2-Coherent Internal Models of Homotopical Type Theory

Joshua Chen

School of Computer Science, University of Nottingham

Abstract • The program of *internal type theory* seeks to develop the categorical model theory of dependent type theory using the language of dependent type theory itself. In the present work we study internal homotopical type theory by relaxing the notion of a category with families (cwf) to that of a *wild*, or *precoherent higher* cwf, and determine coherence conditions that suffice to recover properties expected of models of dependent type theory. The result is a definition of a *split 2-coherent wild cwf*, which admits as instances both the syntax and the "standard model" given by a universe type. This allows us to give a straightforward internalization of the notion of a 2-coherent reflection of homotopical type theory in itself—namely as a 2-coherent wild cwf morphism from the syntax to the standard model. Our theory also easily specializes to give definitions of "low-dimensional" higher cwfs, and conjecturally includes the container higher model as a further instance.

1. Introduction

1.1. Internal type theory

Given a sufficiently expressive logical system $\mathfrak{L}$, it is interesting and productive to ask

To what extent does $\mathfrak{L}$ internalize itself?

In more detail, one seeks to develop a suitable notion of *interpreting structure*, aka *model*, of $\mathfrak{L}$, and to study the theory of such models, entirely within the language and logic of $\mathfrak{L}$ itself.[1] The techniques and perspectives granted by the study of these *inner models* have historically been used to surprising effect, e.g. to prove ZF-relative consistency of the axiom of choice and the continuum hypothesis [Göd38], or to show independence of the Whitehead problem—a statement in homological algebra about short exact sequences of abelian groups—from the traditionally accepted set theoretic foundations of mathematics [She74].

The program of *internal type theory*, first articulated by Dybjer [Dyb96], seeks to develop the same paradigm in the setting of dependent type theory[2] by studying the categorical model theory of intensional Martin-Löf type theory (MLTT) using MLTT. Among the type theory community, this is also known as *internal model theory* of type theory.

The foundational task of internal type theory is then to give a type theoretic definition of the notion of a categorical "model" of type theory, ensuring that it captures all the

[1] Immediate disclaimer: in this paper the term "model" always refers to this logical notion, and not to homotopy theoretic *model structures*. There are, of course, models of homotopy type theory in model structures.

[2] For conciseness, henceforth simply "type theory".

examples we care about. In early work, Buisse and Dybjer [BD08] formalize the type of *1-categories with families* (*1-cwfs*) [Dyb96; Hof97] in MLTT, and discuss approaches to constructing term models and the initial cwf. Later, Ahrens, Lumsdaine and Voevodsky [ALV18] consider a number of other kinds of model in the same setting,[3] and show the equivalence of 1-cwfs with (split) type categories [BG12, Definition 2.2.1; Pit95; Pit01] (aka categories with attributes [Mog91, §6; Hof97]) and representable natural transformations (aka natural models [Awo18]), via relative universes.

Two structures that we would particularly like to exhibit as internal models of a type theory are its *syntax* and, if present, its *universe type*. Altenkirch and Kaposi [AK16] show that if MLTT with uniqueness of identity proofs (UIP) is extended with *quotient inductive-inductive types* (*QIITs*) [Alt+18; KKA19] then the strongly typed term-cwf sketched by Buisse and Dybjer [BD08, §6.1] can be internally constructed, thus yielding an internal syntactic model. Even better, from this inductive construction it immediately follows that internal 1-cwfs satisfy the *initiality principle*, which posits that syntax should give rise to an initial object in an appropriate (higher) category of models. Assuming UIP also allows any universe type in MLTT to be equipped with a canonical 1-cwf structure, giving rise to an internal model known in the literature as the *standard model* [AK16, §4; Kra21a, Example 3].

Now, for any internal definition of model of MLTT which includes as instances both the syntax $\mathcal{S}$ and a universe $\mathcal{U}$, we may ask if the type of model morphisms from $\mathcal{S}$ to $\mathcal{U}$ is definable and inhabited. Elements of such a type may be viewed as *self-interpretations*, or *reflections* of MLTT in itself.[4] By the preceding discussion, in MLTT+$\mathcal{U}$+UIP+QIITs the type of 1-cwf morphisms from the syntax to the standard universe model is definable, and moreover inhabited by eliminating the syntax into the universe. We may colloquially summarize all of this by saying that any model[5] of MLTT+$\mathcal{U}$+UIP+QIITs has an inner syntactic model $\mathcal{S}$ of its MLTT fragment, as well as an internalized reflection function $\mathcal{S} \to \mathcal{U}$.

1.2. Internal homotopical type theory

Can we tell a similar story about inner models of *homotopical* MLTT, i.e. without assuming UIP? Unfortunately, the account given in the previous section depends heavily on the uniqueness of identity proofs—both for the semantics of QIITs, but also for the interpretation of the universe as an internal model. In particular, it is well known that types with their higher identities possess ∞-*groupoidal* structure in the absence of UIP [Lum10; BG11; KL21], which means that the canonical standard model on the universe $\mathcal{U}$ is no longer a 1-cwf since its substitutions $\mathcal{U}(A, B) :\equiv A \to B$ are not generally sets.

We must therefore return to the foundational task of internal type theory, and ask what a good notion of internal model of homotopical type theory might look like. This question is considered by Kraus [Kra21a] in the setting of a two-level type theory (2LTT) [ACKS23] that extends homotopical MLTT with an additional layer of *strict*, aka *non-fibrant*, types, including an equality type former satisfying UIP. With this extra structure it is now possible to define ∞-categories as semi-Segal types [CK17] having

[3] On occasion, assuming univalence.

[4] The self-interpretation—as opposed to just the internal *representation*—of type theory is a fascinating topic of study; Rendel, Ostermann and Hofer [ROH09, §2] give an excellent account of the subtle distinctions involved, albeit in the context of typed lambda calculi.

[5] Taking 1-cwfs, or any of the equivalent notions mentioned previously.

idempotent equivalences [Kra21a, §III]; to answer the question of models by defining ∞-*categories with families*; and to show that the syntax QIIT, the universe standard model, and any slice of an ∞-cwf are all ∞-cwfs.

However, models of 2LTT are somewhat strong extensions of models of MLTT, and it might therefore be argued that ∞-cwfs should not be considered to be inner models of homotopical MLTT in itself.[6] Indeed, Kraus conjectures [Kra21a, §VI] that plain homotopy type theory (HoTT) [Uni13] does not internalize itself, but instead considers the definition of ∞-cwfs to be a step towards showing that 2LTT internalizes HoTT and also itself.

Another potential approach is to work in the axiomatic variation of *simplicial homotopy type theory* [RS17; RS23] developed by Gratzer, Weinberger and Buchholtz [GWB24], in which it is possible to give a straightforward definition of $(\infty, 1)$-categories as Segal or Rezk types. However, in this theory representable presheaves are defined using additional modalities [GWB25], again making it a rather strong extension of plain HoTT.

1.3. Contributions

Thus, in the present work we stick with the original question and fix homotopical MLTT to be the theory of our outer models (i.e. the theory in which we perform our constructions). Since defining a type of ∞-categories and, a fortiori, ∞-categorical models in this theory remains an open problem, we take the notion of a *wild*, or *precoherent higher* category with families as our starting point for "internal model of homotopical MLTT" (Section 4). We then determine coherence conditions such that sufficiently coherent internal cwfs satisfy many good properties expected of any model of type theory (Section 5). In particular, we show that requiring *2-coherence* suffices to equip any wild cwf $\mathscr{C}$ with the expected cloven fibrational structure, which furthermore satisfies a coherent "splitting" property when the category of contexts of $\mathscr{C}$ is either set-level or univalent (Section 6). This generalizes the well known result that 1-cwfs are also full split comprehension categories. In order to do this, we first have to develop some wild category theory (Section 2), as well as a theory of pullbacks in wild categories (Section 3).

The benefit of our approach is that we are able to capture, in a single internally definable notion, both set-level models such as the syntax as well as untruncated higher models such as the universe. We are then able to prove results simultaneously for all such models by stating them as generally as possible in terms of coherence, without resorting to truncation assumptions or restricting to some homotopy n-level. Our theory is still easily applicable to models with "low-dimensional" higher homotopy: in particular, the simple generalization of the theory of 1-cwfs to allow the type presheaf to be valued in 1-types is (almost trivially) an instance of our theory.

1.4. Assumptions and conventions

Logical setting. Our definitions and constructions are in the setting of MLTT with Π, Σ and intensional identity types, without assuming uniqueness of identity proofs. We assume function extensionality as well as η for Π-types throughout. We do not globally

[6] Of course, even the account we gave of inner models of MLTT+UIP in the previous section does not, strictly speaking, provide models of MLTT+UIP in itself. One source of tension in inner model theory is to make the gap between the theory of the outer and inner models (the "host" theory and the "object" theory) as small as possible. How large this gap is allowed to be is typically a matter of some subjective judgment.

assume univalence, but instead take it to be a property that may or may not hold for a given universe type.

Univalent 1-category theory. Of the reader, we assume familiarity with the basics of homotopy type theory as well as the theory of 1-categories in univalent foundations, standard references for which are [Uni13] and [AKS15].

Transport. We frequently work with explicit transports over complicated path concatenations, and use the notation

$$a{\downarrow}^{P}_{e}$$

to denote the transport of an element $a : P(x)$ along an equality $e : x = y$. We will also use, without explicit comment, the equation

$$a{\downarrow}^{P}_{e}{\downarrow}^{P}_{e'} = a{\downarrow}^{P}_{e \cdot e'},$$

which holds by [Uni13, Lemma 2.3.9].

2. Wild Categories

To internalize categorical models of homotopical type theory we first have to define some type of suitable category-like structure that captures both the 1-categorical syntax as well as the ∞-categorical universe. Until the problem of defining general ∞-categorical structures using only features of homotopical MLTT is solved, we consider the approximation given by *wild*, or *precoherent higher*, categories. Variations on this notion appear a number of times in the literature and formalization libraries [CK17; HH24, §3.1; RSPB+; Bau+], typically as an intermediate stage toward defining more coherent structures. We use a particularly simple definition: they are almost precategories [AKS15, Definition 3.1], except that morphisms form *types* instead of sets.

Definition 2.0.1. *Wild categories.* A *wild* or *precoherent higher category* $\mathscr{C}$ consists of:

- A type $\mathscr{C}_0$ of **objects**.
- For all $x, y : \mathscr{C}_0$, a *type* $\mathscr{C}(x, y)$ of **morphisms** from x to y.
- An **composition** operation $\diamond$ of compatible morphisms, together with an **associator** α witnessing associativity of composition

  $$(h \diamond g) \diamond f \overset{\alpha_{f,g,h}}{=\!=} h \diamond g \diamond f$$

  for all morphisms f, g, h.
- **Identity morphisms** id_x for all $x : \mathscr{C}_0$, together with **unitors** λ and ρ witnessing the left and right identity equations

  $$\mathrm{id}_y \diamond f \overset{\lambda_f}{=\!=} f$$
  $$f \diamond \mathrm{id}_x \overset{\rho_f}{=\!=} f$$

  for all $f : \mathscr{C}(x, y)$.

Crucially, no further coherence laws on hom-types are required in the definition of a wild category. We frequently leave the indices on the associator, unitors and identity morphisms implicit.

Examples 2.0.2. *Wild categories.* We are primarily interested in two particular kinds of wild category, which are in fact completely coherent. These are distinguished by the behavior of their hom-types: the "maximally truncated" and the "nontrivially fully coherent".

The first kind consists of the wild categories whose morphisms form sets: any precategory—and hence, set-level or univalent 1-category—is immediately a wild category.

The second kind consists of the type theoretic (sub-) universes. Any universe type $\mathcal{U}$ gives rise to a wild category, also denoted $\mathcal{U}$, with objects $\mathcal{U}_0 :\equiv \mathcal{U}$, and whose hom-types $\mathcal{U}(A, B)$ are the function types $A \to B$. Composition is given by function composition and identity morphisms by identity functions. The associativity and unit laws hold definitionally, i.e. α, λ and ρ are families of trivial equations. This definition applies equally well to any reflective subuniverse [Uni13, Definition 7.7.1],[7] and in this way the m-modal types for any modality m on a universe $\mathcal{U}$ [Uni13, §7.7; RSS20, §1] form a wild category. In particular, we have wild categories whose objects are the n-types in $\mathcal{U}$, which might be seen as prototypical examples of wild $(n, 1)$-categories.

We draw diagrams in wild categories after the familiar notation, with one important difference. Since morphisms in a wild category can form arbitrary types, the commutativity of a given face of a diagram is no longer property, but *data*. Hence we denote commuting faces with the notation

$$\begin{tikzcd} x \arrow[r, bend left=50, "f"{name=U}] \arrow[r, bend right=50, "g"'{name=D}] & y \arrow[from=U, to=D, Rightarrow, phantom, "\Downarrow\gamma"] \end{tikzcd}$$

to make explicit the commutativity witness $\gamma : f = g$. As this notation suggests, equalities between morphisms in a wild category are considered to be *2-cells*, and higher equalities, *higher cells*. We stress that these cells are not explicitly axiomatized as part of the generalized algebraic theory of a wild category, but instead arise out the ambient homotopical type theory.

Remark 2.0.3. *On the semantics of wild categories.* The fact that a wild category is a proper generalization of a precategory therefore relies crucially on the homotopical nature of the identity type. This simple observation suggests that we distinguish the *categorical* and *typal* directions of a wild category: the 1-cells belong to the categorical direction, and the higher cells to the typal. This recalls the point of view—implicit in Rezk [Rez01] and explicated by Joyal and Tierney [JT07]—that Segal spaces (i.e. bisimplicial sets $\Delta^{\mathrm{op}} \times \Delta^{\mathrm{op}} \to \mathrm{Set}$ satisfying the Segal condition) have categorical and *spatial* directions. Indeed, Capriotti and Kraus [CK17, Theorem 4.9] show (a version of) this to hold.[8]

Alternatively, following Lumsdaine [Lum10] and van den Berg and Garner [BG11] in viewing types as Batanin-Leinster weak ω-groupoids, it should also be possible to make

[7] This should also straightforwardly yield a notion of "wild reflective subcategory"; we do not develop this in this paper.

[8] By interpreting their type theoretic proof in the simplicial model of HoTT [KL21].

precise the point of view that wild categories are categories "wildly enriched" in weak ω-groupoids.[9]

2.1. Common concepts

Expectedly, a large number of elementary concepts from univalent 1-category theory [AKS15] and bicategory theory [Ahr+21] straightforwardly transfer into, and are subsumed by, the wild categorical setting. We record them here, as well as in Sections 2.2 and 2.3, for completeness and future reference.

Definition 2.1.1. *Terminal objects.* An object y in a wild category $\mathscr{C}$ is *terminal* if $\mathscr{C}(x,\, y)$ is contractible for all objects $x : \mathscr{C}_0$.

Definition 2.1.2. *Sections and retractions.* The type of *sections* of a morphism $f : \mathscr{C}(x,\, y)$ in a wild category $\mathscr{C}$ is

$$\mathrm{Sect}(f) :\equiv \textstyle\sum_{(s\,:\,\mathscr{C}(y,\, x))},\ f \diamond s = \mathrm{id}.$$

Similarly, the type of *retractions* of f is

$$\mathrm{Retr}(f) :\equiv \textstyle\sum_{(r\,:\,\mathscr{C}(y,\, x))},\ r \diamond f = \mathrm{id}.$$

In contrast to 1-categories, sections and retractions in wild categories are not solely determined by their morphism component in general, but also by the identification of the section-retraction composite with the identity.

Definition 2.1.3. *Whiskering.* Given an equality $\gamma : g = g'$ of morphisms $g, g' : \mathscr{C}(x,\, y)$, for any morphism $f : \mathscr{C}(w,\, x)$ the *right whiskering* $(\gamma * f)$ of γ with f is the canonical equality

$$\mathrm{ap}\,(\_ \diamond f)\,\gamma : g \diamond f = g' \diamond f,$$

and for any $h : \mathscr{C}(y,\, z)$ the *left whiskering* $(h * \gamma)$ of γ with h is the equality

$$\mathrm{ap}\,(h \diamond \_)\,\gamma : h \diamond g = h \diamond g'.$$

Proposition 2.1.4. *Properties of whiskering.* By induction, the following equations hold for right whiskering:

$$\begin{gathered}
\mathrm{refl} * f \equiv \mathrm{refl},\\
\gamma * \mathrm{id} = \rho \cdot \gamma \cdot \rho^{-1},\\
(\gamma * f)^{-1} = \gamma^{-1} * f,\\
(\gamma \cdot \delta) * f = (\gamma * f) \cdot (\delta * f),
\end{gathered}$$

as well as the analogous equations for left whiskering. We also have the following associativity laws expressing "naturality" of α,

$$\begin{aligned}
g * (f * \gamma) &= \alpha^{-1} \cdot ((g \diamond f) * \gamma) \cdot \alpha,\\
(\gamma * g) * f &= \alpha \cdot (\gamma * (g \diamond f)) \cdot \alpha^{-1},\\
(g * \gamma) * f &= \alpha \cdot (g * (\gamma * f)) \cdot \alpha^{-1},
\end{aligned}$$

[9] That is, weakly enriched, but without requiring any coherences on n-cells for $n \geq 2$.

and the interchange law

$$(g * \gamma) \cdot (\delta * f') = (\delta * f) \cdot (g' * \gamma)$$

for all morphisms f, f', g, g' and equalities γ, δ as in

$$\begin{tikzcd}
x \arrow[r, bend left=50, "f"] \arrow[r, bend right=50, "f'"'] \arrow[r, phantom, "\Downarrow\gamma"] & y \arrow[r, bend left=50, "g"] \arrow[r, bend right=50, "g'"'] \arrow[r, phantom, "\Downarrow\delta"] & z .
\end{tikzcd}$$

2.2. 2-coherence

The following two coherence conditions are familiar from bicategory theory.

Definition 2.2.1. *Triangle coherators.* A wild category $\mathscr{C}$ has *triangle coherators* if for all morphisms

$$x \xrightarrow{f} y \xrightarrow{g} z$$

in $\mathscr{C}$, there is an equality

$$\triangle_{f,g} : \alpha \cdot (g * \lambda) = \rho * f$$

filling the triangle

$$\begin{tikzcd}
(g \diamond \mathrm{id}) \diamond f \arrow[rr, Rightarrow, "\alpha"] \arrow[dr, Rightarrow, "\rho * f"'] & & g \diamond \mathrm{id} \diamond f \arrow[dl, Rightarrow, "g * \lambda"] \\
& g \diamond f &
\end{tikzcd}$$

Definition 2.2.2. *Pentagonators.* $\mathscr{C}$ has *pentagon coherators for associators,* or *(α-) pentagonators,* if for all composable chains

$$v \xrightarrow{f} w \xrightarrow{g} x \xrightarrow{h} y \xrightarrow{k} z$$

of morphisms in $\mathscr{C}$, there is an equality

$$⬠_{f,g,h,k} : (\alpha * f) \cdot \alpha \cdot (k * \alpha) = \alpha \cdot \alpha$$

filling the usual pentagon

$$\begin{tikzcd}
& ((k \diamond h) \diamond g) \diamond f \arrow[dl, Rightarrow, "\alpha * f"'] \arrow[dr, Rightarrow, "\alpha"] & \\
(k \diamond h \diamond g) \diamond f \arrow[d, Rightarrow, "\alpha"'] & & (k \diamond h) \diamond g \diamond f \arrow[d, Rightarrow, "\alpha"] \\
k \diamond (h \diamond g) \diamond f \arrow[rr, Rightarrow, "k * \alpha"'] & & k \diamond h \diamond g \diamond f
\end{tikzcd}$$

Definition 2.2.3. *2-coherent wild categories.* We call a wild category *2-coherent* if it has triangle and pentagon coherators.

In the literature, Hart and Hou [HH24] call 2-coherent wild categories *bicategories*; we do not use this terminology as 2-coherent wild categories always have invertible 2-cells (and higher cells). 2-coherent wild categories are also essentially the *wild 2-precategories* of Capriotti and Kraus [CK17]; the difference is that their definition also includes the other two triangle coherators of Proposition 2.2.6 in the data of the type. We also have the following link to the univalent bicategory theory of Ahrens et al. [Ahr+21]:

Proposition 2.2.4. Any 2-coherent wild category is also a *prebicategory* in the sense of Ahrens et al. [Ahr+21, Definition 2.1], by taking the type of 2-cells from f to g to be the equality type $f = g$, and using Proposition 2.1.4.

Examples 2.2.5. *2-coherent higher categories.* Any precategory trivially satisfies all higher equalities between equalities of morphisms. The universe wild categories $\mathcal{U}$ have definitionally unital and associative composition of morphisms, and thus have trivial triangle and pentagon coherators.

