ALGANT
Erasmus Mundus

ALGANT Master Thesis

Localizations of Models of Dependent Type Theory

Author: Matteo Durante

Supervisor: Prof. dr. Denis-Charles Cisinski
Cosupervisor: dr. Hoang Kim Nguyen

Universiteit Leiden

Universität Regensburg

Academic Year 2021-2022

Introduction

We can also rewrite the paper by Kapulkin about LCCC arising from TT using the language of localizations of quasi-categories. There they develop the relevant theory showing that under some conditions the frame associated to a fibration category is locally cartesian closed, but using Cisinski's results we can prove the same theorem directly using a more mainstream theory.

What should be included in such an overview?

1- Cisinski's theory of localizations (of fibration ∞-categories)

2- an introduction to contextual categories: where do they come from? Why are they useful? Check out Voevodsky's papers about C-systems

We explain what dependent type theory is (Martin-Lof's notes from 1984) and why it's an interesting foundation of mathematics. We mention Homotopy Type Theory as an effort to provide homotopical foundations which better model how we think about identities, which explains why intensional identity types are more interesting to us than extensional ones.

We move on to defining contextual categories (1211.2851, 1406.7413, 1507.02648) and what the Pi, Sigma and Id structures are (1406.7413, 1211.2851 Appendix B). To understand what the link between such structures and syntactically presented type theories we refer to 1507.02648, Sec. 1.1, while the statement of the conjectured correspondence is in 1304.0680, Sec. 2.1.

Where does the link between dependent type theories and ∞-categories come from? We see that ∞-categories intuitively model the behavior of type theories and their type constructions, especially when considering Homotopy Type Theory, however this relation is known only partially (references in the intro of 1507.02648). The idea is that the type theory we are interested in should be the internal language of some class of ∞-categories and a precise statement would require us to provide homotopical functors in both directions which induce an equivalence on the associated ∞-categories. The idea is to construct the functor from contextual categories as a localization functor, that is we need to provide a homotopical structure on contextual categories, as they do in 1507.02648 (there should be an older reference) which then provides an associated ∞-category. This is the object of the Initiality Conjecture, stated in 1610.00037, in the hope that such a correspondence will extend to Homotopy Type Theory and some notion of Elementary Higher Toposes, perhaps the one specified in 1805.03805. At the moment we know that HoTT can be interpreted in Higher Toposes with some structure. Current progress: 1709.09519, an upcoming paper by Nguyen-Uemura (HoTTest talk).

Our aim is to show that when taking contextual categories with the structure we specified earlier we obtain a locally cartesian closed ∞-category. To do so we provide a fibrational structure on contextual categories (1304.0680, 1507.02648), which as we anticipate will imply that their simplicial localizations are finitely complete. We also prove that the hypothesis of [Cis19, Thm. 7.6.16] are satisfied, informing that this will be sufficient to prove Kapulkin's main result from 1507.02648.

We then develop the theory of localizations of ∞-categories by Cisinski and specifically develop the results concerning ∞-categories with fibrations and weak equivalences. Localizations of such ∞-categories are finitely complete. The objective is to show [Cis19, Thm. 7.6.16]. How in depth should we go?

Why all of this is interesting: we are proving Kapulkin's result internalizing all of the discussion within the language of ∞-category theory and relying only on its simplicial model.

Acknowledgements

I would like to thank my supervisor, Hoang Kim Nguyen, both for suggesting me this intriguing topic and for helping and guiding me in every step of the creation of this thesis.
I would like to thank my cosupervisor, Denis-Charles Cisinski, for making the theory of ∞-categories accessible to me and others.
I would like to thank all the professors and the staff in Leiden and in Regensburg universities and all the ALGANT consortium to make this experience possible.
Thank you to all the friends I met in Leiden and Regensburg. A special thank you goes to those who supported me during these last few difficult years, for listening and encouraging me when I was feeling unwell.

Thank you to my parents, who have always been there, supporting me both emotionally and economically and always encouraging me to pursue my interests in any field.

Matteo Durante
Regensburg, July 2022

Contents

1 Background

We begin by presenting our dependent type theory and then introduce a class of algebraic models.

1.1 Logical Setting

We assume some familiarity with some kind of dependent type theory, preferably Martin-Löf Type Theory [ML84] or an extension, which should be enough to understand our setting. This has become a popular formal theory of logic to start from when creating an alternative foundation of mathematics since it provides a more nuanced concept of equality, the possibility of creating powerful proof assistants and, thanks to its relation to homotopy theory, cleaner and more general proofs of some theorems, like the Black-Massey theorem, presented in the language of Homotopy Type Theory [Uni13].

Notation 1.1.1. A context Γ is a sequence $a_1 : A_1, \dots, a_n : A_n$ of types A_i with choices of terms $a_i : A_i$, each of them derivable from the previous ones and possibly dependent on them.

$$a_1 : A_1, \dots, a_{i-1} : A_{i-1} \vdash A_i \text{ type}$$
$$x_1 : A_1, \dots, x_{i-1} : A_{i-1} \vdash x_i : A_i.$$

We shall hide the dependency on previous terms in our notation (as we have just done) unless it is needed for clarity, in which case we may write $B(a_1, \dots, a_n)$ for a type B dependent on $a_1 : A_1, \dots, a_n : A_n$ and similarly $b(a_1, \dots, a_n) : B(a_1, \dots, a_n)$ for a dependent term or suppress individual dependencies.

We will be considering the structural rules from [KL12, App. A.1] and some logical ones as presented in [KL12, App. A.2]. Informally, we deal with:

1. *dependent sums* $\Sigma(A, B)$, also called Σ-*types*, and the associated introduction, elimination and computation rules;

2. *dependent products* $\Pi(A, B)$, also called Π-*types*, and the associated introduction, elimination and computation rules;

3. *(intensional) identity types* Id_A, also called Id-*types*, and the associated introduction, elimination and computation rules.

Furthermore, we consider the following two rules:

4. *η-conversion for functions*: for any type A and any type B dependent on $x : A$, given a function $f : \Pi(A, B)$, we have $\lambda(x : A).\, \mathsf{app}(f, x) \equiv f$;

5. *function extensionality*: for any type A and any type B dependent on $x : A$, given functions $f, g : \Pi(A, B)$, if $\mathsf{app}(f, x) = \mathsf{app}(g, x)$ for every $x : A$, then $f = g$.

Definition 1.1.2. We say that a dependent type theory $\mathbf{T}$ has Π_η*-types* if its Π-types satisfy the η-conversion for functions and, if it also provides function extensionality, then we talk about Π-ext-types.

Remark 1.1.3. Observe that when specifying η-conversion we used a *judgemental equality*, while for function extensionality we used a simple equality. The latter means that, given $h : \Pi(x : A, \mathrm{Id}_B(\mathsf{app}(f, x), \mathsf{app}(g, x)))$, we can present a term $\mathsf{ext}(f, g, h)$ of type $\mathsf{Id}_{\Pi(A,B)}(f, g)$.

Remark 1.1.4. It should be noted that function extensionality is a common assumption, however it is sometimes derived instead from other principles, like in Homotopy Type Theory, where it is a consequence of univalence [Uni13, Ch. 4.9]. Also, the other principles we are considering are fairly general and very desirable for any dependent type theory meant as a candidate foundation of mathematics, however they are still not enough for such a role: indeed, we lack introduction rules to construct any type or term starting from the empty context. This is fine, since it means that our results will apply to virtually any type-theoretic foundation of mathematics.

1.2 Contextual Categories

A problem of working with dependent type theory is the handling of the structural rules, namely context substitution, variable binding, variable substitution and so on. To avoid this issue, it is convenient to work within a *semantic model*, which we now present.
There are many algebraic models of dependent type theory in the literature meant to only satisfy these rules, like *category with attributes*, *categories with families* and *comprehension categories*, which are fairly similar among them. Here we shall work with *contextual categories*, which were first explored by Cartmell and Streicher [Car78; Car86; Str91] and later by Voevodsky, under the name *C-systems* [Voe14b]. For informations on the others we refer to [nLab22].

Definition 1.2.1. A *contextual category* C is a small category, also denoted C, with the following data:

1. a length function $l\colon \mathrm{Ob}\,\mathsf{C} \to \mathbb{N}$ or, equivalently, a grading on objects $\mathrm{Ob}\,\mathsf{C} = \coprod_{n\in\mathbb{N}} \mathrm{Ob}_n\,\mathsf{C}$, also called *contexts*;

2. an object $* \in \mathrm{Ob}_0\,\mathsf{C}$, the *empty context*;

3. for each $n \in \mathbb{N}$, a map $\mathsf{ft}_n\colon \mathrm{Ob}_{n+1}\,\mathsf{C} \to \mathrm{Ob}_n\,\mathsf{C}$, often simply denoted ft;

4. for each $n \in \mathbb{N}$ and $X \in \mathrm{Ob}_{n+1}\,\mathsf{C}$, a map $p_X\colon X \to \mathsf{ft}\,X$;

5. for each $n \in \mathbb{N}$, $X \in \mathrm{Ob}_{n+1}\,\mathsf{C}$ and $f\colon Y \to \mathsf{ft}\,X$, an object f^*X and a map $q(f, X)\colon f^*X \to X$;

such that:

1. $*$ is the unique element of $\mathrm{Ob}_0\,\mathsf{C}$;

2. $*$ is terminal;

3. for each n, $X \in \mathrm{Ob}_{n+1}\,\mathsf{C}$ and $f\colon Y \to \mathsf{ft}\,X$, we have $\mathsf{ft}(f^*X) = Y$ and the square

$$\begin{tikzcd} f^*X \arrow[r, "q(f,X)"] \arrow[d, "p_{f^*X}"'] & X \arrow[d, "p_X"] \\ Y \arrow[r, "f"'] & \mathsf{ft}\,X \end{tikzcd}$$

is a pullback;

4. for each $n \in \mathbb{N}$, $X \in \mathrm{Ob}_{n+1}\,\mathsf{C}$ and pair of maps $f\colon Y \to \mathsf{ft}\,X$, $g\colon Z \to Y$, we have $(fg)^*X = g^*f^*X$, $1^*_{\mathsf{ft}\,X}X = X$, $q(fg,X) = q(f,X)\cdot q(g,f^*X)$ and $q(1_{\mathsf{ft}\,X},X) = 1_X$.

Remark 1.2.2. The last condition in the definition means that our choice of pullbacks is functorial, which allows us to see contextual categories as a strict model of dependent type theory, not requiring to keep track of coherency maps. Other models, like comprehension categories, do not have such requirements, which makes them more general and interpretation harder, unless they are first strictified, that is replaced by an equivalent model of the same kind but with functorial pullbacks.

A motivating example for contextual categories is the *syntactic category of a dependent type theory*, which is constructed as follows.

Construction 1.2.3. [Car78, p. 2.6] Given a dependent type theory $\mathbf{T}$, its syntactic category $\mathsf{Syn}(\mathbf{T})$ has:

1. $\mathrm{Ob}_n\,\mathsf{Syn}(\mathbf{T})$ given by contexts $[x_1 : A_1, \ldots, x_n : A_n]$ of length n, modulo judgemental equality and renaming of free variables;

2. maps are *context morphisms*, or *substitutions*, modulo judgemental equalities and renaming of free variables. This means that a map

$$f\colon [x_1 : A_1, \ldots, x_n : A_n] \to [y_1 : B_1, \ldots, y_m : B_m(y_1, \ldots, y_{m-1})]$$

is an equivalence class of tuples of terms $(f_1, \ldots, f_m)$ such that

$$\begin{gathered} x_1 : A_1, \ldots, x_n : A_n \vdash f_1 : B_1, \\ \vdots \\ x_1 : A_1, \ldots, x_n : A_n \vdash f_m : B_m(f_1, \ldots, f_{m-1}), \end{gathered}$$

are all derivable judgements and two such tuples $(f_1, \ldots, f_n), (g_1, \ldots g_n)$ are equivalent if we have

$$x_1 : A_1, \ldots, x_n : A_n \vdash f_i \equiv g_i : B_i(f_1, \ldots, f_{i-1})$$

for every i; we shall henceforth also write $[f_i]$ for f;

3. composition is given by substitution;

4. the identity $\Gamma \to \Gamma$ is given by the variables of Γ, considered as terms, that is we take the sequence $[x_i]$ given by

$$\begin{gathered} x_1 : A_1, \dots, x_n : A_n \vdash x_1 : A_1, \\ \vdots \\ x_1 : A_1, \dots, x_n : A_n \vdash x_n : A_n(x_1, \dots, x_{n-1}); \end{gathered}$$

5. the terminal object is the empty context $[]$;

6. $\mathsf{ft}([x_1 : A_1, \dots, x_{n+1} : A_{n+1}]) = [x_1 : A_1, \dots, x_n : A_n]$;

7. for $\Gamma = [x_1 : A_1, \dots, x_n : A_{n+1}]$, $p_\Gamma : \Gamma \to \mathsf{ft}\,\Gamma$ is the *dependent projection context morphism* $[x_1, \dots, x_n]$, defined by

$$\begin{gathered} x_1 : A_1, \dots, x_{n+1} : A_{n+1} \vdash x_1 : A_1, \\ \vdots \\ x_1 : A_1, \dots, x_{n+1} : A_{n+1} \vdash x_n : A_n(x_1, \dots, x_n) \end{gathered}$$

   and thereby simply forgetting the last variable of Γ;

8. given contexts

$$\begin{aligned} \Gamma &= [x_1 : A_1, \dots, x_{n+1} : A_{n+1}(x_1, \dots, x_n)], \\ \Delta &= [y_1 : B_1, \dots, y_m : B_m(y_1, \dots, y_{m-1})] \end{aligned}$$

   and a map $f = [f_i(y)] : \Delta \to \mathsf{ft}\,\Gamma$ (where y is a vector of variables of length m), the pullback $f^*\Gamma$ is the context

$$[y_1 : B_1, \dots, y_m : B_m(y_1, \dots, y_{m-1}), y_{m+1} : A_{n+1}(f_1(y), \dots, f_n(y))]$$

   for some new variable y_{m+1}, while $q(\Gamma, f) : f^*\Gamma \to \Gamma'$ is specified by $[f_1, \dots, f_n, y_{m+1}]$.

Remark 1.2.4. Given a dependent type theory $\mathbf{T}$, the terms $a : A$ of a type over a context Γ can be recovered (up to definitional equality) from the syntactic category $\mathsf{Syn}(\mathbf{T})$ by looking at sections of the basic dependent projection $p_{[\Gamma, x:A]} : [\Gamma, x : A] \to [\Gamma]$, which indeed act as identities over Γ and furthermore specify a term $\Gamma \vdash a : A$. Given the importance of such maps, we shall often simply write "sections" to refer to sections of basic dependent projections, without specifying which projections unless it creates ambiguity.
The above construction also tells us how to think about the other elements in the definition of contextual categories: basic dependent projections $p_{\Gamma.A.B} : \Gamma.A.B \to \Gamma.A$ represent dependent types B over A in the context Γ, while pulling back along a dependent projection corresponds to switching context and so on for the other objects in the definition.

Notation 1.2.5. We shall also make use of some conventions inspired by this construction. Namely, given a contextual category C and an object $\Gamma \in \mathrm{Ob}_n \mathsf{C}$, we shall write $\Gamma.A_1.\dots.A_k$ and $\Gamma.\Delta$ interchangeably for an object X in $\mathrm{Ob}_{n+k}\mathsf{C}$ with $\mathsf{ft}^k X = \Gamma$ and call it a *context extension of* Γ *of length* k. We shall also write $p_{A_1.\dots.A_k}$ and p_Δ for the composition of the basic dependent projections $p_{\Gamma.A_1.\dots.A_i}$, with i ranging from 1 to k, and the resulting map will be called a *dependent projection*. In the case where $k = 1$, p_{A_1} corresponds to a basic dependent projection and the context extension of Γ will be *simple*, while if $k = 0$ we have $p = 1_\Gamma$ and then the context extension will be *trivial*. We shall also write $1_\Delta, 1_{A_1.\dots.A_k}, \dots, 1_{A_k}$ for $1_{\Gamma.\Delta}$, depending on what we want to emphasize. Greek letters shall be used to indicate context extensions of arbitrary length, while Latin ones will be reserved to simple extensions.
Continuing, given a dependent projection $p_{A_1.\dots.A_k} = p_\Theta$ as above and a context morphism $f\colon \Delta \to \Gamma$, we define inductively $f^*(\Gamma.\Theta) = \Delta.f^*\Theta = \Delta.f^*A_1.\dots.f^*A_k$ and $q(f,\Gamma.\Theta) = q(f,\Theta) = q(f, A_1.\dots.A_k)$ by looking at the pasting of pullback squares

$$\begin{tikzcd}
\Delta.f^*A_1.\dots.f^*A_k \arrow[r, "p_{f^*A_k}"'] \arrow[d, "q(f,A_1.\dots.A_k)"'] \arrow[rrrr, bend left, "p_{f^*A_1.\dots.f^*A_k}"] & \Delta.f^*A_1.\dots.f^*A_{k-1} \arrow[r] \arrow[d, "q(f,A_1.\dots.A_{k-1})"] & \cdots \arrow[r] & \Delta.f^*A_1 \arrow[r, "p_{f^*A_1}"'] \arrow[d, "q(f,A_1)"'] & \Delta \arrow[d, "f"] \\
\Gamma.A_1.\dots.A_k \arrow[r, "p_{A_k}"] \arrow[rrrr, bend right, "p_{A_1.\dots.A_k}"'] & \Gamma.A_1.\dots.A_{k-1} \arrow[r] & \cdots \arrow[r] & \Gamma.A_1 \arrow[r, "p_{A_1}"] & \Gamma
\end{tikzcd},$$

which also shows that

$$q(f, A_1.\dots.A_k) = q(q(f, A_1), A_2.\dots.A_k) = q(q(f, A_1.\dots.A_{k-1}), A_k).$$

As usual, if $k = 0$ we have

$$q(f,\Theta) = f, \quad f^*(\Gamma.\Theta) = \Gamma,$$

while for $k = 1$

$$q(f, A_1) = q(f, \Gamma.A_1), \quad \Delta.f^*A_1 = f^*(\Gamma.A_1),$$

therefore agreeing with the base structure of C.
Furthermore, given a section $a\colon \Gamma \to \Gamma.A$ and a context morphism $f\colon \Delta \to \Gamma$, we also want to specify f^*a, that is the section which we get by switching context. This is given by the map $(1_\Delta, a \cdot f)$ specified by the pullback square

$$\begin{tikzcd}
\Delta \arrow[dr, dashed, "(1_\Delta, a\cdot f)"] \arrow[drr, bend left, "a\cdot f"] \arrow[ddr, equal, bend right] & & \\
& \Delta.f^*A \arrow[r, "q(f,A)"'] \arrow[d, "p_{f^*A}"] & \Gamma.A \arrow[d, "p_A"] \\
& \Delta \arrow[r, "f"'] & \Gamma
\end{tikzcd},$$

and, as shown by the commutative diagram

$$\begin{tikzcd}
\Delta \arrow[r, "f"] \arrow[d, "f^*a"] \arrow[dd, bend right=60, equal] & \Gamma \arrow[d, "a"'] \arrow[dd, bend left=60, equal] \\
\Delta.f^*A \arrow[r, "q(f,A)"'] \arrow[d, "p_{f^*A}"] & \Gamma.A \arrow[d, "p_A"'] \\
\Delta \arrow[r, "f"'] & \Gamma
\end{tikzcd},$$

it corresponds to the pullback of a along $q(f, A)$. By the techniques we provided earlier, we extend this construction to contexts extensions of arbitrary length.
Finally, when reasoning in a syntactic category, to specify a map

$$[f] \colon [\Gamma, a_1 : A_1, \dots, a_n : A_n] \to [\Gamma, b_1 : B_1, \dots, b_m : B_m]$$

stable over Γ (that is acting like the identity on the corresponding variables) we can suppress the associated terms in the tuple. To produce such a map, we shall write then

$$(a_1, \dots, a_n) \mapsto (f_1, \dots, f_m)$$

and clarify how to construct the f_i from the variables in the domain context. We will also write

$$(f_1, \dots, f_n) \equiv (g_1, \dots, g_n)$$

if we have a derivation of the judgment

$$\Gamma, a_1 : A_1, \dots, a_n : A_n \vdash f_i \equiv g_i : B_i(f_1, \dots, f_{i-1})$$

for every i.

