The objective of this thesis is to provide a more modern proof of a result by Kapulkin stating that, given a dependent type theory T with Σ, Id and Π-types, the quasi-categorical localization of its syntactic category is a locally cartesian closed ∞-category.

This goes in the direction of establishing dependent type theory as the internal language of ∞-categories with appropriate structures, which would allow us to reason about dependent type theory through ∞-categories and viceversa. This has been conjectured to be true for a while, however a proper correspondence still requires quite a lot of work.

Henceforth, I will say "localization" to refer to the ∞-categorical one. The modernity shall come from using the theory of localizations of ∞-categories developed by Cisinski.

Dependent type theory is a theory of computations and a foundation of mathematics closely related to computer science. It talks about dependent types A and their terms x over contexts Γ, that is sequences of types and choices of terms each of them derivable from the previous ones. An example is the type of n-dimensional K-vector spaces, which depends on a natural number n and on a field K. Dependent type theory provides structural rules telling us that we can substitute definitionally equal variables and types. There are then logical rules specifying how to construct the objects we talk about, in particular Σ-types $\Sigma(A, B)$, that is the type of pairs (a, b) of elements, Π-types $\Pi(A, B)$, the type of maps from A to B, and Id-types Id_A, the type of proof of equalities between terms of A. They also tell us how to carry out computations.

To study a theory, logicians study its semantics, its models, but here it's hard: structural rules require lots of checks, so providing a model is difficult. A solution is defining a class of algebraic models with simple axioms dealing with the structural rules, so that to introduce a model we only have to check those.

Such a class of models is given by contextual categories, categories specifying a grading on objects, which are to be thought of as contexts, which again are sequences of types and choices of terms each depending on the previous ones. There is a unique and terminal object of length 0, the empty context. We have then a function to reduce such contexts by dropping the last type and, for each object $\Gamma.A$, a basic dependent projection to its reduction exhibiting $\Gamma.A$ as an extension of Γ of length 1 or as a dependent type A in context Γ. Also, we want functorial pullbacks of basic dependent projections preserving them. Functoriality models the associativity of context substitution and allows us to derive from a type A in context Γ a type in context Δ. We may also not ask for functorial pullbacks, but then we would need some coherency maps which would make interpretation harder.

To model the logical rules we need some extra structure. For Id-types, for $\Gamma.A$ we need an identity object, for Π-types we want a Π object, a map describing function evaluation and so on. In particular, a dependent type theory T with some logical rules induces a syntactic category which is contextual and has the corresponding extra structures. It is freely generated by the theory and its objects are contexts, while maps are tuples of terms f_i belonging to B_i and derivable from the domain context.

We want to talk about localizations. The right notion of invertibility in dependent type theory is bi-invertibility, which we now translate in the contextual language. $f\colon\Gamma \to \Delta$ in a contextual category is bi-invertible if it has maps g_1, g_2 in the opposite direction and maps η, ϵ which tell us that they are pointwise left and right inverse to f. Thinking about types as spaces, as suggested by the groupoidal model, identity types are path spaces, hence η and ϵ are more like homotopies. What if we localize at bi-invertible maps? It makes sense to want maps which are invertible in the theory to be invertible in the model. To answer that question we need some tools.

An ∞-category C with weak equivalences W and fibrations Fib is a triple generalizing the 1-categorical notion of fibration category. Essentially, we have some stability under pullbacks of fibrations, trivial fibrations, the factorization condition and other things. As proven in 2013 by Avigad, Kapulkin and Lumsdaine, a contextual category with Σ and Id structures is one if we take bi-invertible maps as weak equivalences and maps isomorphic to dependent projections as fibrations.

We now give a sufficient condition for an ∞-category with weak equivalences and fibrations to have a locally cartesian closed localization. there is a right adjoint to the pullback functor induced by a fibration f

from x to y, both fibrant, on the fibrant slices they define and the right adjoint preserves trivial fibrations, then the localization is also locally cartesian.

We now state the theorem we mentioned at the beginning. Given a categorical model of type theory C, its localization is a locally cartesian closed ∞-category. It is enough to construct a right adjoint to the pullback functor induced by p_A from $\Gamma.A$ to Γ, which we do by sending $\Gamma.A.\Theta$ to $\Gamma.\Pi(A,\Theta)$ and using the function evaluation map as a counit. Here the Π-object on the right is defined by extending the basic Π-structure, which was one among the problems I had to deal with: it is an object of folklore, but nobody fleshed it out.

Why is dependent type theory interesting? It's a foundation for mathematics which has become of interest because it is closely linked to computer science and it allows the creation of proof assistants like Agda and Lean, that is software which allows us to formalize mathematical reasoning, easily check proofs and even write some tedious bits automatically. There are already ongoing projects to formalize all of mathematics, like the math library. Secondly, this theory merges the two strata of classical foundations into one, generalizing both sets and propositions through types. Thirdly, we can reason about proofs within the theory, without a metatheory. Finally, equality is more structured: only objects of the same type can be compared and a proof of equality specifies in what sense two objects are equal.

Why is this interesting? Because it goes towards using ∞-categories to reason about dependent types and, viceversa, using dependent types to reason about ∞-categories, something which folkloristically we know to be doable. In 2016, Kapulkin and Lumsdaine conjectured that the functor induced by localizing contextual categories modeling some dependent type theories would induce equivalences between their ∞-categories of contextual categories and ∞-categories of finitely complete or locally cartesian closed ∞-categories, an equivalence which hopefully will extend to Homotopy Type Theory and Elementary ∞-toposes.