Many coherences involving λ, ρ and α that hold in all bicategories also hold in 2-coherent wild categories—namely, those that do not rely on uniqueness of equality of 2-cells. In particular,

Proposition 2.2.6. In a 2-coherent wild category $\mathscr{C}$, there are witnesses (not necessarily unique) that

- $\lambda_{\mathrm{id}_x} = \rho_{\mathrm{id}_x}$ for all $x : \mathscr{C}_0$, and
- the diagrams of equalities

$$\begin{tikzcd}
(\mathrm{id} \diamond g) \diamond f \arrow[rr, Rightarrow, "\alpha"] \arrow[dr, Rightarrow, "\lambda * f"'] & & \mathrm{id} \diamond g \diamond f \arrow[dl, Rightarrow, "\lambda"] & \text{and} & (g \diamond f) \diamond \mathrm{id} \arrow[rr, Rightarrow, "\alpha"] \arrow[dr, Rightarrow, "\rho"'] & & g \diamond f \diamond \mathrm{id} \arrow[dl, Rightarrow, "g * \rho"] \\
& g \diamond f & & & & g \diamond f &
\end{tikzcd}$$

  commute for all $f : \mathscr{C}(x, y)$ and $g : \mathscr{C}(y, z)$.

We refer to [JY21, Propositions 2.2.4 and 2.2.6] for proofs of these facts.

Non-example 2.2.7. *A wild category lacking 2-coherators.* In [Kra21b, Lemma 8], Kraus uses the circle higher inductive type to construct a wild category $\mathscr{E}$ with an object $x : \mathscr{E}_0$ that refutes $\lambda_{\mathrm{id}_x} = \rho_{\mathrm{id}_x}$. By Proposition 2.2.6, $\mathscr{E}$ must therefore also fail to have triangle or pentagon coherators. In fact, one can see directly by [Uni13, Lemma 6.4.2] that the existence of triangle coherators for $\mathscr{E}$ is a HoTT taboo.

We will also use the following coherence.

Proposition 2.2.8. Suppose that $\gamma : g = \mathrm{id} \diamond f$ in a 2-coherent wild category. Then

$$\begin{tikzcd}
\mathrm{id} \diamond g \arrow[r, Rightarrow, "\mathrm{id} * \gamma"] \arrow[d, Rightarrow, "\lambda"'] & \mathrm{id} \diamond \mathrm{id} \diamond f \arrow[d, Rightarrow, "\mathrm{id} * \lambda"] \\
g \arrow[r, Rightarrow, "\gamma"'] & \mathrm{id} \diamond f
\end{tikzcd}$$

commutes, i.e. $\lambda_g \cdot \gamma =_{(\mathrm{id} \diamond g = \mathrm{id} \diamond f)} (\mathrm{id} * \gamma) \cdot (\mathrm{id} * \lambda_f)$.

Proof. We equivalently prove that $\mathrm{id} * \gamma = \lambda_g \cdot \gamma \cdot (\mathrm{id} * \lambda_f)^{-1}$. Since $\mathrm{id} * \gamma = \lambda_g \cdot \gamma \cdot \lambda_{\mathrm{id}\diamond f}^{-1}$ by properties of whiskering (Proposition 2.1.4), the result follows if $\mathrm{id} * \lambda_f = \lambda_{\mathrm{id}\diamond f}$. Diagram 2.2.8.1 shows how to construct such an equality: its interior is divided into two triangles and a bigon, which commute by the triangle coherator $\triangle_{f,\mathrm{id}}$ and the equalities in Proposition 2.2.6. □

$$\begin{tikzcd}
\mathrm{id} \diamond \mathrm{id} \diamond f \arrow[rr, Rightarrow, bend left=50, "\mathrm{id}*\lambda_f"] \arrow[rr, Rightarrow, bend right=50, "\lambda_{\mathrm{id}\diamond f}"'] & (\mathrm{id} \diamond \mathrm{id}) \diamond f \arrow[l, Rightarrow, "\alpha"'] \arrow[r, Rightarrow, bend left, "\rho_{\mathrm{id}}*f"] \arrow[r, Rightarrow, bend right, "\lambda_{\mathrm{id}}*f"'] & \mathrm{id} \diamond f
\end{tikzcd}$$

Diagram 2.2.8.1

2.3. Univalence

Definition 2.3.1. *Wild equivalences.* A morphism $f : \mathscr{C}(x, y)$ in a wild category $\mathscr{C}$ is a *wild equivalence* if it has both a section and a retraction (i.e. is biinvertible).

The type of wild equivalences from x to y in $\mathscr{C}$ is denoted $x \simeq_{\mathscr{C}} y$. Its elements are also called *$\mathscr{C}$-equivalences* to avoid confusion with type theoretic equivalences.

Lemma 2.3.2. If $f : x \simeq_{\mathscr{C}} y$ is a $\mathscr{C}$-equivalence, then for any $w : \mathscr{C}_0$, the maps

$$\begin{aligned} f \diamond \_ &: \mathscr{C}(w, x) \to \mathscr{C}(w, y) \\ \text{and} \quad \_ \diamond f &: \mathscr{C}(y, z) \to \mathscr{C}(x, z) \end{aligned}$$

are equivalences of hom-types.

Proof. Let s and r be, respectively, a section and retraction of f in $\mathscr{C}$. Then $(s \diamond \_)$ and $(r \diamond \_)$ are, respectively, sections and retractions of $(f \diamond \_)$, while $(\_ \diamond r)$ and $(\_ \diamond s)$ are, respectively, sections and retractions of $(\_ \diamond f)$. □

Definition 2.3.3. *Wild isomorphisms.* We also consider the type of wild *isomorphisms* $x \cong_{\mathscr{C}} y$ in a wild category $\mathscr{C}$, i.e. the type of morphisms $f : \mathscr{C}(x, y)$ having a two-sided inverse $g : \mathscr{C}(y, x)$.

Any two-sided inverse is both a section and a retraction, so there is a canonical map

$$\mathrm{isotoeqv}_{\mathscr{C}} : (x \cong_{\mathscr{C}} y) \to (x \simeq_{\mathscr{C}} y)$$

for any objects $x, y : \mathscr{C}_0$. If $\mathscr{C}$ is a precategory then $\mathrm{isotoeqv}_{\mathscr{C}}$ is an equivalence: its inverse sends a section-retraction pair (s, r) to the two-sided inverse $(r \diamond f \diamond s)$ of f.

Definition 2.3.4. *Dependent identity morphisms.* If $x, y : \mathscr{C}_0$ are objects of a wild category such that $e : x = y$, there is a morphism[10]

$$\mathrm{idd}(e) :\equiv \mathrm{id}_x \downarrow_e^{\mathscr{C}(x,\_)} : \mathscr{C}(x, y).$$

By induction on e, $\mathrm{idd}(e)$ is an isomorphism, with inverse $\mathrm{idd}(e^{-1})$. We thus get a map

$$\mathrm{idd} : x = y \to x \cong_{\mathscr{C}} y.$$

In precategories, idd is essentially idtoiso [Uni13, Definition 9.1.4; AKS15, Lemma 3.4].

Definition 2.3.5. *Wild univalence.* A wild category $\mathscr{C}$ is *univalent* if, for all $x, y : \mathscr{C}_0$, the map

$$\mathrm{idtoeqv}_{\mathscr{C}} :\equiv \mathrm{isotoeqv}_{\mathscr{C}} \circ \mathrm{idd} : x = y \to x \simeq_{\mathscr{C}} y$$

is an equivalence. Its inverse equivalence is denoted $\mathrm{eqvtoid}_{\mathscr{C}}$.

Wild univalence subsumes both 1-categorical univalence (when $\mathscr{C}$ is a precategory) and the univalence axiom (when $\mathscr{C} \equiv \mathscr{U}$ is a universe).

3. Pullbacks in 2-Coherent Wild Categories

In this section we develop the theory of pullbacks in 2-coherent wild categories. These jointly generalize comma objects in (2, 1)-categories (including 1-categorical pullbacks) as well as type theoretic homotopy pullbacks [AKL15].

3.1. Commuting squares

Definition 3.1.1. *Cospans.* A *cospan* in a wild category $\mathscr{C}$ is an element of

$$\mathrm{Cospan}(\mathscr{C}) :\equiv \textstyle\sum_{(A,B,C\,:\,\mathscr{C}_0)} (f : \mathscr{C}(A, C))\,(g : \mathscr{C}(B, C)).$$

We denote cospans graphically as $A \xrightarrow{f} C \xleftarrow{g} B$, or simply by (f, g) when the vertices are understood.

Definition 3.1.2. *Commuting squares.* A *commuting square* in a wild category $\mathscr{C}$ consists of

- a **cospan** $A \xrightarrow{f} C \xleftarrow{g} B$,
- a **source** object $X : \mathscr{C}_0$,
- two **legs** $m_A : \mathscr{C}(X, A)$ and $m_B : \mathscr{C}(X, B)$,
- and an **equality 2-cell** $\gamma : f \diamond m_A = g \diamond m_B$

as in the following diagram in $\mathscr{C}$,

$$\begin{tikzcd} X \arrow[r, "m_B"] \arrow[d, "m_A"'] & B \arrow[d, "g"] \\ A \arrow[r, "f"'] \arrow[ur, equals, "\gamma"'] & C \end{tikzcd} .$$

[10] idd for “*id*entity morphism *d*ependent over an equality”.

First, we consider the type of commuting squares on a cospan with fixed source.

Definition 3.1.3. *Commuting squares, with fixed source and cospan.* Indexing over cospans

$$\mathfrak{c} :\equiv A \xrightarrow{f} C \xleftarrow{g} B$$

and source objects $X : \mathscr{C}_0$, we define the type

$$\mathrm{CommSq}_{\mathfrak{c}}(X) :\equiv \sum\, (m_A : \mathscr{C}(X,\, A))\; (m_B : \mathscr{C}(X,\, B))\; (\gamma : f \diamond m_A = g \diamond m_B)$$

of *commuting squares on* $\mathfrak{c}$ *with source* X.

Avigad, Kapulkin and Lumsdaine [AKL15] also call $\mathrm{CommSq}_{\mathfrak{c}}(X)$ (instantiated in universes) the type of *cones* over $\mathfrak{c}$ with vertex X.

Characterizing the equality of $\mathrm{CommSq}_{\mathfrak{c}}(X)$ is routine.

Proposition 3.1.4. *Equality of* $\mathrm{CommSq}_{\mathfrak{c}}(X)$. Let $\mathfrak{c} \equiv A \xrightarrow{f} C \xleftarrow{g} B$ be a cospan, $X : \mathscr{C}_0$ an object, and

$$\mathfrak{S} \equiv (m_A, m_B, \gamma) \quad \text{and} \quad \mathfrak{S}' \equiv (m_A{}', m_B{}', \gamma')$$

be elements of $\mathrm{CommSq}_{\mathfrak{c}}(X)$ in a wild category $\mathscr{C}$. Then the equality type $\mathfrak{S} = \mathfrak{S}'$ is equivalent to

$$\sum\, (e_A : m_A = m_A{}')\; (e_B : m_B = m_B{}'),\; \gamma = (f * e_A) \cdot \gamma' \cdot (g * e_B)^{-1}.$$

Proof. By a routine application of the fundamental theorem of identity types [Rij22, Theorem 11.2.2], the algebra of Σ-types, and Proposition 2.1.4. □

We have the following operations on commuting squares.

Definition 3.1.5. *Transpose.* For any cospan $A \xrightarrow{f} C \xleftarrow{g} B$ and $X : \mathscr{C}_0$, the *transpose* map

$$\_^{\mathrm{T}} : \mathrm{CommSq}_{(f,g)}(X) \to \mathrm{CommSq}_{(g,f)}(X)$$

is given by

$$(m_A, m_B, \gamma)^{\mathrm{T}} :\equiv (m_B, m_A, \gamma^{-1}).$$

Proposition 3.1.6. *Transpose is an equivalence.* Transpose is involutive: $(\mathfrak{S}^{\mathrm{T}})^{\mathrm{T}} = \mathfrak{S}$ for all $\mathfrak{S}$, and so in particular $\_^{\mathrm{T}}$ is an equivalence.

Definition 3.1.7. *Horizontal and vertical pasting.* From a diagram

$$\begin{tikzcd}
A' \arrow[r, "f'"] \arrow[d, "i"'] & B' \arrow[r, "g'"] \arrow[d, "j"'] & C' \arrow[d, "k"] \\
A \arrow[r, "f"'] \arrow[ur, Rightarrow, "\mathfrak{q}"'] & B \arrow[r, "g"'] \arrow[ur, Rightarrow, "\mathfrak{p}"'] & C
\end{tikzcd}$$

of commuting squares $\mathfrak{Q} :\equiv (i, f', \mathfrak{q})$ and $\mathfrak{P} :\equiv (j, g', \mathfrak{p})$ we get the *horizontal pasting*

$$\mathfrak{Q} \mid \mathfrak{P} \;:\equiv\; \begin{tikzcd}
A' \arrow[r, "g' \diamond f'"] \arrow[d, "i"'] & C' \arrow[d, "k"] \\
A \arrow[r, "g \diamond f"'] \arrow[ur, Rightarrow, "\mathfrak{q} | \mathfrak{p}"'] & C
\end{tikzcd}$$

where $\mathfrak{q} \mid \mathfrak{p} \;:\equiv\; \alpha \cdot (g * \mathfrak{q}) \cdot \alpha^{-1} \cdot (\mathfrak{p} * f') \cdot \alpha$. Similarly, from a diagram

$$\begin{tikzcd}
A' \arrow[r, "i"] \arrow[d, "f'"'] & A \arrow[d, "f"] \\
B' \arrow[r, "j"] \arrow[d, "g'"'] \arrow[ru, Rightarrow, "\mathfrak{q}"'] & B \arrow[d, "g"] \\
C' \arrow[r, "k"'] \arrow[ru, Rightarrow, "\mathfrak{p}"'] & C
\end{tikzcd}$$

of commuting squares $\mathfrak{Q} :\equiv (f', i, \mathfrak{q})$ and $\mathfrak{P} :\equiv (g', j, \mathfrak{p})$ we get the *vertical pasting*

$$\frac{\mathfrak{Q}}{\mathfrak{P}} \;:\equiv\; \begin{tikzcd}
A' \arrow[r, "i"] \arrow[d, "g' \diamond f'"'] & A \arrow[d, "g \diamond f"] \\
C' \arrow[r, "k"'] \arrow[ru, Rightarrow, "\frac{\mathfrak{q}}{\mathfrak{p}}"'] & C
\end{tikzcd}$$

where $\frac{\mathfrak{q}}{\mathfrak{p}} \;:\equiv\; \alpha^{-1} \cdot (\mathfrak{p} * f') \cdot \alpha \cdot (g * \mathfrak{q}) \cdot \alpha^{-1}$.

Definition 3.1.7 is in some ways redundant: a straightforward calculation shows that

$$\frac{\mathfrak{Q}}{\mathfrak{P}} = (\mathfrak{Q}^{\mathrm{T}} \mid \mathfrak{P}^{\mathrm{T}})^{\mathrm{T}},$$

and so one could simply vertically paste squares by horizontally pasting their transposes. However, this equality is only propositional, and it turns out to be more convenient later to use the canonical form of the vertical pasting as we have defined it.

Definition 3.1.8. *Vertical pasting map.* For any $A : \mathscr{C}_0$, $f : \mathscr{C}(A, B)$ and $X : \mathscr{C}_0$, the vertical pasting with

$$\mathfrak{P} \;:\equiv\; \begin{tikzcd}
B' \arrow[r, "j"] \arrow[d, "g'"'] & B \arrow[d, "g"] \\
C' \arrow[r, "k"'] \arrow[ru, Rightarrow, "\mathfrak{p}"'] & C
\end{tikzcd}$$

yields a map $\mathrm{CommSq}_{(j,f)}(X) \to \mathrm{CommSq}_{(k,g\diamond f)}(X)$. That is, we have the family

$$\frac{-}{\mathfrak{P}} : \prod\, (A : \mathscr{C}_0)\, (f : \mathscr{C}(A, B))\, (X : \mathscr{C}_0)\; \mathrm{CommSq}_{(j,f)}(X) \to \mathrm{CommSq}_{(k,g\diamond f)}(X).$$

Morphisms in $\mathscr{C}$ act contravariantly on commuting squares by precomposition at their source.

Definition 3.1.9. *Precomposing squares with morphisms.* For any cospan $\mathfrak{c} \equiv A \xrightarrow{f} C \xleftarrow{g} B$ and $X, Y : \mathscr{C}_0$, there is a *precomposition* map

$$\_ \mathbin{\square} \_ : \mathrm{CommSq}_{\mathfrak{c}}(Y) \to \mathscr{C}(X, Y) \to \mathrm{CommSq}_{\mathfrak{c}}(X)$$

defined by $(m_A, m_B, \gamma) \mathbin{\square} m \;:\equiv\; (m_A \diamond m,\ m_B \diamond m,\ \alpha^{-1} \cdot (\gamma * m) \cdot \alpha)$.

Lemma 3.1.10. *Right action of morphisms on commuting squares.* If $\mathfrak{S} : \mathrm{CommSq}_{\mathfrak{c}}(X)$ is a commuting square in a 2-coherent wild category then

$$\mathfrak{S} \mathbin{\square} \mathrm{id}_X = \mathfrak{S},$$

and for all $f : \mathscr{C}(X,\ Y)$ and $g : \mathscr{C}(Y,\ Z)$,

$$\mathfrak{S} \square (g \diamond f) = \mathfrak{S} \square g \square f.$$

Proof. By calculation, properties of whiskering (Proposition 2.1.4), and coherence—namely, the right identity triangle coherence (Proposition 2.2.6) for the first claim, and the pentagon coherence for the second. □

Corollary 3.1.11. By induction on e and Lemma 3.1.10,

$$\mathfrak{S}\downarrow_e^{\mathrm{CommSq}_{\mathfrak{c}}(\_)} =_{\mathrm{CommSq}_{\mathfrak{c}}(X')} \mathfrak{S} \square \mathrm{idd}(e^{-1})$$

for every $\mathfrak{S} : \mathrm{CommSq}_{\mathfrak{c}}(X)$ and $e : X = X'$.

Now we can characterize the identity of commuting squares on a cospan, allowing their source objects to vary.

Definition 3.1.12. *Commuting squares on a cospan.* The type of commuting squares on a cospan $\mathfrak{c}$ is the total space

$$\mathrm{CommSq}(\mathfrak{c}) :\equiv \textstyle\sum_{(X\,:\,\mathscr{C}_0)} \mathrm{CommSq}_{\mathfrak{c}}(X).$$

Corollary 3.1.13. *Equality of* $\mathrm{CommSq}(\mathfrak{c})$. Suppose that $(X, \mathfrak{S})$ and $(X', \mathfrak{S}')$ are two commuting squares on a cospan $\mathfrak{c} :\equiv (f, g)$. The equality

$$(X, \mathfrak{S}) =_{\mathrm{CommSq}(\mathfrak{c})} (X', \mathfrak{S}')$$

is equivalent to

$$\textstyle\sum_{(e\,:\,X\,=\,X')},\ \mathfrak{S} = \mathfrak{S}' \square \mathrm{idd}(e).$$

Proof. By the equality of Σ-types and Corollary 3.1.11. □

Up to this point we have considered commuting squares with respect to a fixed cospan $(A, B, C, f, g) : \mathrm{Cospan}(\mathscr{C})$. We will also need to compare squares on cospans with propositionally equal but definitionally distinct legs (f, g) and (f', g'). To this end, we index the type of commuting squares over their vertices, and consider the following type family.

Definition 3.1.14. *Commuting squares, with fixed vertices.* For $X, A, B, C : \mathscr{C}_0$, define

$$\mathrm{CommSq}(X, A, B, C) :\equiv \textstyle\sum_{(f\,:\,\mathscr{C}(A,\ C))\,(g\,:\,\mathscr{C}(B,\ C))},\ \mathrm{CommSq}_{(f,g)}(X).$$

That is, $\mathrm{CommSq}(X, A, B, C)$ is the total space of $\mathrm{CommSq}_{(A,B,C,f,g)}(X)$ fibered over $f : \mathscr{C}(A,\ C)$ and $g : \mathscr{C}(B,\ C)$.

We characterize the identity type of $\mathrm{CommSq}(X, A, B, C)$ using the following version of Rijke's structure identity principle [Rij22, Theorem 11.6.2], which may be understood as a dependent form of the fundamental theorem of identity types.

Theorem 3.1.15. *Structure identity principle.* Suppose that A is a type pointed at $a : A$, and $B : A \to \mathscr{U}$ is a type family over A, pointed at $b : B(a)$. Then for any type family

$$R : \textstyle\prod_{(x\,:\,A)} a = x \to B(x) \to \mathscr{U}$$

pointed at $r : R(a, \mathsf{refl}_a, b)$, the canonical family of maps indexed over x and y

$$\prod\,(x : A)\,(y : B(x))\ \ (a, b) = (x, y) \longrightarrow \Sigma\,(p : a = x),\ R(x, p, y)$$

is a family of equivalences if and only if the total space $\Sigma\,(B(a))\,R(a, \mathsf{refl}_a)$ is contractible.