Remark 1.2.6. When we pull back a dependent projection $p_\Delta \colon \Gamma.\Delta \to \Gamma$ along another one p_Ψ, we may also choose to do it the other way around, which induces an isomorphism $\Gamma.\Delta.p_\Delta^*\Psi \to \Gamma.\Psi.p_\Psi^*\Delta$ denoted by $\mathsf{exch}_{\Delta,\Psi}$. Syntactically, this amounts to swapping two context extensions not dependent on one another.

Definition 1.2.7. A *contextual functor* between contextual categories $F \colon \mathsf{C} \to \mathsf{D}$ is a functor on the underlying categories which preserves the grading, basic dependent projections and such that $q(Ff, FX) = F(q(f, X))$.

Remark 1.2.8. Our definition allows us to see contextual categories as models for a (small) essentially algebraic theory [AR94] with sorts indexed by $\mathbb{N} + \mathbb{N} \times \mathbb{N}$. In that context, we get a notion of morphism between models of this theory, which coincides with the one we have just provided. Its category of models will be the category of contextual categories, denoted by CxlCat, which is complete and cocomplete as the category of models of an essentially algebraic theory. The same can be done for the other algebraic models we mentioned.

As we stated, the defining properties of contextual categories are meant to model structural rules, so that we can reason about dependent type theories in terms of pullbacks, sections and so on while eliminating a lot of the bureaucracy needed to provide a model, which can now be done by constructing such a category.

Example 1.2.9. A constructive and formalization-ready approach to producing contextual categories was developed by Voevodsky in [Voe14a; Voe14a; Voe15b; Voe15a], where he starts from a *universe category.*

On the other hand, as noted in [KL12, Par. 1.2], interpreting complicated structures in contextual categories quickly becomes unreadable, meaning that ideally we would like to be able to switch from a syntactic to a semantic (i.e. algebraic) presentation and viceversa freely, so that we may work with the most convenient one for a given situation. This would amount to some *soundness* and *correctness* theorems reminiscent of the ones of *first order predicate logic* establishing an equivalence between the two and allowing us to think of contextual categories as if they were dependent type theories. In categorical semantics, this corresponds to *initiality.*

Initiality Conjecture. *Given a dependent type theory* $\mathbf{T}$*, its syntactic category* $\mathsf{Syn}(\mathbf{T})$ *is initial in the category of contextual categories with the appropriate structure.*

Remark 1.2.10. Here by "appropriate structure" we mean extra algebraic structures meant to model the logical rules of the type theory, like Σ-types, Π-types and Id-types. Indeed, the definition of contextual category is not meant to deal with them. We will introduce the such structures in the next section.

Remark 1.2.11. This conjecture has been partially proven for a simple variant of dependent type theory in [Str91] and a proof formalized in Agda for Martin-Löf Type Theory was provided by de Boer and Brunerie [Boe20], however we still do not have a fully rigorous general statement. Such a statement would first require a general notion of dependent type theory, which has been worked on in [Isa16; Uem19; Bru20; BHL20; NU22]. (ARE YOU SURE ABOUT THESE CITATIONS?)

If we could prove it, then, under the proper conditions, for any contextual category C we would have a contextual functor $\mathsf{Syn}(\mathbf{T}) \to \mathsf{C}$ explaining how to interpret $\mathbf{T}$ in C. This is essentially an algorithmic problem: it reduces to explaining inductively to a computer how to construct the aforementioned functor.

Remark 1.2.12. From the Initiality Conjecture researchers have also derived some homotopical models, which are the great drivers of research in the field.
Specifically, as a concrete example, in [KL12] one can find a model of Homotopy Type Theory produced in the category of ∞-groupoids using Voevodsky's previously mentioned methods. Furthermore, it was shown that ∞*-toposes* can also provide a semantics for it [Shu19]. More recent results concern the *Internal Languages Conjecture* 4.3.

Remark 1.2.13. A reason researchers often assume this conjecture to be true which was briefly is that, thanks to what is known in folklore as the *syntax-emantics adjunction*, it provides for each contextual category an *internal language*, that is a dependent type theory corresponding to it and allowing to think of it as such, as we alluded earlier. We shall not use such instruments and instead restrict ourselves to working with syntactic categories, with the idea that a generalization of our results to arbitrary contextual categories should be straightforward once we have a proof of the conjecture and all of the tools mentioned above. Unfortunately, this will also mean that we will not be able to rely on some results and definitions as originally stated.

Remark 1.2.14. It should be noted that, since CxlCat and its variants are complete, they do have initial objects.

1.3 Logical Structures

We now define the extra structures on contextual categories we mentioned earlier. Our definitions shall be taken from [KL12; KL18].

Definition 1.3.1. A Σ-*structure* on a contextual category C consists of:

1. for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$, an object $\Gamma.\,\Sigma(A,B) \in \mathrm{Ob}_{n+1}\,\mathsf{C}$;
2. for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$, a morphism $\mathsf{pair}_{A,B} \colon \Gamma.A.B \to \Gamma.\,\Sigma(A,B)$ over Γ;
3. for each $\Gamma.A.B, \Gamma.\,\Sigma(A,B).C \in \mathrm{Ob}_{n+2}\,\mathsf{C}$, and $d \colon \Gamma.A.B \to \Gamma.\,\Sigma(A,B).C$ with $p_C \cdot d = \mathsf{pair}_{A,B}$, a section $\mathsf{split}_d \colon \Gamma.\,\Sigma(A,B) \to \Gamma.\Sigma(A,B).C$ such that $\mathsf{split}_d \cdot \mathsf{pair}_{A,B} = d$;
4. where all of the above is compatible with context substitution, that is given a map $f \colon \Delta \to \Gamma$ we have

$$\begin{aligned} f^*(\Gamma.\,\Sigma(A,B)) &= \Delta.\Sigma(f^*A, f^*B), \\ f^*\,\mathsf{pair}_{A,B} &= \mathsf{pair}_{f^*A,f^*B}, \\ f^*\,\mathsf{split}_d &= \mathsf{split}_{f^*d}\,. \end{aligned}$$

Definition 1.3.2. A Id-*structure* on a contextual category C consists of:

1. for each $\Gamma.A \in \mathrm{Ob}_{n+1}\,\mathsf{C}$, an object $\Gamma.A.p_A^*A.\,\mathsf{Id}_A \in \mathrm{Ob}_{n+3}\,\mathsf{C}$;
2. for each $\Gamma.A \in \mathrm{Ob}_{n+1}\,\mathsf{C}$, a morphism $\mathsf{refl}_A \colon \Gamma.A \to \Gamma.A.p_A^*A.\,\mathsf{Id}_A$ such that $p_{\mathsf{Id}_A} \cdot \mathsf{refl}_A = (1_A, 1_A) \colon \Gamma.A \to \Gamma.A.p_A^*A$;
3. for each $\Gamma.A.p_A^*A.\,\mathsf{Id}_A\,.C$ and $d \colon \Gamma.A \to \Gamma.A.p_A^*A.\,\mathsf{Id}_A\,.C$ with $p_C \cdot d = \mathsf{refl}_A$, a section $\mathsf{J}_{C,d} \colon \Gamma.A.p_A^*A.\,\mathsf{Id}_A \to \Gamma.A.p_A^*A.\,\mathsf{Id}_A\,.C$ such that $\mathsf{J}_{C,d} \cdot \mathsf{refl}_A = d$;
4. where all of the above is compatible with context substitution, that is given a map $f \colon \Delta \to \Gamma$ we have

$$\begin{aligned} f^*(\Gamma.A.p_A^*A.\,\mathsf{Id}_A) &= \Delta.f^*A.p_{f^*A}^*(f^*A).\,\mathsf{Id}_{f^*A} \\ f^*\,\mathsf{refl}_A &= \mathsf{refl}_{f^*A} \\ f^*\,\mathsf{J}_{C,d} &= \mathsf{J}_{f^*C,f^*d}. \end{aligned}$$

Definition 1.3.3. A Π-*structure* on a contextual category C consists of:

1. for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$, an object $\Gamma.\,\Pi(A,B) \in \mathrm{Ob}_{n+1}\,\mathsf{C}$;
2. for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$, a map $\mathsf{app}_{A,B}\colon \Gamma.\,\Pi(A,B).p^*_{\Pi(A,B)}A \to \Gamma.A.B$ over Γ, that is such that $p_B \cdot \mathsf{app}_{A,B} = q(\Pi(A,B),A)$;
3. for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$ and section $b\colon\Gamma.A \to \Gamma.A.B$, a section $\lambda_{A,B}(b)\colon\Gamma \to \Gamma.\,\Pi(A,B)$;
4. such that for any sections $k\colon\Gamma \to \Gamma.\,\Pi(A,B)$, $a\colon\Gamma \to \Gamma.A$ the map $\mathsf{app}_{A,B}(k,a)$ defined as the composition of $\mathsf{app}_{A,B}$ with (k,a) specified by the factorization through the pullback

$$\begin{tikzcd}
\Gamma \arrow[dr, dotted, "{(k,a)}"] \arrow[drr, bend left, "a"] \arrow[ddr, bend right, "k"'] & & \\
& \Gamma.\,\Pi(A,B).p^*_{\Pi(A,B)}A \arrow[r, "{q(p_{\Pi(A,B)},A)}"'] \arrow[d, "p_{p^*_{\Pi(A,B)}A}"] & \Gamma.A \arrow[d, "p_A"] \\
& \Gamma.\,\Pi(A,B) \arrow[r, "p_{\Pi(A,B)}"'] & \Gamma
\end{tikzcd},$$

   we have $p_B \cdot \mathsf{app}_{A,B}(k,a) = a$;
5. such that for any $\Gamma.A.B$ and sections $a\colon\Gamma \to \Gamma.A$, $b\colon\Gamma.A \to \Gamma.A.B$ we have
$$\mathsf{app}(\lambda_{A,B}(b),a) = b \cdot a;$$
6. all of the above is compatible with context substitution, that is for any $f\colon\Delta \to \Gamma$ we have
$$\begin{aligned}
f^*(\Gamma,\Pi(A,B)) &= (\Delta,\Pi(f^*A,f^*B)),\\
f^*\lambda_{A,B}(b) &= \lambda_{f^*A,f^*B}(f^*b),\\
f^*(\mathsf{app}_{A,B}(k,a)) &= \mathsf{app}_{f^*A,f^*B}(f^*k,f^*a).
\end{aligned}$$

We shall say that the Π-structure satisfies the Π_η-*rule* if the equation

$$q(p_{\Pi(A,B)},\Pi(A,B)) \cdot \lambda(1_{p^*_{\Pi(A,B)}A}, \mathsf{app}_{A,B}) = 1_{\Gamma.\,\Pi(A,B)}$$

is satisfied, in which case the structure will be called a Π_η-*structure*.
A Π-ext-*structure* on a contextual category with a Π-structure C is an operation giving for each $\Gamma.A.B \in \mathrm{Ob}_{n+2}\,\mathsf{C}$ a map

$$\begin{aligned}
\mathsf{ext}_{A,B}\colon \Gamma.\,\Pi(A,B).p^*_{\Pi(A,B)}\,\Pi(A,B).\,\mathsf{Htp}_{A,B} \to\\
\Gamma.\,\Pi(A,B).p^*_{\Pi(A,B)}\,\Pi(A,B).\,\mathsf{Id}_{\Pi(A,B)}
\end{aligned}$$

over $\Gamma.\,\mathsf{\Pi}(A,B).p^*_{\mathsf{\Pi}(A,B)}\,\mathsf{\Pi}(A,B)$, stably in Γ, where $\mathsf{Htp}_{A,B}$ is a shorthand for the extension

$$\mathsf{\Pi}\Big((p_{\mathsf{\Pi}(A,B)}\cdot p_{p^*_{\mathsf{\Pi}(A,B)}\,\mathsf{\Pi}(A,B)})^* A,\\ (\mathsf{app}_{A,B}\cdot q(p_{\mathsf{\Pi}(A,B)},A), \mathsf{app}_{A,B}\cdot q(q(p_{\mathsf{\Pi}(A,B)},\mathsf{\Pi}(A,B)),A))^*\,\mathsf{Id}_B\Big),$$

representing the type of homotopies from between two terms of $\mathsf{\Pi}(A,B)$. The map then models function extensionality.

Remark 1.3.4. The definition we provided matches [KL18, Def. 2.5], while the one in [KL12, App. B.1.1] is mildly different as it starts from $\mathsf{app}(f,a)$ instead of $\mathsf{app}_{A,B}$ and constructs the latter from the former. We prefer ours because it will allow us to more easily extend the $\mathsf{\Pi}$-structure to arbitrary context extensions and check the necessary properties in Chapter 2.

Example 1.3.5. Given a dependent type theory $\mathbf{T}$, its syntactic category $\mathsf{Syn}(\mathbf{T})$ has all of the logical structures corresponding to its logical rules. We now proceed to show what some of the above maps correspond to in this category.

Remark 1.3.6. In $\mathsf{Syn}(\mathbf{T})$, the map $\mathsf{app}_{A,B}$ corresponds to

$$(f,a)\mapsto(a,\mathsf{app}(f,a)).$$

Also, the association $b\mapsto\lambda_{A,B}(b)$ for sections $\Gamma.A\to\Gamma.A.B$ corresponds to *lambda abstraction*, hence $\lambda_{A,B}(b)$ specifies the term $\lambda(a:A).b(a):\mathsf{\Pi}(A,B)$.
The map on the left of the definition of the $\mathsf{\Pi}_\eta$-rule is the *η-expansion map*, that is it corresponds syntactically to the map

$$f\mapsto\lambda(a:A).\,\mathsf{app}(f,a)$$

over Γ, thus the property means that we have a judgement

$$\Gamma\vdash f\equiv\lambda(a:A).\,\mathsf{app}(f,a):\mathsf{\Pi}(A,B)$$

which corresponds to η-conversion.
The map $(1_{p^*_{\mathsf{\Pi}(A,B)}A},\mathsf{app}_{A,B})$ is specified by the following factorization through the pullback.

$$\begin{tikzcd}
\Gamma.\,\mathsf{\Pi}(A,B).p^*_{\mathsf{\Pi}(A,B)}A \arrow[dr, dashed, "{(1_{p^*_{\mathsf{\Pi}(A,B)}A},\mathsf{app}_{A,B})}"] \arrow[drr, bend left, "\mathsf{app}_{A,B}"] \arrow[ddr, equal] & & \\
& \Gamma.\,\mathsf{\Pi}(A,B).p^*_{\mathsf{\Pi}(A,B)}A.p^*_{\mathsf{\Pi}(A,B)}B \arrow[r, "{q(p_{\mathsf{\Pi}(A,B)},A.B)}"'] \arrow[d, "p_{p^*_{\mathsf{\Pi}(A,B)}A}"] & \Gamma.A.B \arrow[d, "p_A"] \\
& \Gamma.\,\mathsf{\Pi}(A,B).p^*_{\mathsf{\Pi}(A,B)}A \arrow[r, "{q(p_{\mathsf{\Pi}(A,B)},A)}"'] & \Gamma.A
\end{tikzcd}$$

Construction 1.3.7. Let C be a contextual category with a Π-structure. We define for any object $\Gamma.A$ a map $\mathsf{id}_A \colon \Gamma \to \Gamma.\,\Pi(A, p_A^*)$ as $\lambda_{A,p_A^*A}(1_A, 1_A)$.

Construction 1.3.8. Let C be a contextual category with a Π-structure, $f \colon \Gamma.A \to \Gamma.B$ a map over Γ. We want to provide a section $\hat{f} \colon \Gamma \to \Gamma.\,\Pi(A, B)$ corresponding to f. We do so by looking at the commutative diagram

$$\begin{tikzcd}
\Gamma.A \arrow[dr, dotted, "(1_A, f)"] \arrow[drr, bend left, "f"] \arrow[ddr, equal, bend right] & & \\
& \Gamma.A.p_A^*B \arrow[r, "q(p_A, B)"'] \arrow[d, "p_{p_A^*B}"] & \Gamma.B \arrow[d, "p_B"] \\
& \Gamma.A \arrow[r, "p_A"'] & \Gamma
\end{tikzcd},$$

and then taking $\lambda_{A,p_A^*B}(1_A, f)$.
In $\mathsf{Syn}(\mathbf{T})$, using

$$\begin{aligned}
q(p_A, B) \cdot \mathsf{app}_{A,B} \cdot q(p_{p_A^*B}, \lambda_{A,p_A^*B}(1_A, f)) \cdot a &= q(p_A, B) \cdot \mathsf{app}_{A,B}(\lambda_{A,p_A^*B}(1_A, f), a) \\
&= q(p_A, B) \cdot (1_A, f) \cdot a \\
&= f \cdot a,
\end{aligned}$$

we see f acting as

$$\begin{aligned}
& a \\
\scriptstyle q(p_{p_A^*B}, \lambda_{A,p_A^*B}(1_A, f)) &\mapsto (a, \hat{f}) \\
\scriptstyle \mathsf{app}_{A,B} &\mapsto (a, \mathsf{app}(\hat{f}, a)) \\
\scriptstyle q(p_A, B) &\mapsto \mathsf{app}(\hat{f}, a).
\end{aligned}$$

This justifies abusing the notation to write f for $\hat{f}$ and $(1_A, f)$ then acts as $a \mapsto (a, \mathsf{app}(f, a))$, meaning that $\lambda_{A,p_A^*B}(1_A, f)$ corresponds to $\lambda(a : A).\, \mathsf{app}(f, a) : \Pi(A, B)$ in $\mathsf{Syn}(\mathbf{T})$.

Construction 1.3.9. Let C be a contextual category with a Π-structure, $f \colon \Gamma.A.B \to \Gamma.A.B'$ a morphism over $\Gamma.A$, meaning that in $\mathsf{Syn}(\mathbf{T})$ it is a map $b \mapsto \mathsf{app}(f, b)$ for the term $f : \Pi(B, B')$ it induces. We want to construct a morphism

$$\Gamma.\,\Pi(A, f) \colon \Gamma.\,\Pi(A, B) \to \Gamma.\,\Pi(A, B')$$

modeling the postcomposition by f, that is $g \mapsto \lambda(a : A).\, \mathsf{app}(f, \mathsf{app}(g, a))$.
We do so by looking at the commutative diagram

$$\begin{tikzcd}
\Gamma.\,\Pi(A, B).p_{\Pi(A,B)}^*A \arrow[rr, "\mathsf{app}_{A,B}"] \arrow[dr, dotted, "(1_{p_{\Pi(A,B)}^*A}, f \cdot \mathsf{app}_{A,B})"] \arrow[ddr, equal, bend right] & & \Gamma.A.B \arrow[d, "f"] \arrow[dd, bend left, "p_B"] \\
& \Gamma.\,\Pi(A, B).p_{\Pi(A,B)}^*A.p_{\Pi(A,B)}^*B' \arrow[r, "q(p_{\Pi(A,B)}, A.B')"'] \arrow[d, "p_{p_{\Pi(A,B)}^*B'}"] & \Gamma.A.B' \arrow[d, "p_{B'}"] \\
& \Gamma.\,\Pi(A, B).p_{\Pi(A,B)}^*A \arrow[r, "q(p_{\Pi(A,B)}, A)"'] & \Gamma.A
\end{tikzcd}$$

and then applying $\lambda_{p^*_{\Pi(A,B)}A, p^*_{\Pi(A,B)}B'}$ to the map given by the universal property of the pullback, which provides us with the section

$$\lambda_{p^*_{\Pi(A,B)}A, p^*_{\Pi(A,B)}B'}(1_{p^*_{\Pi(A,B)}A}, f \cdot \mathsf{app}_{A,B}) : \Gamma.\, \Pi(A,B) \to \Gamma.\, \Pi(A,B).\, \Pi(A,B').$$

All we have to do now is postcompose with $q(p_{\Pi(A,B)}, \Pi(A,B'))$. Verifying that in $\mathsf{Syn}(\mathbf{T})$ the action on sections is the one we want is straightforward as we have done so far.

