just as a glimpse of the applications in semantics of type theory which provided the proximate motivation for the present development.

▶ **Definition 37** (`comprehension_cat_structure`)**.** A *comprehension category* consists of a category $\mathcal{C}$, a fibration $\mathcal{T}$ over $\mathcal{C}$, and a functor $\chi : \mathcal{T} \to \mathcal{C}/-$ over $\mathcal{C}$ (the 'comprehension') preserving cartesian arrows.

Taking total categories and functors, these form a strictly commuting triangle:

$$\begin{tikzcd} \int \mathcal{T} \arrow[rr, "\chi"] \arrow[dr, "\pi_1"'] & & \int_{c:\mathcal{C}} \mathcal{C}/c \arrow[dl, "\pi_1"] \\ & \mathcal{C} & \end{tikzcd}$$

As such, this is just a rephrasing of the standard definition [7, Theorem 9.3.4], but avoiding mention of equality on objects.[3]

▶ **Definition 38.** A *split type-category* (aka *category with attributes*) consists of a category $\mathcal{C}$; a presheaf Ty on $\mathcal{C}$; an operation assigning to each $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$ an object and map $\pi_A : \Gamma.A \to A$; and operations giving for each $f : \Gamma' \to \Gamma$ and $A : \mathsf{Ty}(\Gamma)$ a map $f.A$ exhibiting π_{f^*A} as a pullback of π_A.

▶ **Problem 39** ([5, Thm. 2.3]). *Any category with attributes induces a comprehension category with the same base.*

▶ **Construction 40** (for Problem 39). The equivalence of Problem 36 turns Ty into a (discrete) fibration. The operations $\Gamma.A$, π_A, and $f.A$ provide the action on objects and arrows of the comprehension functor; while the pullback condition, combined with Proposition 27, ensures that it preserves cartesian maps. ◀

7 Univalence and the Structure Identity Principle

7.1 Displayed univalence

▶ **Definition 41** (`idtoiso_disp`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. Given $c, c' : \mathcal{C}$, $e : c = c'$, $d : \mathcal{D}_c$, $d' : \mathcal{D}_{c'}$, and $e' : d =_e d'$, we write $\mathsf{idtoiso}(e, e') : d \cong_{\mathsf{idtoiso}(e)} d'$ for the canonical displayed isomorphism obtained by identity elimination on e, e'.

Note that we overload the notation $\mathsf{idtoiso}$, using it for both ordinary and displayed categories.

▶ **Definition 42** (`is_univalent_disp`). Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. Say that $\mathcal{D}$ is *univalent* if for any $c, c' : \mathcal{C}$ and $e : c = c'$ and $d : \mathcal{D}_c$ and $d' : \mathcal{D}_{c'}$, the above map $(d =_e d') \to \mathsf{Iso}_{\mathsf{idtoiso}(e)}(d, d')$ is an equivalence.

To verify univalence of a displayed category, it clearly suffices to prove the condition just in the case where e is reflexivity. But displayed isomorphisms over identities are just isomorphisms in the fibre categories, so we have:

▶ **Proposition 43** (`is_univalent_disp_iff_fibers_are_univalent`). *Let $\mathcal{D}$ be a displayed category over $\mathcal{C}$. Then $\mathcal{D}$ is univalent exactly if each of its fibre categories is univalent.*

The key practical application of displayed univalence is in proving that complex categories built up using displayed categories are univalent:

▶ **Theorem 44** (`is_univalent_total_category`). *Let $\mathcal{C}$ be a univalent category, and let $\mathcal{D}$ be a univalent displayed category over $\mathcal{C}$. Then the total category $\int \mathcal{D}$ is univalent.* ◀

However, displayed univalence is a meaningful notion even when the base is not known to be univalent; one has, for instance:

[3] Jacobs' original formulation [6, Definition 4.1] is slightly different.

▶ **Proposition 45** (`isaprop_cartesian_lifts`, `univalent_fibration_is_cloven`). *Let $\mathcal{D}$ be a* univalent *displayed category over $\mathcal{C}$. For any $f : c' \to c$ and $d : \mathcal{D}_c$, if a cartesian lift $(d', \bar{f})$ of f and d exists, then it is unique; that is, the type of cartesian lifts is a proposition. More generally, if $\mathcal{D}$ is a weak (iso-)fibration, then it possesses a unique (iso-)cleaving.*

Proof. The usual classical argument shows that cartesian lifts are unique up to isomorphism. By univalence of $\mathcal{D}$, it follows that they are literally unique.

It follows that the type of (iso-)cleavings of $\mathcal{D}$ is a proposition; and that whenever a suitable lift is known to exist, one can be chosen. Putting these together, the proposition follows. ◀

Similarly, as for ordinary categories, univalence bounds the h-level of the types of objects:

▶ **Proposition 46** (`univalent_disp_cat_has_groupoid_obs`). *Let $\mathcal{D}$ be a univalent displayed category over $\mathcal{C}$. Then for each $c : \mathcal{C}$, the type of objects $\mathcal{D}_c$ is a 1-type.* ◀

7.2 Structure Identity Principle

Theorem 44 generalizes an early-noted consequence of univalence, the so-called *structure identity principle*, as formulated by Aczel. We recall that here for comparison.

▶ **Definition 47** ([9, Def. 9.8.1]). A *standard notion of structure* on a category $\mathcal{C}$ consists of:

1. for each $c : \mathcal{C}$, a type $P(c)$;
2. for each $c, c' : \mathcal{C}$ and $\alpha : P(c)$ and $\beta : P(c')$ and $f : \mathcal{C}(c, c')$, a proposition $H_{\alpha,\beta}(f)$;
3. such that H is suitably closed under composition and identity; and
4. for each $c : \mathcal{C}$, the preorder on $P(c)$ defined by setting $\alpha \leq \alpha'$ if $H_{\alpha,\alpha'}(1_c)$ is a poset.