Lemma 3.1.16. *Equality of* $\mathsf{CommSq}(X, A, B, C)$. Let

$$(f, g, \mathfrak{S}),\ (f', g', \mathfrak{S}') : \mathsf{CommSq}(X, A, B, C)$$

be commuting squares with vertices $X, A, B, C : \mathscr{C}_0$, where $\mathfrak{S} \equiv (m_A, m_B, \gamma)$ and $\mathfrak{S}' \equiv (m_A{}', m_B{}', \gamma')$. The equality $(f, g, \mathfrak{S}) = (f', g', \mathfrak{S}')$ is equivalent to

$$\begin{gathered}\sum\,(e_f : f = f')\,(e_g : g = g')\,(e_A : m_A = m_A{}')\,(e_B : m_B = m_B{}'),\\ \gamma = (e_f * m_A) \cdot (f' * e_A) \cdot \gamma' \cdot (g' * e_B)^{-1} \cdot (e_g * m_B)^{-1}.\end{gathered}$$

This last component is a proof that

$$\begin{tikzcd}
& f \diamond m_A \arrow[dl, equals, "e_f * m_A"'] \arrow[dr, equals, "\gamma"] & \\
f' \diamond m_A \arrow[d, equals, "f' * e_A"'] & & g \diamond m_B \arrow[d, equals, "e_g * m_B"] \\
f' \diamond m_A{}' \arrow[dr, equals, "\gamma'"'] & & g' \diamond m_B \arrow[dl, equals, "g' * e_B"] \\
& g' \diamond m_B{}' &
\end{tikzcd}$$

commutes.

Proof. Use the structure identity principle (Theorem 3.1.15) applied to the pointed type $(\mathscr{C}(A,\ C) \times \mathscr{C}(B,\ C),\ (f, g))$ and the type family

$$\mathsf{CommSq}_{(A,B,C,\_,\_)}(X) : \mathscr{C}(A,\ C) \times \mathscr{C}(B,\ C) \longrightarrow \mathcal{U}$$

pointed at $\mathfrak{S} : \mathsf{CommSq}_{(f,g)}(X)$. We define (in curried form)

$$R : \prod\,(f' : \mathscr{C}(A,\ C))\,(g' : \mathscr{C}(B,\ C))\ \ (f = f') \longrightarrow (g = g') \longrightarrow \mathsf{CommSq}_{(f',g')}(X) \longrightarrow \mathcal{U}$$

such that $R(f', g', e_f, e_g)(k_A, k_B, \delta)$ is the type

$$\begin{gathered}\sum\,(e_A : m_A = k_A)\,(e_B : m_B = k_B),\\ \gamma = (e_f * m_A) \cdot (f' * e_A) \cdot \delta \cdot (g' * e_B)^{-1} \cdot (e_g * m_B)^{-1},\end{gathered}$$

pointed at $(\mathsf{refl}_{m_A}, \mathsf{refl}_{m_B}, \mathsf{refl}_\gamma) : R(f, g, \mathsf{refl}_f, \mathsf{refl}_g, \mathfrak{S})$. Then it's enough to show that $\Sigma\,(\mathsf{CommSq}_{(f,g)}(X))\,R(f, g, \mathsf{refl}_f, \mathsf{refl}_g)$ is contractible, which we have already done in the proof of Proposition 3.1.4. □

3.2. Pullbacks

Definition 3.2.1. *Pullbacks.* Let

$$\mathfrak{P} \;:\equiv\; \begin{tikzcd}
P \arrow[r, "\pi_B"] \arrow[d, "\pi_A"'] & B \arrow[d, "g"] \\
A \arrow[r, "f"'] \arrow[ur, equals, "\mathfrak{p}"'] & C
\end{tikzcd}$$

be a commuting square on $\mathfrak{c} :\equiv A \xrightarrow{f} C \xleftarrow{g} B$ with source P, in a wild category $\mathscr{C}$. By specializing the precomposition map (Definition 3.1.9) at $\mathfrak{P}$, we obtain the family of maps

$$\mathfrak{P} \,\square_{\_}\, \_ : \prod (X : \mathscr{C}_0)\ \mathscr{C}(X, P) \to \mathrm{CommSq}_{\mathfrak{c}}(X).$$

We say that $\mathfrak{P}$ is a *pullback of* $\mathfrak{c}$ if $(\mathfrak{P} \,\square_{\_}\, \_)$ is a family of equivalences, and a *weak pullback of* $\mathfrak{c}$ if $(\mathfrak{P} \,\square_{\_}\, \_)$ is a family of retractions, or split surjections.

Corollary 3.2.2. *Universal property of (weak) pullbacks.* By the characterization of equality of $\mathrm{CommSq}_{\mathfrak{c}}(X)$ (Proposition 3.1.4), for each $X : \mathscr{C}_0$ and commuting square $\mathfrak{S} :\equiv (m_A, m_B, \gamma)$ on $\mathfrak{c}$ with source X, the fiber of $(\mathfrak{P} \,\square_X\, \_)$ at $\mathfrak{S}$ is equivalent to

$$\begin{gathered}\sum (m : \mathscr{C}(X, P))\ (e_A : \pi_A \diamond m = m_A)\ (e_B : \pi_B \diamond m = m_B), \\ \alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha = (f * e_A) \cdot \gamma \cdot (g * e_B)^{-1}.\end{gathered}$$

Thus $\mathfrak{P}$ is a pullback (respectively, a weak pullback) when this type is contractible (respectively, pointed) for every X and $\mathfrak{S}$.

Being a pullback is evidently a property: for any cospan $\mathfrak{c}$ and $P : \mathscr{C}_0$, the predicate

$$\text{is-pullback}_{\mathfrak{c},P}(\mathfrak{P}) :\equiv \prod (X : \mathscr{C}_0)\ \text{is-equiv}\,(\mathfrak{P} \,\square_X\, \_)$$

on $\mathrm{CommSq}_{\mathfrak{c}}(P)$ is propositional.

Proposition 3.2.3. *Pullbacks are closed under transpose.* $\mathfrak{P}^{\mathsf{T}}$ is a (weak) pullback if $\mathfrak{P}$ is.

Proof. By a straightforward calculation,

$$(\mathfrak{P}^{\mathsf{T}} \,\square_X\, \_) = (\_^{\mathsf{T}}) \circ (\mathfrak{P} \,\square_X\, \_)$$

for all $X : \mathscr{C}_0$. Since $\_^{\mathsf{T}}$ is an equivalence (Proposition 3.1.6), $(\mathfrak{P}^{\mathsf{T}} \,\square_X\, \_)$ is an equivalence (respectively, a retraction) when $(\mathfrak{P} \,\square_X\, \_)$ is. □

The following lemma is inspired by the proof of [AKL15, Proposition 4.1.11] and used in the proof of the pullback pasting lemma (Lemma 3.2.5).

Lemma 3.2.4. *Pasting maps of (weak) pullbacks.* A commuting square $\mathfrak{P}$ in a 2-coherent wild category $\mathscr{C}$ is a pullback (respectively, a weak pullback) if and only if the vertical pasting map $\frac{-}{\mathfrak{P}}$ (Definition 3.1.8) is a family of equivalences (respectively, retractions).

Proof. Let $\mathfrak{P} :\equiv (g', j, \mathfrak{p})$ be a commuting square on (k, g) as in Definition 3.1.8. For any $A : \mathscr{C}_0$, $f : \mathscr{C}(A, B)$ and $X : \mathscr{C}_0$, the fiber of $\frac{-}{\mathfrak{P}}_{A,f,X}$ at

$$\mathfrak{X} \;:\equiv\; \begin{tikzcd} X \arrow[r, "m_A"] \arrow[d, "m_{C'}"'] & A \arrow[d, "g \diamond f"] \\ C' \arrow[r, "k"'] \arrow[ur, Rightarrow, "\xi"'] & C \end{tikzcd}$$

is equivalent to the Σ-type

$$\begin{gathered}\sum (m : \mathscr{C}(X, B'))\ (i : \mathscr{C}(X, A))\ (\gamma : j \diamond m = f \diamond i), \\ (e_{C'} : g' \diamond m = m_{C'}) \times (e_A : i = m_A) \\ \times\, (\alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha \cdot (g * \gamma) \cdot \alpha^{-1} = (k * e_{C'}) \cdot \xi \cdot (g \diamond f * e_A)^{-1}),\end{gathered}$$

by the equality characterization of Proposition 3.1.4. Contracting the singleton formed by the components i and e_A, this is equivalent to

$$\sum (m : \mathscr{C}(X, B'))(e_{C'} : g' \diamond m = m_{C'})(e_B : j \diamond m = f \diamond m_A), \\ \alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha = (k * e_{C'}) \cdot (\xi \cdot \alpha) \cdot (g * e_B)^{-1}.$$

But this type is also the fiber of the precomposition map $(\mathfrak{P} \mathbin{\square_X} \_)$ at the commuting square

$$\begin{tikzcd}
X \arrow[r, "f \diamond m_A"] \arrow[d, "m_{C'}"'] & B \arrow[d, "g"] \\
C' \arrow[r, "k"'] \arrow[ur, Rightarrow, "\xi \cdot \alpha"'] & C
\end{tikzcd}$$

obtained by "reparenthesizing" the diagram $\mathfrak{X}$. Thus if $\mathfrak{P}$ is a pullback (respectively, a weak pullback) then by its universal property (Corollary 3.2.2) the fiber $\left(\bar{\bar{\mathfrak{P}}}_{A,f,X}\right)^{-1}(\mathfrak{X})$ is contractible (respectively, pointed).

Conversely, for any $X : \mathscr{C}_0$, we claim that the map

$$\varphi : \mathscr{C}(X, B') \to \mathrm{CommSq}_{(j,\mathrm{id}_B)}(X) \\ \varphi(m) :\equiv (m, j \diamond m, {\lambda_{j \diamond m}}^{-1})$$

is an equivalence, and that the diagram of types and functions

$$\begin{tikzcd}
 & \mathrm{CommSq}_{(j,\mathrm{id}_B)}(X) \arrow[d, "\bar{\bar{\mathfrak{P}}}_{B,\mathrm{id}_B,X}"] \\
\mathscr{C}(X, B') \arrow[ur, "\varphi", "\sim"'] \arrow[dr, "\mathfrak{P} \square_X \_"'] & \mathrm{CommSq}_{(k,g \diamond \mathrm{id}_B)}(X) \arrow[d, "\psi", "\sim"'] \\
 & \mathrm{CommSq}_{(k,g)}(X)
\end{tikzcd}$$

commutes, where ψ is the equivalence $(m_{C'}, m_B, \gamma) \mapsto (m_{C'}, m_B, \gamma \cdot (\rho * m_B))$. That is, $(\mathfrak{P} \mathbin{\square_X} \_)$ is the pre- and post-composition of $\bar{\bar{\mathfrak{P}}}_{B,\mathrm{id}_B,X}$ by equivalences. Thus, if $\bar{\bar{\mathfrak{P}}}$ is a family of equivalences then so is $(\mathfrak{P} \mathbin{\square_\_} \_)$, and if $\bar{\bar{\mathfrak{P}}}$ is a family of retractions then so is $(\mathfrak{P} \mathbin{\square_\_} \_)$.

Now, the map φ is clearly a section of $\mathrm{fst} : \mathrm{CommSq}_{(j,\mathrm{id})}(X) \to \mathscr{C}(X, B')$. We show that it's also a retraction of fst, i.e. that

$$\varphi(m_{B'}) \equiv (m_{B'}, j \diamond m_{B'}, \lambda^{-1}) = (m_{B'}, m_B, \gamma)$$

for all $m_{B'} : \mathscr{C}(X, B')$, $m_B : \mathscr{C}(X, B)$ and $\gamma : j \diamond m_{B'} = \mathrm{id} \diamond m_B$. Taking the straightforward equalities $\mathrm{refl} : m_{B'} = m_{B'}$ and $\gamma \cdot \lambda : j \diamond m_{B'} = m_B$ of the morphism parts of the commuting squares, we lastly need the equality of commutativity witnesses

$$\lambda^{-1} = (j * \mathrm{refl}) \cdot \gamma \cdot (\mathrm{id} * (\gamma \cdot \lambda)^{-1}),$$

or equivalently, that $\lambda \cdot \gamma = (\mathrm{id} * \gamma) \cdot (\mathrm{id} * \lambda)$. This holds by Proposition 2.2.8.

Finally, given $m : \mathscr{C}(X, B')$ we calculate that

$$\mathfrak{S} :\equiv \left(\psi \circ \bar{\bar{\mathfrak{P}}}_{B,\mathrm{id}_B,X} \circ \varphi\right)(m)$$

and

$$\mathfrak{S}' :\equiv \mathfrak{P} \,\square_X\, m$$

are commuting squares of type $\mathrm{CommSq}_{(k,g)}(X)$ with the same morphism components $g' \diamond m$ and $j \diamond m$. The commutativity witness of $\mathfrak{S}$ is

$$\alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha \cdot (g * \lambda^{-1}) \cdot \alpha^{-1} \cdot (\rho * (j \diamond m)),$$

while that of $\mathfrak{S}'$ is

$$\alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha,$$

and these are equal since $(g * \lambda^{-1}) \cdot \alpha^{-1} \cdot (\rho * (j \diamond m)) = \mathrm{refl}$ by the triangle coherator. □

Lemma 3.2.5. *Vertical pullback pasting.* Suppose we have a diagram

$$\begin{tikzcd}
A' \arrow[r, "i"] \arrow[d, "f'"'] & A \arrow[d, "f"] \\
B' \arrow[r, "j"] \arrow[d, "g'"'] \arrow[ur, Rightarrow, "\mathfrak{q}"'] & B \arrow[d, "g"] \\
C' \arrow[r, "k"'] \arrow[ur, Rightarrow, "\mathfrak{p}"'] & C
\end{tikzcd}$$

in a 2-coherent wild category $\mathscr{C}$. Then if $\mathfrak{P} :\equiv (g', j, \mathfrak{p})$ is a pullback of (k, g), the commuting square $\mathfrak{Q} :\equiv (f', i, \mathfrak{q})$ is a pullback of (j, f) if and only if the vertical pasting $\frac{\mathfrak{Q}}{\mathfrak{P}}$ is a pullback of $(k, g \diamond f)$.

Proof. We claim that for any $X : \mathscr{C}_0$, the triangle

$$\begin{tikzcd}
 & \mathrm{CommSq}_{(j,f)}(X) \arrow[dd, "\frac{-}{\mathfrak{P}}_{A,f,X}"] \\
\mathscr{C}(X, A') \arrow[ur, "\mathfrak{Q}\square_X\_"] \arrow[dr, "\frac{\mathfrak{Q}}{\mathfrak{P}}\square_X\_"'] & \\
 & \mathrm{CommSq}_{(k,g\diamond f)}(X)
\end{tikzcd}$$

commutes. Then since $\mathfrak{P}$ is a pullback, the map $\frac{-}{\mathfrak{P}}_{A,f,X}$ is an equivalence (Lemma 3.2.4), and it follows that $(\mathfrak{Q} \,\square\, \_)$ is a family of equivalences if and only if $(\frac{\mathfrak{Q}}{\mathfrak{P}} \,\square\, \_)$ is.

What remains is to construct a homotopy $\left(\frac{\mathfrak{Q}}{\mathfrak{P}} \,\square_X\, \_\right) = \left(\frac{-}{\mathfrak{P}}_{A,f,X}\right) \circ (\mathfrak{Q} \,\square_X\, \_)$ for any X, i.e. a witness that, for any $m : \mathscr{C}(X, A')$, the commuting squares

$$\frac{\mathfrak{Q}}{\mathfrak{P}} \,\square_X\, m \;\equiv\; \left((g' \diamond f') \diamond m,\; i \diamond m,\; \alpha^{-1} \cdot \left(\tfrac{\mathfrak{q}}{\mathfrak{p}} * m\right) \cdot \alpha\right)$$

and

$$\frac{\mathfrak{Q} \,\square_X\, m}{\mathfrak{P}} \;\equiv\; \left(g' \diamond f' \diamond m,\; i \diamond m,\; \frac{\alpha^{-1} \cdot (\mathfrak{q} * m) \cdot \alpha}{\mathfrak{p}}\right)$$

are equal. By Proposition 3.1.4 together with the canonical equalities $\alpha : (g' \diamond f') \diamond m = g' \diamond f' \diamond m$ and $\mathrm{refl} : i \diamond m = i \diamond m$, it's enough to show that

$$\alpha^{-1} \cdot \left(\tfrac{\mathfrak{q}}{\mathfrak{p}} * m\right) \cdot \alpha = (k * \alpha) \cdot \frac{\alpha^{-1} \cdot (\mathfrak{q} * m) \cdot \alpha}{\mathfrak{p}}.$$

$$\begin{tikzcd}
 & & k \diamond (g' \diamond f') \diamond m \arrow[rr, Rightarrow, "k*\alpha"] \arrow[dl, Rightarrow, "\alpha^{-1}"'] & & k \diamond g' \diamond f' \diamond m \arrow[dr, Rightarrow, "\alpha^{-1}"] & & \\
 & (k \diamond g' \diamond f') \diamond m \arrow[dl, Rightarrow, "\alpha^{-1}*m"'] & & & & (k \diamond g') \diamond f' \diamond m \arrow[dr, Rightarrow, "\mathfrak{p}*(f' \diamond m)"] & \\
((k \diamond g') \diamond f') \diamond m \arrow[urrrrr, Rightarrow, "\alpha"] \arrow[d, Rightarrow, "(\mathfrak{p}*f')*m"'] & & & & & & (g \diamond j) \diamond f' \diamond m \arrow[d, Rightarrow, "\alpha"] \\
((g \diamond j) \diamond f') \diamond m \arrow[urrrrrr, Rightarrow, "\alpha"] \arrow[d, Rightarrow, "\alpha*m"'] & & & & & & g \diamond j \diamond f' \diamond m \arrow[d, Rightarrow, "g*\alpha^{-1}"] \\
(g \diamond j \diamond f') \diamond m \arrow[rrrrrr, Rightarrow, "\alpha"] \arrow[dr, Rightarrow, "(g*\mathfrak{q})*m"'] & & & & & & g \diamond (j \diamond f') \diamond m \arrow[dl, Rightarrow, "g*(\mathfrak{q}*m)"] \\
 & (g \diamond f \diamond i) \diamond m \arrow[rrrr, Rightarrow, "\alpha"] \arrow[dr, Rightarrow, "\alpha^{-1}*m"'] & & & & g \diamond (f \diamond i) \diamond m \arrow[dl, Rightarrow, "g*\alpha"] & \\
 & & ((g \diamond f) \diamond i) \diamond m \arrow[dr, Rightarrow, "\alpha"'] & & g \diamond f \diamond i \diamond m \arrow[dl, Rightarrow, "\alpha^{-1}"] & & \\
 & & & (g \diamond f) \diamond i \diamond m & & &
\end{tikzcd}$$

Diagram 3.2.5.1. Construction of $\alpha^{-1} \cdot \left(\frac{\mathfrak{q}}{\mathfrak{p}} * m\right) \cdot \alpha = (k * \alpha) \cdot \frac{\alpha^{-1} \cdot (\mathfrak{q} * m) \cdot \alpha}{\mathfrak{p}}$.

With a little path algebra (noting Proposition 2.1.4) this amounts to showing commutativity of the outer boundary of Diagram 3.2.5.1. By inserting associators α as shown in the interior of the diagram, we decompose the outer shape into a pasting of three commuting pentagons (by the pentagonators) and two commuting squares (by Proposition 2.1.4). Thus the entire diagram commutes. □

Corollary 3.2.6. *Horizontal pullback pasting*. Since the transpose of a pullback is a pullback (Proposition 3.2.3), by taking transposes as appropriate we deduce the more familiar horizontal version of the pullback pasting lemma.

Lemma 3.2.7. *Pullback prism*. Suppose we have a diagram

$$\begin{tikzcd}
 & P \arrow[rr, "\pi_B"] \arrow[dd, "\pi_A"] & & B \arrow[dd, "g"] \\
P' \arrow[rr, "\pi_{B'}"] \arrow[dr, "\pi_A'"'] & & B' \arrow[ur, "h"] \arrow[dr, "g'"'] & \\
 & A \arrow[rr, "f"'] & & C
\end{tikzcd}$$

in a 2-coherent wild category $\mathscr{C}$, such that $c : g \diamond h = g'$ is a commuting triangle, $\mathfrak{p} : f \diamond \pi_A = g \diamond \pi_B$ and $\mathfrak{p}' : f \diamond \pi_A' = g' \diamond \pi_{B'}$, and where the squares $\mathfrak{P} :\equiv (\pi_A, \pi_B, \mathfrak{p}) : \mathrm{CommSq}_{(f,g)}(P)$ and $\mathfrak{P}' :\equiv (\pi_A', \pi_{B'}, \mathfrak{p}') : \mathrm{CommSq}_{(f,g')}(P')$ are both pullbacks. Then there is a contractible type of data consisting of:

- a morphism $m : \mathscr{C}(P', P)$,
- equalities $e : \pi_A \diamond m = \pi_A{}'$ and $\mathfrak{q} : \pi_B \diamond m = h \diamond \pi_{B'}$ completing the boundary of the prism, and
- an equality 3-cell η filling the volume of the completed prism.

Even more, the top face $(m, \pi_{B'}, \mathfrak{q})$ of the completed prism is a pullback of (π_B, h).