Remark 1.3.10. The previous construction is such that $\Gamma.\, \Pi(A, 1_B) = 1_{\Pi(A,B)}$ for every context $\Gamma.A.B$ in C if and only if the Π-structure satisfies the Π_η-rule.

Notation 1.3.11. As noted before, contextual categories with some extra logical structures still are models of small essentially algebraic theory with sorts indexed by $\mathbb{N} + \mathbb{N} \times \mathbb{N}$. Their categories of models are then specified in the literature by writing as subscript of CxlCat the corresponding structures, that is we write $\mathsf{CxlCat}_{\mathsf{Id}}$, $\mathsf{CxlCat}_{\Sigma,\mathsf{Id}}$ and so on.

2 Extended Logical Structures

The aim of this section is to construct from a given contextual category C with extra structure another contextual category $\mathsf{C}^{\mathsf{cxt}}$ with the same structure, but where objects are iterated context extensions, compatibly with a canonical contextual functor $\mathsf{C} \hookrightarrow \mathsf{C}^{\mathsf{cxt}}$ defining an equivalence on the underlying categories, thereby generalizing our structures from simple context extensions to arbitrary ones. Indeed, we may then take a context extension, look at it in $\mathsf{C}^{\mathsf{cxt}}$, apply the construction and then carry it back through the equivalence. This extension shall be heavily exploited in the final part of the thesis. This technique has been mentioned multiple times in the literature to justify various results at a certain level of generality, however nobody actually carried out the necessary constructions.

2.1 Constructions

On the type theoretical side, extensions of logical rules for Id-types to contexts were first explored by Streicher [Str93], then by Gambino and Garner [GG08; Gar09] under the name of *identity contexts*, also called Id-contexts.

Notation 2.1.1. We shall make use of their notation [GG08, p. 4] in the second part of this section and the final chapter of this thesis to study $\mathsf{Syn}(\mathbf{T})$, meaning that we abbreviate the sequence of judgements

$$\begin{gathered}\Gamma \vdash A_1 \;\; \mathsf{type}, \\ \Gamma, a_1 : A_1 \vdash A_2(a_1) \;\; \mathsf{type}, \\ \vdots \\ \Gamma, a_1 : A_1, \ldots, a_{n-1} : A_{n-1}(a_1, \ldots, a_{n-2}) \vdash A_n(a_1, \ldots, a_{n-1}) \;\; \mathsf{type}\end{gathered}$$

by

$$\Gamma \vdash \Delta \;\; \mathsf{context},$$

in which case Δ is a *dependent context in context* Γ and we shall write that $\Delta = A_1, \ldots, A_n$ as a shorthand. Similarly,

$$\Gamma \vdash x : \Delta$$

abbreviates

$$\begin{gathered}\Gamma \vdash a_1 : A_1, \\ \Gamma, a_1 : A_1 \vdash a_2 : A_2(a_1), \\ \vdots \\ \Gamma, a_1 : A_1, \ldots, a_{n-1} : A_{n-1}(a_1, \ldots, a_{n-2}) \vdash a_n : A_n(a_1, \ldots, a_{n-1})\end{gathered}$$

and then x is a *dependent term of* Δ *in context* Γ, which can be thought of as a tuple $(a_1, \dots, a_n)$. Furthermore,

$$\Gamma, x : \Delta \vdash \Theta(x) \ \mathsf{context}$$

specifies the dependency of Θ on the term $x : \Delta$ in context Γ.
It is also possible to introduce further expressions, like

$$\Gamma \vdash \Delta \equiv \Theta \ \mathsf{context}$$

for judgmental equality between dependent contexts and

$$\Gamma \vdash x \equiv x' : \Delta,$$

for judgmental equality between dependent terms, defined as pointwise judgmental equalities. The details are straightforward and therefore omitted.

Lumsdaine claimed that it is possible to provide extensions of the logical structures we are studying at [Lum10, p. 26]. At the time Kapulkin proved the theorem this thesis focuses on, nothing further was available in the literature and only later him and Lumsdaine gave more details on these matters [KL18]. In this section we aim to partially fix that for $\mathsf{\Pi}$-structures specifically, which will be the main contribution of this work. It should be noted that, imitating Gambino and Garner, we may derive the corresponding logical rules for what would then be called $\mathsf{\Pi}$-contexts, however we will not do so and focus on the semantics, believing that the translation should be straightforward and a simple duplication of work. We will later sketch the extension for $\mathsf{\Sigma}$-structures.
We begin by constructing the category of iterated contexts, where we will extend the structures.

Construction 2.1.2. [Lum10, p. 21] Given a contextual category C, we construct a contextual category $\mathsf{C}^{\mathsf{cxt}}$ in the following way:

1. the set $\mathrm{Ob}_n\,\mathsf{C}^{\mathsf{cxt}}$ is given by n-iterated non-trivial context extensions

   $$\Gamma_1.\Gamma_2.\dots.\Gamma_n$$

   in C;

2. morphisms $\Gamma_1.\Gamma_2.\dots.\Gamma_n \to \Delta_1.\Delta_2.\dots.\Delta_m$ are morphisms between them seen as objects of C;

3. $*$ is the only element of $\mathrm{Ob}_0\,\mathsf{C}^{\mathsf{cxt}}$;

4. $\mathsf{ft}(\Gamma_1.\Gamma_2.\dots.\Gamma_n.\Gamma_{n+1}) = \Gamma_1.\Gamma_2.\dots.\Gamma_n$;

5. the map $p_{\Gamma_1.\Gamma_2.\dots.\Gamma_n.\Gamma_{n+1}} : \Gamma_1.\Gamma_2.\dots.\Gamma_{n+1} \to \Gamma_1.\Gamma_2.\dots.\Gamma_n$ is the dependent projection exhibiting $\Gamma_1.\Gamma_2.\dots.\Gamma_{n+1}$ as a context extension of $\Gamma_1.\Gamma_2.\dots.\Gamma_n$;

6. the chosen pullbacks are given by iterating the pullbacks along the basic dependent projections, as in the original contextual category.

As we can see, any object of $\mathsf{C}^{\mathsf{cxt}}$ either is the empty context or is isomorphic to one in $\mathrm{Ob}_1\,\mathsf{C}^{\mathsf{cxt}}$, that is the one which we get by looking at the associated object in C and then taking the dependent projection from it to the terminal object, which exhibits it as a 1-iterated context extension. The isomorphism is then given by the map in $\mathsf{C}^{\mathsf{cxt}}$ corresponding to the identity of the object in C. We now specify a monad $\mathsf{C} \mapsto \mathsf{C}^{\mathsf{cxt}}$ on CxlCat.

The unit $\mathsf{C} \to \mathsf{C}^{\mathsf{cxt}}$ sends every n-object in C to the corresponding n-iterated (simple) context extension and every morphism to the one it represents.

Before we construct the multiplication, let's study this contextual functor. Every n-iterated context in $\mathsf{C}^{\mathsf{cxt}}$ is isomorphic to one in the image of the unit, namely the one which we get by reducing it to an iterated simple context extension, meaning that the functor is essentially surjective. Also, it is fully faithful by construction and therefore it defines an equivalence on the underlying categories.

Let's construct the multiplication. An n-object of $(\mathsf{C}^{\mathsf{cxt}})^{\mathsf{cxt}}$ is an n-iterated context extension where each extension is itself an iterated context extension in C, that is

$$(\Gamma_1.\dots.\Gamma_{i_1}).(\Gamma_{i_1+1}.\dots.\Gamma_{i_2}).\dots.(\Gamma_{i_{n-1}+1}.\dots.\Gamma_{i_n}).$$

Since composing dependent projections gives dependent projections, seeing $\Gamma_{i_{j-1}+1}.\dots.\Gamma_{i_j}$ as a single context extension Δ_j in C, we can naturally map the object of $(\mathsf{C}^{\mathsf{cxt}})^{\mathsf{cxt}}$ to $\Delta_1.\dots.\Delta_n$ in $\mathsf{C}^{\mathsf{cxt}}$ and, again, every morphism in $(\mathsf{C}^{\mathsf{cxt}})^{\mathsf{cxt}}$ corresponds to a unique one in $\mathsf{C}^{\mathsf{cxt}}$ once we specify domain and codomain. By construction, this functor is again contextual and an equivalence of categories.

The monad axioms follow from the fact that, essentially, both unit and multiplication are "identities" on objects and morphisms.

Proposition 2.1.3. The above construction defines a monad on CxlCat.

For the construction of the Id-structure on $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ it is enough to translate the logical rules for Id-context we mentioned earlier. A sketch of the construction on $\mathsf{C}^{\mathsf{cxt}}$ for an arbitrary contextual category with an Id-structure C is available in [KS19, Constr. 2.17].

We are now ready, given a contextual category C, to extend the Π-structure to $\mathsf{C}^{\mathsf{cxt}}$. We shall do so in two parts, first in full generality and then under extra assumptions.

Given $\Gamma.\Delta.\Theta$ in $\mathsf{C}^{\mathsf{cxt}}$, where $l(\Gamma.\Delta.\Theta) = l(\Gamma.\Delta) + n = l(\Gamma) + m + n$ in C, we build $\Gamma.\Pi(\Delta,\Theta)$ by induction on the length of the context extensions involved, taking the one from C in case we are working with objects corresponding to simple extensions.

Construction 2.1.4 (Part 1)**.** If $n = 0$, then

$$\begin{aligned} \Gamma.\Pi(\Delta,\Theta) &= \Gamma, \\ \mathsf{app}_{\Delta,\Theta} &= 1_{\Gamma.\Delta}, \\ \lambda_{\Delta,\Theta}(b) &= 1_{\Gamma}. \end{aligned}$$

Notice that the only possible section b in the latter case is given by $1_{\Gamma.\Theta}$. This is not really necessary to specify the Π-structure, however we shall need it later in Lemma 4.2.1.

We now work with the case where $m > 0$, $n = 1$, thus we can write $\Gamma.\Delta.\Theta = \Gamma.\Delta.B$. In the base case, $m = 1$, we have $\Gamma.\Delta = \Gamma.A$ and therefore have to simply set our structure to coincide with the one in C.

If $n-1 > 0$, we have, for a section $b\colon \Gamma.\Delta \to \Gamma.\Delta.B$, $\Gamma.\Delta = \Gamma.\Delta'.A$ and then set

$$\begin{aligned}
&\Gamma.\,\mathsf{\Pi}(\Delta, B) = \Gamma.\,\mathsf{\Pi}(\Delta'.A, B) = \Gamma.\Pi(\Delta', \Pi(A,B)),\\
&\quad \mathsf{app}_{\Delta,B}\colon \Gamma.\,\mathsf{\Pi}(\Delta', \Pi(A,B)).\Delta'.A \xrightarrow{q(\mathsf{app}_{\Delta',\mathsf{\Pi}(A,B)}, p^*_{\Pi(A,B)}A)}\\
&\qquad\qquad\qquad\qquad\qquad \Gamma.\Delta'.\Pi(A,B).A \xrightarrow{\mathsf{app}_{A,B}} \Gamma.\Delta'.A.B,\\
&\quad \lambda_{\Delta,B}(b)\colon \Gamma \xrightarrow{\lambda_{\Delta',\mathsf{\Pi}(A,B)}(\lambda_{A,B}(b))} \Gamma.\,\mathsf{\Pi}(\Delta', \Pi(A,B)).
\end{aligned}$$

The idea here is to replicate the adjunction $\mathbf{Set}(A \times B, C) \cong \mathbf{Set}(A, \mathbf{Set}(B,C))$. The map $\mathsf{app}_{\Delta,B}$ is then naturally interpreted as a sequence of partial evaluations and the phenomenon is commonly known as *currying-uncurrying*.

We have dealt with the construction for $n = 1$ in C, hence we shall move on to the case where Δ has arbitrary length and induct on n.

We will work only with syntactic categories and in contextual categories where a section $c\colon \Gamma \to \Gamma.A.B$ can be split as two sections $a\colon \Gamma \to \Gamma.A$, $b\colon \Gamma.A \to \Gamma.A.B$. We see that $p_B \cdot c = p_B \cdot b \cdot a = a$. Syntactically, this means that if we pick two terms $a : A, b : B(a)$ over Γ at once, then we can also specify for each choice of a term $a' : A$ over Γ a term $b' : B(a')$ such that $b \equiv b'$ when $a \equiv a'$. We later point out the difficulties encountered without our assumption and exploit our previously presented notation for extended type-theoretic rules to reason about $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ as if we were working with $\mathsf{Syn}(\mathbf{T})$.

Construction 2.1.5 (Part 2)**.** Suppose that $l(\Gamma.\Delta.\Theta) = l(\Gamma.\Delta) + n$, $n > 1$, and we have already provided the relevant constructions up to $n-1$. We again decompose the context as $\Gamma.\Delta.\Theta'.B$.

If we are working with a contextual category C where sections split, we make use of our assumption on a section $y\colon \Gamma.\Delta \to \Gamma.\Delta.\Theta'.B$ to get a factorization as $b \cdot y'$ through $\Gamma.\Delta.\Theta'$.

$$\begin{aligned}
&\Gamma.\,\mathsf{\Pi}(\Delta, \Theta) = \Gamma.\,\mathsf{\Pi}(\Delta, \Theta'.B) = \Gamma.\Pi(\Delta, \mathsf{app}^*_{\Delta,\Theta'} B),\\
&\quad \mathsf{app}_{\Delta,\Theta}\colon \Gamma.\,\mathsf{\Pi}(\Delta,\Theta').\Pi(\Delta, \mathsf{app}^*_{\Delta,\Theta'} B).\Delta \xrightarrow{\mathsf{app}_{\Delta,\mathsf{app}^*_{\Delta,\Theta'} B}}\\
&\qquad\qquad\qquad\qquad \Gamma.\,\mathsf{\Pi}(\Delta,\Theta').\Delta.\,\mathsf{app}^*_{\Delta,\Theta'} B \xrightarrow{q(\mathsf{app}_{\Delta,\Theta'}, B)} \Gamma.\Delta.\Theta'.B,\\
&\quad \lambda_{\Delta,\Theta}(y)\colon \Gamma \xrightarrow{\lambda_{\Delta,\Theta'}(y')} \Gamma.\,\mathsf{\Pi}(\Delta,\Theta') \xrightarrow{\lambda_{p^*_{\mathsf{\Pi}(\Delta,\Theta')}\Delta, \mathsf{app}^*_{\Delta,\Theta'} B}(\mathsf{app}^*_{\Delta,\Theta'} b')} \Gamma.\,\mathsf{\Pi}(\Delta,\Theta').\Pi(\Delta, \mathsf{app}^*_{\Delta,\Theta'} B).
\end{aligned}$$

If we are instead in $\mathsf{Syn}(\mathbf{T})$, we can define $\lambda_{\Delta,\Theta}(y)$ in the following way: from $y'(x) : \Theta(x), b(x, y'(x)) : B(x, y'(x))$ in context $\Gamma, x : \Delta$ we can indeed simply take the section specified by $\lambda(x : \Delta).y(x) : \mathsf{\Pi}(\Delta, \Theta)$ that is the pair

$$\begin{gathered}
\lambda(x : \Delta).y'(x) : \mathsf{\Pi}(\Delta, \Theta'),\\
\lambda(x : \Delta).b'(x, y'(x)) : \mathsf{\Pi}(x : \Delta, B(\mathsf{app}(\lambda(x : \Delta).y'(x), x))),
\end{gathered}$$

where we have made explicit the dependencies of the type B.

Finally, given $a{:}\,\Gamma \to \Gamma.\Delta$, $f{:}\,\Gamma.\Delta \to \Gamma.\Delta.\Theta$, we construct $\mathsf{app}_{\Delta,\Theta}(f,x)$ for sections $f{:}\,\Gamma \to \Gamma.\,\Pi(\Delta,\Theta)$, $x{:}\,\Gamma \to \Gamma.\Delta$ as in the definition of Π-structures 1.3.3.

This fully specifies the data needed for a Π-structure on $\mathsf{C}^{\mathsf{cxt}}$, however we still have to check that it is indeed one, which we will do in the next section. First, however, we sketch the extension for Σ-structures by assuming to have a left inverse $\mathsf{unpair}_{A,B}$ of $\mathsf{pair}_{A,B}$ for any context $\Gamma.A.B$. Such a left inverse can be constructed in $\mathsf{Syn}(\mathbf{T})$ by looking at the projection maps described in [Uni13, Sec. 1.6].

Construction 2.1.6. Let C be a contextual category with a Σ-structure. Given a dependent context $p_\Delta{:}\,\Gamma.\Delta.\Theta \to \Gamma.\Delta$ with $l(\Gamma.\Delta.\Theta) = l(\Gamma.\Delta) + n = l(\Gamma) + m + n$ in C we proceed by induction on $m, n > 0$.

Let's fix $m = 1$. If $n = 1$, like for Π-structures we are forced to take the definition from C, so we can focus on $n > 1$. We have then $\Gamma.\Delta.\Theta = \Gamma.A.B.\Theta'$, so we take

$$\Gamma.\,\Sigma(A,\Theta) = \Gamma.\,\Sigma(A,B.\Theta') = \Gamma.\,\Sigma(A,\Sigma(B,\Theta')),$$
$$\mathsf{pair}_{A,\Theta}{:}\,\Gamma.A.B.\Theta' \xrightarrow{\mathsf{pair}_{B,\Theta'}} \Gamma.A.\,\Sigma(B,\Theta') \xrightarrow{\mathsf{pair}_{A,\Sigma(B,\Theta')}} \Gamma.\,\Sigma(A,\Sigma(B,\Theta')).$$

To deal with the case where $m > 1$ and therefore $\Gamma.\Delta.\Theta = \Gamma.\Delta'.A.\Theta$, we choose

$$\Gamma.\,\Sigma(\Delta,\Theta) = \Gamma.\,\Sigma(\Delta'.A,\Theta) = \Gamma.\,\Sigma(\Delta', A.\Theta),$$
$$\mathsf{pair}_{\Delta,\Theta} = \mathsf{pair}_{\Delta',A.\Theta}\,.$$

To complete the construction, one has to define for all maps

$$z{:}\,\Gamma.\Delta.\Theta \to \Gamma.\,\Sigma(\Delta,\Theta).\Psi$$

such that $p_\Psi \cdot z = \mathsf{pair}_{\Delta,\Theta}$ a section

$$\mathsf{split}_z{:}\,\Gamma.\,\Sigma(\Delta,\Theta) \to \Gamma.\,\Sigma(\Delta,\Theta).\Psi$$

satisfying $\mathsf{split}_z \cdot \mathsf{pair}_{\Delta,\Theta} = z$ and it is here that one uses the morphism $\mathsf{unpair}_{\Delta,\Theta}$.