Items 1–3 can immediately be read as providing an associated displayed category over $\mathcal{C}$ (`disp_cat_from_SIP_data`), whose displayed hom-sets are propositions. The category of (P, H)*-structures*, as defined in [9], is precisely the total category of this displayed category.

With a little thought, item 4 can then be seen as saying that this displayed category is univalent (`is_univalent_disp_from_SIP_data`). Theorem 44 then immediately implies:

▶ **Corollary 48** ([9, Theorem 9.8.2]). *Given a standard notion of structure (P, H) on $\mathcal{C}$, if $\mathcal{C}$ is univalent, then so is the category of (P, H)-structures on $\mathcal{C}$.*

▶ **Example 49** (`is_univalent_disp_functor_alg`). The displayed categories of algebras for an endofunctor or monad (Examples 17, 18) arise from standard notions of structure, and so are univalent.

▶ Remark. The displayed categories arising in this way – univalent, and with all hom-sets propositions – may also be compared to the *amnestic* concrete categories of [1, Def. 3.27(4)].

8 Conclusions

We have introduced displayed categories, and set up their basic theory, along with key examples and applications.

The applications fall into two main groups:

- rephrasing classical definitions to avoid referring to equality of objects;
- allowing categories of multi-component structures, and maps between such categories, to be constucted and reasoned about in a modular, stage-by-stage fashion.

In this paper, we have focused more on the former – for instance, the use of displayed categories as a basis for the development of fibrations in the type-theoretic setting.

We have seen less of the latter, since it is typically tied to specific more involved applications. However, in our own further work (for instance, on the structures considered in [3]), we have found this at least as significant as a payoff of the present work.

Theorem 44, giving univalence of the total category, is especially valuable. Naïve approaches to proving univalence quickly become quite cumbersome even for categories of only moderately complex structures, such as groups. The issue is that identities between such structures translate to a tuples of identities between the components, where the identities of later components are usually heterogeneous, involving accumulated transports along the identities between earlier components.

The displayed-category approach avoids this; one need only work 'fibrewise', over each component in turn. All the necessary wrangling of transports is dealt with once and for all in the proof of Theorem 44.

An instance of this is the proof of univalence of the category of CwF-structures over a fixed univalent base category. Details are beyond the scope of the present article; but it is available in the formalisation as `is_univalent_term_fun_structure`.

Further work

In the present article and formalisation, we have explored only the basic theory and applications of displayed categories. There are many clear directions for further work:

- In [3], we have started a project of giving careful comparisons between the various categorical structures used for semantics of type theory. We touched on this project in Section 6. In forthcoming work, we plan to give full comparisons between categories of such structures, including comprehension categories, type-categories (not necessarily split), and categories with display maps.
- The material on creation of limits in Section 4 should be generalised to a more permissive notion of *displayed limits*, to cover a broader range of examples.
- In the formalisation (though not the article) we study displayed adjunctions and equivalences over a fixed base, and show that these induce adjunctions and equivalences between total categories and fibre categories. This should be generalised to displayed adjunctions/equivalences over adjunctions/equivalences in the base.
- Generally, one should be able to assemble displayed categories into a displayed bicategory over the bicategory of categories. Of course, this would require defining displayed bicategories, and developing the basic theory of bicategories in the type-theoretic setting.
- Displayed categories should also be vieweable as forming some 2-dimensional analogue of a comprehension category, with displayed categories being the 'dependent types' over a base category 'context'. This would provide a new potential guiding example for the 'directed type theory' that various authors have started to explore in recent work.

Acknowledgements. We would like to thank Mike Shulman and the participants of the Stockholm Logic Seminar for very helpful feedback on this article and the work it reports on.

We want to thank the actors of the EUTypes COST Action CA 15123 for providing funding for a research visit in the course of which this article was authored.

References

1. Jiří Adámek, Horst Herrlich, and George E. Strecker. *Abstract and concrete categories: The joy of cats*. Pure and Applied Mathematics (New York). John Wiley & Sons, Inc., New York, 1990. URL: http://www.tac.mta.ca/tac/reprints/articles/17/tr17abs.html.
2. 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.
3. Benedikt Ahrens, Peter LeFanu Lumsdaine, and Vladimir Voevodsky. Categorical structures for type theory in univalent foundations, 2017. To be published in proceedings of Computer Science Logic (CSL) 2017. arXiv:1705.04310.
4. Jean Bénabou. Distributors at work. Notes by Thomas Streicher from lectures given at TU Darmstadt, 2000. URL: http://www.mathematik.tu-darmstadt.de/~streicher/FIBR/DiWo.pdf.
5. Javier Blanco. Relating categorical approaches to type theory, 1991. Master thesis, Univ. Nijmegen.
6. Bart Jacobs. Comprehension categories and the semantics of type dependency. *Theor. Comput. Sci.*, 107(2):169–207, 1993. doi:10.1016/0304-3975(93)90169-T.
7. Bart Jacobs. *Categorical Logic and Type Theory*, volume 141 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, 1999.
8. Tom Leinster. *Basic Category Theory*, volume 143 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 2014.
9. The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. http://homotopytypetheory.org/book, Institute for Advanced Study, 2013.
10. Vladimir Voevodsky, Benedikt Ahrens, Daniel Grayson, et al. *UniMath*: Univalent Mathematics. Available at https://github.com/UniMath.