Proof. From the universal property of $\mathfrak{P}$ we get m, e, $\mathfrak{q}$ and the equality

$$\eta : \alpha^{-1} \cdot (\mathfrak{p} * m) \cdot \alpha = (f * e) \cdot \mathfrak{p}' \cdot (c^{-1} * \pi_{B'}) \cdot \alpha \cdot (g * \mathfrak{q})^{-1}$$

as the center of contraction of the fiber of $(\mathfrak{P} \square_{P'} \_)$ at the commuting square

$$\mathfrak{S} :\equiv \left(\pi_A{}',\ h \diamond \pi_{B'},\ \mathfrak{p}' \cdot (c^{-1} * \pi_{B'}) \cdot \alpha\right)$$

on (f, g). Let $\mathfrak{Q} :\equiv (m, \pi_{B'}, \mathfrak{q})$; then by η and Lemma 3.1.16 it follows that $(f, g \diamond h, \frac{\mathfrak{Q}}{\mathfrak{P}})$ and $(f, g', \mathfrak{P}')$ are equal commuting squares on P', A, B', C. Since $\mathfrak{P}'$ is a pullback, by transport so is $\frac{\mathfrak{Q}}{\mathfrak{P}}$, and by pullback pasting (Lemma 3.2.5) so too is $\mathfrak{Q}$. □

3.3. The truncation level of pullbacks

Definition 3.3.1. *Pullbacks on a cospan.* As observed earlier, the predicate

$$\text{is-pullback}_{\mathfrak{c}, P}(\mathfrak{P}) :\equiv \prod\nolimits_{(X : \mathscr{C}_0)} \text{is-equiv}\,(\mathfrak{P} \square_X \_)$$

on $\text{CommSq}_{\mathfrak{c}}(P)$ is propositional. We obtain the subtype of pullbacks on a cospan $\mathfrak{c}$, with fixed source

$$\text{Pullback}_{\mathfrak{c}}(P) :\equiv \sum\nolimits_{(\mathfrak{P} : \text{CommSq}_{\mathfrak{c}}(P))} \text{is-pullback}_{\mathfrak{c}, P}(\mathfrak{P}),$$

and with arbitrary source,

$$\text{Pullback}(\mathfrak{c}) :\equiv \sum\nolimits_{((P, \mathfrak{P}) : \text{CommSq}(\mathfrak{c}))} \text{is-pullback}_{\mathfrak{c}, P}(\mathfrak{P}).$$

Proposition 3.3.2. $\text{Pullback}(\mathfrak{c})$ *is a set in set-level categories.* If $\mathscr{C}$ is set-level then $\text{Pullback}(\mathfrak{c})$ is a Σ-type of sets and propositions for any cospan $\mathfrak{c}$, and thus also a set.

Proposition 3.3.3. $\text{Pullback}(\mathfrak{c})$ *is a proposition in univalent 2-coherent wild categories.* If $\mathscr{C}$ is a univalent 2-coherent wild category, then $\text{Pullback}(\mathfrak{c})$ is a proposition for any cospan $\mathfrak{c}$.

Proof. In summary, by univalence and the universal property of pullbacks. Suppose that $(P, \mathfrak{P})$ and $(P', \mathfrak{P}')$ are elements of $\text{Pullback}(\mathfrak{c})$. Then $(\mathfrak{P} \square \_)$ and $(\mathfrak{P}' \square \_)$ are equivalences, and from the centers of contraction of $(\mathfrak{P} \square \_)^{-1}(\mathfrak{P}')$ and $(\mathfrak{P}' \square \_)^{-1}(\mathfrak{P})$ we get $m : \mathscr{C}(P, P')$ and $m' : \mathscr{C}(P', P)$ such that

$$e : \mathfrak{P}' \square m = \mathfrak{P} \quad \text{and} \quad e' : \mathfrak{P} \square m' = \mathfrak{P}'.$$

Furthermore, by Lemma 3.1.10

$$\mathfrak{P} \square (m' \diamond m) \;=\; (\mathfrak{P} \square m') \square m \;=\; \mathfrak{P}' \square m \;=\; \mathfrak{P},$$

and so $m' \diamond m = \mathrm{id}_P$ by contractibility of $(\mathfrak{P} \mathbin{\square} \_)^{-1}(\mathfrak{P})$ and Lemma 3.1.10 again.

By a similar argument $m \diamond m' = \mathrm{id}_{P'}$, and so $m : \mathscr{C}(P, P')$ is a $\mathscr{C}$-equivalence. From univalence of $\mathscr{C}$ we now get an equality

$$\mathrm{eqvtoid}_{\mathscr{C}}(m) : P = P',$$

with

$$\mathfrak{P} \quad = \quad \mathfrak{P}' \mathbin{\square} m \quad = \quad \mathfrak{P}' \mathbin{\square} \mathrm{idd}(\mathrm{eqvtoid}_{\mathscr{C}}(m)).$$

By Corollary 3.1.13, this proves $(P, \mathfrak{P}) = (P', \mathfrak{P}')$. □

4. Wild Categories with Families

We can now define precoherent higher internal models of homotopical MLTT. We begin by simply taking Dybjer's generalized algebraic definition of a category with families [Dyb96], and allowing contexts to form wild categories. This notion has previously been briefly considered by Kraus [Kra21a, Definition 5].

4.1. Typed term structures

Definition 4.1.1. *Typed term structures on wild categories.* Let $\mathcal{U}$ be a universe and $\mathscr{C}$ a wild category. A *typed term structure* on $\mathscr{C}$ (valued in $\mathcal{U}$) consists of the following data:

- A wild $\mathcal{U}$-valued presheaf of $\mathscr{C}$-**types** over $\mathscr{C}$, presented as a generalized algebraic theory by the components[11]

$$\begin{aligned} \mathsf{Ty} &: \mathscr{C}_0 \to \mathcal{U} \\ \_[\_]_{\mathsf{T}} &: \mathsf{Ty}\,\Delta \to \mathscr{C}(\Gamma, \Delta) \to \mathsf{Ty}\,\Gamma \end{aligned}$$

  and equations[11] expressing functoriality

$$\begin{aligned} [\mathrm{id}]_{\mathsf{T}} &: A[\mathrm{id}_\Gamma]_{\mathsf{T}} = A && \text{for all} \quad A : \mathsf{Ty}\,\Gamma \\ [\diamond]_{\mathsf{T}} &: A[\tau \diamond \sigma]_{\mathsf{T}} = A[\tau]_{\mathsf{T}}[\sigma]_{\mathsf{T}} && \text{for all} \quad A : \mathsf{Ty}\,\mathrm{E},\ \sigma : \mathscr{C}(\Gamma, \Delta),\ \tau : \mathscr{C}(\Delta, \mathrm{E}). \end{aligned}$$

- A wild $\mathcal{U}$-valued presheaf of $\mathscr{C}$-**terms** over the (wild) category of elements of the $\mathscr{C}$-type presheaf, presented[11] by

$$\begin{aligned} \mathsf{Tm} &: (\Gamma : \mathscr{C}_0) \to \mathsf{Ty}\,\Gamma \to \mathcal{U} \\ \_[\_]_{\mathsf{t}} &: \mathsf{Tm}_\Delta\,A \to (\sigma : \mathscr{C}(\Gamma, \Delta)) \to \mathsf{Tm}_\Gamma\,(A[\sigma]_{\mathsf{T}}) && \text{for all} \quad A : \mathsf{Ty}\,\Delta \end{aligned}$$

  and

$$\begin{aligned} [\mathrm{id}]_{\mathsf{t}} &: a[\mathrm{id}_\Gamma]_{\mathsf{t}} = a{\downarrow}^{\mathsf{Tm}_\Gamma}_{[\mathrm{id}]_{\mathsf{T}}{}^{-1}} && \text{for all} \quad A : \mathsf{Ty}\,\Gamma,\ a : \mathsf{Tm}_\Gamma\,A \\ [\diamond]_{\mathsf{t}} &: a[\tau \diamond \sigma]_{\mathsf{t}} = a[\tau]_{\mathsf{t}}[\sigma]_{\mathsf{t}}{\downarrow}^{\mathsf{Tm}_\Gamma}_{[\diamond]_{\mathsf{T}}{}^{-1}} && \text{for all} \quad A : \mathsf{Ty}\,\mathrm{E},\ a : \mathsf{Tm}_{\mathrm{E}}\,A \\ & && \qquad \sigma : \mathscr{C}(\Gamma, \Delta),\ \tau : \mathscr{C}(\Delta, \mathrm{E}). \end{aligned}$$

The actions $\_[\_]_{\mathsf{T}}$ and $\_[\_]_{\mathsf{t}}$ of the type and term presheaves on morphisms are called *substitution in types* and *substitution in terms*, respectively.

[11] Implicitly quantifying over objects $\Gamma, \Delta, \mathrm{E} : \mathscr{C}_0$ as needed.

We will often denote a typed term structure on a wild category simply by the object parts of its component presheaves $(\mathsf{Ty}, \mathsf{Tm})$. We also frequently elide the first argument of Tm and write, for example, $\mathsf{Tm}\, A$ instead of $\mathsf{Tm}_{\Gamma}\, A$.

Note the following equivalences and equalities in typed term structures $(\mathsf{Ty}, \mathsf{Tm})$ on wild categories $\mathscr{C}$.

Proposition 4.1.2. For every $\Gamma : \mathscr{C}_0$ and $A : \mathsf{Ty}\,\Gamma$, the equation $[\mathrm{id}]_{\mathsf{t}}$ (Definition 4.1.1) implies that the function

$$\begin{aligned} \_[\mathrm{id}]_{\mathsf{t}} \;:\; \mathsf{Tm}\, A &\longrightarrow \mathsf{Tm}\,(A[\mathrm{id}]_{\mathsf{T}}) \\ a &\longmapsto a[\mathrm{id}]_{\mathsf{t}} \end{aligned}$$

is equal to transport in Tm_{Γ} along ${[\mathrm{id}]_{\mathsf{T}}}^{-1}$, and is hence an equivalence.

Definition 4.1.3. Assume objects $\Gamma, \Delta : \mathscr{C}_0$, a $\mathscr{C}$-type $A : \mathsf{Ty}\,\Delta$, and an equality $e : \sigma = \tau$ of morphisms $\sigma, \tau : \mathscr{C}(\Gamma,\, \Delta)$. We write

$$[^{=}e]_{\mathsf{T}} :\equiv \mathrm{ap}\;(A[\_]_{\mathsf{T}})\; e$$

for the induced equality $A[\sigma]_{\mathsf{T}} = A[\tau]_{\mathsf{T}}$. By induction on e, we also have an equality

$$[^{=}e]_{\mathsf{t}} : a[\sigma]_{\mathsf{t}} {\downarrow}^{\mathsf{Tm}}_{[^{=}e]_{\mathsf{T}}} = a[\tau]_{\mathsf{t}}$$

for any $\mathscr{C}$-term $a : \mathsf{Tm}\, A$.

By [Uni13, Lemma 2.2.2], $[^{=}\_]_{\mathsf{T}}$ respects trivial, composite and inverse equalities. Furthermore,

Proposition 4.1.4. For any $\Gamma, \Delta : \mathscr{C}_0$, $A : \mathsf{Ty}\,\Delta$, $a : \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}})$ and morphisms $\sigma, \tau : \mathscr{C}(\Gamma,\, \Delta)$ such that $e : \sigma = \tau$,

$$a {\downarrow}^{\mathsf{Tm}\,(A[\_]_{\mathsf{T}})}_{e} = a {\downarrow}^{\mathsf{Tm}}_{[^{=}e]_{\mathsf{T}}}$$

by [Uni13, Lemma 2.3.10].

Definition 4.1.5. Suppose that $A, A' : \mathsf{Ty}\,\Delta$ are $\mathscr{C}$-types such that $e : A = A'$. For any $\sigma : \mathscr{C}(\Gamma,\, \Delta)$, we write

$$e[\sigma]_{\mathsf{T}} :\equiv \mathrm{ap}\;(\_[\sigma]_{\mathsf{T}})\; e$$

for the induced equality $A[\sigma]_{\mathsf{T}} = A'[\sigma]_{\mathsf{T}}$.

Similarly, if $a, a' : \mathsf{Tm}\, A$ with $e : a = a'$, we write

$$e[\sigma]_{\mathsf{t}} :\equiv \mathrm{ap}\;(\_[\sigma]_{\mathsf{t}})\; e$$

for the induced equality $a[\sigma]_{\mathsf{t}} = a'[\sigma]_{\mathsf{t}}$.

Proposition 4.1.6. *Substitution in transported terms.* If $e : A =_{\mathsf{Ty}\,\Delta} A'$, then for any $a : \mathsf{Tm}\, A$ and morphism $\sigma : \mathscr{C}(\Gamma,\, \Delta)$,

$$(a {\downarrow}^{\mathsf{Tm}}_{e})[\sigma]_{\mathsf{t}} = a[\sigma]_{\mathsf{t}} {\downarrow}^{\mathsf{Tm}}_{e[\sigma]_{\mathsf{T}}}$$

by induction on e.

Proposition 4.1.7. $[\diamond]_{\mathrm{T}}$ *is a natural isomorphism.* Suppose that $\sigma, \sigma' : \mathscr{C}(\Gamma, \Delta)$ are morphisms such that $e : \sigma = \sigma'$. By induction on e, we have that the square

$$\begin{tikzcd}
A[\tau \diamond \sigma]_{\mathrm{T}} \arrow[r, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] \arrow[d, Rightarrow, "{[^{=}\tau * e]_{\mathrm{T}}}"'] & A[\tau]_{\mathrm{T}}[\sigma]_{\mathrm{T}} \arrow[d, Rightarrow, "{[^{=}e]_{\mathrm{T}}}"] \\
A[\tau \diamond \sigma']_{\mathrm{T}} \arrow[r, Rightarrow, "{[\diamond]_{\mathrm{T}}}"'] & A[\tau]_{\mathrm{T}}[\sigma']_{\mathrm{T}}
\end{tikzcd}$$

canonically commutes for all $A : \mathrm{Ty}\,\mathrm{E}$ and $\tau : \mathscr{C}(\Delta, \mathrm{E})$, and also that

$$\begin{tikzcd}
A[\sigma \diamond \varrho]_{\mathrm{T}} \arrow[r, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] \arrow[d, Rightarrow, "{[^{=}e * \varrho]_{\mathrm{T}}}"'] & A[\sigma]_{\mathrm{T}}[\varrho]_{\mathrm{T}} \arrow[d, Rightarrow, "{[^{=}e]_{\mathrm{T}}[\varrho]_{\mathrm{T}}}"] \\
A[\sigma' \diamond \varrho]_{\mathrm{T}} \arrow[r, Rightarrow, "{[\diamond]_{\mathrm{T}}}"'] & A[\sigma']_{\mathrm{T}}[\varrho]_{\mathrm{T}}
\end{tikzcd}$$

canonically commutes for all $A : \mathrm{Ty}\,\Delta$ and $\varrho : \mathscr{C}(\mathrm{B}, \Gamma)$.

The following two definitions are analogous to the conditions for a pseudofunctor between weak (2, 1)-categories.[12] In the case that $\mathscr{C}$ is 2-coherent, they improve the wild presheaf Ty of a typed term structure on $\mathscr{C}$ to what might be called a *wild weak (2, 1)-presheaf.*

Definition 4.1.8. *Type triangulators.* A typed term structure on a wild category $\mathscr{C}$ is said to have *type triangulators* if for all morphisms $\sigma : \mathscr{C}(\Gamma, \Delta)$ and $\mathscr{C}$-types $A : \mathrm{Ty}\,\Delta$ the following triangles commute:

$$\begin{tikzcd}
A[\mathrm{id} \diamond \sigma]_{\mathrm{T}} \arrow[rr, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] \arrow[dr, Rightarrow, "{[^{=}\lambda]_{\mathrm{T}}}"'] & & A[\mathrm{id}]_{\mathrm{T}}[\sigma]_{\mathrm{T}} \arrow[dl, Rightarrow, "{[\mathrm{id}]_{\mathrm{T}}[\sigma]_{\mathrm{T}}}"] \\
& A[\sigma]_{\mathrm{T}} &
\end{tikzcd}
\quad \text{and} \quad
\begin{tikzcd}
A[\sigma \diamond \mathrm{id}]_{\mathrm{T}} \arrow[rr, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] \arrow[dr, Rightarrow, "{[^{=}\rho]_{\mathrm{T}}}"'] & & A[\sigma]_{\mathrm{T}}[\mathrm{id}]_{\mathrm{T}} \arrow[dl, Rightarrow, "{[\mathrm{id}]_{\mathrm{T}}}"] \\
& A[\sigma]_{\mathrm{T}} &
\end{tikzcd}$$

Definition 4.1.9. *Type pentagonators.* A typed term structure on a wild category $\mathscr{C}$ has *type pentagonators* if for all morphisms

$$\Gamma \xrightarrow{\varrho} \Delta \xrightarrow{\sigma} \mathrm{E} \xrightarrow{\tau} \mathrm{Z}$$

and $\mathscr{C}$-types $A : \mathrm{Ty}\,\mathrm{Z}$, the following pentagon commutes:

$$\begin{tikzcd}
& A[\tau \diamond \sigma \diamond \varrho]_{\mathrm{T}} \arrow[dl, Rightarrow, "{[^{=}\alpha^{-1}]_{\mathrm{T}}}"'] \arrow[dr, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] & \\
A[(\tau \diamond \sigma) \diamond \varrho]_{\mathrm{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathrm{T}}}"'] & & A[\tau]_{\mathrm{T}}[\sigma \diamond \varrho]_{\mathrm{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathrm{T}}}"] \\
A[\tau \diamond \sigma]_{\mathrm{T}}[\varrho]_{\mathrm{T}} \arrow[rr, Rightarrow, "{[\diamond]_{\mathrm{T}}[\varrho]_{\mathrm{T}}}"'] & & A[\tau]_{\mathrm{T}}[\sigma]_{\mathrm{T}}[\varrho]_{\mathrm{T}}
\end{tikzcd}$$

[12] See e.g. [JY21, §4.1].

4.2. Context extension structures and wild cwfs

Definition 4.2.1. *Context extension structures.* Assume a typed term structure $(\mathsf{Ty}, \mathsf{Tm})$ on a wild category $\mathscr{C}$. A *context extension structure* on $(\mathscr{C}, \mathsf{Ty}, \mathsf{Tm})$ is given[13] by the following components

$$\begin{aligned} \_.\_ &: (\Gamma : \mathscr{C}_0) \to \mathsf{Ty}\,\Gamma \to \mathscr{C}_0 \\ \mathsf{p} &: (A : \mathsf{Ty}\,\Gamma) \to \mathscr{C}(\Gamma.A, \Gamma) \\ \mathsf{q} &: (A : \mathsf{Ty}\,\Gamma) \to \mathsf{Tm}_{\Gamma.A}\,(A[\mathsf{p}_A]_\mathsf{T}) \\ \_,\_ &: (\sigma : \mathscr{C}(\Gamma, \Delta)) \to \mathsf{Tm}_\Gamma\,(A[\sigma]_\mathsf{T}) \to \mathscr{C}(\Gamma, \Delta.A) \quad \text{for all} \quad A : \mathsf{Ty}\,\Delta \end{aligned}$$

and equations (note Definition 4.1.3)

$$\begin{aligned} \mathsf{p}\beta &: \mathsf{p}_A \diamond (\sigma, a) = \sigma && \text{and} \\ \mathsf{q}\beta &: \mathsf{q}_A[\sigma, a]_\mathsf{t} = a\downarrow^{\mathsf{Tm}}_{[{}^{=}\mathsf{p}\beta]_\mathsf{T}{}^{-1}\cdot[\diamond]_\mathsf{T}} && \text{for all} \quad \sigma : \mathscr{C}(\Gamma, \Delta), \\ & && \qquad\qquad A : \mathsf{Ty}\,\Delta,\ a : \mathsf{Tm}_\Gamma\,(A[\sigma]_\mathsf{T}) \\ ,\eta &: (\mathsf{p}_A, \mathsf{q}_A) = \mathrm{id}_{\Gamma.A} && \text{for all} \quad A : \mathsf{Ty}\,\Gamma \\ ,\diamond &: (\tau, a) \diamond \sigma = (\tau \diamond \sigma, a[\sigma]_\mathsf{t}\downarrow^{\mathsf{Tm}}_{[\diamond]_\mathsf{T}{}^{-1}}) && \text{for all} \quad \sigma : \mathscr{C}(\Gamma, \Delta),\ \tau : \mathscr{C}(\Delta, \mathrm{E}), \\ & && \qquad\qquad A : \mathsf{Ty}\,\mathrm{E},\ a : \mathsf{Tm}_\Delta\,(A[\tau]_\mathsf{T}). \end{aligned}$$

We call p the *display map*, and q the *generic term* of the context extension structure.

We sometimes elide the argument $A : \mathsf{Ty}\,\Gamma$ to the display map p and the generic term q of a context extension structure. When we need to be concise, we denote the display map $\Gamma.A \xrightarrow{\mathsf{p}_A} \Gamma$ by $\Gamma.A \twoheadrightarrow \Gamma$.

Definition 4.2.2. *Cwf structures on wild categories.* If $\mathscr{C}$ is a wild category, a *cwf structure* on $\mathscr{C}$ consists of:

- a terminal object $\blacklozenge : \mathscr{C}_0$,
- a typed term structure $(\mathsf{Ty}, \mathsf{Tm})$ on $\mathscr{C}$, and
- a context extension structure on $(\mathscr{C}, \mathsf{Ty}, \mathsf{Tm})$,

which model the structural rules of a Martin-Löf type theory over $\mathscr{C}$.

Definition 4.2.3. *Wild categories with families.* A *wild category with families (wild cwf)* is a wild category $\mathscr{C}$ together with a cwf structure on $\mathscr{C}$. In this case, we call $\mathscr{C}$ the *category of contexts* of the wild cwf, its objects *contexts*, and its morphisms *substitutions*.