2.2 Properties

In this section we verify that the desired properties hold for our constructions.

Lemma 2.2.1. Given a dependent type theory with Π-types $\mathbf{T}$, the data provided defines a Π-structure on $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ which is compatible with the natural contextual functor $\mathsf{Syn}(\mathbf{T}) \hookrightarrow \mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$.

Proof. We have to show that it is a Π-structure, which we will do inductively by verifying that at every step our proposed construction maintains the desired properties. The compatibility with the contextual functor will then follow directly from the way we defined the base case.

Let's consider an object $\Gamma.\Delta.\Theta$ in $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ such that $l(\Gamma.\Delta.\Theta) = l(\Gamma.\Delta) + n = l(\Gamma) + m + n$ in $\mathsf{Syn}(\mathbf{T})$. The only interesting case is the one where least one of m, n is > 1: indeed, the desired properties in the other cases are follow directly from the fact that they hold in $\mathsf{Syn}(\mathbf{T})$.

We start as before by working on $m > 1$, $n = 1$, so we write $\Gamma.\Delta.\Theta = \Gamma.\Delta.B = \Gamma.\Delta'.A.B$ and assume that the desired properties hold for shorter context extensions. Picking a section $b\colon \Gamma.\Delta \to \Gamma.\Delta.B$, we can then write

$$\begin{aligned}
p_B \cdot \mathsf{app}_{\Delta,B} &= p_B \cdot \mathsf{app}_{A,B} \cdot q(\mathsf{app}_{\Delta',\Pi(A,B)}, p^*_{\Pi(A,B)}A) \\
&= q(p_{\Pi(A,B)}, A) \cdot q(\mathsf{app}_{\Delta',\Pi(A,B)}, p^*_{\Pi(A,B)}A) \\
&= q(p_{\Pi(A,B)} \cdot \mathsf{app}_{\Delta',\Pi(A,B)}, A) \\
&= q(q(p_{\Pi(\Delta',\Pi(A,B))}, \Delta'), A) \\
&= q(p_{\Pi(\Delta',\Pi(A,B))}, \Delta'.A) \\
&= q(p_{\Pi(\Delta,B)}, \Delta) \\
p_B \cdot \mathsf{app}_{\Delta,B}(f,a) &= p_B \cdot \mathsf{app}_{\Pi(\Delta,B)} \cdot (f,a) \\
&= q(p_{\Pi(\Delta,B)}, \Delta) \cdot (f,a) \\
&= a \\
p_{\Pi(\Delta,B)} \cdot \lambda_{\Delta,B}(b) &= p_{\Pi(\Delta',\Pi(A,B))} \cdot \lambda_{\Delta',\Pi(A,B)}(\lambda_{A,B}(b)) \\
&= 1_\Gamma.
\end{aligned}$$

To justify $\mathsf{app}_{\Delta,B}(\lambda_{\Delta,B}(b), x) = b \cdot x$ we shall work type-theoretically. The section b is such that

$$x \equiv (x', a) \mapsto (x', a, b(x', a)) \equiv (x, b(x)),$$

thus applying $\lambda_{A,B}$ and then $\lambda_{\Delta',\Pi(A,B)}$ we get

$$\lambda(x' : \Delta').(\lambda(a : A).b(x', a)) \equiv \lambda(x : \Delta).b(x) : \Gamma.\,\Pi(\Delta, B).$$

Also, $\mathsf{app}_{\Delta,B}$ acts as

$$\begin{aligned}
&(f, x) \\
\overset{\text{split } x}{}\; &\equiv (f, x', a) \\
\overset{q(\mathsf{app}_{\Delta',\Pi(A,B)}, p^*_{\Pi(A,B)}A)}{}\; &\mapsto (x', \mathsf{app}(f, x'), a) \\
\overset{\mathsf{app}_{A,B}}{}\; &\mapsto (x', a, \mathsf{app}(\mathsf{app}(f, x'), a)) \\
&\equiv (y, \mathsf{app}(f, y)),
\end{aligned}$$

thus in this case we get

$$\begin{aligned}
&(\lambda(x : \Delta).b(x), x) \\
&\qquad \equiv (\lambda(x' : \Delta').(\lambda(a : A).b(x', a)), x', a) \\
&\qquad \mapsto (x', \lambda(a : A).b(x', a), a) \\
&\qquad \mapsto (x', a, b(x', a)) \\
&\qquad \equiv (x, b(x)),
\end{aligned}$$

where the inductive hypothesis has been used in the second step.

We now check inductively on the length of Θ.

$$\begin{aligned}
p_\Theta \cdot \mathsf{app}_{\Delta,\Theta} &= p_{\Theta'} \cdot p_B \cdot q(\mathsf{app}_{\Delta,\Theta'}, B) \cdot \mathsf{app}_{\Delta,\mathsf{app}^*_{\Delta,\Theta'} B} \\
&= p_{\Theta'} \cdot \mathsf{app}_{\Delta,\Theta'} \cdot p_{\mathsf{app}^*_{\Delta,\Theta'} B} \cdot \mathsf{app}_{\Delta,\mathsf{app}^*_{\Delta,\Theta'} B} \\
&= q(p_{\Pi(\Delta,\Theta)}, \Delta) \cdot q(p_{\Pi(\Delta,\mathsf{app}^*_{\Delta,\Theta'} B)}, \Delta) \\
&= q(p_{\Pi(\Delta,\Theta')} \cdot p_{\Pi(\Delta,\mathsf{app}^*_{\Delta,\Theta'} B)}, \Delta) \\
&= q(p_{\Pi(\Delta,\Theta').\Pi(\Delta,\mathsf{app}^*_{\Delta,\Theta'} B)}, \Delta) \\
&= q(p_{\Pi(\Delta,\Theta)}, \Delta) \\
p_\Theta \cdot \mathsf{app}_{\Delta,\Theta}(f,x) &= p_\Theta \cdot \mathsf{app}_{\Delta,\Theta} \cdot (f,x) \\
&= q(p_{\Pi(\Delta,\Theta)}, \Delta) \cdot (f,x) \\
&= x \\
p_{\Pi(\Delta,\Theta)} \cdot \lambda_{\Delta,\Theta}(y) &= p_{\Pi(\Delta,\Theta')} \cdot p_{\Pi(p^*_{\Pi(\Delta,\Theta')}\Delta,\mathsf{app}^*_{\Delta,\Theta'} B')} \cdot \lambda_{p^*_{\Pi(\Delta,\Theta')}\Delta,\mathsf{app}^*_{\Delta,\Theta'} B}(\mathsf{app}^*_{\Delta,\Theta'} b) \cdot \lambda_{\Delta,\Theta'}(y) \\
&= p_{\Pi(\Delta,\Theta')} \cdot \lambda_{\Delta,\Theta'}(y) \\
&= 1_\Gamma.
\end{aligned}$$

Again, to justify $\mathsf{app}_{\Delta,\Theta}(\lambda_{\Delta,\Theta}(y),x) = y \cdot x$ we use type-theoretic reasoning. Splitting $y : \Theta(x)$ in context $\Gamma, x : \Delta$ as $(y'(x), b(x,y'))$, we know that $y{:}\,\Gamma.\Delta \to \Gamma.\Delta.\Theta$ acts as

$$x \mapsto (x, y(x)) \equiv (x, y(x), b(x, y'(x))),$$

thus our construction provides us with

$$\lambda(x:\Delta).y(x) \equiv (\lambda(x:\Delta).y'(x), \lambda(x:\Delta).b(x,y'(x))).$$

Also, $\mathsf{app}_{\Delta,\Theta}$ acts internally as

$$\begin{aligned}
&(f,x) \\
{\scriptstyle \text{split } f}\ &\equiv (f', f'', x) \\
{\scriptstyle \mathsf{app}_{\Delta,\mathsf{app}^*_{\Delta,\Theta'} B}}\ &\mapsto (f', x, \mathsf{app}(f'', x)) \\
{\scriptstyle q(\mathsf{app}_{\Delta,\Theta'},B)}\ &\mapsto (x, \mathsf{app}(f', x), \mathsf{app}(f'', x)) \\
&\equiv (x, \mathsf{app}(f,x)),
\end{aligned}$$

which here translates to

$$\begin{aligned}
&(\lambda(x:\Delta).y(x), x) \\
&\qquad \equiv (\lambda(x:\Delta).y'(x), \lambda(x:\Delta).b(x,y'(x)), x) \\
&\qquad \mapsto (\lambda(x:\Delta).y'(x), x, b(x,y'(x))) \\
&\qquad \mapsto (x, y'(x), b(x,y'(x))) \\
&\qquad \equiv (x, y(x)),
\end{aligned}$$

where the inductive hypothesis has been used in the second step.

To conclude the proof one would still need to verify that the construction is compatible with context substitution. $\square$

Remark 2.2.2. The above theorem can be generalized to contextual categories with a $\mathsf{\Pi}$-structure C as long as sections of dependent projections can be factored as described, in which case this hypothesis is used when we split dependent terms. The proof is essentially the same, albeit harder to read.

We shall also need the following lemma.

Lemma 2.2.3 ([Kap17, Lem. 5.6])**.** Given a dependent type theory with $\mathsf{\Pi}$-types $\mathbf{T}$ and an iterated context extension $\Gamma.\Delta.\Theta.\Psi$ in $\mathsf{Syn}(\mathbf{T})$, the contexts

$$\Gamma.\,\mathsf{\Pi}(\Delta,\Theta.\Psi), \quad \Gamma.\,\mathsf{\Pi}(\Delta,\Theta).\,\mathsf{\Pi}(p^*_{\mathsf{\Pi}(\Delta,\Theta)}\Delta, \mathsf{app}^*_{\Delta,\Theta}\,\Psi)$$

are equal in $\mathsf{Syn}(\mathbf{T})$. Also, $\Gamma.\,\mathsf{\Pi}(\Delta,p_\Psi) = p_{\mathsf{\Pi}(\Delta,\mathsf{app}^*_{\Delta,\Theta}\,\Psi)}$.

Proof. For the first claim, it is enough to notice that the two contexts reduce to the same one in $\mathsf{Syn}(\mathbf{T})$ after applying the inductive construction we defined on $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ to reduce Ψ.

For the second claim instead we consider the chain of equalities

$$\begin{aligned}\Gamma.\,\mathsf{\Pi}(\Delta,p_\Psi) &= q(p_{\mathsf{\Pi}(\Delta,\Theta.\Psi)},\mathsf{\Pi}(\Delta,\Theta))\cdot\lambda_{\Delta,\Theta}(1_{p^*_{\mathsf{\Pi}(\Delta,\Theta)}\Delta}, p_\Psi\cdot\mathsf{app}_{\Delta,\Theta.\Psi})\\ &= p_{\mathsf{\Pi}(\Delta,\mathsf{app}^*_{\Delta,\Theta}\,\Psi)}\cdot p_{\mathsf{\Pi}(\Delta,\Theta)}\cdot\lambda_{\Delta,\Theta}(1_{p^*_{\mathsf{\Pi}(\Delta,\Theta)}\Delta}, p_\Psi\cdot\mathsf{app}_{\Delta,\Theta.\Psi})\\ &= p_{\mathsf{\Pi}(\Delta,\mathsf{app}^*_{\Delta,\Theta}\,\Psi)},\end{aligned}$$

which can be checked type-theoretically. □

We conclude by presenting the following propositions without proof. It is worth noting that the original statements mention contextual categories, but to do so they rely on the Initiality Conjecture 1.2 and here our focus is on syntactic categories.

Lemma 2.2.4 ([Gar09, Prop. 3.3.1])**.** Given a dependent type theory with Id-types $\mathbf{T}$, the contextual category $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ carries a natural Id-structure compatible with the natural contextual functor $\mathsf{Syn}(\mathbf{T}) \hookrightarrow \mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$.

Lemma 2.2.5 ([KS19, Lemma 2.28])**.** Given a dependent type theory with Id-, $\mathsf{\Pi}_\eta$-types structures and function extensionality $\mathbf{T}$, the latter can also be lifted from $\mathsf{Syn}(\mathbf{T})$ to $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ compatibly with the natural contextual functor $\mathsf{Syn}(\mathbf{T}) \hookrightarrow \mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$.

Remark 2.2.6. Lumsdaine notes in [Lum10] that such extended structures are not all compatible with the

3 Localizations of ∞-Categories

To prove that by localizing a syntactic category of a dependent type theory with some logical rules we get a locally cartesian closed ∞-category we need a theory of localizations of ∞-categories, which in our case will be quasi-categories. We shall provide such a theory as developed by Cisinski in [Cis19, Ch. 7] with the aim of proving Theorem 3.3.8, which will do the heavy lifting in showing Theorem 4.2.4. Those familiar with the theory may skip the entire chapter keeping in mind the aforementioned theorem, while those who do not know it may read it for a quick tour. We recommend however to consult the original source, which provides all of the intermediate results and details we gloss over in our sketched proofs.
Our exposition is divided in three parts: first we define localizations and show some results derived directly from it, while later we present some facts which allow us to give a better description of the localization when the starting ∞-category has a fibrational structure and finally we introduce some conditions under which localizing induces an equivalence of ∞-categories, from which we derive Theorem 3.3.8.

Notation 3.0.1. We denote generic simplicial sets by $A, B \ldots$, while for ∞-categories we use a calligraphic font.

3.1 Universal Property

Localizations are defined in any context by a universal property of a certain form. Here we present ours and see what we can derive from it without extra assumptions.

Definition 3.1.1. Let C be a simplicial set and $W \subset C$ a simplicial subset. Given an ∞-category $\mathcal{D}$, we define $\underline{\mathrm{Hom}}_W(C, \mathcal{D})$ to be the full simplicial subset of $\underline{\mathrm{Hom}}(C, \mathcal{D})$ whose objects are the morphisms $f: C \to \mathcal{D}$ sending the 1-simplices in W to isomorphisms.

Remark 3.1.2. The above definition induces a canonical pullback square

$$\begin{tikzcd} \underline{\mathrm{Hom}}_W(C, \mathcal{D}) \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \underline{\mathrm{Hom}}(C, \mathcal{D}) \arrow[d] \\ \underline{\mathrm{Hom}}_W(W, \mathcal{D}) \arrow[r] & \underline{\mathrm{Hom}}(W, \mathcal{D}) \end{tikzcd}$$

given by the inclusion $W \to C$.

Definition 3.1.3. Given a simplicial set C and a simplicial subset W, a *localization of C by W* is a morphism $\gamma: C \to L(C)$ in **sSet** such that:

1. $L(C)$ is an ∞-category;
2. γ sends the 1-simplices of W to isomorphisms in $L(C)$;
3. for any ∞-category $\mathcal{D}$ there is an equivalence of ∞-categories
$$\underline{\mathrm{Hom}}(L(C), \mathcal{D}) \to \underline{\mathrm{Hom}}_W(C, \mathcal{D})$$
given by precomposing with γ.

We will often refer to $L(C)$ as the localization of C by W, without providing γ explicitly.

Proposition 3.1.4. The localization of C by W always exists and is essentially unique.

Proof. We begin by proving that a localization exists in the case where $W = C$.
In this context, $\underline{\mathrm{Hom}}_W(C, \mathcal{D}) \cong \underline{\mathrm{Hom}}(C, \mathcal{D}^{\cong})$ canonically, where $\mathcal{D}^{\cong}$ is the maximal subgroupoid of $\mathcal{D}$. Factoring $C \to \Delta^0$ in the Kan model structure, we find an anodyne map $C \to C'$. We then choose this map for γ and set $L(C) = C'$.
Remember that for any anodyne map $A \to B$ we get a trivial fibration $\underline{\mathrm{Hom}}(B, \mathcal{D}^{\cong}) \to \underline{\mathrm{Hom}}(A, \mathcal{D}^{\cong})$. Looking then at the commutative diagram

$$\begin{tikzcd}
\mathrm{Hom}(L(C), \mathcal{D}^{\cong}) \arrow[r, two heads, "\gamma^*", "\sim"'] \arrow[d, "\cong"'] & \mathrm{Hom}_W(C, \mathcal{D}^{\cong}) \arrow[d, "\cong"] \\
\underline{\mathrm{Hom}}(L(C), \mathcal{D}) \arrow[r, "\gamma^*"'] & \underline{\mathrm{Hom}}_W(C, \mathcal{D})
\end{tikzcd},$$

we see that the lower γ^* is a trivial fibration, thus we have constructed a valid localization of C by W.

We now move on to the general case. First of all, notice that as a particular case of the previous one we get that localizing Δ^1 at its non-trivial morphism our construction provides $L(\Delta^1) = J \sim \Delta^0$, while γ is the inclusion $\Delta^1 \to J$. Taking then $W \subset C$, we consider the commutative diagram

$$\begin{tikzcd}
\coprod_{f \in W_1} \Delta^1 \arrow[r] \arrow[d] & C \arrow[d] \arrow[ddr, bend left, "\gamma"] & \\
\coprod_{f \in W_1} J \arrow[r] & C' \arrow[ul, phantom, "\ulcorner", very near start] \arrow[dr, "\sim"'] & \\
& & L(C)
\end{tikzcd},$$

where $C' \xrightarrow{\sim} L(C)$ is an inner anodyne map obtained by taking the fibrant replacement of C' in the Joyal model structure. This can be done functorially via the small object argument.
For any ∞-category $\mathcal{D}$, we get a trivial fibration $\underline{\mathrm{Hom}}(L(C), \mathcal{D}) \xrightarrow{\sim} \underline{\mathrm{Hom}}(C', \mathcal{D})$ and a pullback square

$$\begin{tikzcd}
\underline{\mathrm{Hom}}(C', \mathcal{D}) \arrow[r] \arrow[d] & \underline{\mathrm{Hom}}(C, \mathcal{D}) \arrow[d] \\
\prod_{f \in W_1} \underline{\mathrm{Hom}}(J, \mathcal{D}) \arrow[r] & \prod_{f \in W_1} \underline{\mathrm{Hom}}(\Delta^1, \mathcal{D})
\end{tikzcd},$$

which together with the pullback

$$\begin{tikzcd}
\underline{\mathrm{Hom}}_W(C,\mathcal{D}) \arrow[r] \arrow[d] & \underline{\mathrm{Hom}}(C,\mathcal{D}) \arrow[d] \\
\Pi_{f\in W_1}\underline{\mathrm{Hom}}(\Delta^1,\mathcal{D}^{\cong}) \arrow[r] & \Pi_{f\in W_1}\underline{\mathrm{Hom}}(\Delta^1,\mathcal{D})
\end{tikzcd}$$

implies by pasting that

$$\begin{tikzcd}
\underline{\mathrm{Hom}}(C',\mathcal{D}) \arrow[r] \arrow[d] & \underline{\mathrm{Hom}}_W(C,\mathcal{D}) \arrow[d] \\
\Pi_{f\in W_1}\underline{\mathrm{Hom}}(J,\mathcal{D}) \arrow[r, two heads, "\sim"] & \Pi_{f\in W_1}\underline{\mathrm{Hom}}(\Delta^1,\mathcal{D}^{\cong})
\end{tikzcd}$$

is also a pullback and therefore the upper arrow is a trivial fibration. Composing it with the other one we get $\gamma^*\colon \underline{\mathrm{Hom}}(L(C),\mathcal{D}) \to \underline{\mathrm{Hom}}_W(C,\mathcal{D})$, which is then a trivial fibration and therefore an equivalence of ∞-categories.