We usually denote a wild cwf by its category of contexts.
Of course, every 1-cwf is a wild cwf.

Example 4.2.4. *Universe cwfs.* If a universe wild category $\mathcal{U}$ has Σ-types that satisfy the η-rule, then it supports a canonical wild cwf structure given as follows.

- The terminal context $\blacklozenge$ is the unit type $\mathbb{1} : \mathcal{U}$.

[13] Again, implicitly generalizing over $\Gamma, \Delta, \mathrm{E} : \mathscr{C}_0$ as needed.

- The typed term structure is as follows. $\mathcal{U}$-types in context Γ are Γ-indexed type families

$$\begin{aligned}\mathsf{Ty} &: \mathcal{U} \to \mathcal{U}^+\\ \mathsf{Ty}\,\Delta &:\equiv \Delta \to \mathcal{U},\end{aligned}$$

  while $\mathcal{U}$-terms $a : \mathsf{Tm}_\Delta\, A$ are sections of $A : \mathsf{Ty}\,\Delta$

$$\mathsf{Tm}_\Delta\, A :\equiv \Pi\, \Delta\, A.$$

  Substitution of $\sigma : \mathcal{U}(\Gamma, \Delta)$ in $\mathcal{U}$-types $A : \mathsf{Ty}\,\Delta$ and $\mathcal{U}$-terms a is given by precomposition

$$\begin{aligned}A[\sigma]_\mathsf{T} &:\equiv A \circ \sigma,\\ a[\sigma]_\mathsf{t} &:\equiv a \circ \sigma.\end{aligned}$$

  This action is definitionally functorial—that is, $[\mathrm{id}]_\mathsf{T}$, $[\diamond]_\mathsf{T}$, $[\mathrm{id}]_\mathsf{t}$ and $[\diamond]_\mathsf{t}$ are all families of trivial identities.
- The context extension structure is given by dependent pairing. The extended context $\Delta.A$ is $\Sigma\,\Delta\,A$, and p and q are the functions fst and snd respectively. For $\sigma : \mathcal{U}(\Gamma, \Delta)$ and $t : \mathsf{Tm}_\Gamma\,(A \circ \sigma)$, the extended substitution $(\sigma, t) : \mathcal{U}(\Gamma, \Sigma\,\Delta\,A)$ is given by

$$(\sigma, t)(\gamma) :\equiv (\sigma(\gamma), t(\gamma)).$$

  Again, the equations for context extension structures hold definitionally. In particular, the η-rule for Σ-types is used for , η.

We refer to the resulting wild cwf as the *universe cwf*.

Variations of this canonical universe cwf structure appear throughout the literature as the "standard model".

Example 4.2.5. *Subuniverse cwfs and the 1-cwf of sets.* The construction of the typed term and context extension structures of Example 4.2.4 works equally well for any *sub*universe wild category (Examples 2.0.2) that has a terminal object and is closed under Σ-types with η. In particular, the 1-cwf of sets $\mathrm{Set}_\mathcal{U}$ is a subuniverse cwf of $\mathcal{U}$.

Definition 4.2.6. *Univalent wild cwfs.* A wild cwf $\mathscr{C}$ is called *univalent* if its category of contexts is univalent.

Example 4.2.7. *Univalent wild cwfs.* Any subuniverse of a univalent universe $\mathcal{U}$ yields a univalent cwf. In particular, $\mathrm{Set}_\mathcal{U}$ and $\mathcal{U}$ are univalent cwfs if $\mathcal{U}$ is a univalent universe.

4.3. Structural properties of wild cwfs

From now on we assume that $\mathscr{C}$ is a wild cwf.

Lemma 4.3.1. *Substitutions into extended contexts are pairs.* Let $\Gamma, \Delta : \mathscr{C}_0$ be contexts, and $A : \mathsf{Ty}\,\Delta$ a $\mathscr{C}$-type. There is an equivalence

$$\begin{tikzcd}\mathscr{C}(\Gamma, \Delta.A) \arrow[r, bend left, "\sigma \mapsto (\mathsf{p}_A \diamond \sigma,\, \mathsf{q}_A[\sigma]_\mathsf{t} \downarrow^{\mathsf{Tm}}_{[\diamond]_\mathsf{T}{}^{-1}})"] \arrow[r, phantom, "\simeq"] & \sum_{(\sigma : \mathscr{C}(\Gamma, \Delta))} \mathsf{Tm}\,(A[\sigma]_\mathsf{T}), \arrow[l, bend left, "(\sigma, a) \mapsfrom (\sigma, a)"]\end{tikzcd}$$

where the reverse function sends a pair (σ, a) to the extended substitution $(\sigma\,, a)$ given by the context extension structure (Definition 4.2.1).

Proof. For one composition, it's enough to show that for all $\sigma : \mathscr{C}(\Gamma,\ \Delta.A)$,

$$(\mathsf{p}_A \diamond \sigma\,, \mathsf{q}_A[\sigma]_{\mathsf{t}\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}) \overset{,\,\diamond^{-1}}{=\!=\!=} (\mathsf{p}_A\,, \mathsf{q}_A) \diamond \sigma \overset{,\,\eta * \sigma}{=\!=\!=} \mathrm{id} \diamond \sigma \overset{\lambda}{=\!=} \sigma.$$

For the other, we show the equality of pairs

$$(\mathsf{p}_A \diamond (\sigma, a), \mathsf{q}_A[\sigma\,, a]_{\mathsf{t}\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}) = (\sigma, a).$$

Equality of the first components is given by $\mathsf{p}\beta$ (Definition 4.2.1), and for the second components we have that

$$\begin{aligned}
&\mathsf{q}_A[\sigma\,, a]_{\mathsf{t}\downarrow^{\mathsf{Tm}}_{[\diamond]_{\mathsf{T}}{}^{-1}}\downarrow^{\mathsf{Tm}\,(A[\_]_{\mathsf{T}})}_{\mathsf{p}\beta}} \\
= \ &\mathsf{q}_A[\sigma\,, a]_{\mathsf{t}\downarrow^{\mathsf{Tm}}_{[\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}\mathsf{p}\beta]_{\mathsf{T}}}} && \text{(by Proposition 4.1.4)} \\
= \ &a && \text{(by p}\beta\text{ and properties of transport).} \qquad \square
\end{aligned}$$

Corollary 4.3.2. *Elimination principle for* $\mathscr{C}(\Gamma,\ \Delta.A)$. By Lemma 4.3.1, to construct a section of a family of types P over $\mathscr{C}(\Gamma,\ \Delta.A)$, it's enough to give an element of $P((\sigma\,, a))$ for every $\sigma : \mathscr{C}(\Gamma,\ \Delta)$ and $a : \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}})$.

Corollary 4.3.3. *Equality of substitutions into extended contexts.* If σ and τ are substitutions from Γ to $\Delta.A$, then by Lemma 4.3.1, the fact that equivalences induce equivalent identity types [Uni13, Theorem 2.11.1], and Proposition 4.1.4, an equality $\sigma = \tau$ is equivalent to a pair of equalities

$$e : \mathsf{p} \diamond \sigma = \mathsf{p} \diamond \tau \quad \text{and} \quad \mathsf{q}[\sigma]_{\mathsf{t}\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}e]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}}.$$

An alternative but equivalent formulation is the following—for substitutions of the form $(\sigma, a), (\tau, b) : \mathscr{C}(\Gamma,\ \Delta.A)$,

$$\big((\sigma, a) =_{\mathscr{C}(\Gamma,\ \Delta.A)} (\tau, b)\big) \ \simeq\ \big(\textstyle\sum_{(e\,:\,\sigma = \tau)},\ a\downarrow^{\mathsf{Tm}}_{[^{=}e]_{\mathsf{T}}} = b\big).$$

To see this, write φ for the forward equivalence of Lemma 4.3.1. In the proof of Lemma 4.3.1 we showed that the equalities of pairs $\varphi((\sigma, a)) = (\sigma, a)$ and $\varphi((\tau, b)) = (\tau, b)$ hold. Then $\varphi((\sigma, a)) = \varphi((\tau, b))$ is equivalent to the equality type of pairs $(\sigma, a) = (\tau, b)$, which by Proposition 4.1.4 is equivalent to the Σ-type as claimed.

Lemma 4.3.4. *Terms are sections of display maps.* For all contexts $\Gamma : \mathscr{C}_0$ and $\mathscr{C}$-types $A : \mathsf{Ty}\,\Gamma$, there is an equivalence

$$\mathsf{Tm}\,A \simeq \mathsf{Sect}(\mathsf{p}_A)$$

whose forward map sends the $\mathscr{C}$-term a to the section $(\mathrm{id}\,, a[\mathrm{id}]_{\mathsf{t}})$ of p_A, witnessed by $\mathsf{p}\beta$.

Proof. We have that

$$
\begin{aligned}
&\mathrm{Sect}(\mathrm{p}_A) \\
\equiv\ & \textstyle\sum (\sigma : \mathscr{C}(\Gamma, \Gamma.A)),\ \mathrm{p}_A \diamond \sigma = \mathrm{id} \\
\simeq\ & \textstyle\sum (u : \Sigma\, (\sigma : \mathscr{C}(\Gamma, \Gamma)), \mathsf{Tm}\,(A[\mathrm{id}]_{\mathsf{T}})),\ \mathrm{p}_A \diamond (\mathrm{fst}\, u, \mathrm{snd}\, u) = \mathrm{id} && \text{(by Lemma 4.3.1)} \\
\simeq\ & \textstyle\sum (u : \Sigma\, (\sigma : \mathscr{C}(\Gamma, \Gamma)), \mathsf{Tm}\,(A[\mathrm{id}]_{\mathsf{T}})),\ \mathrm{fst}\, u = \mathrm{id} && \text{(by p}\beta\text{ (Definition 4.2.1))} \\
\simeq\ & \textstyle\sum (u : \Sigma\, (\sigma : \mathscr{C}(\Gamma, \Gamma)), \sigma = \mathrm{id}),\ \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}}) && \text{(assoc. of } \Sigma \text{ and comm. of } \times\text{)} \\
\simeq\ & \mathsf{Tm}\,(A[\mathrm{id}]_{\mathsf{T}}) && \text{(contractibility of singletons)} \\
\simeq\ & \mathsf{Tm}\, A && \text{(by the inverse of the equivalence } \_[\mathrm{id}]_{\mathsf{t}} \text{ (Proposition 4.1.2)).}
\end{aligned}
$$

Tracing the composition of this chain of equivalences, we compute that its inverse is equal to the map

$$
\begin{aligned}
\mathsf{Tm}\, A &\longrightarrow \mathrm{Sect}(\mathrm{p}_A) \\
a &\longmapsto ((\mathrm{id}, a[\mathrm{id}]_{\mathsf{t}}), \mathrm{p}\beta).
\end{aligned}
\qquad \square
$$

Analogues of Lemmas 4.3.1 and 4.3.4 were already observed for set-level 1-cwfs by Dybjer [Dyb96] and by Hofmann [Hof97] in a traditional 1-categorical setting. In that setting, these properties essentially follow from the fact that context extension structures on ($\mathscr{C}$, Ty, Tm) are choices of representing objects for particular presheaves on slices of $\mathscr{C}$.

It may seem slightly surprising that the fully coherent homotopical versions of the same properties hold for arbitrary, even noncoherent, wild cwfs. On the other hand, given that a context extension structure for 1-cwfs essentially spells out the universal property of representability of a certain presheaf (and, relatedly, that the equivalent natural models [Awo18] have a relatively simple axiomatization in terms of representability of pullbacks of presheaves), it is perhaps to be expected that certain consequences would carry over immediately to higher generalizations even without imposing additional coherence conditions.

5. 2-Coherence for Context Extension

We now construct and analyze in more detail a characterization of the equality of substitutions into extended contexts.

First, consider the case where σ and τ are substitutions from Γ to *arbitrary* contexts Δ, with $A : \mathsf{Ty}\,\Delta$, $a : \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}})$ and $b : \mathsf{Tm}\,(A[\tau]_{\mathsf{T}})$. From the equivalence

$$\Big(\textstyle\sum (\sigma : \mathscr{C}(\Gamma, \Delta)),\ \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}})\Big) \xrightarrow[(\_,\_)]{\sim} \mathscr{C}(\Gamma, \Delta.A)$$

of Lemma 4.3.1, we obtain

$$\Big((\sigma, a) =_{\Sigma\,(\mathscr{C}(\Gamma, \Delta.A))\,(\mathsf{Tm}\,(A[\_]_{\mathsf{T}}))} (\tau, b)\Big) \xrightarrow[\mathrm{ap}\,(\_,\_)]{\sim} \Big((\sigma, a) =_{\mathscr{C}(\Gamma, \Delta.A)} (\tau, b)\Big).$$

Precomposing this with

$$
\begin{aligned}
\Big(\textstyle\sum (e : \sigma = \tau),\ a \downarrow^{\mathsf{Tm}}_{[^{=}e]_{\mathsf{T}}} = b\Big) &\xrightarrow[\mathrm{id} \times \varphi]{\sim} \textstyle\sum (e : \sigma = \tau),\ a \downarrow^{\mathsf{Tm}\,(A[\_]_{\mathsf{T}})}_{e} = b \\
&\xrightarrow[\mathrm{pair}^{=}]{\sim} \Big((\sigma, a) =_{\Sigma\,(\mathscr{C}(\Gamma, \Delta.A))\,(\mathsf{Tm}\,(A[\_]_{\mathsf{T}}))} (\tau, b)\Big),
\end{aligned}
$$

where

$$\varphi : \prod\,(e : \sigma = \tau)\ \left(a{\downarrow}^{\mathsf{Tm}}_{[{}^{=}e]_{\mathsf{T}}} = b \xrightarrow{\sim} a{\downarrow}^{\mathsf{Tm}\,(A[\_]_{\mathsf{T}})}_{e} = b\right)$$

is the family of equivalences induced by Proposition 4.1.4 and $\mathsf{pair}^{=}$ is the standard characterization of the equality of Σ-types, we get an equivalence

$$\begin{gathered} \mathsf{sub}_0^{=} : \left(\sum\,(e : \sigma = \tau),\ a{\downarrow}^{\mathsf{Tm}}_{[{}^{=}e]_{\mathsf{T}}} = b\right) \xrightarrow{\sim} (\sigma, a) =_{\mathscr{C}(\Gamma,\,\Delta.A)} (\tau, b) \\ \mathsf{sub}_0^{=}\,(e, e') :\equiv \mathsf{ap}\,(\_,\_)\,(\mathsf{pair}^{=}\,(e, \varphi_e e')). \end{gathered}$$

Lemma 5.0.1. $\mathsf{p\beta}$ *is a natural isomorphism.* By definition, $\mathsf{sub}_0^{=}\,(\mathsf{refl}, \mathsf{refl}) \equiv \mathsf{refl}$. Hence for all substitutions $\sigma, \tau : \mathscr{C}(\Gamma,\ \Delta.A)$, $\mathscr{C}$-terms $a : \mathsf{Tm}\,(A[\sigma]_{\mathsf{T}})$ and $b : \mathsf{Tm}\,(A[\tau]_{\mathsf{T}})$, and equalities $e : \sigma = \tau$ and $e' : a_{\downarrow\,[{}^{=}e]_{\mathsf{T}}} = b$, we have that the square

$$\begin{tikzcd} \mathsf{p}_A \diamond (\sigma, a) \arrow[r, Rightarrow, "\mathsf{p\beta}"] \arrow[d, Rightarrow, "\mathsf{p}_A * \mathsf{sub}_0^{=}\,(e{,}e')"'] & \sigma \arrow[d, Rightarrow, "e"] \\ \mathsf{p}_A \diamond (\tau, b) \arrow[r, Rightarrow, "\mathsf{p\beta}"'] & \tau \end{tikzcd}$$

canonically commutes by induction on e and e'. Equivalently,

$$\mathsf{p}_A * \mathsf{sub}_0^{=}\,(e, e') = \mathsf{p\beta} \cdot e \cdot \mathsf{p\beta}^{-1}.$$

Definition 5.0.2. η*-equality of substitutions.* For all $A : \mathsf{Ty}\,\Delta$ and $\sigma : \mathscr{C}(\Gamma,\ \Delta.A)$, denote by

$$\begin{gathered} \eta_{\sigma}^{\mathsf{sub}} : (\mathsf{p}_A \diamond \sigma, \mathsf{q}_A[\sigma]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}) = \sigma \\ \eta_{\sigma}^{\mathsf{sub}} :\equiv\ ,\diamond^{-1} \cdot (\,,\eta * \sigma) \cdot \lambda \end{gathered}$$

the equality in the first part of the proof of Lemma 4.3.1. This is an η-rule for substitutions into extended contexts.

Definition 5.0.3. *Equality of substitutions into extended contexts, revisited.* Suppose that $\sigma, \tau : \mathscr{C}(\Gamma,\ \Delta.A)$ are substitutions into an extended context. We define an equivalence

$$\mathsf{sub}_{\sigma,\tau}^{=} : \left(\sum\,(e : \mathsf{p} \diamond \sigma = \mathsf{p} \diamond \tau),\ \mathsf{q}[\sigma]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}\cdot[{}^{=}e]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}}\right) \xrightarrow{\sim} \sigma = \tau$$

as follows: if

$$e : \mathsf{p}_A \diamond \sigma = \mathsf{p}_A \diamond \tau$$

and

$$e' : \mathsf{q}[\sigma]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}\cdot[{}^{=}e]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}},$$

then take $\mathsf{sub}_{\sigma,\tau}^{=}\,(e, e')$ to be the composite

$$\sigma \overset{\eta_{\sigma}^{\mathsf{sub}\,-1}}{=\!=\!=} (\mathsf{p} \diamond \sigma, \mathsf{q}[\sigma]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}) \overset{\mathsf{sub}_0^{=}\,(e,e'')}{=\!=\!=\!=} (\mathsf{p} \diamond \tau, \mathsf{q}[\tau]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}) \overset{\eta_{\tau}^{\mathsf{sub}}}{=\!=\!=} \tau,$$

where $e'' : \mathsf{q}[\sigma]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}\,\downarrow\,[{}^{=}e]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}\,\downarrow\,[\diamond]_{\mathsf{T}}{}^{-1}}$ is canonically constructed from e'. This definition yields an equivalence, being essentially the composition of $\mathsf{sub}_0^{=}$ with the equivalence given by path-composing with $\eta_{\sigma}^{\mathsf{sub}\,-1}$ and $\eta_{\tau}^{\mathsf{sub}}$.

Now, it is natural to ask if a β-rule holds for the first argument of $\mathsf{sub}^{=}_{\sigma,\tau}$, i.e. if, for all σ and τ, the composition

$$\big(\textstyle\sum_{(\mathsf{p}\diamond\sigma=\mathsf{p}\diamond\tau)} \mathsf{q}[\sigma]_{\mathsf{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}\cdot[{}^{=}\_]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}}\big) \xrightarrow{\mathsf{sub}^{=}_{\sigma,\tau}} \sigma = \tau \xrightarrow{\mathsf{p}\,*\,\_} \mathsf{p}\diamond\sigma = \mathsf{p}\diamond\tau$$

is equal to the first projection. By Lemma 5.0.1, we calculate that

$$\mathsf{p} * \mathsf{sub}^{=}_{\sigma,\tau}(e,e') = (\mathsf{p} * \eta_{\sigma}^{\mathsf{sub}^{-1}}) \cdot \mathsf{p}\beta \cdot e \cdot \mathsf{p}\beta^{-1} \cdot (\mathsf{p} * \eta_{\tau}^{\mathsf{sub}})$$

for all e and e'. The desire to have this expression be equal to e motivates the next definition, whence Proposition 5.0.5 immediately follows.

Definition 5.0.4. *Coherators for* η^{sub}. We say that a wild cwf $\mathscr{C}$ *has coherators for* η^{sub} if for all $\Gamma, \Delta : \mathscr{C}_0$, $A : \mathsf{Ty}\,\Delta$ and $\sigma : \mathscr{C}(\Gamma,\ \Delta.A)$, we have that

$$\mathsf{p}_A * \eta_{\sigma}^{\mathsf{sub}} = \mathsf{p}\beta$$

as 2-cells of type $\mathsf{p}_A \diamond (\mathsf{p}_A \diamond \sigma, \mathsf{q}_A[\sigma]_{\mathsf{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}}) = \mathsf{p}_A \diamond \sigma$.

Proposition 5.0.5. β*-reduction for* $\mathsf{sub}^{=}_{\sigma,\tau}$. Suppose $\sigma, \tau : \mathscr{C}(\Gamma,\ \Delta.A)$ are equal substitutions, witnessed by $e : \mathsf{p}_A \diamond \sigma = \mathsf{p}_A \diamond \tau$ and $e' : \mathsf{q}[\sigma]_{\mathsf{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}\cdot[{}^{=}e]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathsf{q}[\tau]_{\mathsf{t}}$. If $\mathscr{C}$ has coherators for η^{sub}, then

$$\mathsf{p}_A * \mathsf{sub}^{=}_{\sigma,\tau}(e,e') = e.$$

In fact, a wild cwf $\mathscr{C}$ has coherators for η^{sub} if its category of contexts has triangle coherators (Definition 2.2.1), and it further satisfies the following coherence condition:

Definition 5.0.6. *Coherators for context extension.* A wild cwf $\mathscr{C}$ *has coherators for context extension* if, for all $\Gamma, \Delta : \mathscr{C}_0$, $A : \mathsf{Ty}\,\Delta$ and $\sigma : \mathscr{C}(\Gamma,\ \Delta.A)$, the following diagrams of equalities commute:

$$\begin{tikzcd}
 & \mathsf{p}_A \diamond (\mathsf{p}_A , \mathsf{q}_A) \arrow[dr, Rightarrow, "\mathsf{p}\beta"] & \\
\mathsf{p}_A \diamond \mathrm{id} \arrow[ur, Rightarrow, "\mathsf{p}_A * , \eta"] \arrow[rr, Rightarrow, "\rho"'] & & \mathsf{p}_A
\end{tikzcd}$$

and

$$\begin{tikzcd}
\mathsf{p}_A \diamond (\mathsf{p}_A , \mathsf{q}_A) \diamond \sigma \arrow[r, Rightarrow, "\alpha^{-1}"] \arrow[d, Rightarrow, "\mathsf{p}_A * , \diamond"'] & (\mathsf{p}_A \diamond (\mathsf{p}_A , \mathsf{q}_A)) \diamond \sigma \arrow[d, Rightarrow, "\mathsf{p}\beta * \sigma"] \\
\mathsf{p}_A \diamond (\mathsf{p}_A \diamond \sigma, \mathsf{q}_A[\sigma]_{\mathsf{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}}) \arrow[r, Rightarrow, "\mathsf{p}\beta"'] & \mathsf{p}_A \diamond \sigma
\end{tikzcd}$$

Lemma 5.0.7. If a wild cwf $\mathscr{C}$ has triangle coherators as well as coherators for context extension, then it has coherators for η^{sub}. Thus, the conclusion of Proposition 5.0.5 also holds if $\mathscr{C}$ has coherators for context extension.

Proof. Having coherators for η^{sub} is equivalent to having the outer boundary of Diagram 5.0.7.1 commute for all $\Gamma, \Delta : \mathscr{C}_0$, $A : \mathsf{Ty}\,\Delta$ and $\sigma : \mathscr{C}(\Gamma,\ \Delta.A)$. We show that this holds by pasting together the regions shown in the interior of the diagram, where the topmost interior square is filled by associativity of whiskering (Proposition 2.1.4), the rightmost triangle by the triangle coherator, and the regions marked $\circlearrowright$ using the coherators for context extension. □

$$\begin{tikzcd}
\mathrm{p}_A \diamond (\mathrm{p}_A , \mathrm{q}_A) \diamond \sigma \arrow[rrr, Rightarrow, "\mathrm{p}{*}(\,,\eta{*}\sigma)"] \arrow[ddd, Rightarrow, "\mathrm{p}{*}\,,\,\diamond"'] \arrow[dr, Rightarrow, "\alpha^{-1}"] & & & \mathrm{p}_A \diamond \mathrm{id} \diamond \sigma \arrow[ddd, Rightarrow, "\mathrm{p}{*}\lambda"] \\
& (\mathrm{p}_A \diamond (\mathrm{p}_A , \mathrm{q}_A)) \diamond \sigma \arrow[r, Rightarrow, "(\mathrm{p}{*}\,,\eta){*}\sigma"] \arrow[ddrr, Rightarrow, bend right, "\mathrm{p}\beta{*}\sigma"] & (\mathrm{p}_A \diamond \mathrm{id}) \diamond \sigma \arrow[ur, Rightarrow, "\alpha"] \arrow[ddr, Rightarrow, "\rho{*}\sigma"'] & \\
\arrow[r, phantom, "\circlearrowright"] & & \arrow[r, phantom, "\circlearrowright"] & \\
\mathrm{p}_A \diamond (\mathrm{p}_A \diamond \sigma , \mathrm{q}_A[\sigma]_{\mathrm{t}\downarrow[\diamond]_{\mathrm{T}}{}^{-1}}) \arrow[rrr, Rightarrow, "\mathrm{p}\beta"'] & & & \mathrm{p}_A \diamond \sigma
\end{tikzcd}$$

Diagram 5.0.7.1. Coherators for η^{sub} from coherators for context extension.

Presumably, the coherence conditions for context extension structures would arise out of the universal properties of sufficiently coherent representable presheaves à la a formulation via wild natural models. In any case, we can now define:

Definition 5.0.8. *Structurally 2-coherent wild cwfs.* We say that a wild cwf $\mathscr{C}$ is *(structurally) 2-coherent* if $\mathscr{C}$ has

- a 2-coherent wild category of contexts (Definition 2.2.3),
- type triangulators (Definition 4.1.8) and type pentagonators (Definition 4.1.9), and
- coherators for context extension (Definition 5.0.6).

Examples 5.0.9. *2-coherent internal cwfs.* Any set-level 1-cwf is immediately 2-coherent, and it is straightforward to check that the universe cwfs have type triangulators, type pentagonators and coherators for context extension.

Conjecture 5.0.10. *The container model.* We also expect that the higher *container model* of Altenkirch and Kaposi [AK21] is 2-coherent, to be shown by forthcoming work of Damato and Altenkirch [DA24].

6. Context Comprehension in 2-Coherent Wild Cwfs

A central feature of the categorical semantics of dependent type theory is that types form a cartesian fibration over contexts that is preserved by context extension. This is summed up in the notion of a *comprehension category* [Jac93], and it is widely known that 1-cwfs are equivalent to full split comprehension categories [Bla91; ALN24]. In this section we show that analogous results hold for 2-coherent wild cwfs.

6.1. The universal property of context extension

Lemma 6.1.1. *2-coherent substitution in types is weak pullback.* Suppose that $\mathscr{C}$ is a 2-coherent wild cwf, $\sigma : \mathscr{C}(\Gamma,\ \Delta)$ is a substitution, and $A : \mathsf{Ty}\ \Delta$ is a $\mathscr{C}$-type. Then there is a substitution

$$\sigma^{\cdot A} :\equiv (\sigma \diamond \mathrm{p}_{A[\sigma]_\mathsf{T}} , \mathrm{q}_{A[\sigma]_\mathsf{T}} \downarrow^{\mathsf{Tm}}_{[\diamond]_\mathsf{T}{}^{-1}})$$

from $\Gamma.A[\sigma]_\mathsf{T}$ to $\Delta.A$, such that the square

$$\mathfrak{P}_{\sigma, A} \quad :\equiv \quad \begin{tikzcd} \Gamma.A[\sigma]_\mathsf{T} \arrow[r, "\sigma^{\cdot A}"] \arrow[d, "\mathrm{p}"'] & \Delta.A \arrow[d, "\mathrm{p}"] \\ \Gamma \arrow[r, "\sigma"'] \arrow[ur, Rightarrow, "\mathrm{p}\beta^{-1}"'] & \Delta \end{tikzcd}$$

is a weak pullback in $\mathscr{C}$. That is, for any $\mathrm{B} : \mathscr{C}_0$ and commuting square $\mathfrak{S} :\equiv (\tau, \varrho, \gamma)$ with source B as in

$$\begin{tikzcd} \mathrm{B} \arrow[r, "\varrho"] \arrow[d, "\tau"'] & \Delta.A \arrow[d, "\mathrm{p}"] \\ \Gamma \arrow[r, "\sigma"'] \arrow[ur, Rightarrow, "\gamma"'] & \Delta \end{tikzcd} ,$$

the fiber $(\mathfrak{P}_{\sigma, A} \,\square\, \_)^{-1}(\mathfrak{S})$ is pointed, i.e. there is a mediating substitution

$$\mu_{\sigma,A,\mathfrak{S}} : \mathscr{C}(\mathrm{B}, \Gamma.A[\sigma]_\mathsf{T})$$

such that

$$\theta_{\sigma,A,\mathfrak{S}} : \mathfrak{P}_{\sigma, A} \,\square\, \mu_{\sigma,A,\mathfrak{S}} = \mathfrak{S}.$$

Proof. We claim that a mediating substitution is given by

$$\mu_{\sigma,A,\mathfrak{S}} :\equiv (\tau , \mathrm{q}_A[\varrho]_\mathrm{t} \downarrow^{\mathsf{Tm}}_{[\diamond]_\mathsf{T}{}^{-1} \cdot [{}^{=}\gamma]_\mathsf{T}{}^{-1} \cdot [\diamond]_\mathsf{T}}),$$

where $\mathrm{q}_A[\varrho]_\mathrm{t} : \mathsf{Tm}\, A[\mathrm{p}_A]_\mathsf{T}[\varrho]_\mathsf{T}$ is transported in the family Tm_B along

$$A[\mathrm{p}]_\mathsf{T}[\varrho]_\mathsf{T} \overset{[\diamond]_\mathsf{T}{}^{-1}}{=\!=\!=} A[\mathrm{p} \diamond \varrho]_\mathsf{T} \overset{[{}^{=}\gamma]_\mathsf{T}{}^{-1}}{=\!=\!=} A[\sigma \diamond \tau]_\mathsf{T} \overset{[\diamond]_\mathsf{T}}{=\!=\!=} A[\sigma]_\mathsf{T}[\tau]_\mathsf{T}.$$

For brevity, denote $\mu_{\sigma,A,\mathfrak{S}}$ by μ. By Proposition 3.1.4, constructing $\theta : \mathfrak{P}_{\sigma, A} \,\square\, \mu = \mathfrak{S}$ is equivalent to constructing witnesses

$$\delta : \mathrm{p}_{A[\sigma]_\mathsf{T}} \diamond \mu = \tau$$

and

$$\epsilon : \sigma^{\cdot A} \diamond \mu = \varrho$$

such that

$$\alpha^{-1} \cdot (\mathrm{p}\beta^{-1} * \mu) \cdot \alpha = (\sigma * \delta) \cdot \gamma \cdot (\mathrm{p}_A * \epsilon)^{-1}.$$

Let $\delta :\equiv \mathrm{p}\beta$. Using the equivalence $\mathrm{sub}^{=}_{\_,\,\_}$ (Definition 5.0.3), we define

$$\epsilon :\equiv \mathrm{sub}^{=}_{(\sigma^{\cdot A} \diamond \mu), \varrho}(\epsilon_0, \epsilon_1),$$

where

$$\epsilon_0 : \mathrm{p}_A \diamond \sigma^{\cdot A} \diamond \mu = \mathrm{p}_A \diamond \varrho$$

and

$$\epsilon_1 : \mathrm{q}_A[\sigma^{\cdot A} \diamond \mu]_\mathrm{t} \downarrow^{\mathsf{Tm}}_{[\diamond]_\mathsf{T}{}^{-1} \cdot [{}^{=}\epsilon_0]_\mathsf{T} \cdot [\diamond]_\mathsf{T}} = \mathrm{q}_A[\varrho]_\mathrm{t}$$

are the 2-cells[14] constructed as follows.

First, let ϵ_0 be the concatenation of equalities

$$\mathrm{p}_A \diamond \sigma^{\cdot A} \diamond \mu \overset{\alpha^{-1}\cdot(\mathrm{p}\beta * \mu)\cdot\alpha}{=\!=\!=\!=\!=\!=} \sigma \diamond \mathrm{p}_{A[\sigma]_{\mathsf{T}}} \diamond \mu \overset{\sigma * \mathrm{p}\beta}{=\!=\!=} \sigma \diamond \tau \overset{\gamma}{=\!=} \mathrm{p}_A \diamond \varrho .$$

Now calculate that

$$\begin{aligned}
& \mathrm{q}_A[\sigma^{\cdot A} \diamond \mu]_{\mathrm{t}} \\
= {} & \mathrm{q}_A[\sigma^{\cdot A}]_{\mathrm{t}}[\mu]_{\mathrm{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}} && \text{(by } [\diamond]_{\mathrm{t}}\text{)} \\
\equiv {} & \mathrm{q}_A[\sigma \diamond \mathrm{p}_{A[\sigma]_{\mathsf{T}}}, \mathrm{q}_{A[\sigma]_{\mathsf{T}}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}}]_{\mathrm{t}}[\mu]_{\mathrm{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}} \\
= {} & (\mathrm{q}_{A[\sigma]_{\mathsf{T}}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}\mathrm{p}\beta]_{\mathsf{T}}{}^{-1}\cdot[\diamond]_{\mathsf{T}}})[\mu]_{\mathrm{t}\downarrow[\diamond]_{\mathsf{T}}{}^{-1}} && \text{(by q}\beta\text{)} \\
= {} & \mathrm{q}_{A[\sigma]_{\mathsf{T}}}[\mu]_{\mathrm{t}\downarrow([\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}\mathrm{p}\beta]_{\mathsf{T}}{}^{-1}\cdot[\diamond]_{\mathsf{T}})[\mu]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}{}^{-1}} && \text{(by Proposition 4.1.6)} \\
= {} & \mathrm{q}_A[\varrho]_{\mathrm{t}\downarrow e} && \text{(by q}\beta\text{)}
\end{aligned}$$

where the transports are all in $\mathsf{Tm}_{\,\mathrm{B}}$, and e is the composition

$$e :\equiv [\diamond]_{\mathsf{T}}{}^{-1} \cdot [^{=}\gamma]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}} \cdot [^{=}\mathrm{p}\beta]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}} \cdot ([\diamond]_{\mathsf{T}}{}^{-1} \cdot [^{=}\mathrm{p}\beta]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}})[\mu]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}{}^{-1}.$$

So to construct ϵ_1, we may as well show that

$$\mathrm{q}_A[\varrho]_{\mathrm{t}\downarrow e\cdot[\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}\epsilon_0]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}} = \mathrm{q}_A[\varrho]_{\mathrm{t}}.$$

We do this by showing that the equality

$$\widetilde{e} :\equiv e \cdot [\diamond]_{\mathsf{T}}{}^{-1} \cdot [^{=}\epsilon_0]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}$$

is in fact equal to the trivial identity. Some path algebra shows that $\widetilde{e}$ is equal to the outer boundary of Diagram 6.1.1.1. This boundary commutes, since we can fill the interior of the diagram with the following commuting regions:

- (1), which commutes straightforwardly,
- (2) and (4), which commute by Proposition 4.1.7, and
- (3) and (5), which are filled by the type pentagonators.

This shows that $\widetilde{e} = \mathrm{refl}$, which completes the proof ϵ_1 that

$$\mathrm{q}_A[\sigma^{\cdot A} \diamond \mu]_{\mathrm{t}\downarrow^{\mathsf{Tm}}_{[\diamond]_{\mathsf{T}}{}^{-1}\cdot[^{=}\epsilon_0]_{\mathsf{T}}\cdot[\diamond]_{\mathsf{T}}}} = \mathrm{q}_A[\varrho]_{\mathrm{t}\downarrow\widetilde{e}} = \mathrm{q}_A[\varrho]_{\mathrm{t}},$$

and thus also the proof $\epsilon :\equiv \mathrm{sub}^{=}_{(\sigma^{\cdot A}\diamond\mu),\varrho}(\epsilon_0, \epsilon_1)$ that

$$\sigma^{\cdot A} \diamond \mu = \varrho .$$

Finally, what remains is to show that

$$\alpha^{-1} \cdot (\mathrm{p}\beta^{-1} * \mu) \cdot \alpha = (\sigma * \delta) \cdot \gamma \cdot (\mathrm{p}_A * \epsilon)^{-1}.$$

But by Lemma 5.0.7 we have that $(\mathrm{p}_A * \epsilon)^{-1} = \epsilon_0^{-1}$ on the right hand side, and the equality then follows by calculation. □

[14] Since $\mathscr{C}$-terms correspond to display maps in $\mathscr{C}$ (Lemma 4.3.4), we are justified in also calling ϵ_1 a 2-cell.

$$\begin{tikzcd}
 & & A[\mathrm{p}_A]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[dl, equal, "{[\diamond]_{\mathsf{T}}^{-1}}"'] \arrow[dr, equal, "{[\diamond]_{\mathsf{T}}}"] & & \\
 & A[\mathrm{p}_A \diamond \varrho]_{\mathsf{T}} \arrow[dl, equal, "{[{}^{=}\gamma]_{\mathsf{T}}^{-1}}"'] & (1) & A[\mathrm{p}_A \diamond \varrho]_{\mathsf{T}} \arrow[dr, equal, "{[{}^{=}\gamma]_{\mathsf{T}}}"] & \\
A[\sigma \diamond \tau]_{\mathsf{T}} \arrow[rrrr, equal] \arrow[d, equal, "{[\diamond]_{\mathsf{T}}}"'] & & & & A[\sigma \diamond \tau]_{\mathsf{T}} \arrow[d, equal, "{[{}^{=}\sigma{*}\mathrm{p}\beta]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}} \arrow[d, equal, "{[{}^{=}\mathrm{p}\beta]_{\mathsf{T}}^{-1}}"'] & & (2) & & A[\sigma \diamond \mathrm{p}_{A[\sigma]_{\mathsf{T}}} \diamond \mu]_{\mathsf{T}} \arrow[dllll, equal, "{[\diamond]_{\mathsf{T}}}" description] \arrow[d, equal, "{[{}^{=}\alpha]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\mathrm{p}_{A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}} \diamond \mu \arrow[d, equal, "{[\diamond]_{\mathsf{T}}}"'] & & (3) & & A[(\sigma \diamond \mathrm{p}_{A[\sigma]_{\mathsf{T}}}) \diamond \mu]_{\mathsf{T}} \arrow[ddllll, equal, "{[\diamond]_{\mathsf{T}}}" description] \arrow[d, equal, "{[{}^{=}\mathrm{p}\beta{*}\mu]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\mathrm{p}_{A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\mu]_{\mathsf{T}} \arrow[d, equal, "{[\diamond]_{\mathsf{T}}^{-1}[\mu]_{\mathsf{T}}}"'] & & & & A[(\mathrm{p}_A \diamond \sigma^{\cdot A}) \diamond \mu]_{\mathsf{T}} \arrow[ddlll, equal, "{[\diamond]_{\mathsf{T}}}" description] \arrow[d, equal, "{[{}^{=}\alpha]_{\mathsf{T}}^{-1}}"] \\
A[\sigma \diamond \mathrm{p}_{A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\mu]_{\mathsf{T}} \arrow[dr, equal, "{[{}^{=}\mathrm{p}\beta]_{\mathsf{T}}^{-1}[\mu]_{\mathsf{T}}}"'] & & (4) & & A[\mathrm{p}_A \diamond \sigma^{\cdot A} \diamond \mu]_{\mathsf{T}} \arrow[dl, equal, "{[\diamond]_{\mathsf{T}}^{-1}}"] \\
 & A[\mathrm{p}_A \diamond \sigma^{\cdot A}]_{\mathsf{T}}[\mu]_{\mathsf{T}} \arrow[dr, equal, "{[\diamond]_{\mathsf{T}}[\mu]_{\mathsf{T}}}"'] & (5) & A[\mathrm{p}_A]_{\mathsf{T}}[\sigma^{\cdot A} \diamond \mu]_{\mathsf{T}} \arrow[dl, equal, "{[\diamond]_{\mathsf{T}}^{-1}}"] & \\
 & & A[\mathrm{p}_A]_{\mathsf{T}}[\sigma^{\cdot A}]_{\mathsf{T}}[\mu]_{\mathsf{T}} & &
\end{tikzcd}$$

Diagram 6.1.1.1. $e \cdot [\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\epsilon_0]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}} = \mathrm{refl}$.

Theorem 6.1.2. *2-coherent substitution in types is pullback.* The weak pullbacks $\mathfrak{P}_{\sigma, A}$ of Lemma 6.1.1 are pullbacks.

Proof. By Lemma 6.1.1 we have that, for any $\mathrm{B} : \mathscr{C}_0$, the map

$$\mu_{\sigma,A} : \mathrm{CommSq}_{(\sigma,\mathrm{p}_A)}(\mathrm{B}) \to \mathscr{C}(\mathrm{B}, \Gamma.A[\sigma]_{\mathsf{T}})$$
$$\mu_{\sigma,A}((\tau, \varrho, \gamma)) :\equiv (\tau, \mathrm{q}_A[\varrho]_{\mathsf{t}} \downarrow^{\mathsf{Tm}}_{[\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\gamma]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}}})$$

is a section of the precomposition map $(\mathfrak{P}_{\sigma, A} \square_{\mathrm{B}} \_)$. We show that it's a retraction of the same, and therefore that $(\mathfrak{P}_{\sigma, A} \square_{\_} \_)$ is a family of equivalences.

That is, for $m : \mathscr{C}(\mathrm{B}, \Gamma.A[\sigma]_{\mathsf{T}})$, we want the equality of substitutions

$$\mu_{\sigma,A}(\mathfrak{P}_{\sigma, A} \square m) = m.$$

By Corollary 4.3.3 and a calculation similar to the one in the proof of Lemma 6.1.1 we have that

$$\mu_{\sigma,A}(\mathfrak{P}_{\sigma, A} \square m) = (\mathrm{p} \diamond m, \mathrm{q}_{A[\sigma]_{\mathsf{T}}}[m]_{\mathsf{t} \downarrow e}),$$

where

$$e :\equiv ([\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\mathrm{p}\beta]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}})[m]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}{}^{-1} \cdot [\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\alpha^{-1} \cdot (\mathrm{p}\beta * m) \cdot \alpha]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}.$$

On the other hand,

$$m = (\mathrm{p} \diamond m, \mathrm{q}_{A[\sigma]_{\mathsf{T}}}[m]_{\mathsf{t} \downarrow [\diamond]_{\mathsf{T}}{}^{-1}})$$

by Lemma 4.3.1, and thus by Corollary 4.3.3 again it's enough to show that $e = {[\diamond]_{\mathsf{T}}}^{-1}$.