We now move on to proving that the localization is essentially unique. For this, we notice that γ^* establishes then an isomorphism between $\pi_0(k(\underline{\mathrm{Hom}}_W(C,-)))$ and $\pi_0(\underline{\mathrm{Hom}}(L(C),-)) = ho(\mathbf{sSet})(L(C),-)$ with respect to the Joyal model structure, thus by Yoneda the pair $(L(\mathcal{C}),\gamma)$ satisfying the universal property is unique up to unique isomorphism in $ho(\mathbf{sSet})$ and up to a contractible space of equivalences in $\mathbf{sSet}$. $\square$

Remark 3.1.5. Through our construction above, one can always choose $L(C)$ so that γ is a bijection on objects because $C' \to L(C)$ is an inner anodyne extension and therefore a retract of a countable composition of sums of pushouts of maps which are the identity on objects, that is the inner horn inclusions.

Remark 3.1.6. Writing **weCat** for the category of *relative categories* [BK12], there are various ways to construct a localization functor $\mathbf{weCat} \to \mathbf{Cat}_\infty$ in the literature and Kapulkin refers to simplicial localization when stating Theorem 4.2.4 in [Kap14], however it was shown that they are all weakly equivalent (see [Bar16, Ex. 1.6.3]), meaning that they are linked by natural transformations whose arrows are equivalences of ∞-categories. In particular, they correspond homotopically to a localization functor defined as $(\mathcal{C},W) \mapsto L(N(\mathcal{C}))$ on objects, where we are localizing $N(\mathcal{C})$ at the 1-simplices specified by W.

Remark 3.1.7. Given an ∞-category $\mathcal{C}$ and a simplicial subset W, the functor $ho(\mathcal{C}) \to ho(L(\mathcal{C}))$ exhibits $ho(L(\mathcal{C}))$ as the 1-categorical localization of $\mathcal{C}$ at $\mathrm{Arr}(\tau(W))$, as can be seen by using the universal property.
On the other hand, given a 1-category $\mathcal{C}$ and considered a set of morphisms W, not necessarily the induced map $L(N(\mathcal{C})) \to N(L(\mathcal{C}))$ is an isomorphism. Indeed, $L(N(\mathcal{C}))$ can have much better properties, as can be seen for example from Proposition 3.2.13, and in fact localizing nerves of 1-categories gives every ∞-category as shown in [Cis19, Prop. 7.3.15]. It should be noted that here we used $L(-)$ to refer both to the 1-categorical and the ∞-categorical localization.

Remark 3.1.8. In this context, we may define $\overline{W}$, the saturation of W in C, as the maximal simplicial subset of C whose morphisms are the ones which become invertible in $L(C)$, thus we have $\overline{W} \cong k(L(C)) \times_{L(C)} C$ canonically, meaning that we can draw the following pullback square.

$$\begin{tikzcd} \overline{W} \arrow[r, hook] \arrow[d] & C \arrow[d, "\gamma"] \\ k(L(C)) \arrow[r, hook] & L(C) \end{tikzcd}$$

We have inclusions $Sk_1(W) \subset W \subset \overline{W}$ and, for any ∞-category $\mathcal{D}$, this induces equalities

$$\underline{\mathrm{Hom}}_{Sk_1(W)}(C, \mathcal{D}) = \underline{\mathrm{Hom}}_W(C, \mathcal{D}) = \underline{\mathrm{Hom}}_{\overline{W}}(C, \mathcal{D}),$$

implying that $(L(C), \gamma)$ is also the localization of C by $Sk_1(W)$ and the one by $\overline{W}$. It can be noted that the inclusion $\overline{W} \to C$ is a fibration with respect to the Joyal model structure as it is the pullback of one, implying that if C is an is an ∞-category then so is $\overline{W}$.
We shall say that W is *saturated* if $W = \overline{W}$.

Remark 3.1.9. Given a universe $\mathbf{U}$ and a simplicial subset W of a $\mathbf{U}$-small simplicial set C, let $\gamma: C \to L(C)$ be the associated localization. Then the functor

$$\gamma^*: \underline{\mathrm{Hom}}(L(C)^{\mathrm{op}}, \mathcal{S}) \to \underline{\mathrm{Hom}}(C^{\mathrm{op}}, \mathcal{S})$$

is fully faithful because it is obtained as

$$\underline{\mathrm{Hom}}(L(C)^{\mathrm{op}}, \mathcal{S}) \simeq \underline{\mathrm{Hom}}_{W^{\mathrm{op}}}(C^{\mathrm{op}}, \mathcal{S}) \to \underline{\mathrm{Hom}}(C^{\mathrm{op}}, \mathcal{S}).$$

It also has a left adjoint $\gamma_!$ and a right adjoint γ_* as shown in [Cis19, Ch. 6]. Full faithfulness then implies that for any presheaf $F: L(C)^{\mathrm{op}} \to \mathcal{S}$ the unit map $F \to \gamma_*\gamma^*(F)$ is invertible and, by adjunction, the same goes for the counit map $\gamma_!\gamma^*(F) \to F$.
Furthermore, its essential image consists exactly of those presheafs $F: C^{\mathrm{op}} \to \mathcal{S}$ such that, for any morphism $u: x \to y$ in W, the map $Fu: Fy \to Fx$ is invertible in $\mathcal{S}$. Indeed, restricting $\gamma_!$ and γ_* to $\underline{\mathrm{Hom}}_{W^{\mathrm{op}}}(C^{\mathrm{op}}, \mathcal{S})$, we see that they are left and right adjoint to the equivalence of ∞-categories induced by γ^*, meaning that the counit map $\gamma^*\gamma_*(F) \to F$ and the unit map $F \to \gamma^*\gamma_!(F)$ are invertible.

Proposition 3.1.10. Given a simplicial set C and a simplicial subset W, the localization functor $\gamma: C \to L(C)$ is final and cofinal. In particular, if $e: \Delta^0 \to C$ encodes a final or a cofinal object, so does $\gamma(e)$.

Proof. First of all, the functor γ^{op} is also a localization, so it suffices to prove that γ is final. To do this, first we fix a universe $\mathbf{U}$ such that C is $\mathbf{U}$-small and then we remember that there is an adjunction $\gamma^*: \underline{\mathrm{Hom}}(L(C)^{\mathrm{op}}, \mathcal{S}) \rightleftarrows \underline{\mathrm{Hom}}(C^{\mathrm{op}}, \mathcal{S}) : \gamma_*$ and, by the above remark, the unit of the adjunction is invertible, hence $F \simeq \gamma_*\gamma^*(F)$ for any presheaf $F: L(\mathcal{C})^{\mathrm{op}} \to \mathcal{S}$. This gives us

$$\lim_{C^{\mathrm{op}}} F \simeq \lim_{C^{\mathrm{op}}} \gamma_*\gamma^*(F) \simeq \lim_{L(C)^{\mathrm{op}}} \gamma^*(F),$$

which is enough to prove that γ is final by [Cis19, Thm. 6.4.5]. $\square$

Proposition 3.1.11. Let's fix a universe $\mathbf{U}$, a $\mathbf{U}$-small simplicial set C and a simplicial subset W. Consider a morphism $f: C \to \mathcal{D}$, where $\mathcal{D}$ is a small ∞-category. Then f exhibits $\mathcal{D}$ as the localization of C by W if and only if the following conditions hold:

1. the morphism f sends the 1-simplices of W to invertible maps of $\mathcal{D}$;

2. the morphism f is essentially surjective;

3. the morphism f^* induces an equivalence of ∞-categories

$$f^*: \underline{\mathrm{Hom}}(\mathcal{D}^{\mathrm{op}}, \mathcal{S}) \to \underline{\mathrm{Hom}}_{W^{\mathrm{op}}}(C^{\mathrm{op}}, \mathcal{S}).$$

Proof. If f is a localization then the conditions are satisfied (for (2) look at the construction in Prop. 3.1.4). For the converse, let's pick a localization $\gamma: C \to L(C)$ and, through condition (1), we get a factorization $g: L(C) \to \mathcal{D}$ such that $g \cdot \gamma \simeq f$, giving us a triangle

$$\begin{tikzcd} \underline{\mathrm{Hom}}(\mathcal{D}^{\mathrm{op}}, \mathcal{E}) \arrow[rr, "g^*"] \arrow[dr, "f^*"'] & & \underline{\mathrm{Hom}}(L(C)^{\mathrm{op}}, \mathcal{E}) \arrow[dl, "\gamma^*"] \\ & \underline{\mathrm{Hom}}_{W^{\mathrm{op}}}(C^{\mathrm{op}}, \mathcal{E}) & \end{tikzcd}$$

commuting up to J-homotopy for any ∞-category $\mathcal{E}$. Picking $\mathcal{E} = \mathcal{S}$, γ^* and f^* are equivalences of ∞-categories, the latter by (3). It follows by 2-out-of-3 that g^* is one too, therefore the same applies to its left adjoint $g_!$, which is then fully faithful. This is equivalent to g being fully faithful by [Cis19, Prop. 6.1.15] and, since f is essentially surjective by (2), the same goes for g. It follows that g is an equivalence of ∞-categories. In the above triangle g^* is then an equivalence for any choice of $\mathcal{E}$ and the same applies to f^* by 2-out-of-3. We conclude by using $(-)^{\mathrm{op}}$. □

Proposition 3.1.12. Let $f: \mathcal{C} \to \mathcal{D}$ be a functor between ∞-categories with a right adjoint $g: \mathcal{D} \to \mathcal{C}$ and suppose that we are given simplicial subsets $V \subset \mathcal{C}$, $W \subset \mathcal{D}$ such that $f(V) \subset W$, $g(W) \subset V$. We can lift them to an adjunction $\overline{f}: L(\mathcal{C}) \rightleftarrows L(\mathcal{D}) : \overline{g}$ such that the diagrams

$$\begin{tikzcd} \mathcal{C} \arrow[r, "f"] \arrow[d, "\gamma_{\mathcal{C}}"'] & \mathcal{D} \arrow[d, "\gamma_{\mathcal{D}}"] \\ L(\mathcal{C}) \arrow[r, "\overline{f}"'] & L(\mathcal{D}) \end{tikzcd}, \qquad \begin{tikzcd} \mathcal{D} \arrow[r, "g"] \arrow[d, "\gamma_{\mathcal{D}}"'] & \mathcal{C} \arrow[d, "\gamma_{\mathcal{C}}"] \\ L(\mathcal{D}) \arrow[r, "\overline{g}"'] & L(\mathcal{C}) \end{tikzcd}$$

commute up to J-homotopy.

Proof. Let's write $\underline{\mathrm{Hom}}_V^W(\mathcal{C}, \mathcal{D})$ for the full subcategory of $\underline{\mathrm{Hom}}(\mathcal{C}, \mathcal{D})$ whose objects are functors ϕ such that $\phi(V) \subset W$. The equivalence $\gamma_{\mathcal{C}}^*: \mathrm{Hom}(L(\mathcal{C}), L(\mathcal{D})) \to \underline{\mathrm{Hom}}_V(\mathcal{C}, L(\mathcal{D}))$ allows us to construct a functor $\underline{\mathrm{Hom}}_V^W(\mathcal{C}, \mathcal{D}) \to \underline{\mathrm{Hom}}_V(\mathcal{C}, L(\mathcal{D})) \to \underline{\mathrm{Hom}}(L(\mathcal{C}), L(\mathcal{D}))$ which associates to any ϕ as above a functor $\overline{\phi}$ making the square

$$\begin{tikzcd} \mathcal{D} \arrow[r, "\phi"] \arrow[d, "\gamma_{\mathcal{D}}"'] & \mathcal{C} \arrow[d, "\gamma_{\mathcal{C}}"] \\ L(\mathcal{D}) \arrow[r, "\overline{\phi}"'] & L(\mathcal{C}) \end{tikzcd}$$

commute up to J-homotopy.
The proof works by observing that our map also lifts natural transformations functorially, which allows us to show the triangle identities for the lifted unit and counit. $\square$

Proposition 3.1.13. Let $u\colon \mathcal{C} \to \mathcal{D}$ be a functor between ∞-categories with a fully faithful right adjoint v and consider $W \subset \mathcal{C}$, the subcategory of maps of $\mathcal{C}$ which become invertible in $\mathcal{D}$. Then u exhibits $\mathcal{D}$ as the localization of $\mathcal{C}$ by W.

Proof. Given a localization $\gamma\colon \mathcal{C} \to L(\mathcal{C})$ by W, we get a functor $\gamma \cdot v\colon \mathcal{D} \to L(\mathcal{C})$ which, paired with the $\overline{u}$ obtained from the construction in the previous proof, lifts the adjunction $u \dashv v$ to the localizations (where $L(\mathcal{D}) = \mathcal{D}$ as we localize at the identities). The functoriality of the lifting guarantees that the counit is still invertible, which allows us to conclude that $\gamma \cdot v$ is fully faithful.
Essential surjectivity follows from the fact that, for any object c in $\mathcal{C}$, the unit η_c is such that $\epsilon_{u(c)} \cdot u(\eta_c) = 1_{u(c)}$ and, since ϵ is invertible, so is $u(\eta_c)$, thus $\gamma(\eta_c)$ is invertible in $L(\mathcal{C})$ and shows that $(\gamma_{\mathcal{C}} \cdot v)(u(c)) = \gamma_{\mathcal{C}}(vu(c)) \simeq \gamma_{\mathcal{C}}(c) = c$. Here we used that $L(\mathcal{C})_0 = \mathcal{C}_0$, which is permissible up to equivalence as noted in Remark 3.1.5. $\square$

3.2 Fibrational Structures

As anticipated, here we study the localizations of ∞-categories $\mathcal{C}$ with a fibrational structure, which allows for a more explicit description of $L(\mathcal{C})$ through a *right calculus of fractions*. However, we will not be providing all of the results necessary for this and refer to [Cis19, Ch. 7.2-7.4].
This generalizes previous work by Szumilo and Kapulkin [Szu14; KS15], who proved similar results for the simplicial localization of a *fibration category* $\mathcal{P}$ (see Definition 4.1.7) by studying the associated ∞*-category of frames*, which provides a concrete model of $L(N(\mathcal{P}))$ avoiding inexplicit fibrant replacements.

Definition 3.2.1. An ∞*-category with weak equivalences and fibrations* $\mathcal{C}$ is a triple $(\mathcal{C}, W, \mathrm{Fib})$ where $\mathcal{C}$ is an ∞-category with a final object, $W \subset \mathcal{C}$ is a subcategory with the 2-out-of-3 property and $\mathrm{Fib} \subset \mathcal{C}$ a subsimplicial set such that:

1. for any morphism $p\colon x \to y$ in Fib (and W) with y, y' fibrant, there is in $\mathcal{C}$ a pullback square
$$\begin{tikzcd} x' \arrow[r, "u"] \arrow[d, "p'"'] & x \arrow[d, "p"] \\ y' \arrow[r, "v"'] & y \end{tikzcd}$$
where p' also lies in Fib (and W);

2. for any map $f\colon x \to y$ with fibrant codomain can be factored as a map in W followed by one in Fib.

By *fibrant object* we mean an object whose map to the terminal one is in Fib. If every object in $\mathcal{C}$ is fibrant then we call it an ∞*-category of fibrant objects*.

We shall call *weak equivalences* the maps in W and *fibrations* the ones in Fib. Maps which are both shall be referred to as *trivial fibrations*.

Construction 3.2.2. Any finitely complete ∞-category $\mathcal{C}$ can be given the structure of an ∞-category with weak equivalences and fibrations by setting $W = k(\mathcal{C})$, $\mathrm{Fib} = \mathcal{C}$, which we will be doing henceforth.

Construction 3.2.3. For any ∞-category with weak equivalences and fibrations $\mathcal{C}$ and a fibrant object c, we can give to the slice category $\mathcal{C}/c$ the structure of an ∞-category with weak equivalences and fibrations by specifying as weak equivalences the morphisms which are mapped to weak equivalences of $\mathcal{C}$ by the projection $\mathcal{C}/c \to \mathcal{C}$ and similarly for the fibrations.

Construction 3.2.4. For any ∞-category with weak equivalences and fibrations $\mathcal{C}$, its full subcategory given by fibrant objects is canonically an ∞-category of fibrant objects. We shall denote it by $\mathcal{C}_f$ and its weak equivalences are given by $W_f = W \cap \mathcal{C}_f$, its fibrations by $\mathrm{Fib}_f = \mathrm{Fib} \cap \mathcal{C}_f$. Furthermore, we write γ_f for the localization functor $\mathcal{C}_f \to L(\mathcal{C}_f)$.

Lemma 3.2.5 (Brown's Lemma)**.** For any map $f\colon x \to y$ between fibrant objects in an ∞-category with weak equivalences and fibrations $\mathcal{C}$, there exists a commutative diagram of the form

$$\begin{tikzcd}
 & x \arrow[dl, equal] \arrow[d, "s", "\sim"'] \arrow[dr, "f"] & \\
x & z \arrow[l, two heads, "\sim", "p"'] \arrow[r, two heads, "q"'] & y
\end{tikzcd},$$

where s is a weak equivalence, p a trivial fibration and q a fibration.

Proof. Since x and y are fibrant, the pullback of $x \to e$ and $y \to e$ exists and it corresponds to $x \times y$. The maps 1_x, f define a cone over our cospan inducing a map $g\colon x \to x \times y$, which we factor the latter as a weak equivalence $s\colon x \to z$ followed by a fibration $\pi\colon z \to x \times y$. We get then the desired maps $p = p_x \cdot \pi$, $q = p_y \cdot \pi$, where p_x, p_y denote the projections $x \times y \to x$, $x \times y \to y$ respectively. ☐

Corollary 3.2.6. Let $\mathcal{C}$ be an ∞-category with weak equivalences and fibrations, $\mathcal{D}$ an ∞-category and $V \subset \mathcal{D}$ a subcategory with the 2-out-of-3 property. If F sends trivial fibrations between fibrant objects into V, then it also sends weak equivalences between fibrant objects into V.

Proof. Looking at the commutative diagram

$$\begin{tikzcd}
x \arrow[dd, equal] \arrow[dr, "\sim", "s"'] \arrow[drr, bend left, "f", "\sim"'] & & \\
 & z \arrow[dl, two heads, "p"', "\sim"] \arrow[r, two heads, "q"] & y \\
x \arrow[urr, bend right, "f"', "\sim"] & &
\end{tikzcd},$$

given by Brown's Lemma 3.2.5, we see that Fp lies in V and therefore the same goes for Fs. Also, since f and s are weak equivalences we know that q is too, hence the latter is a trivial fibration. It follows that Fq is in V and the same goes for $Ff = Fq \cdot Fs$. ☐

Construction 3.2.7. Given an ∞-category with weak equivalences and fibrations $\mathcal{C}$ and a fibrant object z in it, we write $\mathcal{C}(z)$ for $(\mathcal{C}/z)_f$, that is the full subcategory $\mathcal{C}/z$ given by the fibrations $x \to z$ of $\mathcal{C}$; we shall refer to $\mathcal{C}(z)$ as the *fibrant slice of* $\mathcal{C}$ *over* z. For any morphism $f: x \to y$ between fibrant objects, we have a left exact functor

$$f^*: \mathcal{C}(y) \to \mathcal{C}(x), \quad (z \to y) \mapsto (z \times_y x \to x)$$

induced by pulling back along f. The existence follows from the fact that pullbacks along fibrations with fibrant codomain exist, while left exactness comes from limits commuting and weak equivalences being preserved as a consequence of Corollary 3.2.6.