By path algebra this amounts to showing the commutativity of a diagram of equalities that looks like the one formed by regions (3), (4) and (5) of Diagram 6.1.1.1, but where we replace μ with m. Commutativity of this diagram then follows as in the proof of Lemma 6.1.1, i.e. by Proposition 4.1.7 and type pentagonators. □

6.2. Cleavings of 2-coherent wild cwfs

Definition 6.2.1. *Cleavings.* Let $\mathscr{C}$ be a 2-coherent wild cwf. A *cleaving of* $\mathscr{C}$ is an assignment

$$\mathfrak{cl} : \prod\,(\Gamma, \Delta : \mathscr{C}_0)\,(\sigma : \mathscr{C}(\Gamma, \Delta))\,(A : \mathsf{Ty}\,\Delta)\ \mathrm{Pullback}_{(\sigma, \mathrm{p}_A)}(\Gamma.A[\sigma]_{\mathsf{T}})$$

of pullbacks

$$\mathfrak{cl}_{\Gamma,\Delta}(\sigma, A) \;:\equiv\; \begin{tikzcd} \Gamma.A[\sigma]_{\mathsf{T}} \arrow[r, "\ell_{\sigma,A}"] \arrow[d, two heads] & \Delta.A \arrow[d, two heads] \\ \Gamma \arrow[r, "\sigma"'] \arrow[ur, Rightarrow, "\mathfrak{p}_{\sigma,A}"'] & \Delta \end{tikzcd}$$

to each cospan in $\mathscr{C}$ of the form $\Gamma \xrightarrow{\sigma} \Delta \xleftarrow{\mathrm{p}_A} \Delta.A$. We call the component $\ell_{\sigma,A}$ of the pullback $\mathfrak{cl}_{\Gamma,\Delta}(\sigma, A)$ the *chosen lift of* σ *at* A.

The upshot of Theorem 6.1.2 is then that every 2-coherent wild cwf has a cleaving

$$\mathfrak{cl}_{\Gamma,\Delta}(\sigma, A) :\equiv \mathfrak{P}_{\sigma, A},$$

which we call the *type substitution cleaving*. In particular, $\sigma^{\cdot A}$ is the chosen lift of a substitution $\sigma : \mathscr{C}(\Gamma, \Delta)$ at $A : \mathsf{Ty}\,\Delta$.

With the type substitution cleaving, the chosen lift of the identity is the canonical dependent identity.

Proposition 6.2.2. Let $e : A = A'$ be an equality of $\mathscr{C}$-types $A, A' : \mathsf{Ty}\,\Gamma$. Then

$$\mathrm{idd}(\mathrm{ap}\,(\Gamma.\_)\;e) = (\mathrm{p}_A\,,\,\mathrm{q}_A \downarrow^{\mathsf{Tm}}_{e[\mathrm{p}_A]_{\mathsf{T}}}).$$

Proof. By induction on e it's enough to show that $\mathrm{idd}(\mathrm{refl}_{\Gamma.A}) = (\mathrm{p}_A\,,\,\mathrm{q}_A)$, which holds by , η (Definition 4.2.1). □

Lemma 6.2.3. The type substitution cleaving of any 2-coherent wild cwf satisfies

$$\mathrm{id}^{\cdot A} = \mathrm{idd}(\mathrm{ap}\,(\Gamma.\_)\;[\mathrm{id}]_{\mathsf{T}})$$

for any $\Gamma : \mathscr{C}_0$ and $A : \mathsf{Ty}\,\Gamma$.

Proof. By Proposition 6.2.2 it's enough to show that $\mathrm{id}^{\cdot A} = (\mathrm{p}_{A[\mathrm{id}]_{\mathsf{T}}}\,,\,\mathrm{q}_{A[\mathrm{id}]_{\mathsf{T}}} \downarrow [\mathrm{id}]_{\mathsf{T}}[\mathrm{p}_{A[\mathrm{id}]_{\mathsf{T}}}]_{\mathsf{T}})$, which holds by Corollary 4.3.3 and the left type triangulator. □

Definition 6.2.4. *Split cleavings of 2-coherent wild cwfs.* A cleaving $\mathfrak{cl}$ of a 2-coherent wild cwf $\mathscr{C}$ is called *split* if for all substitutions

$$\mathrm{B} \xrightarrow{\tau} \Gamma \xrightarrow{\sigma} \Delta$$

and $\mathscr{C}$-types $A : \mathrm{Ty}\,\Delta$, the equality type

$$\big(\mathrm{B}.A[\sigma \diamond \tau]_{\mathrm{T}},\ \mathfrak{cl}_{\mathrm{B},\Delta}(\sigma \diamond \tau, A)\big) = \big(\mathrm{B}.A[\sigma]_{\mathrm{T}}[\tau]_{\mathrm{T}},\ \mathfrak{cl}_{\mathrm{B},\Gamma}(\tau, A[\sigma]_{\mathrm{T}}) \mid \mathfrak{cl}_{\Gamma,\Delta}(\sigma, A)\big)$$

of the pullbacks

$$\mathfrak{cl}_{\mathrm{B},\Delta}(\sigma \diamond \tau, A) \equiv \begin{tikzcd} \mathrm{B}.A[\sigma \diamond \tau]_{\mathrm{T}} \arrow[r, "\ell_{\sigma \diamond \tau, A}"] \arrow[d, two heads] & \Delta.A \arrow[d, two heads] \\ \mathrm{B} \arrow[r, "\sigma \diamond \tau"'] \arrow[ur, Rightarrow, "\mathfrak{p}_{\sigma \diamond \tau, A}"'] & \Delta \end{tikzcd}$$

and

$$\mathfrak{cl}_{\mathrm{B},\Gamma}(\tau, A[\sigma]_{\mathrm{T}}) \mid \mathfrak{cl}_{\Gamma,\Delta}(\sigma, A) \equiv \begin{tikzcd} \mathrm{B}.A[\sigma]_{\mathrm{T}}[\tau]_{\mathrm{T}} \arrow[r, "\ell_{\tau, A[\sigma]_{\mathrm{T}}}"] \arrow[d, two heads] & \Gamma.A[\sigma]_{\mathrm{T}} \arrow[r, "\ell_{\sigma, A}"] \arrow[d, two heads] & \Delta.A \arrow[d, two heads] \\ \mathrm{B} \arrow[r, "\tau"'] \arrow[ur, Rightarrow, "\mathfrak{p}_{\tau, A[\sigma]_{\mathrm{T}}}"'] & \Gamma \arrow[r, "\sigma"'] \arrow[ur, Rightarrow, "\mathfrak{p}_{\sigma, A}"'] & \Delta \end{tikzcd}$$

on $\mathrm{B} \xrightarrow{\sigma \diamond \tau} \Delta \xleftarrow{\mathrm{p}_A} \Delta.A$ is contractible.

A split cleaving of a 2-coherent wild cwf, in our sense, may be considered a higher version of a splitting of a full comprehension category. As is to be expected, set-level internal cwfs have split type substitution cleavings. We now show that the type substitution cleavings of *univalent* 2-coherent wild cwfs are also always split.

Lemma 6.2.5. Suppose $\mathscr{C}$ is a 2-coherent wild cwf. For all substitutions $\mathrm{B} \xrightarrow{\tau} \Gamma \xrightarrow{\sigma} \Delta$ and $\mathscr{C}$-types $A : \mathrm{Ty}\,\Delta$,

$$(\mathrm{B}.A[\sigma \diamond \tau]_{\mathrm{T}},\ \mathfrak{P}_{\sigma \diamond \tau, A}) = (\mathrm{B}.A[\sigma]_{\mathrm{T}}[\tau]_{\mathrm{T}},\ \mathfrak{P}_{\tau, A[\sigma]_{\mathrm{T}}} \mid \mathfrak{P}_{\sigma, A}).$$

Proof. By Corollary 3.1.13 it's enough to give

$$e : \mathrm{B}.A[\sigma \diamond \tau]_{\mathrm{T}} = \mathrm{B}.A[\sigma]_{\mathrm{T}}[\tau]_{\mathrm{T}}$$

such that

$$\mathfrak{P}_{\sigma \diamond \tau, A} = (\mathfrak{P}_{\tau, A[\sigma]_{\mathrm{T}}} \mid \mathfrak{P}_{\sigma, A}) \mathbin{\square} \mathrm{idd}(e).$$

Take

$$e :\equiv \mathrm{ap}\,(\mathrm{B}.\_)\,[\diamond]_{\mathrm{T}},$$

then by Proposition 6.2.2 we may as well show that

$$\mathfrak{P}_{\sigma \diamond \tau, A} = (\mathfrak{P}_{\tau, A[\sigma]_{\mathrm{T}}} \mid \mathfrak{P}_{\sigma, A}) \mathbin{\square} (\mathrm{p}\,,\, \mathrm{q}_{\downarrow\,[\diamond]_{\mathrm{T}}[\mathrm{p}]_{\mathrm{T}}}),$$

or, equivalently, give three equalities

$$\delta : \mathrm{p}_{A[\sigma \diamond \tau]_{\mathrm{T}}} = \mathrm{p}_{A[\sigma]_{\mathrm{T}}[\tau]_{\mathrm{T}}} \diamond (\mathrm{p}\,,\, \mathrm{q}_{\downarrow\,[\diamond]_{\mathrm{T}}[\mathrm{p}]_{\mathrm{T}}}),$$
$$\epsilon : (\sigma \diamond \tau)^{\cdot A} = (\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathrm{T}}}) \diamond (\mathrm{p}\,,\, \mathrm{q}_{\downarrow\,[\diamond]_{\mathrm{T}}[\mathrm{p}]_{\mathrm{T}}})$$

and

$$\eta : \mathfrak{p}_{\sigma \diamond \tau, A} = \big((\sigma \diamond \tau) * \delta\big) \cdot \alpha^{-1} \cdot \big((\mathfrak{p}_{\tau, A[\sigma]_{\mathrm{T}}} \mid \mathfrak{p}_{\sigma, A}) * (\mathrm{p}\,,\, \mathrm{q}_{\downarrow\,[\diamond]_{\mathrm{T}}[\mathrm{p}]_{\mathrm{T}}})\big) \cdot \alpha \cdot (\mathrm{p}_A * \epsilon^{-1}).$$

Take $\delta :\equiv \mathsf{p}\beta^{-1}$. We will define $\epsilon :\equiv \mathsf{sub}^{=}(\epsilon_0, \epsilon_1)$, where $\mathsf{sub}^{=}$ is the equivalence defined at Definition 5.0.3, and where we seek equalities

$$\epsilon_0 : \mathsf{p} \diamond (\sigma \diamond \tau)^{\cdot A} = \mathsf{p} \diamond (\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond (\mathsf{p}, \mathsf{q}_{\downarrow [\diamond]_{\mathsf{T}}[\mathsf{p}]_{\mathsf{T}}})$$

and

$$\epsilon_1 : \mathsf{q}[(\sigma \diamond \tau)^{\cdot A}]_{\mathsf{t}\downarrow [\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\epsilon_0]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}} = \mathsf{q}[(\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond (\mathsf{p}, \mathsf{q}_{\downarrow [\diamond]_{\mathsf{T}}[\mathsf{p}]_{\mathsf{T}}})]_{\mathsf{t}}.$$

Now, from Proposition 5.0.5 we have that

$$\mathsf{p}_A * \epsilon^{-1} = (\mathsf{p}_A * \epsilon)^{-1} = \epsilon_0^{-1},$$

and by rearranging the type of η we may take

$$\epsilon_0 :\equiv \mathfrak{p}_{\sigma\diamond\tau,A}{}^{-1} \cdot \big((\sigma \diamond \tau) * \delta\big) \cdot \alpha^{-1} \cdot \big((\mathfrak{p}_{\tau,A[\sigma]_{\mathsf{T}}} \mid \mathfrak{p}_{\sigma,A}) * (\mathsf{p}, \mathsf{q}_{\downarrow [\diamond]_{\mathsf{T}}[\mathsf{p}]_{\mathsf{T}}})\big) \cdot \alpha.$$

What remains, then, is to construct ϵ_1. By applying $\mathsf{q}\beta$ and $[\diamond]_{\mathsf{t}}$ to reduce the generic terms on the left and right, we calculate that its type is equivalent to

$$\mathsf{q}_{A[\sigma\diamond\tau]_{\mathsf{t}}} {\downarrow}^{\mathsf{Tm\,B}}_{e_1} = \mathsf{q}_{A[\sigma\diamond\tau]_{\mathsf{t}}} {\downarrow}^{\mathsf{Tm\,B}}_{e_2},$$

where the left and right hand sides are transported, respectively, over equalities

$$e_1 :\equiv [\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\mathsf{p}\beta^{-1}]_{\mathsf{T}} \cdot [{}^{=}\epsilon_0]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}$$

and

$$e_2 :\equiv [\diamond]_{\mathsf{T}}[\mathsf{p}]_{\mathsf{T}} \cdot [{}^{=}\mathsf{p}\beta^{-1}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}} \cdot \widetilde{e}\,[\mathsf{p}, \mathsf{q}_{\downarrow [\diamond]_{\mathsf{T}}[\mathsf{p}]_{\mathsf{T}}}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}{}^{-1},$$

and where

$$\widetilde{e} :\equiv [\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\mathsf{p}\beta^{-1}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}} \cdot \big([\diamond]_{\mathsf{T}}{}^{-1} \cdot [{}^{=}\mathsf{p}\beta^{-1}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}\big)[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}}{}^{-1}.$$

It's now enough to show that $e_1 = e_2$. This amounts to giving a filling of Diagram 6.2.5.1, which we divide into three regions filled with coherence cells as shown in Diagram 6.2.5.2 and Diagram 6.2.5.3. □

$$\begin{tikzcd}
& & A[(\sigma \diamond \tau) \diamond \mathrm{p}]_{\mathsf{T}} \arrow[dr, Rightarrow, "{[^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"] & & \\
& A[\sigma \diamond \tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}} \arrow[ur, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}}"] & & A[\mathrm{p} \diamond (\sigma \diamond \tau)^{\cdot A}]_{\mathsf{T}} \arrow[dr, Rightarrow, "{[^{=}\mathrm{p}\beta]_{\mathsf{T}}}"] & \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}} \arrow[ur, Rightarrow, "{[\diamond]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}}"] \arrow[d, Rightarrow, "{[^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"'] & & & & A[(\sigma \diamond \tau) \diamond \mathrm{p}]_{\mathsf{T}} \arrow[d, Rightarrow, "{[^{=}(\sigma{\diamond}\tau){*}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p} \diamond \varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] & & \text{(I)} & & A[(\sigma \diamond \tau) \diamond \mathrm{p} \diamond \varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[^{=}\alpha^{-1}]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}[\varrho] \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}[\varrho]}"'] & & & & A[((\sigma \diamond \tau) \diamond \mathrm{p}) \diamond \varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[^{=}\alpha{*}\varrho]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\tau \diamond \mathrm{p}]_{\mathsf{T}}[\varrho] \arrow[d, Rightarrow, "{[^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & & & & A[(\sigma \diamond \tau \diamond \mathrm{p}) \diamond \varrho]_{\mathsf{T}} \arrow[llll, Rightarrow] \arrow[d, Rightarrow, "{[^{=}(\sigma{*}\mathrm{p}\beta^{-1}){*}\varrho]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\mathrm{p} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho] \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & & \text{(II)} & & A[(\sigma \diamond \mathrm{p} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[^{=}\alpha^{-1}{*}\varrho]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho] \arrow[d, Rightarrow, "\xi"'] & & & & A[((\sigma \diamond \mathrm{p}) \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} \arrow[llll, Rightarrow] \arrow[d, Rightarrow, "{[^{=}(\mathrm{p}\beta^{-1}{*}\tau^{\cdot A[\sigma]_{\mathsf{T}}}){*}\varrho]_{\mathsf{T}}}"] \\
A[\mathrm{p}]_{\mathsf{T}}[\sigma^{\cdot A}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho] \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}[\varrho]_{\mathsf{T}}}"'] & & & & A[((\mathrm{p} \diamond \sigma^{\cdot A}) \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[^{=}\alpha{*}\varrho]_{\mathsf{T}}}"] \\
A[\mathrm{p}]_{\mathsf{T}}[\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho] \arrow[dr, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}}"'] & & \text{(III)} & & A[(\mathrm{p} \diamond \sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} \arrow[dl, Rightarrow, "{[^{=}\alpha]_{\mathsf{T}}}"] \\
& A[\mathrm{p}]_{\mathsf{T}}[(\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} & & A[\mathrm{p} \diamond (\sigma^{\cdot A} \diamond \tau^{\cdot A[\sigma]_{\mathsf{T}}}) \diamond \varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] &
\end{tikzcd}$$

Diagram 6.2.5.1. The pasting proof of $e_1 = e_2$ splits into three regions, which are filled with the cells shown in Diagram 6.2.5.2 and Diagram 6.2.5.3. We abbreviate the substitution $(\mathrm{p}, \mathrm{q}_{\downarrow [\diamond]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}})$ by ϱ and the substitution $([\diamond]_{\mathsf{T}}{}^{-1} \cdot [^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}} \cdot [\diamond]_{\mathsf{T}})[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}$ by ξ.

$$\begin{tikzcd}
 & & & & A[(\sigma\diamond\tau)\diamond\mathrm{p}]_{\mathsf{T}} \arrow[drr, Rightarrow, "{[{}^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"] \arrow[ddrrr, equals] & & & \\
 & & A[\sigma\diamond\tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}} \arrow[urr, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}}"] \arrow[dl, Rightarrow, "{[\diamond]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}}"'] & & (1) & & A[\mathrm{p}\diamond(\sigma\diamond\tau)^{\cdot A}]_{\mathsf{T}} \arrow[dr, Rightarrow, "{[{}^{=}\mathrm{p}\beta]_{\mathsf{T}}}"] & \\
 & A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}} \arrow[ddl, Rightarrow, "{[{}^{=}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"'] & & & & & & A[(\sigma\diamond\tau)\diamond\mathrm{p}]_{\mathsf{T}} \arrow[dll, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}"'] \arrow[dd, Rightarrow, "{[{}^{=}(\sigma{\diamond}\tau){*}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}"] \\
 & & (2) & A[\sigma]_{\mathsf{T}}[\tau\diamond\mathrm{p}]_{\mathsf{T}} \arrow[ull, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[dd, Rightarrow, "{[{}^{=}\tau{*}\mathrm{p}\beta^{-1}]_{\mathsf{T}}}" description] & & A[\sigma\diamond\tau\diamond\mathrm{p}]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[dd, Rightarrow, "{[{}^{=}\sigma{*}(\tau{*}\mathrm{p}\beta^{-1})]_{\mathsf{T}}}" description] & (3) & \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p}\diamond\varrho]_{\mathsf{T}} \arrow[dd, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] & & & & (2) & & & A[(\sigma\diamond\tau)\diamond\mathrm{p}\diamond\varrho]_{\mathsf{T}} \arrow[dll, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}"] \arrow[dd, Rightarrow, "{[{}^{=}\alpha^{-1}]_{\mathsf{T}}}"] \\
 & & & A[\sigma]_{\mathsf{T}}[\tau\diamond\mathrm{p}\diamond\varrho]_{\mathsf{T}} \arrow[ulll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] & & A[\sigma\diamond\tau\diamond\mathrm{p}\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] & & \\
A[\sigma]_{\mathsf{T}}[\tau]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}{}^{-1}[\varrho]_{\mathsf{T}}}"'] & (1) & & & (2) & & (4) & A[((\sigma\diamond\tau)\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[{}^{=}\alpha{*}\varrho]_{\mathsf{T}}}"] \\
A[\sigma]_{\mathsf{T}}[\tau\diamond\mathrm{p}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} & & & A[\sigma]_{\mathsf{T}}[(\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[lll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] \arrow[uu, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}" description] & & A[\sigma\diamond(\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] \arrow[uu, Rightarrow, "{[{}^{=}\sigma{*}\alpha]_{\mathsf{T}}}" description] & & A[(\sigma\diamond\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}"]
\end{tikzcd}$$