Definition 3.2.8. Let $\mathcal{C}$, $\mathcal{D}$ be ∞-categories with weak equivalences and fibrations. A functor $F: \mathcal{C} \to \mathcal{D}$ is *left exact* if it has the following properties:

1. the functor F preserves final objects;
2. the functor F sends (trivial) fibrations between fibrant objects to (trivial) fibrations;
3. the functor F preserves any pullback square in $\mathcal{C}$

$$\begin{tikzcd} x' \arrow[r, "u"] \arrow[d, "p'"'] & x \arrow[d, "p"] \\ y' \arrow[r, "v"'] & y \end{tikzcd}$$

   where p is a fibration and y, y' are fibrant objects.

Remark 3.2.9. By Brown's Lemma, a left exact functor preserves weak equivalences between fibrant objects.

Remark 3.2.10. When considering a functor F between finitely complete ∞-categories, left exactness is equivalent to preserving finite limits.

Proposition 3.2.11. Given an ∞-category with weak equivalences and fibrations $\mathcal{C}$, the localization $L(\mathcal{C}_f)$ has finite limits and the localization functor $\mathcal{C}_f \to L(\mathcal{C}_f)$ is left exact. Moreover, for any ∞-category $\mathcal{D}$ with finite limits and any left exact functor $f: \mathcal{C}_f \to \mathcal{D}$, the induced functor $\overline{F}: L(\mathcal{C}_f) \to \mathcal{D}$ is left exact.

Proof. We know by 3.1.10 that $L(\mathcal{C}_f)$ has a final object, hence to show completeness it is enough to prove that it also has pullbacks. To prove this, Cisinski develops the theory of the aforementioned right calculus of fractions, through which he shows that any morphism in $L(\mathcal{C}_f)$ can be seen as a composition $\gamma_f(p) \cdot \gamma_f(s)^{-1}$, where s is a trivial fibration. The proof also shows that all pullback squares in $L(\mathcal{C}_f)$ are isomorphic to images of pullback squares in $\mathcal{C}_f$ in which all maps are fibrations. $\square$

Proposition 3.2.12. Let $\mathbf{U}$ be a universe and $\mathcal{C}$ a $\mathbf{U}$-small ∞-category with weak equivalences and fibrations. For any ∞-category $\mathcal{D}$ with $\mathbf{U}$-small colimits and any functor $F: \mathcal{C} \to \mathcal{D}$, we have an isomorphism

$$(\gamma_f)_! \iota^*(F) \simeq \bar{\iota}^* \gamma_!(F)$$

induced by the square

$$\begin{tikzcd}
\mathcal{C}_f \arrow[r, "\iota"] \arrow[d, "\gamma_f"'] & \mathcal{C} \arrow[d, "\gamma"] \\
L(\mathcal{C}_f) \arrow[r, "\bar{\iota}"'] & L(\mathcal{C})
\end{tikzcd} ,$$

where ι is the canonical inclusion, $\bar{\iota}$ is induced by the universal property applied to $\gamma \cdot \iota$ and we have commutativity up to J-homotopy.

Proof. We only need to prove that the evaluation of the canonical map $(\gamma_f)_!\iota^*(F) \to \bar{\iota}^*\gamma_!(F)$ at any object x of $\mathcal{C}_f$ is invertible. This evaluation is equivalent by [Cis19, Prop. 6.4.9] to the map

$$\operatorname{colim}_{\mathcal{C}_f/\gamma_f(x)} i^*(F)/\gamma_f(x) \to \operatorname{colim}_{\mathcal{C}/\gamma(x)} F/\gamma(x),$$

where $F/\gamma(x)$ is defined by composing F with the canonical projection $\mathcal{C}/\gamma(x) \to \mathcal{C}$ and similarly for $i^*(F)/\gamma_f(x)$. Using the commutativity of the square above, we get that the desired map is indeed invertible for all x by proving that the functor $\mathcal{C}_f/\gamma_f(x) \to \mathcal{C}/\gamma(x)$ is final, which is true by [Cis19, Thm. 7.5.16]. □

Proposition 3.2.13. Let $\mathcal{C}$ be an ∞-category with weak equivalences and fibrations. The canonical functor $\bar{\iota}: L(\mathcal{C}_f) \to L(\mathcal{C})$ is an equivalence of ∞-categories, hence the ∞-category $L(\mathcal{C})$ is finitely complete and the localization functor $\gamma: \mathcal{C} \to L(\mathcal{C})$ is left exact.

Proof. We already know that $\bar{\iota}$ is essentially surjective as every object in $\mathcal{C}$ is weakly equivalent to one in $\mathcal{C}_f$ and the localization functors are essentially surjective themselves, thus it is enough to prove that it is fully faithful. To do this, we may fix a universe $\mathbf{U}$ such that $\mathcal{C}$ is $\mathbf{U}$-small, prove that the functor

$$\bar{\iota}_!: \underline{\mathrm{Hom}}(L(\mathcal{C}_f), \mathcal{S}) \to \underline{\mathrm{Hom}}(L(\mathcal{C}), \mathcal{S})$$

is fully faithful and use [Cis19, Prop. 6.1.15]. Remember that this full faithfulness condition is equivalent to the unit map $1 \to \bar{\iota}^*\bar{\iota}_!$ of the adjunction $\bar{\iota}_! \dashv \bar{\iota}^*$ being invertible.

We know that $\bar{\iota}_!$ and $\bar{\iota}^*$ both have right adjoints, thus they preserve colimits. Also, every $\mathcal{S}$-valued functor indexed by a $\mathbf{U}$-small ∞-category can be obtained as a colimit of representable ones, hence it is enough to check that the condition holds for any representable functor F. Also, γ_f is essentially surjective, meaning that it is sufficient to check that the map $(\gamma_f)_! \to \bar{\iota}^*\bar{\iota}_!(\gamma_f)_!$ (which we get by precomposing the unit with $(\gamma_f)_!$) is invertible.

We have then the chain of isomorphisms

$$\begin{aligned}
(\gamma_f)_! &\simeq (\gamma_f)_!\bar{\iota}^*\bar{\iota}_! \\
&\simeq \bar{\iota}^*\gamma_f\iota_! \\
&\simeq \bar{\iota}^*\bar{\iota}_!(\gamma_f)_!,
\end{aligned}$$

where the first comes from the full faithfulness of ι, the second from Prop. 3.2.12 and the last one from the fact that $\bar{\iota} \cdot \gamma_f \simeq \gamma \cdot \iota$.

The second claim follows directly from the first one and Prop. 3.2.11. □

Remark 3.2.14. Here we see that the theory of localizations of ∞-categories with weak equivalences and fibrations provides much better results the 1-categorical equivalent, embodied by the aforementioned fibration categories: indeed, these are particular cases of the ∞-analogue, however their homotopy categories, i.e. their 1-categorical localizations by weak equivalences, are almost never finitely complete.

Construction 3.2.15. Given an ∞-category $\mathcal{C}$ with weak equivalences and fibrations, we can get another one $\overline{\mathcal{C}}$ with the same underlying ∞-category and class of fibrations, but where the weak equivalences are given by the saturation $\overline{W}$ as described in Remark 3.1.8. We have that $L(\mathcal{C}) \simeq L(\overline{\mathcal{C}})$, hence in general we can substitute $\mathcal{C}$ by $\overline{\mathcal{C}}$ with no issues. Also, the substitution commutes with the formation of slices over fibrant objects, that is, for any fibrant object x of $\mathcal{C}$, a map in $\mathcal{C}/x$ induces invertible in $L(\mathcal{C}/x)$ if and only if its image becomes invertible in $L(\mathcal{C})$.

Remark 3.2.16. Let $\mathcal{C}$ be an ∞-category with weak equivalences W and $F\colon \mathcal{C} \to \mathcal{D}$ be a functor. The precomposition functor $\gamma^*\colon \underline{\mathrm{Hom}}(L(\mathcal{C}), \mathcal{D}) \to \underline{\mathrm{Hom}}(\mathcal{C}, \mathcal{D})$ does not have a left adjoint in general, but we may ask whether $\mathrm{Hom}(F, \gamma^*(-))$ is representable in $\underline{\mathrm{Hom}}(L(\mathcal{C}), \mathcal{D})$. If it is, a representative is denoted by $\mathbf{R}F\colon L(\mathcal{C}) \to \mathcal{D}$ and is called the *right derived functor of F*. Beware that to be precise one would have to specify the natural transformation $F \to \mathbf{R}F \cdot \gamma$ exhibiting it as such. Dually, a representative of $\mathrm{Hom}(\gamma^*(-), F)$ is the *left derived functor of F*. These notions admit a theory which reproduces the corresponding 1-categorical one.

Proposition 3.2.17. If $F\colon \mathcal{C} \to \mathcal{D}$ sends weak equivalences to isomorphisms, then the functor $\overline{F}\colon L(\mathcal{C}) \to \mathcal{D}$, associated to F by the universal property of $L(\mathcal{C})$, is the right derived functor of F.

Proof. Let's fix a universe $\mathbf{U}$ such that $\mathcal{C}$ and $\mathcal{D}$ are $\mathbf{U}$-small and let $G\colon L(\mathcal{C}) \to \mathcal{D}$ be any functor. Then the invertible map $\overline{F} \cdot \gamma \simeq F$ and the equivalence of ∞-categories $\underline{\mathrm{Hom}}(L(\mathcal{C}), \mathcal{D}) \simeq \underline{\mathrm{Hom}}_W(\mathcal{C}, \mathcal{D})$ induce invertible maps

$$\mathrm{Hom}(\overline{F}, G) \simeq \mathrm{Hom}(\overline{F} \cdot \gamma, G \cdot \gamma) \simeq \mathrm{Hom}(F, G \cdot \gamma)$$

in $\mathcal{S}$, functorially in G. □

One can not always obtain a left derived functor as in the previous proposition because not all weak equivalences may be mapped to isomorphisms, however in the 1-categorical case, when working with model categories, we can define one by taking for every object x a fibrant-cofibrant replacement x' and then set $\overline{F}x = Fx'$. The following construction replicates this.

Construction 3.2.18. Let $\mathcal{C}$ be an ∞-category with weak equivalences and fibrations. Any functor $F\colon \mathcal{C} \to \mathcal{D}$ sending weak equivalences between fibrant objects to invertible maps then has a right derived functor $\mathbf{R}F$, which may be constructed as follows.
First we choose a quasi-inverse $R\colon L(\mathcal{C}) \to L(\mathcal{C}_f)$ of the equivalence of ∞-categories specified in Prop. 3.2.13, then we pick a functor $\overline{F}\colon L(\mathcal{C}_f) \to \mathcal{D}$ with a natural isomorphism $j\colon \overline{F} \cdot \gamma_f \to F \cdot \iota$. Finally, we set $\mathbf{R}F = \overline{F} \cdot R$.

As anticipated, what we are doing in this construction is selecting for every object in $\mathcal{C}$ a fibrant replacement, exactly like when we talk about right derived functors in the context of model categories. Also, for any other functor $G\colon \mathcal{D} \to \mathcal{E}$, we have that $G\cdot\mathbf{R}F = \mathbf{R}(G\cdot F)$. It still needs to be verified that the construction does represent $\mathrm{Hom}(F, \gamma^*(-))$, which is done in [Cis19, Rem. 7.5.25].

Definition 3.2.19. Given an ∞-category with weak equivalences and fibrations $\mathcal{C}$ and an ∞-category with weak equivalences $\mathcal{D}$, let's consider a functor $F\colon \mathcal{C} \to \mathcal{D}$ preserving weak equivalences between fibrant objects of $\mathcal{C}$. We call the *right derived functor of F* the right derived functor of the composition

$$\mathcal{C} \xrightarrow{F} \mathcal{D} \xrightarrow{\gamma_{\mathcal{D}}} L(\mathcal{D}),$$

where $\gamma_{\mathcal{D}}$ is the localization functor of of $\mathcal{D}$ at its weak equivalences. This right derived functor of F is denoted by $\mathbf{R}F$, that is $\mathbf{R}F = \mathbf{R}(\gamma_{\mathcal{D}}\cdot F)\colon L(\mathcal{C}) \to L(\mathcal{D})$, which makes sense since we can apply Construction 3.2.18.

Remark 3.2.20. The right derived functor of any left exact functor exists by Corollary 3.2.6. Specifically, the one of the localization functor $\gamma\colon \mathcal{C} \to L(\mathcal{C})$ for an ∞-category with weak equivalences and fibrations $\mathcal{C}$ is the identity, meaning that the construction is compatible with composition.

Proposition 3.2.21. For any left exact functor $F\colon \mathcal{C} \to \mathcal{D}$ between ∞-categories with weak equivalences and fibrations, the right derived functor $\mathbf{R}F\colon L(\mathcal{C}) \to L(\mathcal{D})$ is left exact.

Proof. We have a square

$$\begin{tikzcd} L(\mathcal{C}_f) \arrow[r, "\overline{F}"] \arrow[d] & L(\mathcal{D}_f) \arrow[d] \\ L(\mathcal{C}) \arrow[r, "\mathbf{R}F"'] & L(\mathcal{D}) \end{tikzcd}$$

commuting up to J-homotopy, where the vertical maps are equivalences of ∞-categories and $\overline{F}$ is the functor obtained by restricting F to the subcategories of fibrant objects $\mathcal{C}_f$ and $\mathcal{D}_f$. It therefore suffices to show that $\overline{F}$ is left exact, but this follows from Proposition 3.2.11. □

3.3 Right Approximation Property

Here we study some sufficient conditions under which functors induce equivalences of ∞-categories after localizing. We shall use this to prove Theorem 3.3.8, which specifies some sufficient conditions guaranteeing that the localization of an ∞-category is locally cartesian closed and generalizes [Kap17, Thm. 5.3].

Remark 3.3.1. For the remainder of this chapter, given an ∞-category $\mathcal{C}$, subcategories of weak equivalences $W \subset \mathcal{C}$ are such that the inclusion $W \to \mathcal{C}$ is an inner fibration. This

means that a simplex $x\colon \Delta^n \to \mathcal{C}$ lies in W if and only if its edges $x|_{\Delta^{\{i,i+1\}}}\colon \{i, i+1\} \to \mathcal{C}$ lie in W for all $0 \le i < n$.
W then contains all invertible maps of $\mathcal{C}$ if and only if the aforementioned inclusion is an isofibration.

Definition 3.3.2. Let $\mathcal{C}$, $\mathcal{D}$ be ∞-categories with subcategories of weak equivalences $V \subset \mathcal{C}$, $W \subset \mathcal{D}$. A functor $f\colon \mathcal{C} \to \mathcal{D}$ has the *right approximation property* if the following conditions hold:

1. a morphism in $\mathcal{C}$ is in V if and only if its image under f is in W;

2. given objects c, d in $\mathcal{C}$, $\mathcal{D}$ respectively and a map $\psi\colon d \to f(c)$ in $\mathcal{D}$, there is a map $\phi\colon c' \to c$ in $\mathcal{C}$ and a weak equivalence $u\colon d \to f(c')$ in $\mathcal{D}$ such that the triangle

$$\begin{tikzcd} d \arrow[r, "\psi"] \arrow[d, "u"'] & f(c) \\ f(c') \arrow[ur, "f(\phi)"'] & \end{tikzcd}$$

   commutes.

As an example, we prove the following.

Proposition 3.3.3. A functor $f\colon \mathcal{C} \to \mathcal{D}$ between ∞-categories such that the induced functor on the homotopy categories $ho(f)\colon ho(\mathcal{C}) \to ho(\mathcal{D})$ is an equivalence of categories has the right approximation property if we take $V = k(\mathcal{C})$, $W = k(\mathcal{D})$ as weak equivalences.

Proof. Consider a map $\psi\colon d \to f(c)$. Since $ho(f)$ is essentially surjective, there exists an invertible map $d \to ho(f)(c') = f(c')$ in $ho(\mathcal{D})$, which comes from an invertible map $d \to f(c')$ of $\mathcal{D}$. In $ho(\mathcal{D})$ we can then complete this to a triangle

$$\begin{tikzcd} d \arrow[r, "{[\psi]}"] \arrow[d, "\sim"'] & f(c) \\ f(c') \arrow[ur, "{[\phi]}"'] & \end{tikzcd}$$

and, since $ho(f)$ is fully faithful, $[\phi]$ can be lifted to $[\tilde{\phi}]\colon c' \to c$ in $ho(\mathcal{C})$. This gives us a commutative triangle

$$\begin{tikzcd} d \arrow[r, "\psi"] \arrow[d, "\sim"'] & f(c) \\ f(c') \arrow[ur, "f(\tilde{\phi})"'] & \end{tikzcd}$$

in $\mathcal{D}$. □

Example 3.3.4. Given an ∞-category with weak equivalences and fibrations $\mathcal{C}$, the inclusion $\mathcal{C}_f \to \mathcal{C}$ has the right approximation property.

Example 3.3.5. Given a saturated ∞-category with weak equivalences and fibrations $\mathcal{C}$, the localization functor $\mathcal{C} \to L(\mathcal{C})$ has the right approximation property as per [Cis19, Ex. 7.6.4].

Theorem 3.3.6. Let $f\colon \mathcal{C} \to \mathcal{D}$ be a functor between ∞-categories with finite limits. If f commutes with them, then the following conditions are equivalent:

1. the functor f is an equivalence of ∞-categories;

2. the functor $ho(f)\colon ho(\mathcal{C}) \to ho(\mathcal{D})$ is an equivalence of categories;

3. the functor f has the right approximation property.

Proof. We trivially have that (1) implies (2) and, by Proposition 3.3.3, (3) follows from (2), hence we only have to show that (3) gives (1).
Let's assume then that f has the right approximation property. Given a final object e of $\mathcal{C}$, $f(e)$ is still final in $\mathcal{D}$ by Proposition 3.1.10, thus for any object d of $\mathcal{D}$ we have a map $d \to f(e)$ and, by the right approximation property, we get a commutative triangle with an isomorphism $d \to f(c)$ for some c in $\mathcal{C}$, which gives us essential surjectivity.
We are still missing full faithfulness. To prove it, we use that the right approximation property implies that we have an equivalence of ∞-groupoids $k(f)\colon k(\mathcal{C}) \to k(\mathcal{D})$, as per [Cis19, Lem. 7.6.7], and that for any object c of $\mathcal{C}$ the map $\mathcal{C}/c \to \mathcal{D}/f(c)$ induced on the slices still has the right approximation property, which is [Cis19, Prop. 7.6.7], therefore again we get an equivalence of ∞-groupoids $k(\mathcal{C}/c) \to k(\mathcal{D}/f(c))$.
Keeping these facts in mind, let's look at the projection $\mathcal{C}/c \to \mathcal{C}$. This functor is conservative, thus the square

$$\begin{tikzcd} k(\mathcal{C}/c) \arrow[r] \arrow[d] & \mathcal{C}/c \arrow[d] \\ k(\mathcal{C}) \arrow[r] & \mathcal{C} \end{tikzcd}$$

is a pullback. We observe by pasting that the pullback of $k(\mathcal{C}/c) \to k(\mathcal{C})$ along $c'\colon \Delta^0 \to k(\mathcal{C})$ is $\mathcal{C}(c', c)$, as it is clear from the diagram

$$\begin{tikzcd} \mathcal{C}(x,y) \arrow[r] \arrow[d] & k(\mathcal{C}/c) \arrow[r] \arrow[d] & \mathcal{C}/c \arrow[d] \\ \Delta^0 \arrow[r, "c'"'] & k(\mathcal{C}) \arrow[r] & \mathcal{C} \end{tikzcd}.$$

In the same way, we get that the pullback of $k(\mathcal{D}/f(c)) \to k(\mathcal{D})$ along $f(c')\colon \Delta^0 \to k(\mathcal{D})$ is $\mathcal{D}(f(c'), f(c))$. Since we have a commutative square

$$\begin{tikzcd} k(\mathcal{C}/c) \arrow[r] \arrow[d] & k(\mathcal{D}/f(c)) \arrow[d] \\ k(\mathcal{C}) \arrow[r] & k(\mathcal{D}) \end{tikzcd},$$

where the horizontal maps are equivalences of ∞-groupoids, the induced map $\mathcal{C}(c', c) \to \mathcal{D}(f(c'), f(c))$ is again an equivalence of ∞-groupoids, which is what we wanted.
Since f is essentially surjective and fully faithful, it is an equivalence of ∞-categories. □

Corollary 3.3.7. Let $\mathcal{C}$ be an ∞-category with weak equivalences and fibrations and consider a localization functor $\gamma: \mathcal{C} \to L(\mathcal{C})$. For any fibrant object x of $\mathcal{C}$, the canonical functor $\mathcal{C}/x \to L(\mathcal{C})/\gamma(x)$, $t \mapsto \gamma(t)$, induces an equivalence of ∞-categories $L(\mathcal{C}/x) \simeq L(\mathcal{C})/\gamma(x)$.