Region (I)

$$\begin{tikzcd}
A[\sigma]_{\mathsf{T}}[\tau\diamond\mathrm{p}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[ddd, Rightarrow, "{\mathrm{p}\beta^{-1}[\varrho]_{\mathsf{T}}}"'] & & A[\sigma]_{\mathsf{T}}[(\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[dd, Rightarrow, "{[{}^{=}\mathrm{p}\beta^{-1}{*}\varrho]_{\mathsf{T}}}" description] & & A[\sigma\diamond(\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[dd, Rightarrow, "{[{}^{=}\sigma{*}(\mathrm{p}\beta^{-1}{*}\varrho)]_{\mathsf{T}}}" description] & & A[(\sigma\diamond\tau\diamond\mathrm{p})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}"'] \arrow[ddd, Rightarrow, "{[{}^{=}(\sigma{*}\mathrm{p}\beta^{-1}){*}\varrho]_{\mathsf{T}}}"] \\
 & (2) & & (2) & & (3) & \\
 & & A[\sigma]_{\mathsf{T}}[(\mathrm{p}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[dll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] & & A[\sigma\diamond(\mathrm{p}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] & & \\
A[\sigma]_{\mathsf{T}}[\mathrm{p}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[dd, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & & & (1) & & & A[(\sigma\diamond\mathrm{p}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[ull, Rightarrow, "{[{}^{=}\alpha]_{\mathsf{T}}}"'] \arrow[dlll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] \arrow[dd, Rightarrow, "{[{}^{=}\alpha^{-1}{*}\varrho]_{\mathsf{T}}}"] \\
 & (1) & & A[\sigma\diamond\mathrm{p}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[ulll, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] \arrow[dr, Rightarrow, "{[{}^{=}\alpha^{-1}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] & & (2) & \\
A[\sigma]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} & & A[\sigma\diamond\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] & & A[(\sigma\diamond\mathrm{p})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] & & A[((\sigma\diamond\mathrm{p})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[ll, Rightarrow, "{[\diamond]_{\mathsf{T}}}"]
\end{tikzcd}$$

Region (II)

Diagram 6.2.5.2. Filling regions (I) and (II) of Diagram 6.2.5.1 with coherence cells. Regions marked (1) are filled using type pentagonators (Definition 4.1.9); those marked (2), by naturality of $[\diamond]_{\mathsf{T}}$ (Proposition 4.1.7); (3), by associativity of whiskering (Proposition 2.1.4); and (4), by the pentagon associator of the category of contexts (Definition 4.1.9).

$$\begin{tikzcd}
A[\sigma]_{\mathsf{T}}[\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}}^{-1}{[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & A[\sigma\diamond\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[l, Rightarrow, "{[\diamond]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] \arrow[dl, Rrightarrow, no head] & A[(\sigma\diamond\mathrm{p})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[l, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] \arrow[d, Rightarrow, "{[^{=}\mathrm{p}\beta^{-1}*\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] \arrow[dr, phantom, "(2)"] & A[((\sigma\diamond\mathrm{p})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[l, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[d, Rightarrow, "{[^{=}(\mathrm{p}\beta^{-1}*\tau^{\cdot A[\sigma]_{\mathsf{T}}})*\varrho]_{\mathsf{T}}}"] \\
A[\sigma\diamond\mathrm{p}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "\mathrm{p}\beta^{-1}{[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & (2) & A[(\mathrm{p}\diamond\sigma^{\cdot A})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[dll, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] \arrow[d, Rightarrow, "{[^{=}\alpha]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] \arrow[dr, phantom, "(2)"] & A[((\mathrm{p}\diamond\sigma^{\cdot A})\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[l, Rightarrow, "{[\diamond]_{\mathsf{T}}}"'] \arrow[d, Rightarrow, "{[^{=}\alpha*\varrho]_{\mathsf{T}}}"] \\
A[\mathrm{p}\diamond\sigma^{\cdot A}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[d, Rightarrow, "{[\diamond]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"'] & & A[\mathrm{p}\diamond\sigma^{\cdot A}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[ddl, Rightarrow, "{[\diamond]_{\mathsf{T}}[\varrho]_{\mathsf{T}}}"] & A[(\mathrm{p}\diamond\sigma^{\cdot A}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[l, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] \arrow[d, Rightarrow, "{[^{=}\alpha]_{\mathsf{T}}}"] \\
A[\mathrm{p}]_{\mathsf{T}}[\sigma^{\cdot A}]_{\mathsf{T}}[\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[dr, Rightarrow, "{[\diamond]_{\mathsf{T}}}^{-1}{[\varrho]_{\mathsf{T}}}"'] & (1) & (1) & A[\mathrm{p}\diamond(\sigma^{\cdot A}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} \arrow[dl, Rightarrow, "{[\diamond]_{\mathsf{T}}}"] \\
& A[\mathrm{p}]_{\mathsf{T}}[\sigma^{\cdot A}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}}]_{\mathsf{T}}[\varrho]_{\mathsf{T}} \arrow[r, Rightarrow, "{[\diamond]_{\mathsf{T}}}^{-1}"'] & A[\mathrm{p}]_{\mathsf{T}}[(\sigma^{\cdot A}\diamond\tau^{\cdot A[\sigma]_{\mathsf{T}}})\diamond\varrho]_{\mathsf{T}} &
\end{tikzcd}$$

Diagram 6.2.5.3. Filling region (III) of Diagram 6.2.5.1 with coherence cells. Regions marked (1) are filled using type pentagonators (Definition 4.1.9); regions marked (2), by naturality of $[\diamond]_{\mathsf{T}}$ (Proposition 4.1.7).

Theorem 6.2.6. *Split comprehension for set-level and univalent cwfs.* The type substitution cleaving $\mathfrak{P}_{\sigma,A}$ of any set-level or univalent 2-coherent wild cwf $\mathscr{C}$ is split.

Proof. By Propositions 3.3.2 and 3.3.3 and Lemma 6.2.5, the equality

$$(\mathrm{B}.A[\sigma \diamond \tau]_\mathrm{T},\ \mathfrak{P}_{\sigma\diamond\tau,A}) = (\mathrm{B}.A[\sigma]_\mathrm{T}[\tau]_\mathrm{T},\ \mathfrak{P}_{\tau,A[\sigma]_\mathrm{T}} \mid \mathfrak{P}_{\sigma,A})$$

of pullbacks is an inhabited proposition for all appropriately typed σ, τ and A when $\mathscr{C}$ is set-level, or 2-coherently univalent. □

Definition 6.2.7. We call a 2-coherent wild cwf *split* if its type substitution cleaving is split.

Corollary 6.2.8. Thus, the syntax cwf QIIT of Altenkirch and Kaposi [AK16] as well as any univalent universe cwf $\mathscr{U}$ is split 2-coherent.

7. Discussion

We have given a unified account of the cloven fibrational structure of internal models of homotopical dependent type theory, in such a way so as to include set-level models such as the syntax as well as the canonical higher models given by universe types.

While the “splitness” of a cleaving of a 2-coherent wild cwf (Definition 6.2.4) is propositional and ensures that composition of lifts is coherently unique, the same cannot currently be said for lifts of identity substitutions. Even so, the type of morphisms of split 2-coherent wild cwfs is internally definable, which opens the possibility of internally speaking of “transfers” of constructions that respect cloven fibrations of internal models. Indeed, this is a large part of the motivation of the present work: the theory developed here is intended to provide the correct formal setting in which to investigate constructions of semisimplicial and other Reedy fibrant inverse diagrams [KS17] in internal models of homotopical type theory [CK21; CK24].

Separately from questions of infinite higher coherent constructions, we hope that our theory can still be useful by immediately specializing to yield notions of 1-truncated “2-cwfs”. For instance, by modifying the definition of a 2-coherent wild cwf to additionally require that (1) the category of contexts $\mathscr{C}$ is a precategory, (2) the presheaf of $\mathscr{C}$-types is valued in 1-types, and (3) the presheaf of $\mathscr{C}$-terms is set-valued, we obtain a simple higher generalization of the notion of a 1-cwf, which conjecturally includes the container higher model of type theory [AK21] as an instance. Then via Rezk completion and our results, any instance of such a higher cwf should be equivalent to a split one. A less naïve approach would be to use univalent bicategory theory [Ahr+21], noting Proposition 2.2.4, to develop a full theory of 2-cwfs.

As a final remark, we have developed wild categories with families for their anticipated applicability to specific further internal constructions, but we also expect the study of wild *natural models* [Awo18] to prove complementarily fruitful.

References

[ACKS23] Danil Annenkov, Paolo Capriotti, Nicolai Kraus, and Christian Sattler. "Two-level type theory and applications". In: *Mathematical Structures in Computer Science* 33.8 (2023), pp. 688–743. DOI: 10.1017/S0960129523000130. Available on the arXiv:1705.03307 [cs.LO].

[Ahr+21] Benedikt Ahrens, Dan Frumin, Marco Maggesi, Niccolò Veltri, and Niels van der Weide. "Bicategories in univalent foundations". In: *Mathematical Structures in Computer Science* 31.10 (Nov. 2021), pp. 1232–1269. DOI: 10.1017/S0960129522000032. Available on the arXiv:1903.01152 [math.CT].

[AK16] Thorsten Altenkirch and Ambrus Kaposi. "Type Theory in Type Theory Using Quotient Inductive Types". In: *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. POPL '16. St. Petersburg, FL, USA: Association for Computing Machinery, 2016, pp. 18–29. ISBN: 9781450335492. DOI: 10.1145/2837614.2837638. URL: https://doi.org/10.1145/2837614.2837638.

[AK21] Thorsten Altenkirch and Ambrus Kaposi. "A container model of type theory". http://real.mtak.hu/id/eprint/130775. 27th International Conference on Types for Proofs and Programs (TYPES). 2021.

[AKL15] Jeremy Avigad, Krzysztof Kapulkin, and Peter Lefanu Lumsdaine. "Homotopy limits in type theory". In: *Mathematical Structures in Computer Science* 25.5 (2015), pp. 1040–1070. DOI: 10.1017/S0960129514000498. Available on the arXiv:1304.0680 [math.LO].

[AKS15] Benedikt Ahrens, Krzysztof Kapulkin, and Michael Shulman. "Univalent categories and the Rezk completion". In: *Mathematical Structures in Computer Science* 25.5 (2015), pp. 1010–1039. DOI: 10.1017/S0960129514000486. Available on the arXiv:1303.0584 [math.CT].

[ALN24] Benedikt Ahrens, Peter LeFanu Lumsdaine, and Paige Randall North. "Comparing Semantic Frameworks for Dependently-Sorted Algebraic Theories". In: *Programming Languages and Systems (Proc. APLAS 2024)*. Ed. by Oleg Kiselyov. Singapore: Springer Nature Singapore, Oct. 28, 2024, pp. 3–22. ISBN: 978-981-97-8943-6.

[Alt+18] Thorsten Altenkirch, Paolo Capriotti, Gabe Dijkstra, Nicolai Kraus, and Fredrik Nordvall Forsberg. "Quotient Inductive-Inductive Types". In: *Foundations of Software Science and Computation Structures*. Ed. by Christel Baier and Ugo Dal Lago. Cham: Springer International Publishing, 2018, pp. 293–310. ISBN: 978-3-319-89366-2. Available on the arXiv:1612.02346 [cs.LO].

[ALV18] Benedikt Ahrens, Peter LeFanu Lumsdaine, and Vladimir Voevodsky. "Categorical structures for type theory in univalent foundations". In: *Logical Methods in Computer Science* Volume 14, Issue 3, 25 (Sept. 2018). ISSN: 1860-5974. DOI: 10.23638/LMCS-14(3:18)2018. URL: https://lmcs.episciences.org/4801.

[Awo18] Steve Awodey. "Natural models of homotopy type theory". In: *Mathematical Structures in Computer Science* 28.2 (2018), pp. 241–286. DOI: 10.1017/S0960129516000268. Available on the arXiv:1406.3219 [math.CT].

[Bau+] Andrej Bauer, Jason Gross, Peter LeFanu Lumsdaine, Mike Shulman, Matthieu Sozeau, Bas Spitters, et al. *WildCat*. The HoTT Library. URL: https://github.com/HoTT/Coq-HoTT/tree/master/theories/WildCat. Directory of Coq code. Last accessed 2025-02-06.

[BD08] Alexandre Buisse and Peter Dybjer. "Towards Formalizing Categorical Models of Type Theory in Type Theory". In: *Electronic Notes in Theoretical Computer Science* 196 (2008). Proceedings of the Second International Workshop on Logical Frameworks and Meta-Languages: Theory and Practice (LFMTP 2007), pp. 137–151. ISSN: 1571-0661. DOI: https://doi.org/10.1016/j.entcs.2007.09.023. URL: https://www.sciencedirect.com/science/article/pii/S1571066108000431.

[BG11] Benno van den Berg and Richard Garner. "Types are weak ω-groupoids". In: *Proceedings of the London Mathematical Society* 102.2 (2011), pp. 370–394. DOI: https://doi.org/10.1112/plms/pdq026. eprint: https://londmathsoc.onlinelibrary.wiley.com/doi/pdf/10.1112/plms/pdq026. URL: https://londmathsoc.onlinelibrary.wiley.com/doi/abs/10.1112/plms/pdq026.

[BG12] Benno van den Berg and Richard Garner. "Topological and Simplicial Models of Identity Types". In: *ACM Trans. Comput. Logic* 13.1 (Jan. 2012). ISSN: 1529-3785. DOI: 10.1145/2071368.2071371. URL: https://doi.org/10.1145/2071368.2071371. Available on the arXiv:1007.4638 [math.LO].

[Bla91] Javier Blanco. "Relating categorical approaches to type dependency". Masters thesis. University of Nijmegen, 1991.

[CK17] Paolo Capriotti and Nicolai Kraus. "Univalent Higher Categories via Complete Semi-Segal Types". In: *Proceedings of the ACM on Programming Languages* 2.POPL '18 (Dec. 2017), 44:1–44:29. ISSN: 2475-1421. DOI: 10.1145/3158132. URL: http://doi.acm.org/10.1145/3158132. Available on the arXiv:1707.03693 [math.CT].

[CK21] Joshua Chen and Nicolai Kraus. "Semisimplicial types in internal categories with families". https://types21.liacs.nl/download/semisimplicial-types-in-internal-categories-with-families/. 27th International Conference on Types for Proofs and Programs (TYPES) (June 16, 2021). 2021.

[CK24] Joshua Chen and Nicolai Kraus. *Constructing inverse diagrams in homotopical type theory*. Talk given at the Working Group 6 meeting of the European Research Network on Formal Proofs. Leuven, Belgium, Apr. 4, 2024. URL: https://europroofnet.github.io/wg6-leuven/programme/#chen-kraus.

[DA24] Stefania Damato and Thorsten Altenkirch. "Coherences for the Container Model of Type Theory". https://hott-uf.github.io/2024/abstracts/HoTTUF_2024_paper_9.pdf. Workshop on Homotopy Type Theory/Univalent Foundations. 2024.

[Dyb96] Peter Dybjer. "Internal type theory". In: *Types for Proofs and Programs*. Ed. by Stefano Berardi and Mario Coppo. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 120–134. ISBN: 978-3-540-70722-6.

[Göd38] Kurt Gödel. "The Consistency of the Axiom of Choice and of the Generalized Continuum-Hypothesis". In: *Proceedings of the National Academy of Sciences* 24.12 (1938), pp. 556–557. DOI: 10.1073/pnas.24.12.556. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.24.12.556. URL: https://www.pnas.org/doi/abs/10.1073/pnas.24.12.556.

[GWB24] Daniel Gratzer, Jonathan Weinberger, and Ulrik Buchholtz. *Directed univalence in simplicial homotopy type theory*. 2024. arXiv: 2407.09146 [cs.LO]. URL: https://arxiv.org/abs/2407.09146.

[GWB25] Daniel Gratzer, Jonathan Weinberger, and Ulrik Buchholtz. *The Yoneda embedding in simplicial type theory*. 2025. arXiv: 2501.13229 [cs.LO]. URL: https://arxiv.org/abs/2501.13229.

[HH24] Perry Hart and Kuen-Bang Hou. *Coslice Colimits in Homotopy Type Theory*. 2024. arXiv: 2411.15103 [cs.LO]. URL: https://arxiv.org/abs/2411.15103.

[Hof97] Martin Hofmann. "Syntax and Semantics of Dependent Types". In: *Semantics and Logics of Computation*. Cambridge University Press, 1997, pp. 79–130.

[Jac93] Bart Jacobs. “Comprehension categories and the semantics of type dependency”. In: *Theoretical Computer Science* 107.2 (1993), pp. 169–207. ISSN: 0304-3975. DOI: 10.1016/0304-3975(93)90169-T. URL: https://doi.org/10.1016/0304-3975(93)90169-T.

[JT07] André Joyal and Myles Tierney. “Quasi-categories vs Segal spaces”. In: *Categories in algebra, geometry and mathematical physics*. Vol. 431. Contemp. Math. Amer. Math. Soc., Providence, RI, 2007, pp. 277–326. ISBN: 978-0-8218-3970-6; 0-8218-3970-5. DOI: 10.1090/conm/431/08278. URL: https://doi.org/10.1090/conm/431/08278. Available on the arXiv:math/0607820 [math.AT].

[JY21] Niles Johnson and Donald Yau. *2-Dimensional Categories*. Oxford University Press, Jan. 2021. ISBN: 9780198871378. DOI: 10.1093/oso/9780198871378.001.0001. URL: https://doi.org/10.1093/oso/9780198871378.001.0001. Available on the arXiv:2002.06055 [math.CT].

[KKA19] Ambrus Kaposi, András Kovács, and Thorsten Altenkirch. “Constructing Quotient Inductive-inductive Types”. In: *Proc. ACM Program. Lang.* 3.POPL (Jan. 2019), 2:1–2:24. ISSN: 2475-1421. DOI: 10.1145/3290315. URL: http://doi.acm.org/10.1145/3290315.

[KL21] Krzysztof Kapulkin and Peter Lefanu Lumsdaine. “The simplicial model of Univalent Foundations (after Voevodsky)”. In: *Journal of the European Mathematical Society* 23.6 (2021), pp. 2071–2126. Available on the arXiv:1211.2851 [math.LO].

[Kra21a] Nicolai Kraus. “Internal ∞-Categorical Models of Dependent Type Theory: Towards 2LTT Eating HoTT”. In: *Proceedings of the 36th Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS ’21. Rome, Italy: Association for Computing Machinery, 2021. ISBN: 9781665448956. DOI: 10.1109/LICS52264.2021.9470667. URL: https://doi.org/10.1109/LICS52264.2021.9470667.

[Kra21b] Nicolai Kraus. *Internal ∞-Categorical Models of Dependent Type Theory: Towards 2LTT Eating HoTT*. 2021. arXiv: 2009.01883 [cs.LO]. URL: https://arxiv.org/abs/2009.01883. Extended version of conference paper of the same title.

[KS17] Nicolai Kraus and Christian Sattler. “Space-Valued Diagrams, Type-Theoretically (Extended Abstract)”. In: *ArXiv e-prints* (2017). URL: https://arxiv.org/abs/1704.04543.

[Lum10] Peter LeFanu Lumsdaine. “Weak ω-categories from intensional type theory”. In: *Logical Methods in Computer Science* Volume 6, Issue 3, 24 (Sept. 2010). ISSN: 1860-5974. DOI: 10.2168/LMCS-6(3:24)2010. URL: https://lmcs.episciences.org/1062.

[Mog91] Eugenio Moggi. “A category-theoretic account of program modules”. In: *Mathematical Structures in Computer Science* 1.1 (1991), pp. 103–139. DOI: 10.1017/S0960129500000074.

[Pit01] Andrew M. Pitts. “Categorical logic”. In: *Handbook of Logic in Computer Science: Volume 5: Logic and Algebraic Methods*. USA: Oxford University Press, Inc., 2001, pp. 39–123. ISBN: 0198537816.

[Pit95] Andrew M. Pitts. *Categorical logic*. Cambridge University Computer Laboratory, May 1995. Available on the author’s webpage: https://www.cl.cam.ac.uk/~amp12/papers/catl/catl.pdf.

[Rez01] Charles Rezk. “A Model for the Homotopy Theory of Homotopy Theory”. In: *Transactions of the American Mathematical Society* 353.3 (2001), pp. 973–1007. ISSN: 00029947. URL: http://www.jstor.org/stable/221843.

[Rij22] Egbert Rijke. *Introduction to Homotopy Type Theory*. 2022. arXiv: 2212.11082 [math.LO].

[ROH09] Tillmann Rendel, Klaus Ostermann, and Christian Hofer. “Typed Self-Representation”. In: *Proceedings of the 30th ACM SIGPLAN Conference on Programming Language Design and Implementation*. PLDI ’09. Dublin, Ireland: Association for Computing Machinery, 2009, pp. 293–303. ISBN: 9781605583921. DOI: 10.1145/1542476.1542509. URL: https://doi.org/10.1145/1542476.1542509.

[RS17] Emily Riehl and Michael Shulman. “A type theory for synthetic ∞-categories”. In: *Higher Structures* 1 (1 2017), pp. 147–224. DOI: 10.21136/HS.2017.06.

[RS23] Emily Riehl and Michael Shulman. *A type theory for synthetic ∞-categories*. 2023. arXiv: 1705.07442 [math.CT].

[RSPB+] Egbert Rijke, Elisabeth Stenholm, Jonathan Prieto-Cubides, Fredrik Bakke, et al. *Wild category theory*. The agda-unimath library. URL: https://unimath.github.io/agda-unimath/wild-category-theory.html. Agda module. Last accessed 2025-02-06.

[RSS20] Egbert Rijke, Michael Shulman, and Bas Spitters. “Modalities in homotopy type theory”. In: *Logical Methods in Computer Science* Volume 16, Issue 1, 2 (Jan. 2020). ISSN: 1860-5974. DOI: 10.23638/LMCS-16(1:2)2020. URL: https://lmcs.episciences.org/3826. Available on the arXiv:1706.07526 [math.CT].

[She74] Saharon Shelah. “Infinite abelian groups, Whitehead problem and some constructions”. In: *Israel Journal of Mathematics* 18 (3 1974).

[Uni13] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study: https://homotopytypetheory.org/book, 2013.