Proof. By Construction 3.2.15, we can assume that $\mathcal{C}$ is saturated. Our objective is to show that the induced functor $\phi: L(\mathcal{C}/x) \to L(\mathcal{C})/\gamma(x)$ has the right approximation property and it preserves finite limits, which will allow us to apply Theorem 3.3.6 and conclude.
To show rcondition (1) we only need to prove that ϕ is conservative, which can be reduced to showing that a map in $\mathcal{C}/x$ becomes invertible in $L(\mathcal{C}/x)$ if and only if it becomes an isomorphism in $L(\mathcal{C})$. This however is true by saturation of $\mathcal{C}$. We still need to check condition (2), which can be done on $\mathcal{C}/x \to L(\mathcal{C})/\gamma(x)$, but this follows from the fact that γ has it, as mentioned in Example 3.3.5.
To apply Theorem 3.3.6 we still need to show that ϕ preserves finite limits. To do this, we use the fact that $\mathcal{C}/x$ has the structure of an ∞-category with weak equivalences and fibrations. Given that γ is left exact, the functor $\mathcal{C}/x \to L(\mathcal{C})/\gamma(x)$ maps weak equivalences to isomorphisms and we can apply Proposition 3.2.17 to prove that ϕ is its right derived functor. Finally, through Proposition 3.2.21 we get that ϕ is also left exact. □

Theorem 3.3.8. Let $\mathcal{C}$ be an ∞-category with weak equivalences and fibrations. Given a fibrant object x, consider the associated ∞-category $\mathcal{C}(x)$ as specified in Construction 3.2.7. Assume that, for any fibration between fibrant objects $p: x \to y$, the pullback functor

$$p^*: \mathcal{C}(y) \to \mathcal{C}(x), \quad (y' \to y) \mapsto (y' \times_y x \to x)$$

has a right adjoint $p_*: \mathcal{C}(x) \to \mathcal{C}(y)$ preserving trivial fibrations. Then, for any map $p: x \to y$ in $L(\mathcal{C})$, the pullback functor $p^*: \mathcal{C}(y) \to \mathcal{C}(x)$ has a right adjoint.

Proof. Given a localization functor $\gamma: \mathcal{C} \to L(\mathcal{C})$, one reduces the problem to proving that, for any fibration between fibrant objects $p: x \to y$, the pullback functor

$$\gamma(p)^*: L(\mathcal{C})/\gamma(y) \to L(\mathcal{C})/\gamma(x)$$

has a right adjoint.
By Corollary 3.2.6, any functor preserving trivial fibrations between fibrant objects also preserves weak equivalences. Also, since $p^*: \mathcal{C}(y) \to \mathcal{C}(x)$ has both a right and a left adjoint named p_* and $p_!$ (the latter given by post-composing with p) which do preserve them, by Proposition 3.1.12 we have a pair of adjunctions on the localizations, namely

$$\overline{p}^*: L(\mathcal{C}(y)) \underset{\perp}{\rightleftarrows} L(\mathcal{C}(x)) : \overline{p}_*,$$

$$\overline{p}_!: L(\mathcal{C}(x)) \underset{\perp}{\rightleftarrows} L(\mathcal{C}(y)) : \overline{p}^*.$$

Given that $(\mathcal{C}/z)_f = \mathcal{C}(z)$ for all fibrant objects z of $\mathcal{C}$, by Proposition 3.2.13 we have that $L(\mathcal{C}(z)) \simeq L(\mathcal{C}/z)$ and, by Corollary 3.3.7, we also know that $L(\mathcal{C}/z) \simeq L(\mathcal{C})/\gamma(z)$, hence $L(\mathcal{C}(z)) \simeq L(\mathcal{C})/\gamma(z)$. Notice that $\overline{p}_!$ is equivalent to $\gamma(p)_! \colon L(\mathcal{C})/\gamma(x) \to L(\mathcal{C})/\gamma(y)$ and, by essential uniqueness of the adjoints, this also applies to $\gamma(p)^*$ and $\overline{p}^*$, therefore $\gamma(p)^*$ has a right adjoint induced by $\overline{p}_*$. $\square$

4 Localizations of Syntactic Categories

In this chapter we finally talk about localizing syntactic categories. To do this, we first specify a class of maps we localize at, explaining the rationale behind our choice, and then from this we present a fibrational structure which will allow us to apply the results from the previous section to prove Theorem 4.2.4. The main difference between our proof and Kapulkin's original in one [Kap14] will be our usage of extended logical structures, mimicking his later methods from [Kap17]. We conclude with a section summarizing what we have done.

4.1 Bi-Invertibility

As anticipated, we introduce a notion of weak equivalence in our context.

Definition 4.1.1. Given a contextual category with Id-structure C, a morphism $f\colon \Gamma.A \to \Gamma.B$ over Γ is *simply bi-invertible over* Γ if there exist:

1. a morphism $g_1\colon \Gamma.B \to \Gamma.A$;
2. a section $\eta\colon \Gamma.A \to \Gamma.A.(1_A, g_1 f)^* \mathsf{Id}_A$;
3. a morphism $g_2\colon \Gamma.B \to \Gamma.A$;
4. a section $\epsilon\colon \Gamma.B \to \Gamma.B.(1_B, f g_2)^* \mathsf{Id}_B$.

We now generalize the above definition to arbitrary context extensions in $\mathsf{Syn}(\mathbf{T})$ by working as usual with $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ to provide the notion given in [Kap17, Def. 1.4].

Definition 4.1.2. Given a dependent type theory with Id-types $\mathbf{T}$, a morphism $f\colon \Gamma.\Delta \to \Gamma.\Theta$ over Γ in $\mathsf{Syn}(\mathbf{T})$ is *bi-invertible over* Γ if it is simply bi-invertible over Γ as a morphism between the two corresponding simple context extensions of Γ in $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$. It is then called *bi-invertible* if Γ is the empty context.

Remark 4.1.3. A morphism $f\colon \Gamma.\Delta \to \Gamma.\Theta$ bi-invertible over Γ is also bi-invertible since we can extend a section $\Gamma.\Delta \to \Gamma.\Delta.(1_\Delta, gf)^* \mathsf{Id}_\Delta$ to a section $\Gamma.\Delta \to \Gamma.\Delta.(1_{\Gamma.\Delta}, gf)^* \mathsf{Id}_{\Gamma.\Delta}$ thanks to the map $\mathsf{refl}_\Gamma\colon \Gamma \to \Gamma.p_\Gamma^*\Gamma.\mathrm{Id}_\Gamma$.

Remark 4.1.4. Our definition of bi-invertible map is less general than the one generally presented (which concerns arbitrary contextual categories with an Id-structure) because that one relies on strong Σ-types, which we do not assume, or on the Id-structure on iterated context extensions, however we did not provide the construction for arbitrary contextual categories.

Remark 4.1.5. Our interest in simply bi-invertible morphisms stems from the fact that simple ones model the right notion of invertible map in dependent type theory: indeed, from the required data for a simply bi-invertible map f over Γ we can provide a section $\Gamma \to \Gamma.\mathsf{isHIso}(f)$ (MAYBE ACTUALLY DO IT?) as defined in [KL12, Def. B.3.3], which itself is the translation into the language of contextual categories of the notion of bi-invertible map in type theory [Uni13, Def 4.3.1].
It is important to note that, while type theory has no way to encode internally the concept of isomorphism of the contextual model, it does have its own internal notion of isomorphism. However, given a type-theoretic map f, the associated type $\mathsf{isIso}(f)$ is not, in general, a *mere proposition* [Uni13, Def. 3.3.1], unlike $\mathsf{isHIso}(f)$ [Uni13, Thm. 4.3.2], which makes the latter preferable. Also, every bi-invertible map can be given the structure of an type-theoretic isomorphism and viceversa, hence they are closely related.

Remark 4.1.6. Assuming the Initiality Conjecture 1.2, localizing a contextual category with an Id-structure at bi-invertible maps gives us an ∞-category modeling a dependent type theory with Id-types. It has been conjectured that such a type theory should provide the internal language of ∞-categories, but a precise statement of this correspondence has not yet been produced and only a few results in this direction have been proven so far. A conjecture of this kind would amount to an equivalence between an ∞-category of contextual categories with some extra structures and an ∞-category of structured ∞-categories induced by the localization at bi-invertible maps, which means that first have to determine what properties the localization has. Notable in this sense is the already mentioned Internal Languages Conjecture 4.3.

To study the localizations of syntactic categories we want to apply the results from the previous section, which require us to specify a fibrational structure by also providing a class of fibrations.

Definition 4.1.7 ([Bro73])**.** A *fibration category* is a triple $(\mathcal{P}, W, \mathrm{Fib})$ where $\mathcal{P}$ is a category and W, Fib are subcategories such that:

1. $\mathcal{P}$ has a terminal object;
2. maps to the terminal object lie in Fib;
3. Fib and $W \cap \mathrm{Fib}$ are closed under pullback along any map in $\mathcal{P}$;
4. every map in $\mathcal{P}$ can be factored as a map in W followed by one in Fib;
5. W has the 2-out-of-6 property.

Remark 4.1.8. Seeing $\mathcal{P}$, W and Fib in the above definition as ∞-categories by applying the nerve functor, we notice that the triple has canonically the structure of an ∞-category of fibrant objects, hence we shall adopt the conventions we used in that context.

Remark 4.1.9 ([Shu14])**.** Our definition differs slightly from the original one by Brown because it asks for W to be closed under the 2-out-of-6 property instead of the more classical 2-out-of-3, but, as shown by Cisinski, if all of the other axioms are satisfied then the following are equivalent:

1. W has the 2-out-of-6 property;

2. W has the 2-out-of-3 and is saturated, that is a morphism in $\mathcal{P}$ becomes invertible in $\mathbf{Ho}(\mathcal{P})$ if and only if it lies in W.

See [Rad06, Thm. 7.2.7]. Another difference is that Brown only requires factorizations of the diagonal maps $X \to X \times X$, but then he derives our factorization condition from the other properties.

Now we have enough to specify the fibrational structure.

Proposition 4.1.10 ([AKL15, Thm. 3.2.5])**.** *Any contextual category with Σ-, Id-, Nat- and $\mathsf{1}$-structures C carries the structure of a fibration category, where the class of fibrations is given by all maps isomorphic to dependent projections and weak equivalences correspond to all of the bi-invertible ones.*

The above result can be however generalized.

Proposition 4.1.11. *A contextual category with Σ- and Id- structures C carries the structure of a fibration category given by the same classes of maps as above.*

Proof. It suffices to note that at no point the proof uses the other structures. □

Remark 4.1.12. We shall refer to the above classes of maps as weak equivalences and fibrations even in absence of a Σ-structure.

Remark 4.1.13. The results as stated rely on the Initiality Conjecture 1.2 (since in their reasoning the authors used internal languages) and their proof makes use of strong Σ-types, which they adopted to say that every dependent projection is isomorphic to a basic one. We can avoid relying on strong Σ-types by constructing the fibrational structure on $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ (where the condition on dependent projections holds by construction) to later carry it back to $\mathsf{Syn}(\mathbf{T})$ through the equivalence, while we do not need the Initiality Conjecture because we can argue in $\mathsf{Syn}(\mathbf{T})^{\mathsf{cxt}}$ using the dependent type theory $\mathbf{T}$.

Corollary 4.1.14. *Given a dependent type theory with Σ- and Id-types $\mathbf{T}$, the associated ∞-category $L(N(\mathsf{Syn}(\mathbf{T})))$ is finitely complete.*

Proof. It follows directly from Proposition 4.1.11 and Proposition 3.2.13. □

4.2 Local Cartesian Closure

We are now ready to provide a few results needed to show that, given a dependent type theory with Σ-, Id-, Π_η-types and function extensionality $\mathbf{T}$, the hypothesis of Theorem 3.3.8 are satisfied by $\mathsf{Syn}(\mathbf{T})$ with respect to the above fibrational structure. Henceforth, we shall write C for $\mathsf{Syn}(\mathbf{T})$.

Lemma 4.2.1 ([Kap14, Thm. 9.3.15])**.** *For any dependent projection $p_\Delta : \Gamma.\Delta \to \Gamma$ in C, the pullback functor $p_\Delta^* : \mathsf{C}(\Gamma) \to \mathsf{C}(\Gamma.\Delta)$ between the fibrant slices admits a right adjoint.*

Proof. Let's set $(p_\Delta)_*(\Gamma.\Delta.\Theta) = \Gamma.\mathsf{\Pi}(\Delta,\Theta)$. Our counit shall be given by

$$\epsilon_{\Gamma.\Delta.\Theta}\colon \Gamma.\Delta.p_\Delta^*\,\mathsf{\Pi}(\Delta,\Theta) \xrightarrow{\mathsf{exch}_{\Delta,\mathsf{\Pi}(\Delta,\Theta)}} \Gamma.\mathsf{\Pi}(\Delta,\Theta).p_{\mathsf{\Pi}(\Delta,\Theta)}^*\Delta \xrightarrow{\mathsf{app}_{\Delta,\Theta}} \Gamma.\Delta.\Theta$$

and it is then sufficient to prove that, for any context morphism $f\colon \Gamma.\Delta.p_\Delta^*\Psi \to \Gamma.\Delta.\Theta$ over $\Gamma.\Delta$, there is a unique $\tilde{f}\colon \Gamma.\Psi \to \Gamma.\mathsf{\Pi}(\Delta,\Theta)$ making the diagram

$$\begin{tikzcd}
\Gamma.\Delta.p_\Delta^*\Psi \arrow[d, dotted, "p_\Delta^*(\tilde{f})"'] \arrow[dr, "f"] & \\
\Gamma.\Delta.p_\Delta^*\,\mathsf{\Pi}(\Delta,\Theta) \arrow[r, "\epsilon_{\Gamma.\Delta.\Theta}"'] & \Gamma.\Delta.\Theta
\end{tikzcd}$$

commute. This will then uniquely define how the right adjoint acts on the morphisms.
We start by specifying the unit $\eta_{\Gamma.\Psi}\colon \Gamma.\Psi \to \Gamma.\mathsf{\Pi}(\Delta, p_\Delta^*\Psi)$.
Let's consider the commutative square

$$\begin{tikzcd}
\Gamma.\Psi.p_\Psi^*\,\mathsf{\Pi}(\Delta,p_\Delta^*\Psi) \arrow[r, "q(p_\Psi,\mathsf{\Pi}(\Delta,p_\Delta^*\Psi))"] \arrow[d, "p_{\mathsf{\Pi}(\Delta,p_\Delta^*\Psi)}"'] & \Gamma.\mathsf{\Pi}(\Delta,p_\Delta^*\Psi) \arrow[d, "p_{\mathsf{\Pi}(\Delta,p_\Delta^*\Psi)}"] \\
\Gamma.\Psi \arrow[r, "p_\Psi"'] & \Gamma
\end{tikzcd}$$

where the map $q(p_\Psi, \mathsf{\Pi}(\Delta, p_\Delta^*\Psi))$ acts as $(y,g) \mapsto g$. If we can provide a section of the left map corresponding to $y \mapsto (y, \lambda(x:\Delta).y)$ we are done as we can then compose it with $q(p_\Psi, \mathsf{\Pi}(\Delta, p_\Delta^*\Psi))$ to get our unit $y \mapsto \lambda(x:\Delta).y$.
To construct it we pull back the map $\mathsf{id}_\Psi\colon \Gamma \to \Gamma.[\Psi,\Psi]$ from Construction 1.3.7 along $p_{p_\Psi^*\Delta}$, getting a section

$$p_{p_\Psi^*\Delta}^*\,\mathsf{id}_\Psi\colon \Gamma.\Psi.p_\Psi^*\Delta \to \Gamma.\Psi.p_\Psi^*\Delta.p_{p_\Psi^*\Delta}^*\Psi.$$

We then apply $\lambda_{\Delta,\Psi}$, thereby getting a section

$$\lambda_{\Delta,\Psi}(p_{p_\Psi^*\Delta}^*\,\mathsf{id}_\Psi) = \lambda_{\Delta,\Psi}(1_{p_\Psi^*\Delta}, p_{p_\Psi^*\Delta})\colon \Gamma.\Psi \to \Gamma.\Psi.p_\Psi^*\,\mathsf{\Pi}(\Delta, p_\Delta^*\Psi),$$

which is exactly what we were looking for.
Let us define our lift $\tilde{f}$ as the composite $\Gamma.\mathsf{\Pi}(\Delta,f)\cdot\eta_{\Gamma.\Psi}$, where $\Gamma.\mathsf{\Pi}(\Delta,f)$ was specified in Construction 1.3.9. The commutativity of the above triangle follows by type-theoretic reasoning from

$$\begin{aligned}
&(x,y)\\
\overset{\eta_{\Gamma.\Psi}}{}&\mapsto (x, \lambda(x:\Delta).y)\\
\overset{\Gamma.\mathsf{\Pi}(\Delta,f)}{}&\mapsto (x, \lambda(x:\Delta).\,\mathsf{app}(f, \mathsf{app}(\lambda(x:\Delta).y, x)))\\
\overset{\beta\text{-reduction}}{}&\equiv (x, \lambda(x:\Delta).\,\mathsf{app}(f,y))\\
\overset{\mathsf{exch}_{\mathsf{\Pi}(\Delta,\Psi),\Delta}}{}&\mapsto (\lambda(x:\Delta).\,\mathsf{app}(f,y), x)\\
\overset{\mathsf{app}_{\Delta,\Psi}}{}&\mapsto (x, \mathsf{app}(\lambda(x:\Delta).\,\mathsf{app}(f,y), x))\\
\overset{\beta\text{-reduction}}{}&\equiv (x, \mathsf{app}(f,y))
\end{aligned}$$

and our description of $f\colon \Gamma.\Delta.p_\Delta^*\Psi \to \Gamma.\Delta.\Theta$ in Remark 1.3.8, while the uniqueness of $\tilde{f}$ comes from the fact that we can not construct another derivation of the form

$$\Gamma, x : \Delta, y : \Psi \vdash g : \Pi(\Delta, \Theta)$$

such that

$$\Gamma, x : \Delta, y : \Psi \vdash \mathsf{app}(g, x) \equiv \mathsf{app}(f, y) : \Theta(x).$$

Indeed, we can derive

$$\begin{aligned} & y \\ & \mapsto g \\ \overset{\eta\text{-conversion}}{} & \equiv \lambda(x : \Delta).\, \mathsf{app}(g, x) \\ & \equiv \lambda(x : \Delta).\, \mathsf{app}(f, y) \\ & \equiv \mathsf{app}(\tilde{f}, y), \end{aligned}$$

which concludes the proof.
This also shows that $(p_\Delta)_*(f) = \Gamma.\,\Pi(\Delta, f)$, meaning that the construction of $\Gamma.\,\Pi(\Delta, f)$ is functorial. □

We know that every fibration in C is isomorphic to a dependent projection, so this tells us that every fibration induces an adjunction between fibrant slices.
To apply Theorem 3.3.8, we need to show that $(p_\Delta)_*$ is left exact (actually, we need to prove that it preserves trivial fibrations, but it essentially requires the same amount of work), which we shall do in two steps.

Lemma 4.2.2 ([KL18, Lem. 2.29])**.** Consider a bi-invertible map $f\colon \Gamma.\Delta.\Theta \to \Gamma.\Delta.\Psi$ over $\Gamma.\Delta$ in C. The map $\Gamma.\,\Pi(\Delta, f)$ is then bi-invertible over Γ.

Proof. We shall construct an homotopical right inverse to $\Gamma.\,\Pi(\Delta, f)$ by working with g_1 and η since the other part of the construction involving g_2 and ϵ is essentially identical. To do so, we consider the commutative diagram

$$\begin{tikzcd} \Gamma.\Delta.\Psi.(1_\Psi, fg_1)^*\,\mathsf{Id}_\Psi \arrow[r, "q((1_\Psi, fg_1), \mathsf{Id}_\Psi)"] \arrow[d, "p_{(1_\Psi, fg_1)^*\,\mathsf{Id}_\Psi}"] & \Gamma.\Delta.\Psi.p_\Psi^*\Psi.\,\mathsf{Id}_\Psi \arrow[d, "p_{\mathsf{Id}_\Psi}"] \\ \Gamma.\Delta.\Psi \arrow[u, bend left, "\eta"] \arrow[r, "(1_\Psi, fg_1)"'] & \Gamma.\Delta.\Psi.p_\Psi^*\Psi \end{tikzcd}$$

in $\mathsf{C}(\Gamma.\Delta)$ and apply the right adjoint $(p_\Delta)_*$, which gives us

$$\begin{aligned} (p_\Delta)_*(q((1_\Psi, fg_1), \mathsf{Id}_\Psi) \cdot \eta)\colon \Gamma.\,\Pi(\Delta, \Psi) \to & \\ \Gamma.\,\Pi(\Delta, \Psi.p_\Psi^*\Psi.\,\mathsf{Id}_\Psi) &= \Gamma.\,\Pi(\Delta, \Psi).p_{\Pi(\Delta,\Psi)}^*\,\Pi(\Delta, \Psi).\,\mathsf{Htp}_{\Delta,\Psi} \end{aligned}$$

in $\mathsf{C}(\Gamma)$. Postcomposing with the function extensionality map

$$\mathsf{ext}_{\Delta,\Psi}\colon \Gamma.\,\Pi(\Delta, \Psi).p_{\Pi(\Delta,\Psi)}^*\,\Pi(\Delta, \Psi).\,\mathsf{Htp}_{\Delta,\Psi} \to \Gamma.\,\Pi(\Delta, \Psi).p_{\Pi(\Delta,\Psi)}^*\,\Pi(\Delta, \Psi).\,\mathsf{Id}_{\Pi(\Delta,\Psi)},$$

we obtain a morphism

$$h\colon \Gamma.\,\mathsf{\Pi}(\Delta,\Psi) \to \Gamma.\,\mathsf{\Pi}(\Delta,\Psi).p^*_{\mathsf{\Pi}(\Delta,\Psi)}\,\mathsf{\Pi}(\Delta,\Psi).\,\mathsf{Id}_{\mathsf{\Pi}(\Delta,\Psi)}$$

fitting in the commutative diagram

$$\begin{tikzcd}
\Gamma.\,\mathsf{\Pi}(\Delta,\Psi) \arrow[rrd, bend left, "h"] \arrow[rd, dotted] \arrow[rdd, bend right, equal] & & \\
& \Gamma.\,\mathsf{\Pi}(\Delta,\Psi).(1_{\Psi(\Delta,\Psi)},\Gamma.\,\mathsf{\Pi}(\Delta,(1_\Psi,fg_1))^*\,\mathsf{Id}_{\mathsf{\Pi}(\Delta,\Psi)} \arrow[r] \arrow[d, "p_{(1_{\Psi(\Delta,\Psi)},\Gamma.\,\mathsf{\Pi}(\Delta,(1_\Psi,fg_1))^*\,\mathsf{Id}_{\mathsf{\Pi}(\Delta,\Psi)}}"] & \Gamma.\,\mathsf{\Pi}(\Delta,\Psi).p^*_{\mathsf{\Pi}(\Delta,\Psi)}\,\mathsf{\Pi}(\Delta,\Psi).\,\mathsf{Id}_{\mathsf{\Pi}(\Delta,\Psi)} \arrow[d, "p_{\mathsf{Id}_{\mathsf{\Pi}(\Delta,\Psi)}}"] \\
& \Gamma.\,\mathsf{\Pi}(\Delta,\Psi) \arrow[r, "(1_{\Psi(\Delta,\Psi)}{,}\Gamma.\,\mathsf{\Pi}(\Delta{,}(1_\Psi{,}fg_1)))"'] & \Gamma.\,\mathsf{\Pi}(\Delta,\Psi).p^*_{\mathsf{\Pi}(\Delta,\Psi)}\,\mathsf{\Pi}(\Delta,\Psi)
\end{tikzcd}$$

and thereby inducing the factorization shown, which is what we needed. □

Lemma 4.2.3 ([Kap14, Thm. 9.3.15])**.** In the above conditions, the functor $(p_\Delta)_*\colon \mathsf{C}(\Gamma.\Delta) \to \mathsf{C}(\Gamma)$ is left exact.

Proof. As a right adjoint, $(p_\Delta)_*$ preserves limits and in particular pullbacks and the terminal object. Also, by Lemma 2.2.3, it preserves dependent projections and Lemma 4.2.2 tells us that the same goes for weak equivalences, which concludes the proof. □

Again, the above extends to all fibrations in C and it allows to prove our desired result.

Theorem 4.2.4 ([Kap14, Thm. 9.3.17])**.** Given a dependent type theory with $\mathsf{\Sigma}$-, Id-, $\mathsf{\Pi}_\eta$-types and function extensionality $\mathbf{T}$, the ∞-category $L(N(\mathsf{Syn}(\mathbf{T})))$ is locally cartesian closed.

Proof. We already know that it is finitely complete by Corollary 4.1.14. The hypothesis of Theorem 3.3.8 are satisfied by Lemma 4.2.1 and Lemma 4.2.3. □

4.3 Conclusions

We wrap up by summarizing what we have done and presenting developments which followed Kapulkin's result.
In Chapter 1 we introduced dependent type theories, specifying the logical rules we would consider throughout, and then contextual categories, a class of algebraic models for type-theoretic structural rules. As a motivating example, we introduced the syntactic category of a dependent type theory $\mathsf{Syn}(\mathbf{T})$. Later, we presented extra structures on contextual categories meant to interpret logical rules and some constructions derived from $\mathsf{\Pi}$-structures, showing what they correspond to in $\mathsf{Syn}(\mathbf{T})$. We were careful to explain the role of the Initiality Conjecture 1.2 in the field, referring the state of the art, and in Remark 1.2.12 we showed some results derived from it and our theorem, like the Internal Languages Conjecture 1.2.12, which we will get to at the end of this section.
Going back to our summary, in Chapter 2 we moved on to extending the logical structures so that we may talk about $\Gamma.\,\mathsf{\Pi}(\Delta,\Theta)$, $\Gamma.\Delta.p^*_\Delta\Delta.\,\mathsf{Id}_\Delta$ and so on even when Δ,Θ are not simple context extensions, focusing specifically on $\mathsf{\Pi}$-structures (see Lemma 2.2.1). This

was instrumental in our proof of Theorem 4.2.4 and a deviation from Kapulkin's approach in [Kap14], but identical to the one he took instead in [Kap17].

In Chapter 3 we presented a theory of localizations of ∞-categories referring to Cisinski's work **??**. Our objective was to provide some sufficient conditions under which the localization of an ∞-category would be locally cartesian closed and to do this we introduced ∞-categories with weak equivalences and fibrations 3.2.1, which generalize model categories and, more specifically, fibration categories 4.1.11. The conditions were then given by Theorem 3.3.8.

Finally, in Chapter 4 we introduced, for a class of dependent type theories $\mathbf{T}$, a notion of bi-invertible map in $\mathsf{Syn}(\mathbf{T})$ describing the weak equivalences we intended to localize at and then a fibrational structure, allowing us to refer to the results of the previous section when talking about $L(N(\mathsf{Syn}(\mathbf{T})))$. To prove that this latter ∞-category is locally cartesian closed we aimed to apply Theorem 3.3.8, for which we provided a right adjoint to the pullback functor between fibrant slices $p_{\Delta}^{*}\colon \mathsf{Syn}(\mathbf{T})(\Gamma) \to \mathsf{Syn}(\mathbf{T})(\Gamma.\Delta)$ through Lemma 4.2.1 and then showed that it is left exact, leading up to Theorem 4.2.4, our objective.

A generalization of this result produced by Kapulkin in [Kap17] applies to all contextual categories with the logical structures meant to model the considered logical rules and a similar one concerns finitely complete ∞-categories as a corollary of Szumilo's work on fibration categories [Szu14] or, more recently, Theorem 3.2.13. This led to the formulation of the Internal Languages Conjecture [KL16, Conj. 3.7], which we now briefly explain as anticipated.

Internal Languages Conjecture. *The horizontal functors structures*

$$\begin{array}{ccc} \mathsf{CxlCat}_{\Sigma,1,\mathsf{Id},\Pi\text{-}\mathsf{ext}} & \longrightarrow & \mathsf{LCCC}_{\infty} \\ \downarrow & & \downarrow \\ \mathsf{CxlCat}_{\Sigma,1,\mathsf{Id}} & \longrightarrow & \mathsf{Lex}_{\infty} \end{array}$$

given by simplicial localization are DK-equivalences, where the homotopical categories involved are the ones of contextual categories with the indicated logical structures and those of properly structured ∞-categories. Specifically, Lex_{∞} *and* LCCC_{∞} *have respectively finitely complete and locally cartesian closed ∞-categories as objects, while their morphisms are left exact functors.*

A candidate proof was provided by Nguyen and Uemura [NU22] and this conjecture means that providing an ∞-category with the appropriate properties is equivalent to producing a contextual category with the specified logical structures and, therefore, a dependent type theory with the corresponding logical rules (its internal language) through which one may equivalently reason about the ∞-category and viceversa, establishing a strong link between the two theories. However, so far we have been unable to explicitly construct the corresponding contextual category from an ∞-category which is not locally presentable, meaning that we do not have a concrete definition of the homotopical inverses to the localization functors.

There is currently an ongoing project to extend the above diagram by introducing a simplicial localization functor $\mathsf{CxlCat}_{\mathsf{HoTT}} \to \mathsf{ElTopos}_\infty$ which should itself be a DK-equivalence, and indeed the idea is that the "correct" definition of elementary ∞-topos should be one which would make the statement true (see [Ras18] for a candidate and [Ras21] for a characterization of univalence in ∞-categories). This would allow further advances in the study of Homotopy Type Theory and its extensions.
Throughout our work we didn't mention what would happen if instead of taking intensional Id-types we chose *extensional* ones. This case is, in our opinion, less interesting as such identity types are less structured and they lack an homotopical interpretation, while also making equality undecidable, however Seely [See84] managed to relate it to locally cartesian closed categories and to construct for each of them the associated dependent type theory.

Bibliography

[AKL15] J. Avigad, C. Kapulkin and P. L. Lumsdaine. 'Homotopy limits in type theory'. In: *Mathematical Structures in Computer Science* 25.5 (Jan. 2015), pp. 1040–1070. DOI: 10.1017/s0960129514000498. URL: https://doi.org/10.1017%2Fs0960129514000498.

[AR94] J. Adámek and J. Rosicky. *Locally presentable and accessible categories*. Vol. 189. Cambridge University Press, 1994.

[Bar16] C. Barwick. 'On the algebraic K-theory of higher categories'. In: *Journal of Topology* 9.1 (Jan. 2016), pp. 245–347. DOI: 10.1112/jtopol/jtv042. URL: https://doi.org/10.1112%2Fjtopol%2Fjtv042.

[BHL20] A. Bauer, P. G. Haselwarter and P. L. Lumsdaine. *A general definition of dependent type theories*. 2020. DOI: 10.48550/ARXIV.2009.05539. URL: https://arxiv.org/abs/2009.05539.

[BK12] C. Barwick and D. M. Kan. 'Relative categories: another model for the homotopy theory of homotopy theories'. In: *Indagationes Mathematicae* 23.1-2 (2012), pp. 42–68.

[Boe20] M. de Boer. 'A proof and formalization of the initiality conjecture of dependent type theory'. PhD thesis. Department of Mathematics, Stockholm University, 2020.

[Bro73] K. S. Brown. 'Abstract homotopy theory and generalized sheaf cohomology'. In: *Transactions of the American Mathematical Society* 186 (1973), pp. 419–458.

[Bru20] G. Brunerie. *A formalization of general type theories in Agda*. 2020. URL: https://github.com/guillaumebrunerie/general-type-theories.

[Car78] J. Cartmell. *Generalised algebraic theories and contextual categories*. University of Oxford, 1978.

[Car86] J. Cartmell. 'Contextual Category and Generalized Algebraic Theories'. In: *Annals of Pure and Applied Logic* 32 (1986).

[Cis19] D. Cisinski. *Higher Categories and Homotopical Algebra*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2019. ISBN: 9781108473200. URL: https://books.google.it/books?id=RawqvQEACAAJ.

[Gar09] R. Garner. 'Two-dimensional models of type theory'. In: *Mathematical Structures in Computer Science* 19.4 (Aug. 2009), pp. 687–736. DOI: 10.1017/s0960129509007646. URL: https://doi.org/10.1017%2Fs0960129509007646.

[GG08] N. Gambino and R. Garner. ‘The identity type weak factorisation system’. In: *Theoretical Computer Science* 409.1 (Dec. 2008), pp. 94–109. DOI: 10.1016/j.tcs.2008.08.030. URL: https://doi.org/10.1016%2Fj.tcs.2008.08.030.

[Isa16] V. Isaev. *Algebraic Presentations of Dependent Type Theories*. 2016. DOI: 10.48550/ARXIV.1602.08504. URL: https://arxiv.org/abs/1602.08504.

[Kap14] K. Kapulkin. ‘Joyal’s Conjecture in Homotopy Type Theory’. PhD thesis. University of Pittsburgh, 2014.

[Kap17] K. Kapulkin. ‘Locally cartesian cclosed quasi-categories from type theory’. In: *Journal of Topology* 10.4 (Nov. 2017), pp. 1029–1049. DOI: 10.1112/topo.12031. URL: https://doi.org/10.1112%2Ftopo.12031.

[KL12] C. Kapulkin and P. L. Lumsdaine. *The Simplicial Model of Univalent Foundations (after Voevodsky)*. 2012. DOI: 10.48550/ARXIV.1211.2851. URL: https://arxiv.org/abs/1211.2851.

[KL16] C. Kapulkin and P. L. Lumsdaine. *The homotopy theory of type theories*. 2016. DOI: 10.48550/ARXIV.1610.00037. URL: https://arxiv.org/abs/1610.00037.

[KL18] C. Kapulkin and P. L. Lumsdaine. *Homotopical inverse diagrams in categories with attributes*. 2018. DOI: 10.48550/ARXIV.1808.01816. URL: https://arxiv.org/abs/1808.01816.

[KS15] C. Kapulkin and K. Szumiło. *Quasicategories of Frames of Cofibration Categories*. 2015. DOI: 10.48550/ARXIV.1506.08681. URL: https://arxiv.org/abs/1506.08681.

[KS19] K. Kapulkin and K. Szumiło. ‘Internal languages of finitely complete $(\infty, 1)$-categories’. In: *Selecta Mathematica* 25.2 (2019), pp. 1–46.

[Lum10] P. L. Lumsdaine. *Higher categories from type theories*. Carnegie Mellon University, 2010.

[ML84] P. Martin-Löf and G. Sambin. *Intuitionistic type theory*. Vol. 9. Bibliopolis Naples, 1984.

[nLab22] nLab authors. *categorical model of dependent types*. http://ncatlab.org/nlab/show/categorical%20model%20of%20dependent%20types. Revision 61. July 2022.

[NU22] H. K. Nguyen and T. Uemura. *∞-type theories*. 2022. DOI: 10.48550/ARXIV.2205.00798. URL: https://arxiv.org/abs/2205.00798.

[Rad06] A. Radulescu-Banu. *Cofibrations in Homotopy Theory*. 2006. DOI: 10.48550/ARXIV.MATH/0610009. URL: https://arxiv.org/abs/math/0610009.

[Ras18] N. Rasekh. *A Theory of Elementary Higher Toposes*. 2018. DOI: 10.48550/ARXIV.1805.03805. URL: https://arxiv.org/abs/1805.03805.

[Ras21] N. Rasekh. *Univalence in Higher Category Theory*. 2021. DOI: 10.48550/ARXIV.2103.12762. URL: https://arxiv.org/abs/2103.12762.

[See84] R. Seely. 'Locally cartesian closed categories and type theory'. In: *Mathematical proceedings of the Cambridge philosophical society*. Vol. 95. 1. Cambridge University Press. 1984, pp. 33–48.

[Shu14] M. Shulman. 'Univalence for inverse diagrams and homotopy canonicity'. In: *Mathematical Structures in Computer Science* 25.5 (Nov. 2014), pp. 1203–1277. DOI: 10.1017/s0960129514000565. URL: https://doi.org/10.1017%2Fs0960129514000565.

[Shu19] M. Shulman. *All* $(\infty, 1)$*-toposes have strict univalent universes*. 2019. arXiv: 1904.07004 [math.AT].

[Str91] T. Streicher. *Semantics of type theory. Progress in Theoretical Computer Science*. 1991.

[Str93] T. Streicher. 'Investigations into intensional type theory'. In: *Habilitiation Thesis, Ludwig Maximilian Universität* (1993).

[Szu14] K. Szumiło. *Two Models for the Homotopy Theory of Cocomplete Homotopy Theories*. 2014. DOI: 10.48550/ARXIV.1411.0303. URL: https://arxiv.org/abs/1411.0303.

[Uem19] T. Uemura. *A General Framework for the Semantics of Type Theory*. 2019. DOI: 10.48550/ARXIV.1904.04097. URL: https://arxiv.org/abs/1904.04097.

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

[Voe14a] V. Voevodsky. *A C-system defined by a universe category*. 2014. DOI: 10.48550/ARXIV.1409.7925. URL: https://arxiv.org/abs/1409.7925.

[Voe14b] V. Voevodsky. *Subsystems and regular quotients of C-systems*. 2014. DOI: 10.48550/ARXIV.1406.7413. URL: https://arxiv.org/abs/1406.7413.

[Voe15a] V. Voevodsky. *Martin-Lof identity types in the C-systems defined by a universe category*. 2015. DOI: 10.48550/ARXIV.1505.06446. URL: https://arxiv.org/abs/1505.06446.

[Voe15b] V. Voevodsky. *Products of families of types in the C-systems defined by a universe category*. 2015. DOI: 10.48550/ARXIV.1503.07072. URL: https://arxiv.org/abs/1503.07072.

Declaration of Authorship

Ich habe die Arbeit selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und bisher keiner anderen Prüfungsbehörde vorgelegt. Außerdem bestätige ich hiermit, dass die vorgelegten Druckexemplare und die vorgelegte elektronische Version der Arbeit identisch sind und dass ich von den in §26 Abs. 6 vorgesehenen Rechtsfolgen Kenntnis habe.

Regensburg, Germany

Matteo Durante